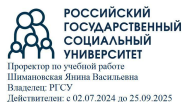


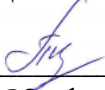


**Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Российский государственный социальный университет»**



Подписано
цифровой
подписью:
Шимановская
Янина Васильевна
Дата: 2024.07.26
11:25:23 +03'00'

УТВЕРЖДАЮ
Декан факультета политических и
социальных технологий

 / С.В. Пивнева
«28» февраля 2024 г.

**МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ
«ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ, СЕТИ И ТЕЛЕКОММУНИКАЦИИ»**

**Направление подготовки
«Информационные системы и технологии»**

**Направленность
«Интеллектуальные информационные системы и технологии»**

**ОБРАЗОВАТЕЛЬНАЯ ПРОГРАММА ВЫСШЕГО ОБРАЗОВАНИЯ - ПРОГРАММА
БАКАЛАВРИАТА**

Форма обучения

Очная, заочная

Москва, 2024 г.

Методические материалы по дисциплине (модулю) «Вычислительные системы, сети и телекоммуникации» разработаны на основании федерального государственного образовательного стандарта высшего образования – *бакалавриата* по направлению подготовки 09.03.02 *Информационные системы и технологии*, утвержденного приказом Министерства образования и науки Российской Федерации от 19.09.2017 № 926, учебного плана по основной профессиональной образовательной программе высшего образования – программе бакалавриата по направлению подготовки 09.03.02 *Информационные системы и технологии* (далее – «ОПОП»).

Методические материалы по дисциплине (модулю) разработаны канд. тех. наук, доцентом С.М. Бобровским.

Методические материалы по дисциплине (модулю) обсуждены и утверждены на заседании кафедры информационных технологий, искусственного интеллекта и общественно-социальных технологий цифрового общества факультета социальных и политических технологий.

Протокол № 11 от «28» февраля 2024 г.

Заведующий кафедрой
канд. пед. наук, доцент



(подпись)

С.В. Пивнева

СОДЕРЖАНИЕ

1. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ К ЛЕКЦИОННЫМ И ЛАБОРАТОРНЫМ ЗАНЯТИЯМ	4
1.1. Методические материалы к проведению лекционных занятий по дисциплине (модулю)	4
1.2. Методические материалы по подготовке к лабораторным занятиям по дисциплине (модулю).....	9
2. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ) И САМОСТОЯТЕЛЬНОЙ РАБОТЕ.....	16
3. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНИВАНИЯ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ И (ИЛИ) ОПЫТА ДЕЯТЕЛЬНОСТИ, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ.....	25
3.1. Организационные основы применения балльно-рейтинговой системы оценки успеваемости обучающихся по дисциплине (модулю)	26
3.2. Проведение текущего контроля успеваемости обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося	26
3.3. Проведение промежуточной аттестации обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося.....	27
ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ.....	29

1. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ К ЛЕКЦИОННЫМ И ЛАБОРАТОРНЫМ ЗАНЯТИЯМ

1.1. Методические материалы к проведению лекционных занятий по дисциплине (модулю)

Лекция - один из методов обучения, одна из основных системообразующих форм организации учебного процесса в вузе. Лекционное занятие представляет собой систематическое, последовательное, монологическое изложение педагогическим работником учебного материала, как правило, теоретического характера. Такое занятие представляет собой элемент технологии представления учебного материала путем логически стройного, систематически последовательного и ясного изложения.

Цель лекции – организация целенаправленной познавательной деятельности обучающихся по овладению программным материалом дисциплины (модуля). Чтение курса лекций позволяет дать связанное, последовательное изложение материала в соответствии с новейшими данными науки, сообщить слушателям основное содержание предмета в целостном, систематизированном виде. В ряде случаев лекция выполняет функцию основного источника информации: при отсутствии учебников и учебных пособий, чаще по новым курсам; в случае, когда новые научные данные по той или иной теме не нашли отражения в учебниках; отдельные разделы и темы очень сложны для самостоятельного изучения. В таких случаях только лектор может методически помочь обучающимся в освоении сложного материала.

Возможные формы проведения лекций:

- Вводная лекция – один из наиболее важных и трудных видов лекции при чтении систематических курсов. От успеха этой лекции во многом зависит успех усвоения всего курса. Она может содержать: определение дисциплины (модуля); краткую историческую справку о дисциплине (модуле); цели и задачи дисциплины (модуля), ее роль в общей системе обучения и связь со смежными дисциплинами (модулями); основные проблемы (понятия и определения) данной науки; основную и дополнительную учебную литературу; особенности самостоятельной работы обучающихся над дисциплиной (модулем) и формы участия в научно-исследовательской работе; отчетность по курсу.

- Информационная лекция ориентирована на изложение и объяснение обучающимся научной информации, подлежащей осмыслению и запоминанию. Это самый традиционный тип лекций в практике высшей школы.

- Заключительная лекция предназначена для обобщения полученных знаний и раскрытия перспектив дальнейшего развития данной науки.

- Обзорная лекция – это систематизация научных знаний на высоком уровне, допускающая большое число ассоциативных связей в процессе осмысления информации, излагаемой при раскрытии внутрипредметной и межпредметной связей, исключая детализацию и конкретизацию. Как правило, стержень излагаемых теоретических положений составляет научно-понятийная и концептуальная основа всего курса или крупных его разделов.

- Лекция-беседа - непосредственный контакт педагогического работника с аудиторией - диалог. По ходу лекции педагогический работник задает вопросы для выяснения мнений и уровня осведомленности обучающихся по рассматриваемой проблеме.

- Лекция-дискуссия - свободный обмен мнениями в ходе изложения лекционного материала. Педагогический работник активизирует участие в обсуждении отдельными вопросами, сопоставляет между собой различные мнения и тем самым развивает дискуссию, стремясь направить ее в нужное русло.

- Лекция с применением обратной связи включает в себе то, что в начале и конце каждого раздела лекции задаются вопросы. Первый - для того, чтобы узнать, насколько обучающиеся ориентируются в излагаемом материале, вопрос в конце раздела предназначен для выяснения степени усвоения только что изложенного материала. При

неудовлетворительных результатах контрольного опроса педагогический работник возвращается к уже прочитанному разделу, изменив при этом методику подачи материала.

- Проблемная лекция опирается на логику последовательно моделируемых проблемных ситуаций путем постановки проблемных вопросов или предъявления проблемных задач. Проблемный вопрос - это диалектическое противоречие, требующее для своего решения размышления, сравнения, поиска, приобретения и применения новых знаний. Проблемная задача содержит дополнительную вводную информацию и при необходимости некоторые ориентиры поиска ее решения.

- Программированная лекция - консультация – педагогический работник сам составляет и предлагает обучающимся вопросы. На подготовленные вопросы педагогический работник сначала просит ответить обучающихся, а затем проводит анализ и обсуждение неправильных ответов. В лекциях можно использовать наглядные материалы, а также подготовить презентацию. Что касается презентации, то в качестве визуальной поддержки ее можно органично интегрировать во все вышеупомянутые лекции. В то же время лекцию-презентацию возможно выделить и в качестве самостоятельной формы. Лекция-презентация должна отражать суть основных и (или) проблемных вопросов лекции, на которые особо следует обратить внимание обучающихся. В условиях применения активного метода проведения занятий презентация представляется весьма удачным способом донесения информации до слушателей. Единственное, на что следует обратить внимание при подготовке слайдов, - это их оформление и текст. Слайд не должен быть перегружен картинками и лишней информацией, которая будет отвлекать от основного аспекта того или иного вопроса лекции. Во время лекции можно задавать вопросы аудитории в отношении того или иного слайда, тем самым еще больше вовлекая обучающихся в проблематику.

Краткое содержание лекционных занятий

Наименование разделов	Содержание учебного материала
РАЗДЕЛ 1. ОБЩИЕ ПРИНЦИПЫ ПОСТРОЕНИЯ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ	Вычислительные машины и системы. Общие принципы построения компьютера. История и тенденции развития вычислительной техники. Основные характеристики компьютеров. Классификация компьютеров. Основные классы вычислительных машин. Большие компьютеры. Малые компьютеры. Микрокомпьютеры. Принципы построения компьютера. Структурные схемы и взаимодействие устройств компьютера. Вычислительные системы. Классификация вычислительных систем. Архитектура вычислительных систем. Классификация архитектур вычислительных систем. Типовые структуры вычислительных систем. Кластерные технологии и их развитие. Организация функционирования вычислительных систем. Функциональная и структурная организация информационных систем. Архитектурные особенности вычислительных систем различных классов.

	Многомашинные и многопроцессорные ВС. Высокопараллельные многопроцессорные вычислительные системы. Ассоциативные и потоковые ВС.
РАЗДЕЛ 2. ИНФОРМАЦИОННО-ЛОГИЧЕСКИЕ ОСНОВЫ ПОСТРОЕНИЯ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ	Информационно-логические основы построения вычислительных машин. Представление информации в вычислительных машинах. Системы счисления. Арифметические основы ЭВМ. Особенности представления информации в ПК. Логические основы построения вычислительной машины. Кодирование информации. Элементная база. Электронные технологии и элементы, применяемые в ЭВМ. Основные блоки ПК и их назначение. Микропроцессор. Системная шина. Элементы конструкции ПК. Основная память. Внешняя память. Источник питания. Внешние устройства. Функциональные характеристики ПК. Производительность, быстродействие, тактовая частота. Тип и базовые характеристики МП. Тип и емкость оперативной памяти. Статическая и динамическая оперативная память. Кэш-память. Наличие, виды и емкость кэш-памяти. Типы системного, локальных и периферийных интерфейсов. Внешние запоминающие устройства. Емкость и характеристики накопителей на магнитных дисках. Системные платы и чипсеты. Интерфейсная система ПК. Шины расширений. Системы визуального отображения информации (видеосистемы). Видеоконтроллеры. Клавиатура. Графический манипулятор мышь. Принтеры. Сканеры. Мультимедийные устройства ввода-вывода.
РАЗДЕЛ 3. ФУНКЦИОНИРОВАНИЕ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ	Программное управление - основа автоматизации вычислительного процесса. Организация функционирования ЭВМ с магистральной архитектурой. Организация работы ЭВМ при выполнении задания пользователя. Режимы работы компьютеров. Однопрограммный режим. Многопрограммный режим. Особенности управления основной памятью ЭВМ. Система прерываний ЭВМ. Адресация регистров и ячеек памяти в ПК.

	Параллельные вычисления. Инструментальные средства контроля и диагностики ЭВМ.
РАЗДЕЛ 4. ОСНОВЫ ПОСТРОЕНИЯ КОМПЬЮТЕРНЫХ СЕТЕЙ	Основы построения компьютерных сетей. Основные сведения о компьютерных сетях. Классификация и архитектура информационно-вычислительных сетей. Виды информационно-вычислительных сетей. Топология сети. Сетевые операционные системы. Технологии передачи данных. Коммутация каналов. Коммутация пакетов. Техническое обеспечение информационно-вычислительных сетей. Серверы и рабочие станции. Модемы и сетевые карты. Интеграционное оборудование. Средства масштабирования сетей. Повторители. Мосты. Коммутаторы. Маршрутизаторы, протоколы маршрутизации и шлюзы.
РАЗДЕЛ 5. СЕТИ И СЕТЕВЫЕ ТЕХНОЛОГИИ	Модель взаимодействия открытых систем. Сети и сетевые технологии нижних уровней. Физический уровень связи и уровень канала данных. Физический уровень OSI. Канальный уровень OSI. Подуровень MAC. Подуровень LLC. Взаимодействие MAC и LLC. Основные понятия TCP/IP. Сравнение стека протоколов TCP/IP с моделью OSI. Протоколы ARP и RARP. Протоколы IP, UDP и TCP. Классы IP-адресов. Формирование масок подсетей. Протокол DHCP и маршрутизация. Механизм аренды IP-адреса. IP-маршрутизация. Функционирование службы DNS. Имена службы DNS. Распознавание имен. WINS и разрешение имен NetBIOS. Управление сетями TCP/IP и протоколы прикладного уровня. Развитие протокола TCP/IP. Протокол IPv6.
РАЗДЕЛ 6. ТЕХНОЛОГИЯ БЕСПРОВОДНЫХ СЕТЕЙ	Технология беспроводных сетей. Беспроводные среды передачи. Требования к беспроводным сетям. Типы беспроводных сетей. Беспроводные персональные сети.

	Беспроводные локальные сети. Беспроводные городские сети. Беспроводные глобальные сети.
РАЗДЕЛ 7. ЛОКАЛЬНЫЕ ВЫЧИСЛИТЕЛЬНЫЕ СЕТИ	Локальные вычислительные сети. Виды локальных вычислительных сетей. Одноранговые локальные сети. Серверные локальные сети. Сети, использующие в качестве каналов связи линии электропитания. Структурированная кабельная система и логическая структуризация ЛКС. Устройства межсетевого интерфейса. Способы повышения производительности ЛВС. Организация и функционирование виртуальных локальных компьютерных сетей. Корпоративные компьютерные сети. Корпоративные информационные системы. Особенности архитектуры корпоративных компьютерных сетей. Программное и информационное обеспечение сетей. Программное обеспечение информационно-вычислительных сетей. Информационное обеспечение сетей.
РАЗДЕЛ 8. ГЛОБАЛЬНАЯ ИНФОРМАЦИОННАЯ СЕТЬ ИНТЕРНЕТ	Глобальная информационная сеть Интернет. Общие сведения о сети Интернет. Протоколы общения компьютеров в сети. Система адресации в Интернете. Варианты общения пользователя с Интернетом. Подключение компьютера для работы в Интернете. Базовые пользовательские технологии работы в Интернете. Передача файлов с помощью протокола FTP. Telnet. Электронная почта. Основные технологии работы в WWW. Гипертекстовые технологии Интернета. Обзорщики Интернета и поисковые системы.
РАЗДЕЛ 9. СИСТЕМЫ ТЕЛЕКОММУНИКАЦИЙ	Системы телекоммуникаций. Системы и каналы передачи данных. Системы передачи данных и их характеристики. Линии и каналы связи. Цифровые каналы связи. Российские сети передачи информации. Системы оперативной связи. Телефонная связь. Офисные АТС. Разновидности офисных АТС. Системы сотовой связи. Стандарты сотовой связи. Сервисы сотовой связи. Спутниковые навигационные системы GPS.

1.2. Методические материалы по подготовке к лабораторным занятиям по дисциплине (модулю)

Лабораторные занятия - одна из форм учебного занятия, направленная на развитие самостоятельности обучающихся и приобретение умений и навыков. Данные учебные занятия углубляют, расширяют, детализируют полученные ранее знания. Лабораторное занятие предполагает выполнение обучающимися по заданию и под руководством преподавателей одной или нескольких работ.

Цель лабораторных занятий состоит в развитии познавательных способностей, самостоятельности мышления и творческой активности обучающихся; углублении, расширении, детализировании знаний, полученных на лекции в обобщенной форме, и содействии выработке навыков профессиональной деятельности. В отдельных случаях на лабораторных занятиях руководителем занятия сообщаются дополнительные знания.

Для достижения поставленных целей и решения требуемого перечня задач лабораторные занятия проводятся традиционными технологиями или с использованием активных и интерактивных образовательных технологий.

При подготовке и работе во время проведения лабораторных занятий следует обратить внимание на следующие моменты: на процесс предварительной подготовки, на работу во время занятия, обработку полученных результатов, исправление полученных замечаний.

Предварительная подготовка к лабораторному занятию заключается в изучении теоретического материала в отведенное для самостоятельной работы время, ознакомление с инструктивными материалами с целью осознания задач занятия, техники безопасности при работе с приборами, веществами.

Работа во время проведения занятия включает:

- консультирование студентов преподавателями и вспомогательным персоналом с целью предоставления исчерпывающей информации, необходимой для самостоятельного выполнения предложенных преподавателем задач.
- самостоятельное выполнение заданий согласно обозначенной рабочей программой дисциплины (модуля) тематики.

Задания для самостоятельной работы к Разделу 1

Вопросы для самостоятельной работы к Разделу 1

1. История и тенденции развития вычислительной техники.
2. Основные характеристики компьютеров. Классификация компьютеров.
3. Большие компьютеры. Малые компьютеры. Микрокомпьютеры.
4. Принципы построения компьютера. Структурные схемы и взаимодействие устройств компьютера.
5. Вычислительные системы. Классификация вычислительных систем.
6. Архитектура вычислительных систем. Классификация архитектур вычислительных систем.
7. Кластерные технологии и их развитие.
8. Функциональная и структурная организация информационных систем.
9. Многомашинные и многопроцессорные ВС.
10. Ассоциативные и потоковые ВС.

Литература для самостоятельного изучения к Разделу 1

Основная литература

1. Замятина, О. М. Вычислительные системы, сети и телекоммуникации. Моделирование сетей : учебное пособие для вузов / О. М. Замятина. — Москва : Издательство Юрайт, 2022. — 159 с. — (Высшее образование). — ISBN 978-5-534-00335-2. — URL : <https://urait.ru/bcode/490257>

2. Сети и телекоммуникации : учебник и практикум для вузов / К. Е. Самуйлов [и др.] ; под редакцией К. Е. Самуйлова, И. А. Шалимова, Д. С. Кулябова. — Москва : Издательство Юрайт, 2023. — 363 с. — (Высшее образование). — ISBN 978-5-534-00949-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511092> (дата обращения: 08.03.2023).

Дополнительная литература

1. Новожилов, О. П. Архитектура ЭВМ и систем в 2 ч. Часть 1 : учебное пособие для вузов / О. П. Новожилов. — Москва : Издательство Юрайт, 2023. — 276 с. — (Высшее образование). — ISBN 978-5-534-07717-9. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/516640> (дата обращения: 08.03.2023).

2. Новожилов, О. П. Архитектура ЭВМ и систем в 2 ч. Часть 2 : учебное пособие для вузов / О. П. Новожилов. — Москва : Издательство Юрайт, 2023. — 246 с. — (Высшее образование). — ISBN 978-5-534-07718-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/516641> (дата обращения: 08.03.2023).

3. Толстобров, А. П. Архитектура ЭВМ : учебное пособие для вузов / А. П. Толстобров. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 163 с. — (Высшее образование). — ISBN 978-5-534-16839-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/531870> (дата обращения: 24.05.2023).

Задания для самостоятельной работы к Разделу 2

Вопросы для самостоятельной работы к Разделу 2

1. Представление информации в вычислительных машинах.
2. Системы счисления. Кодирование информации.
3. Элементная база. Электронные технологии и элементы, применяемые в ЭВМ.
4. Основные блоки ПК и их назначение. Микропроцессор. Системная шина.
5. Основная память. Внешняя память.
6. Функциональные характеристики ПК. Производительность, быстродействие, тактовая частота.
7. Статическая и динамическая оперативная память. Кэш-память.
8. Типы системного, локальных и периферийных интерфейсов.
9. Внешние запоминающие устройства.
10. Системные платы и чипсеты.
11. Интерфейсная система ПК. Шины расширений.
12. Системы визуального отображения информации (видеосистемы). Видеоконтроллеры.

Литература для самостоятельного изучения к Разделу 2

Основная литература

1. Замятина, О. М. Вычислительные системы, сети и телекоммуникации. Моделирование сетей : учебное пособие для вузов / О. М. Замятина. — Москва : Издательство Юрайт, 2022. — 159 с. — (Высшее образование). — ISBN 978-5-534-00335-2. — URL : <https://urait.ru/bcode/490257>

2. Сети и телекоммуникации : учебник и практикум для вузов / К. Е. Самуйлов [и др.] ; под редакцией К. Е. Самуйлова, И. А. Шалимова, Д. С. Кулябова. — Москва : Издательство Юрайт, 2023. — 363 с. — (Высшее образование). — ISBN 978-5-534-00949-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511092> (дата обращения: 08.03.2023).

Дополнительная литература

1. Новожилов, О. П. Архитектура ЭВМ и систем в 2 ч. Часть 1 : учебное пособие для вузов / О. П. Новожилов. — Москва : Издательство Юрайт, 2023. — 276 с. — (Высшее образование). — ISBN 978-5-534-07717-9. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/516640> (дата обращения: 08.03.2023).

2. Новожилов, О. П. Архитектура ЭВМ и систем в 2 ч. Часть 2 : учебное пособие для вузов / О. П. Новожилов. — Москва : Издательство Юрайт, 2023. — 246 с. — (Высшее образование). — ISBN 978-5-534-07718-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/516641> (дата обращения: 08.03.2023).

3. Толстобров, А. П. Архитектура ЭВМ : учебное пособие для вузов / А. П. Толстобров. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 163 с. — (Высшее образование). — ISBN 978-5-534-16839-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/531870> (дата обращения: 24.05.2023).

Здания для самостоятельной работы к Разделу 3

Вопросы для самостоятельной работы к Разделу 3

1. Организация функционирования ЭВМ с магистральной архитектурой.
2. Организация работы ЭВМ при выполнении задания пользователя.
3. Режимы работы компьютеров. Однопрограммный режим. Многопрограммный режим.
4. Особенности управления основной памятью ЭВМ.
5. Система прерываний ЭВМ.
6. Адресация регистров и ячеек памяти в ПК.
7. Параллельные вычисления.

Литература для самостоятельного изучения к Разделу 3

Основная литература

1. Замятина, О. М. Вычислительные системы, сети и телекоммуникации. Моделирование сетей : учебное пособие для вузов / О. М. Замятина. — Москва : Издательство Юрайт, 2022. — 159 с. — (Высшее образование). — ISBN 978-5-534-00335-2. — URL : <https://urait.ru/bcode/490257>

2. Сети и телекоммуникации : учебник и практикум для вузов / К. Е. Самуйлов [и др.] ; под редакцией К. Е. Самуйлова, И. А. Шалимова, Д. С. Кулябова. — Москва : Издательство Юрайт, 2023. — 363 с. — (Высшее образование). — ISBN 978-5-534-00949-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511092> (дата обращения: 08.03.2023).

Дополнительная литература

1. Новожилов, О. П. Архитектура ЭВМ и систем в 2 ч. Часть 1 : учебное пособие для вузов / О. П. Новожилов. — Москва : Издательство Юрайт, 2023. — 276 с. — (Высшее образование). — ISBN 978-5-534-07717-9. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/516640> (дата обращения: 08.03.2023).

2. Новожилов, О. П. Архитектура ЭВМ и систем в 2 ч. Часть 2 : учебное пособие для вузов / О. П. Новожилов. — Москва : Издательство Юрайт, 2023. — 246 с. — (Высшее образование). — ISBN 978-5-534-07718-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/516641> (дата обращения: 08.03.2023).

3. Толстобров, А. П. Архитектура ЭВМ : учебное пособие для вузов / А. П. Толстобров. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 163 с. — (Высшее образование). — ISBN 978-5-534-16839-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/531870> (дата обращения: 24.05.2023).

Задания для самостоятельной работы к Разделу 4

Вопросы для самостоятельной работы к Разделу 4

1. Классификация и архитектура информационно-вычислительных сетей.
2. Виды информационно-вычислительных сетей. Топология сети.
3. Сетевые операционные системы.
4. Технологии передачи данных. Коммутация каналов. Коммутация пакетов.
5. Техническое обеспечение информационно-вычислительных сетей.
6. Серверы и рабочие станции.
7. Повторители. Мосты.
8. Коммутаторы.
9. Маршрутизаторы, протоколы маршрутизации и шлюзы.

Литература для самостоятельного изучения к Разделу 4

Основная литература

1. Замятина, О. М. Вычислительные системы, сети и телекоммуникации. Моделирование сетей : учебное пособие для вузов / О. М. Замятина. — Москва : Издательство Юрайт, 2022. — 159 с. — (Высшее образование). — ISBN 978-5-534-00335-2. — URL : <https://urait.ru/bcode/490257>
2. Сети и телекоммуникации : учебник и практикум для вузов / К. Е. Самуйлов [и др.] ; под редакцией К. Е. Самуйлова, И. А. Шалимова, Д. С. Кулябова. — Москва : Издательство Юрайт, 2023. — 363 с. — (Высшее образование). — ISBN 978-5-534-00949-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511092> (дата обращения: 08.03.2023).

Дополнительная литература

1. Новожилов, О. П. Архитектура ЭВМ и систем в 2 ч. Часть 1 : учебное пособие для вузов / О. П. Новожилов. — Москва : Издательство Юрайт, 2023. — 276 с. — (Высшее образование). — ISBN 978-5-534-07717-9. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/516640> (дата обращения: 08.03.2023).
2. Новожилов, О. П. Архитектура ЭВМ и систем в 2 ч. Часть 2 : учебное пособие для вузов / О. П. Новожилов. — Москва : Издательство Юрайт, 2023. — 246 с. — (Высшее образование). — ISBN 978-5-534-07718-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/516641> (дата обращения: 08.03.2023).
3. Толстобров, А. П. Архитектура ЭВМ : учебное пособие для вузов / А. П. Толстобров. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 163 с. — (Высшее образование). — ISBN 978-5-534-16839-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/531870> (дата обращения: 24.05.2023).

Задания для самостоятельной работы к Разделу 5

Вопросы для самостоятельной работы к Разделу 5

1. Модель взаимодействия открытых систем.
2. Физический уровень OSI.
3. Канальный уровень OSI. Подуровень MAC. Подуровень LLC.
4. Основные понятия TCP/IP. Сравнение стека протоколов TCP/IP с моделью OSI.
5. Протоколы ARP и RARP.
6. Протоколы IP, UDP и TCP.
7. Классы IP-адресов.
8. Протокол DHCP и маршрутизация.
9. Функционирование службы DNS.

10. WINS и разрешение имен NetBIOS.
11. Развитие протокола TCP/IP. Протокол IPv6.

Литература для самостоятельного изучения к Разделу 5

Основная литература

1. Замятина, О. М. Вычислительные системы, сети и телекоммуникации. Моделирование сетей : учебное пособие для вузов / О. М. Замятина. — Москва : Издательство Юрайт, 2022. — 159 с. — (Высшее образование). — ISBN 978-5-534-00335-2. — URL : <https://urait.ru/bcode/490257>
2. Сети и телекоммуникации : учебник и практикум для вузов / К. Е. Самуйлов [и др.] ; под редакцией К. Е. Самуйлова, И. А. Шалимова, Д. С. Кулябова. — Москва : Издательство Юрайт, 2023. — 363 с. — (Высшее образование). — ISBN 978-5-534-00949-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511092> (дата обращения: 08.03.2023).

Дополнительная литература

1. Новожилов, О. П. Архитектура ЭВМ и систем в 2 ч. Часть 1 : учебное пособие для вузов / О. П. Новожилов. — Москва : Издательство Юрайт, 2023. — 276 с. — (Высшее образование). — ISBN 978-5-534-07717-9. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/516640> (дата обращения: 08.03.2023).
2. Новожилов, О. П. Архитектура ЭВМ и систем в 2 ч. Часть 2 : учебное пособие для вузов / О. П. Новожилов. — Москва : Издательство Юрайт, 2023. — 246 с. — (Высшее образование). — ISBN 978-5-534-07718-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/516641> (дата обращения: 08.03.2023).
3. Толстобров, А. П. Архитектура ЭВМ : учебное пособие для вузов / А. П. Толстобров. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 163 с. — (Высшее образование). — ISBN 978-5-534-16839-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/531870> (дата обращения: 24.05.2023).

Вопросы для самостоятельной работы к Разделу 6

1. Технология беспроводных сетей. Беспроводные среды передачи.
2. Требования к беспроводным сетям. Типы беспроводных сетей.
3. Беспроводные персональные сети. Беспроводные локальные сети.
4. Беспроводные городские сети. Беспроводные глобальные сети.

Литература для самостоятельного изучения к Разделу 6

Основная литература

1. Замятина, О. М. Вычислительные системы, сети и телекоммуникации. Моделирование сетей : учебное пособие для вузов / О. М. Замятина. — Москва : Издательство Юрайт, 2022. — 159 с. — (Высшее образование). — ISBN 978-5-534-00335-2. — URL : <https://urait.ru/bcode/490257>
2. Сети и телекоммуникации : учебник и практикум для вузов / К. Е. Самуйлов [и др.] ; под редакцией К. Е. Самуйлова, И. А. Шалимова, Д. С. Кулябова. — Москва : Издательство Юрайт, 2023. — 363 с. — (Высшее образование). — ISBN 978-5-534-00949-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511092> (дата обращения: 08.03.2023).

Дополнительная литература

1. Новожилов, О. П. Архитектура ЭВМ и систем в 2 ч. Часть 1 : учебное пособие для вузов / О. П. Новожилов. — Москва : Издательство Юрайт, 2023. — 276 с. — (Высшее образование). — ISBN 978-5-534-07717-9. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/516640> (дата обращения: 08.03.2023).

2. Новожилов, О. П. Архитектура ЭВМ и систем в 2 ч. Часть 2 : учебное пособие для вузов / О. П. Новожилов. — Москва : Издательство Юрайт, 2023. — 246 с. — (Высшее образование). — ISBN 978-5-534-07718-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/516641> (дата обращения: 08.03.2023).

3. Толстобров, А. П. Архитектура ЭВМ : учебное пособие для вузов / А. П. Толстобров. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 163 с. — (Высшее образование). — ISBN 978-5-534-16839-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/531870> (дата обращения: 24.05.2023).

Задания для самостоятельной работы к Разделу 7

Вопросы для самостоятельной работы к Разделу 7

1. Локальные вычислительные сети. Виды локальных вычислительных сетей.
2. Одноранговые локальные сети. Серверные локальные сети.
3. Сети, использующие в качестве каналов связи линии электропитания.
4. Структурированная кабельная система и логическая структуризация ЛКС.
5. Устройства межсетевого интерфейса.
6. Организация и функционирование виртуальных локальных компьютерных сетей.
7. Корпоративные информационные системы. Особенности архитектуры корпоративных компьютерных сетей.
8. Программное и информационное обеспечение сетей.

Литература для самостоятельного изучения к Разделу 7

Основная литература

1. Замятина, О. М. Вычислительные системы, сети и телекоммуникации. Моделирование сетей : учебное пособие для вузов / О. М. Замятина. — Москва : Издательство Юрайт, 2022. — 159 с. — (Высшее образование). — ISBN 978-5-534-00335-2. — URL : <https://urait.ru/bcode/490257>

2. Сети и телекоммуникации : учебник и практикум для вузов / К. Е. Самуйлов [и др.] ; под редакцией К. Е. Самуйлова, И. А. Шалимова, Д. С. Кулябова. — Москва : Издательство Юрайт, 2023. — 363 с. — (Высшее образование). — ISBN 978-5-534-00949-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511092> (дата обращения: 08.03.2023).

Дополнительная литература

1. Новожилов, О. П. Архитектура ЭВМ и систем в 2 ч. Часть 1 : учебное пособие для вузов / О. П. Новожилов. — Москва : Издательство Юрайт, 2023. — 276 с. — (Высшее образование). — ISBN 978-5-534-07717-9. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/516640> (дата обращения: 08.03.2023).

2. Новожилов, О. П. Архитектура ЭВМ и систем в 2 ч. Часть 2 : учебное пособие для вузов / О. П. Новожилов. — Москва : Издательство Юрайт, 2023. — 246 с. — (Высшее образование). — ISBN 978-5-534-07718-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/516641> (дата обращения: 08.03.2023).

3. Толстобров, А. П. Архитектура ЭВМ : учебное пособие для вузов / А. П. Толстобров. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 163 с. — (Высшее образование). — ISBN 978-5-534-16839-6. — Текст : электронный //

Здания для самостоятельной работы к Разделу 8

Вопросы для самостоятельной работы к Разделу 8

1. **Глобальная информационная сеть Интернет. Протоколы общения компьютеров в сети.**
2. **Система адресации в Интернете.**
3. **Подключение компьютера для работы в Интернете.**
4. **Базовые пользовательские технологии работы в Интернете.**
5. **Передача файлов с помощью протокола FTP. Telnet.**
6. **Электронная почта. Основные технологии работы в WWW.**
7. **Гипертекстовые технологии Интернета. Обзорщики Интернета и поисковые системы.**

Литература для самостоятельного изучения к Разделу 8

Основная литература

1. Замятина, О. М. Вычислительные системы, сети и телекоммуникации. Моделирование сетей : учебное пособие для вузов / О. М. Замятина. — Москва : Издательство Юрайт, 2022. — 159 с. — (Высшее образование). — ISBN 978-5-534-00335-2. — URL : <https://urait.ru/bcode/490257>
2. Сети и телекоммуникации : учебник и практикум для вузов / К. Е. Самуйлов [и др.] ; под редакцией К. Е. Самуйлова, И. А. Шалимова, Д. С. Кулябова. — Москва : Издательство Юрайт, 2023. — 363 с. — (Высшее образование). — ISBN 978-5-534-00949-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511092> (дата обращения: 08.03.2023).

Дополнительная литература

1. Новожилов, О. П. Архитектура ЭВМ и систем в 2 ч. Часть 1 : учебное пособие для вузов / О. П. Новожилов. — Москва : Издательство Юрайт, 2023. — 276 с. — (Высшее образование). — ISBN 978-5-534-07717-9. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/516640> (дата обращения: 08.03.2023).
2. Новожилов, О. П. Архитектура ЭВМ и систем в 2 ч. Часть 2 : учебное пособие для вузов / О. П. Новожилов. — Москва : Издательство Юрайт, 2023. — 246 с. — (Высшее образование). — ISBN 978-5-534-07718-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/516641> (дата обращения: 08.03.2023).
3. Толстобров, А. П. Архитектура ЭВМ : учебное пособие для вузов / А. П. Толстобров. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 163 с. — (Высшее образование). — ISBN 978-5-534-16839-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/531870> (дата обращения: 24.05.2023).

Задания для самостоятельной работы к Разделу 9

Вопросы для самостоятельной работы к Разделу 9

1. Системы телекоммуникаций.
2. Системы передачи данных и их характеристики.
3. Линии и каналы связи. Цифровые каналы связи.
4. Телефонная связь. Офисные АТС. Разновидности офисных АТС.
5. Системы сотовой связи. Стандарты сотовой связи. Сервисы сотовой связи.
6. Спутниковые навигационные системы GPS.

Литература для самостоятельного изучения к Разделу 9

Основная литература

1. Замятина, О. М. Вычислительные системы, сети и телекоммуникации. Моделирование сетей : учебное пособие для вузов / О. М. Замятина. — Москва : Издательство Юрайт, 2022. — 159 с. — (Высшее образование). — ISBN 978-5-534-00335-2. — URL : <https://urait.ru/bcode/490257>
2. Сети и телекоммуникации : учебник и практикум для вузов / К. Е. Самуйлов [и др.] ; под редакцией К. Е. Самуйлова, И. А. Шалимова, Д. С. Кулябова. — Москва : Издательство Юрайт, 2023. — 363 с. — (Высшее образование). — ISBN 978-5-534-00949-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511092> (дата обращения: 08.03.2023).

Дополнительная литература

1. Новожилов, О. П. Архитектура ЭВМ и систем в 2 ч. Часть 1 : учебное пособие для вузов / О. П. Новожилов. — Москва : Издательство Юрайт, 2023. — 276 с. — (Высшее образование). — ISBN 978-5-534-07717-9. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/516640> (дата обращения: 08.03.2023).
2. Новожилов, О. П. Архитектура ЭВМ и систем в 2 ч. Часть 2 : учебное пособие для вузов / О. П. Новожилов. — Москва : Издательство Юрайт, 2023. — 246 с. — (Высшее образование). — ISBN 978-5-534-07718-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/516641> (дата обращения: 08.03.2023).
3. Толстобров, А. П. Архитектура ЭВМ : учебное пособие для вузов / А. П. Толстобров. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 163 с. — (Высшее образование). — ISBN 978-5-534-16839-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/531870> (дата обращения: 24.05.2023).

2. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ) И САМОСТОЯТЕЛЬНОЙ РАБОТЕ

Освоение обучающимся дисциплины (модуля) «Технологии возможностей и безбарьерной среды» предполагает изучение материалов дисциплины на аудиторных занятиях и в ходе самостоятельной работы. Аудиторные занятия проходят в форме лекций, семинаров, практических и лабораторных занятий. Самостоятельная работа включает разнообразный комплекс видов и форм работы обучающихся.

Для успешного освоения дисциплины (модуля) и достижения поставленных целей необходимо внимательно ознакомиться с рабочей программы дисциплины (модуля), доступной в электронной информационно-образовательной среде РГСУ.

Следует обратить внимание на списки основной и дополнительной литературы, на предлагаемые преподавателем ресурсы информационно-телекоммуникационной сети Интернет. Эта информация необходима для самостоятельной работы обучающегося.

При подготовке к аудиторным занятиям необходимо помнить особенности каждой формы его проведения.

Подготовка к учебному занятию лекционного типа заключается в следующем.

С целью обеспечения успешного обучения обучающийся должен готовиться к лекции, поскольку она является важнейшей формой организации учебного процесса, поскольку:

- знакомит с новым учебным материалом;
- разъясняет учебные элементы, трудные для понимания;
- систематизирует учебный материал;
- ориентирует в учебном процессе.

С этой целью:

- внимательно прочитайте материал предыдущей лекции;

- ознакомьтесь с учебным материалом по учебнику и учебным пособиям с темой прочитанной лекции;
- внесите дополнения к полученным ранее знаниям по теме лекции на полях лекционной тетради;
- запишите возможные вопросы, которые вы зададите лектору на лекции по материалу изученной лекции;
- постарайтесь уяснить место изучаемой темы в своей подготовке;
- узнайте тему предстоящей лекции (по тематическому плану, по информации лектора) и запишите информацию, которой вы владеете по данному вопросу.

Подготовка к занятию семинарского типа.

При подготовке и работе во время проведения занятий семинарского типа следует обратить внимание на следующие моменты: на процесс предварительной подготовки, на работу во время занятия, обработку полученных результатов, исправление полученных замечаний.

Предварительная подготовка к учебному занятию семинарского типа заключается в изучении теоретического материала в отведенное для самостоятельной работы время, ознакомление с инструктивными материалами с целью осознания задач практического занятия, техники безопасности при работе с приборами, веществами.

Работа во время проведения учебного занятия семинарского типа включает:

- консультирование студентов преподавателями и вспомогательным персоналом с целью предоставления исчерпывающей информации, необходимой для самостоятельного выполнения предложенных преподавателем задач.
- самостоятельное выполнение заданий согласно обозначенной рабочей программой дисциплины (модуля) тематики.

Самостоятельная работа.

Самостоятельная работа - планируемая учебная, учебно-исследовательская, научно-исследовательская работа студентов, выполняемая во внеаудиторное (аудиторное) время по заданию и при методическом руководстве преподавателя, но без его непосредственного участия (при частичном непосредственном участии преподавателя, оставляющем ведущую роль за работой студентов).

Для более углубленного изучения темы задания для самостоятельной работы рекомендуется выполнять параллельно с изучением данной темы. При выполнении заданий по возможности используйте наглядное представление материала.

Самостоятельная работа студентов в ВУЗе является важным видом учебной и научной деятельности студента. Самостоятельная работа студентов играет значительную роль в рейтинговой технологии обучения.

К современному специалисту в области медицины общество предъявляет достаточно широкий перечень требований, среди которых немаловажное значение имеет наличие у выпускников определенных навыков (компетенций) и умения самостоятельно добывать знания из различных источников, систематизировать полученную информацию, давать оценку конкретной профессиональной ситуации. Формирование такого умения происходит в течение всего периода обучения.

В процессе самостоятельной работы студент приобретает необходимые для будущей специальности компетенции, навыки самоорганизации, самоконтроля, самоуправления, саморефлексии и становится активным самостоятельным субъектом учебной деятельности.

Самостоятельная работа студентов должна оказывать важное влияние на формирование личности будущего специалиста, его компетентность. Каждый студент самостоятельно определяет режим своей работы и меру труда, затрачиваемого на овладение учебным содержанием по каждой дисциплине. Он выполняет внеаудиторную работу по личному индивидуальному плану, в зависимости от его подготовки, времени и других условий.

Виды самостоятельной работы.

Работа с литературой.

При работе с книгой необходимо подобрать литературу, научиться правильно ее читать, вести записи. Для подбора литературы в библиотеке используются алфавитный и систематический каталоги. Важно помнить, что рациональные навыки работы с книгой - это всегда большая экономия времени и сил. Правильный подбор учебников рекомендуется преподавателем, читающим лекционный курс. Необходимая литература может быть также указана в методических разработках по данному курсу. Изучая материал по учебнику, следует переходить к следующему вопросу только после правильного уяснения предыдущего, описывая на бумаге все выкладки и вычисления (в том числе те, которые в учебнике опущены или на лекции даны для самостоятельного вывода). При изучении любой дисциплины большую и важную роль играет самостоятельная индивидуальная работа. Особое внимание следует обратить на определение основных понятий курса. Студент должен подробно разбирать примеры, которые поясняют такие определения, и уметь строить аналогичные примеры самостоятельно. Нужно добиваться точного представления о том, что изучаешь. Полезно составлять опорные конспекты. При изучении материала по учебнику полезно в тетради (на специально отведенных полях) дополнять конспект лекций. Там же следует отмечать вопросы, выделенные студентом для консультации с преподавателем. Выводы, полученные в результате изучения, рекомендуется в конспекте выделять, чтобы они при перечитывании записей лучше запоминались. Опыт показывает, что многим студентам помогает составление листа опорных сигналов, содержащего важнейшие и наиболее часто употребляемые формулы и понятия. Такой лист помогает запомнить формулы, основные положения лекции, а также может служить постоянным справочником для студента. Различают два вида чтения: первичное и вторичное. Первичное - это внимательное, неторопливое чтение, при котором можно остановиться на трудных местах.

Задача вторичного чтения - полное усвоение смысла целого (по счету это чтение может быть и не вторым, а третьим или четвертым). Самостоятельная работа с учебниками и книгами (а также самостоятельное теоретическое исследование проблем, обозначенных преподавателем на лекциях) - это важнейшее условие формирования у себя научного способа познания.

Чтение научного текста является частью познавательной деятельности. Ее цель - извлечение из текста необходимой информации. От того, насколько осознанно читающим собственная внутренняя установка при обращении к печатному слову (найти нужные сведения, усвоить информацию полностью или частично, критически проанализировать материал и т.п.) во многом зависит эффективность осуществляемого действия.

Методические рекомендации по составлению конспекта:

1. Внимательно прочитайте текст. Уточните в справочной литературе непонятные слова. При записи не забудьте вынести справочные данные на поля конспекта;
2. Выделите главное, составьте план;
3. Кратко сформулируйте основные положения текста, отметьте аргументацию автора;
4. Законспектируйте материал, четко следуя пунктам плана. При конспектировании старайтесь выразить мысль своими словами. Записи следует вести четко, ясно.
5. Грамотно записывайте цитаты. Цитируя, учитывайте лаконичность, значимость мысли. В тексте конспекта желательно приводить не только тезисные положения, но и их доказательства. При оформлении конспекта необходимо стремиться к емкости каждого предложения. Мысли автора книги следует излагать кратко, заботясь о стиле и выразительности написанного. Число дополнительных элементов конспекта должно быть логически обоснованным, записи должны распределяться в определенной последовательности, отвечающей логической структуре произведения. Для уточнения и дополнения необходимо оставлять поля. Овладение навыками конспектирования требует от студента целеустремленности, повседневной самостоятельной работы.

Методические материалы по самостоятельному решению задач

При самостоятельном решении задач нужно обосновывать каждый этап решения, исходя из теоретических положений курса. Если студент видит несколько путей решения

проблемы (задачи), то нужно сравнить их и выбрать самый рациональный. Полезно до начала вычислений составить краткий план решения проблемы (задачи). Решение проблемных задач или примеров следует излагать подробно, вычисления располагать в строгом порядке, отделяя вспомогательные вычисления от основных. Решения при необходимости нужно сопровождать комментариями, схемами, чертежами и рисунками. Следует помнить, что решение каждой учебной задачи должно доводиться до окончательного логического ответа, которого требует условие, и по возможности с выводом. Полученный ответ следует проверить способами, вытекающими из существа данной задачи. Полезно также (если возможно) решать несколькими способами и сравнить полученные результаты. Решение задач данного типа нужно продолжать до приобретения твердых навыков в их решении.

Методические материалы к выполнению реферата

Реферат (от лат. *refere* – сообщать) – краткое изложение в письменном виде или в форме публикации доклада, содержания научного труда (трудов), литературы по теме. Работа над рефератом условно разделяется на выбор темы, подбор литературы, подготовку и защиту плана; написание теоретической части и всего текста с указанием библиографических данных используемых источников, подготовку доклада, выступление с ним. Тематика рефератов полностью связана с основными вопросами изучаемого курса.

Список литературы к темам не дается, и обучающиеся самостоятельно ведут библиографический поиск, причем им не рекомендуется ограничиваться университетской библиотекой.

Важно учитывать, что написание реферата требует от обучающихся определенных усилий. Особое внимание следует уделить подбору литературы, методике ее изучения с целью отбора и обработки собранного материала, обоснованию актуальности темы и теоретического уровня обоснованности используемых в качестве примеров фактов какой-либо деятельности.

Выбрав тему реферата, начав работу над литературой, необходимо составить план. Изучая литературу, продолжается обдумывание темы, осмысливание прочитанного, делаются выписки, сопоставляются точки зрения разных авторов и т.д. Реферативная работа сводится к тому, чтобы в ней выделились две взаимосвязанные стороны: во-первых, ее следует рассматривать как учебное задание, которое должен выполнить обучаемый, а во-вторых, как форму научной работы, творческого воображения при выполнении учебного задания.

Наличие плана реферата позволяет контролировать ход работы, избежать формального переписывания текстов из первоисточников.

Оформление реферата включает титульный лист, оглавление и краткий список использованной литературы. Список использованной литературы размещается на последней странице рукописи или печатной форме реферата. Реферат выполняется в письменной или печатной форме на белых листах формата А4 (210 x 297 мм). Шрифт Times New Roman, кегель 14, через 1,5 интервала при соблюдении следующих размеров текста: верхнее поле – 25 мм, нижнее – 20 мм, левое – 30 мм, правое – 15 мм. Нумерация страниц производится сверху листа, по центру. Титульный лист нумерации не подлежит.

Рефераты должны быть написаны простым, ясным языком, без претензий на наукообразность. Следует избегать сложных грамматических оборотов, непривычных терминов и символов. Если же такие термины и символы все-таки приводятся, то необходимо разъяснять их значение при первом упоминании в тексте реферата.

Объем реферата предполагает тщательный отбор информации, необходимой для краткого изложения вопроса. Важнейший этап – редактирование готового текста реферата и подготовка к обсуждению. Обсуждение требует хорошей ориентации в материале темы, умения выделить главное, поставить дискуссионный вопрос, привлечь внимание слушателей к интересной литературе, логично и убедительно изложить свои мысли.

Рефераты обязательно подлежат защите. Процедура защиты начинается с определения оппонентов защищающего свою работу. Они стремятся дать основательный анализ работы обучающимся, обращают внимание на положительные моменты и недостатки реферата, дают общую оценку содержанию, форме преподнесения материала, характеру использованной

литературы. Иногда они дополняют тот или иной раздел реферата. Последнее особенно ценно, ибо говорит о глубоком знании обучающимся-оппонентом изучаемой проблемы.

Обсуждение не ограничивается выслушиванием оппонентов. Другие обучающиеся имеют право уточнить или опровергнуть какое-либо утверждение. Преподаватель предлагает любому обучающемуся задать вопрос по существу доклада или попытаться подвести итог обсуждению.

Алгоритм работы над рефератом

1. Выбор темы

Тема должна быть сформулирована грамотно (с литературной точки зрения);

В названии реферата следует поставить четкие рамки рассмотрения темы;

Желательно избегать слишком длинных названий;

Следует по возможности воздерживаться от использования в названии спорных с научной точки зрения терминов, излишней наукообразности, а также чрезмерного упрощения формулировок.

2. Реферат следует составлять из пяти основных частей: введения; основной части; заключения; списка литературы; приложений.

3. Основные требования к введению:

Во введении не следует концентрироваться на содержании; введение должно включать краткое обоснование актуальности темы реферата, где требуется показать, почему данный вопрос может представлять научный интерес и есть ли связь представляемого материала с современностью. Таким образом, тема реферата должна быть актуальна либо с научной точки зрения, либо с современных позиций.

Очень важно выделить цель, а также задачи, которые требуется решить для выполнения цели.

Введение должно содержать краткий обзор изученной литературы, в котором указывается взятый из того или иного источника материал, кратко анализируются изученные источники, показываются их сильные и слабые стороны;

Объем введения составляет две страницы текста.

4. Требования к основной части реферата:

Основная часть содержит материал, отобранный для рассмотрения проблемы;

Также основная часть должна включать в себя собственно мнение обучающихся и сформулированные самостоятельные выводы, опирающиеся на приведенные факты;

Материал, представленный в основной части, должен быть логически изложен и распределен по параграфам, имеющим свои названия;

В изложении основной части необходимо использовать сноски (в первую очередь, когда приводятся цифры и чьи-то цитаты);

Основная часть должна содержать иллюстративный материал (графики, таблицы и т. д.);

Объем основной части составляет около 10 страниц.

5. Требования к заключению:

В заключении формулируются выводы по параграфам, обращается внимание на выдвинутые во введении задачи и цели;

Заключение должно быть четким, кратким, вытекающим из содержания основной части.

6. Требования к оформлению списка литературы (по ГОСТу):

Необходимо соблюдать правильность последовательности записи источников: сначала следует писать фамилию, а после инициалы; название работы не ставится в кавычки; после названия сокращенно пишется место издания; затем идет год издания; наконец, называется процитированная страница.

Критерии оценки реферата

Обучающийся, защищающий реферат, должен рассказать о его актуальности, поставленных целях и задачах, изученной литературе, структуре основной части, сделанных в ходе работы выводах.

По окончании выступления ему может быть задано несколько вопросов по представленной проблеме.

Оценка складывается из соблюдения требований к реферату, грамотного раскрытия темы, умения четко рассказывать о представленном реферате, способности понять суть задаваемых по работе вопросов и найти точные ответы на них.

Методические материалы к выполнению эссе

Эссе – литературное произведение небольшого объема, обычно прозаическое, свободной композиции, передающее индивидуальные впечатления, суждения, соображения автора о той или иной проблеме, теме, о том или ином событии или явлении. Это вид самостоятельной исследовательской работы обучающихся, с целью углубления и закрепления теоретических знаний и освоения практических навыков. Цель эссе состоит в развитии самостоятельного творческого мышления и письменного изложения собственных мыслей. При написании эссе обучающийся должен представить развернутый письменный ответ на теоретический или практический актуальный вопрос, объявленный преподавателем в аудитории непосредственно перед ее написанием. В процессе написания эссе разрешается пользоваться нормативно-правовыми актами, конспектом лекций (в печатном виде). Использование интернет-ресурсов не допускается. Темы эссе преподаватель предлагает из числа тех, которые обучающиеся уже рассматривали на лекциях или семинарских занятиях, исходя из содержания заданий в составе оценочных средств. По решению преподавателя, в качестве темы эссе может быть выбрана одна или несколько тем, которые могут быть распределены между обучающимися по желанию.

Требования к выполнению эссе:

1. Проводится письменно.
2. Эссе выполняется на компьютере (гарнитура Times New Roman, шрифт 14) через 1,5 интервала с полями: верхнее, нижнее – 2; правое – 3; левое – 1,5. Отступ первой строки абзаца – 1,25. Сноски – постраничные. Таблицы и рисунки встраиваются в текст работы. При этом обязательный заголовок таблицы надо размещать над табличным полем, а рисунки сопровождать подрисовочными подписями. При включении в эссе нескольких таблиц и/или рисунков их нумерация обязательна. Обязательна и нумерация страниц. Их целесообразно проставлять внизу страницы – по середине или в правом углу. Номер страницы не ставится на титульном листе, но в общее число страниц он включается. Объем эссе, без учета приложений, не должен превышать 5 страниц. Значительное превышение установленного объема является недостатком работы и указывает на то, что обучающийся не сумел отобрать и переработать необходимый материал.

3. Работа должна содержать собственные умозаключения по сути поставленной проблемы, включать самостоятельно проведенный анализ по сути этой проблемы, выводы, обобщающие авторскую позицию по поставленной проблеме.

Критерии оценки эссе:

«Отлично» – исключительные знания материала, абсолютное понимание сути, безукоризненное знание основных понятий и положений, логически и лексически грамотно изложенный, содержательный, аргументированный, конкретный и исчерпывающий ответ.

«Хорошо» – глубокие знания материала, правильное понимание сути, знание основных понятий и положений, содержательный, полный и конкретный ответ.

«Удовлетворительно» – твердые, но недостаточно полные знания, верное понимание сути, в целом правильный ответ.

«Неудовлетворительно» – непонимание сущности задания, грубые ошибки в ответе.

Методические материалы по выполнению тестирования.

Тестовые задания содержат вопросы и 3-4 варианта ответа по базовым положениям изучаемой темы/раздела, составлены с расчетом на знания, полученные обучающимся в процессе изучения темы/раздела.

Тестовые задания выполняются в письменной форме и сдаются преподавателю, ведущему дисциплину (модуль). На выполнение тестовых заданий обучающимся отводится 45 минут.

При обработке результатов оценочной процедуры используются: критерии оценки по содержанию и качеству полученных ответов, ключи, оценочные листы.

Критерии оценки теста:

«Зачтено» - если обучающийся правильно выполнил не менее 2/3 всей работы или допустил не более одной грубой ошибки и двух недочетов, не более одной грубой и одной негрубой ошибки, не более трех негрубых ошибок, одной негрубой ошибки и трех недочетов, при наличии четырех-пяти недочетов.

«Не зачтено» - если число ошибок и недочетов превысило норму для оценки 3 или правильно выполнено менее 2/3 всей работы.

Методические материалы по выполнению доклада.

Рекомендуется следующая структура доклада:

1. титульный лист, содержание доклада;
2. краткое изложение;
3. цели и задачи;
4. изложение характера исследований и рассмотренных проблем, гипотезы, спорные вопросы;
5. источники информации, методы сбора и анализа данных, степень их полноты и достоверности;
6. анализ и толкование полученных в работе результатов;
7. выводы и оценки;
8. библиография и приложения.

Время выступления докладчика не должно превышать 10 минут.

Основные требования к оформлению доклада:

- титульный лист должен включать название доклада, наименование предметной (цикловой) комиссии, фамилию обучающегося;
- все использованные литературные источники сопровождаются библиографическим описанием;
- приводимая цитата из источника берется в кавычки (оформляются сноски);
- единицы измерения должны применяться в соответствии с действующими стандартами;
- все названия литературных источников следует приводить в соответствии с новейшими изданиями;
- рекомендуется включение таблиц, графиков, схем, если они отражают основное содержание или улучшают ее наглядность;
- названия фирм, учреждений, организаций и предприятий должны именоваться так, как они указываются в источнике;

Критерии оценки доклада

При выполнении доклада обучающийся должен продемонстрировать умение кратко излагать прочитанный материал, а также умение обобщать и анализировать материал по теме доклада.

Презентация

Методические материалы к презентациям

1. Объем презентации 10 -20 слайдов.
2. На титульном слайде должно быть отражено:
 - наименование факультета;
 - тема презентации;

- фамилия, имя, отчество, направление подготовки/ специальность, направленность (профиль)/ специализация, форма обучения, номер группы автора презентации;
- фамилия, имя, отчество, степень, звание, должность руководитель работы;
- год выполнения работы.

3. В презентации должны быть отражены обоснование актуальности представляемого материала, цели и задачи работы.

4. Содержание презентации должно включать наиболее значимый материал доклада, а также, при необходимости, таблицы, диаграммы, рисунки, фотографии, карты, видео – вставки, звуковое сопровождение.

5. Заключительный слайд должен содержать информацию об источниках информации для презентации.

Критерии оценки презентации

1. Объём презентации 10 -20 слайдов.
2. Правильность оформления титульного слайда.
3. Актуальность отобранного материала, обоснованность формулировки цели и задач работы.

4. Наглядность и логичность презентации, обоснованность использования таблиц, диаграмм, рисунков, фотографий, карт, видео – вставок, звукового сопровождения; правильный выбор шрифтов, фона, других элементов дизайна слайда.

5. Объём и качество источников информации (не менее 2-х интернет – источников и не менее 2-х литературных источников).

Методические материалы по подготовке к опросу

Самостоятельная работа обучающихся включает подготовку к опросу на практических занятиях. Для этого обучающийся изучает лекции, основную и дополнительную литературу, публикации, информацию из Интернет-ресурсов.

Тема и вопросы к практическим занятиям, вопросы для самоконтроля содержатся в рабочей программе и доводятся до студентов заранее.

Для подготовки к опросу обучающемуся необходимо ознакомиться с материалом, посвященным теме практического занятия, в учебнике или другой рекомендованной литературе, конспекте лекции, обратить внимание на усвоение основных понятий дисциплины, выявить неясные вопросы и подобрать дополнительную литературу для их освещения.

Критерии оценки опроса

«Отлично»:

- дан полный, развернутый ответ на поставленный вопрос;
- в ответе прослеживается четкая структура, логическая последовательность, отражающая сущность раскрываемых понятий, теорий, явлений;
- знание по предмету демонстрируются на фоне понимания его в системе данной науки и междисциплинарных связей;
- свободное владение терминологией;
- ответы на дополнительные вопросы четкие, краткие;

«Хорошо»:

- дан полный, развернутый ответ на поставленный вопрос, показано умение выделять существенные и несущественные признаки, причинно-следственные связи;
- ответ недостаточно логичен с единичными ошибками в частностях, исправленные студентом с помощью преподавателя;
- единичные ошибки в терминологии;
- ответы на дополнительные вопросы правильные, недостаточно полные и четкие.

«Удовлетворительно»:

- ответ не полный, с ошибками в деталях, умение раскрыть значение обобщённых знаний не показано, речевое оформление требует поправок, коррекции;

– логика и последовательность изложения имеют нарушения, студент не способен самостоятельно выделить существенные и несущественные признаки и причинно-следственные связи;

- ошибки в раскрываемых понятиях, терминах;
- студент не ориентируется в теме, допускает серьезные ошибки;
- студент не может ответить на большую часть дополнительных вопросов.

«Неудовлетворительно»:

– ответ представляет собой разрозненные знания с существенными ошибками по вопросу;

– присутствуют фрагментарность, нелогичность изложения, студент не осознает связь обсуждаемого вопроса с другими объектами дисциплины, речь неграмотная;

- незнание терминологии;
- ответы на дополнительные вопросы неправильные.

Методические материалы по выполнению практического задания

При выполнении практического задания обучающийся придерживается следующего алгоритма:

1. Записать дату, тему и цель задания;
2. Ознакомиться с правилами и условия выполнения практического задания;
3. Повторить теоретические задания, необходимые для рациональной работы и других практических действий, используя конспекты лекций и рекомендованную литературу, представленной в программе;
4. Выполнить работу по предложенному алгоритму действий;
5. Обобщить результаты работы, сформулировать выводы / дать ответы на контрольные вопросы;

Работа должна быть выполнена грамотно, с соблюдением культуры изложения. При использовании данных из учебных, методических пособий и другой литературы, периодических изданий, Интернет-источников должны иметься ссылки на вышеперечисленные.

Критерии оценки практического задания:

«Отлично» – правильный ответ, дается четкое обоснование принятому решению; рассуждения четкие последовательные логические; используются ссылки на полученные при изучении дисциплины знания; правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Хорошо» – правильный ответ, дается обоснование принятому решению; но с не существенными ошибками, в рассуждениях отсутствует логическая последовательность; используются ссылки на полученные при изучении дисциплины знания, правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Удовлетворительно» – правильный ответ, допускаются грубые ошибки в обосновании принятого решения; рассуждения не последовательные сумбурные; используются ссылки на полученные при изучении дисциплины знания; используются формулы, процедуры, понятия, имеющие прямое значение для подтверждения принятого решения, однако, при обращении к ним допускаются серьезные ошибки, студент не может правильно ими воспользоваться.

«Неудовлетворительно, не зачтено» – ответ неверный, отсутствует обоснование принятому решению; студент демонстрирует полное непонимание сути вопроса.

Для оценки решения ситуационной задачи (аналитического задания):

Оценка «отлично» выставляется, если задача решена грамотно, ответы на вопросы сформулированы четко. Эталонный ответ полностью соответствует решению студента, которое хорошо обосновано теоретически.

Оценка «хорошо» выставляется, если задача решена, ответы на вопросы сформулированы не достаточно четко. Решение студента в целом соответствует эталонному ответу, но не достаточно хорошо обосновано теоретически.

Оценка «удовлетворительно» выставляется, если задача решена не полностью, ответы не содержат всех необходимых обоснований решения.

Оценка «неудовлетворительно» выставляется, если задача не решена или имеет грубые теоретические ошибки в ответе на поставленные вопросы.

Методические материалы по выполнению лабораторного задания

При выполнении лабораторного задания обучающийся придерживается следующего алгоритма:

1. Записать дату, тему и цель задания;
2. Ознакомиться с правилами и условиями выполнения задания;
3. Повторить теоретические задания, необходимые для рациональной работы и других практических действий, используя конспекты лекций и рекомендованную литературу, представленную в программе;
4. Выполнить работу по предложенному алгоритму действий;
5. Обобщить результаты работы, сформулировать выводы / дать ответы на контрольные вопросы;

Работа должна быть выполнена грамотно, с соблюдением культуры изложения. При использовании данных из учебных, методических пособий и другой литературы, периодических изданий, Интернет-источников должны иметься ссылки на вышеперечисленные.

Критерии оценки лабораторного задания:

«Отлично» – правильный ответ, дается четкое обоснование принятому решению; рассуждения четкие последовательные логические; используются ссылки на полученные при изучении дисциплины знания; правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Хорошо» – правильный ответ, дается обоснование принятому решению; но с не существенными ошибками, в рассуждениях отсутствует логическая последовательность; используются ссылки на полученные при изучении дисциплины знания, правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Удовлетворительно» – правильный ответ, допускаются грубые ошибки в обосновании принятого решения; рассуждения не последовательные сумбурные; используются ссылки на полученные при изучении дисциплины знания; используются формулы, процедуры, понятия, имеющие прямое значение для подтверждения принятого решения, однако, при обращении к ним допускаются серьезные ошибки, студент не может правильно ими воспользоваться.

«Неудовлетворительно, не зачтено» – ответ неверный, отсутствует обоснование принятому решению; студент демонстрирует полное непонимание сути вопроса.

Методические указания для подготовки к промежуточной аттестации.

Изучение учебных дисциплин (модулей) завершается зачетом/зачетом с оценкой или экзаменом. Подготовка к промежуточной аттестации способствует закреплению, углублению и обобщению знаний, получаемых, в процессе обучения, а также применению их к решению практических задач. Готовясь к экзамену, студент ликвидирует имеющиеся пробелы в знаниях, углубляет, систематизирует и упорядочивает свои знания. На зачете или экзамене студент демонстрирует то, что он освоил в процессе обучения по дисциплине (модулю).

Вначале следует просмотреть весь материал по дисциплине (модулю), отметить для себя трудные вопросы. Обязательно в них разобраться. В заключение еще раз целесообразно повторить основные положения, используя при этом листы опорных сигналов. Систематическая подготовка к занятиям в течение семестра позволит использовать время промежуточной аттестации для систематизации знаний.

3. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНИВАНИЯ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ И (ИЛИ) ОПЫТА ДЕЯТЕЛЬНОСТИ, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ

3.1. Организационные основы применения балльно-рейтинговой системы оценки успеваемости обучающихся по дисциплине (модулю)

Оценка качества освоения обучающимися дисциплины (модуля) реализуется в формате балльно-рейтинговой системы оценки успеваемости обучающихся (БРСО).

БРСО в ходе текущего контроля успеваемости и промежуточной аттестации осуществляется по 100-балльной шкале.

Академический рейтинг обучающегося по дисциплине (модулю) складывается из результатов:

- текущего контроля успеваемости (максимальный текущий рейтинг обучающегося 80 рейтинговых баллов);
- промежуточной аттестации (максимальный рубежный рейтинг обучающегося 20 рейтинговых баллов).

Условия оценки освоения обучающимся дисциплины (модуля) в формате БРСО доводятся преподавателем до сведения обучающихся на первом учебном занятии, а также размещены в свободном доступе в электронной информационно-образовательной среде Университета.

3.2. Проведение текущего контроля успеваемости обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося

В течение учебного семестра до промежуточной аттестации на основании утвержденной рабочей программы дисциплины (модуля) формируется текущий рейтинг обучающегося. Текущий рейтинг обучающегося складывается как сумма рейтинговых баллов, полученных им в течение учебного семестра по всем видам учебных занятий по учебной дисциплине.

В процессе текущего контроля оцениваются следующие действия обучающегося, направленные на освоение компетенций в рамках изучения учебной дисциплины:

- академическая активность (посещаемость учебных занятий, самостоятельное изучение содержания учебной дисциплины в электронной информационно-образовательной среде, соблюдение сроков сдачи практических заданий и текущих контрольных мероприятий и др.);
- выполнение и сдача текущих и итогового практических заданий (эссе, рефераты, творческие задания, кейс-задания, лабораторные работы, расчетные задания и др., активное участие в групповых интерактивных занятиях (дискуссии, WiKi-проекты и др.), защита проектов и др.);
- прохождение рубежей текущего контроля, включая соблюдение графика их прохождения в электронной информационно-образовательной среде.

Для планирования расчета текущего рейтинга обучающегося используются следующие пропорции:

Вид учебного действия	Максимальная рейтинговая оценка, баллов
академическая активность	10
практические задания	40
<i>из них: текущие практические задания</i>	20
<i>итоговое практическое задание</i>	20
рубежи текущего контроля	30
<i>ИТОГО:</i>	80

В течение учебного семестра по дисциплине (модулю) обучающимся должен быть накоплен текущий рейтинг не менее 52 рейтинговых баллов (65% от максимального значения текущего рейтинга).

Необходимыми условиями допуска обучающегося к промежуточной аттестации по дисциплине являются положительное прохождение обучающимся не менее 65% рубежей текущего контроля с накоплением не менее 65% максимального рейтингового балла за каждый рубеж текущего контроля и положительное выполнение итогового практического задания с накоплением не менее 65% максимального рейтингового балла, установленного за итоговое практическое задание.

Невыполнение вышеуказанных условий является текущей академической задолженностью, которая должна быть ликвидирована обучающимся до контрольного мероприятия промежуточной аттестации.

Сведения о наличии у обучающихся текущей академической задолженности, сроках и порядке добора рейтинговых баллов для её ликвидации доводятся до обучающихся педагогическим работником.

В случае неликвидации текущей академической задолженности, педагогический работник обязан во время контрольного мероприятия промежуточной аттестации поставить обучающемуся 0 рейтинговых баллов. В этом случае ликвидация текущей академической задолженности возможна в периоды проведения повторной промежуточной аттестации.

3.3. Проведение промежуточной аттестации обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося

Промежуточная аттестация по дисциплине (модулю) проводится в соответствии с Положением о промежуточной аттестации обучающихся по основным профессиональным образовательным программам высшего образования – программ специалитета в Российском государственном социальном университете и Положением о балльно-рейтинговой системе оценки успеваемости обучающихся по основным профессиональным образовательным программам высшего образования - программам бакалавриата, программам специалитета, программам магистратуры в Российском государственном социальном университете.

На промежуточную аттестацию отводится 20 рейтинговых баллов.

Ответы обучающегося на контрольном мероприятии промежуточной аттестации оцениваются педагогическим работником по 20 - балльной шкале, а итоговая оценка по дисциплине (модулю) выставляется по пятибалльной системе.

Критерии выставления оценки определяются Положением о балльно-рейтинговой системе оценки успеваемости обучающихся по основным профессиональным образовательным программам высшего образования - программам бакалавриата, программам специалитета, программам магистратуры в Российском государственном социальном университете.

В процессе определения рубежного рейтинга обучающегося используется следующая шкала:

Рубежный рейтинг	Критерии оценки освоения обучающимся учебной дисциплины в ходе контрольных мероприятий промежуточной аттестации
19-20 рейтинговых баллов	обучающийся глубоко и прочно усвоил программный материал, исчерпывающе, последовательно, грамотно и логически стройно его излагает, тесно увязывает с задачами и будущей деятельностью, не затрудняется с ответом при видоизменении задания, свободно справляется с задачами и практическими заданиями, правильно обосновывает принятые решения, умеет самостоятельно обобщать и излагать материал, не допуская
16-18 рейтинговых баллов	обучающийся твердо знает программный материал, грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос, может правильно применять теоретические положения и владеет необходимыми умениями и навыками при выполнении практических

13-15 рейтинговых баллов	обучающийся освоил основной материал, но не знает отдельных деталей, допускает неточности, недостаточно правильные формулировки, нарушает последовательность в изложении программного материала и испытывает затруднения в выполнении практических заданий
1-12 рейтинговых баллов	обучающийся не знает значительной части программного материала, допускает существенные ошибки, с большими затруднениями выполняет практические задания
0 рейтинговых баллов	не аттестован

Если результат контроля успеваемости в рамках проведения контрольных мероприятий промежуточной аттестации (рубежный рейтинг обучающегося) неудовлетворительный (получено менее 13 рейтинговых баллов), то промежуточная аттестация по учебной дисциплине (модулю) невозможна даже при наличии высокого текущего рейтинга, полученного по итогам текущего контроля по учебной дисциплине (модулю).

ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

№ п/п	Содержание изменения	Реквизиты документа об утверждении изменения	Дата введения изменения
1.			
2.			
3.			
4.			



**Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Российский государственный социальный университет»**

УТВЕРЖДАЮ

декан факультета политических и социальных

технологий  /Пивнева С.В./

«28» февраля 2024 года.

**МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ
ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ**

Направление подготовки
«Информационные системы и технологии»

Направленность
«Интеллектуальные информационные системы и технологии»

**ОБРАЗОВАТЕЛЬНАЯ ПРОГРАММА ВЫСШЕГО ОБРАЗОВАНИЯ - ПРОГРАММА
БАКАЛАВРИАТА**

Форма обучения

Очная, заочная

Москва, 2024 г.

Методические материалы по дисциплине (модуля) «Объектно-ориентированное программирование» разработаны на основании федерального государственного образовательного стандарта высшего образования – *бакалавриата* по направлению подготовки *09.03.02 Информационные системы и технологии*, утвержденного приказом Министерства образования и науки Российской Федерации от 19.09.2017 № 926, учебного плана по основной профессиональной образовательной программе высшего образования – программе бакалавриата по направлению подготовки *09.03.02 Информационные системы и технологии* (далее – «ОПОП»).

Методические материалы по дисциплине (модуля) «Объектно-ориентированное программирование» разработаны рабочей группой в составе: канд. техн. наук, доцент Шаховской А.В.

Методические материалы по дисциплине (модуля) обсуждена и утверждены на заседании кафедры информационных технологий, искусственного интеллекта и общественно-социальных технологий цифрового общества.

Протокол № 11 от «28» февраля 2024 года

Заведующий кафедрой
канд. пед. наук, доцент


_____ С.В. Пивнева
(подпись)

СОДЕРЖАНИЕ

1. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ К ЛЕКЦИОННЫМ, ЛАБОРАТОРНЫМ ЗАНЯТИЯМ.....	4
1.1. Методические материалы к проведению лекционных занятий по дисциплине (модулю)	4
1.2. Методические материалы по подготовке к практическим занятиям по дисциплине (модулю)	6
1.4. Методические материалы по подготовке к лабораторным занятиям по дисциплине (модулю)	10
2. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ) И САМОСТОЯТЕЛЬНОЙ РАБОТЕ	11
3. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНИВАНИЯ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ И (ИЛИ) ОПЫТА ДЕЯТЕЛЬНОСТИ, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ.....	20
3.1. Организационные основы применения балльно-рейтинговой системы оценки успеваемости обучающихся по дисциплине (модулю)	20
3.2. Проведение текущего контроля успеваемости обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося.....	20
3.3. Проведение промежуточной аттестации обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося.....	21
ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ	23

1. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ К ЛЕКЦИОННЫМ, ЛАБОРАТОРНЫМ ЗАНЯТИЯМ

1.1. Методические материалы к проведению лекционных занятий по дисциплине (модулю)

Лекция - один из методов обучения, одна из основных системообразующих форм организации учебного процесса в вузе. Лекционное занятие представляет собой систематическое, последовательное, монологическое изложение педагогическим работником учебного материала, как правило, теоретического характера. Такое занятие представляет собой элемент технологии представления учебного материала путем логически стройного, систематически последовательного и ясного изложения.

Цель лекции – организация целенаправленной познавательной деятельности обучающихся по овладению программным материалом дисциплины (модуля). Чтение курса лекций позволяет дать связанное, последовательное изложение материала в соответствии с новейшими данными науки, сообщить слушателям основное содержание предмета в целостном, систематизированном виде. В ряде случаев лекция выполняет функцию основного источника информации: при отсутствии учебников и учебных пособий, чаще по новым курсам; в случае, когда новые научные данные по той или иной теме не нашли отражения в учебниках; отдельные разделы и темы очень сложны для самостоятельного изучения. В таких случаях только лектор может методически помочь обучающимся в освоении сложного материала.

Возможные формы проведения лекций:

- Вводная лекция – один из наиболее важных и трудных видов лекции при чтении систематических курсов. От успеха этой лекции во многом зависит успех усвоения всего курса. Она может содержать: определение дисциплины (модуля); краткую историческую справку о дисциплине (модуле); цели и задачи дисциплины (модуля), ее роль в общей системе обучения и связь со смежными дисциплинами (модулями); основные проблемы (понятия и определения) данной науки; основную и дополнительную учебную литературу; особенности самостоятельной работы обучающихся над дисциплиной (модулем) и формы участия в научно-исследовательской работе; отчетность по курсу.

- Информационная лекция ориентирована на изложение и объяснение обучающимся научной информации, подлежащей осмыслению и запоминанию. Это самый традиционный тип лекций в практике высшей школы.

- Заключительная лекция предназначена для обобщения полученных знаний и раскрытия перспектив дальнейшего развития данной науки.

- Обзорная лекция – это систематизация научных знаний на высоком уровне, допускающая большое число ассоциативных связей в процессе осмысления информации, излагаемой при раскрытии внутрипредметной и межпредметной связей, исключая детализацию и конкретизацию. Как правило, стержень излагаемых теоретических положений составляет научно-понятийная и концептуальная основа всего курса или крупных его разделов.

- Лекция-беседа - непосредственный контакт педагогического работника с аудиторией - диалог. По ходу лекции педагогический работник задает вопросы для выяснения мнений и уровня осведомленности обучающихся по рассматриваемой проблеме.

- Лекция-дискуссия - свободный обмен мнениями в ходе изложения лекционного материала. Педагогический работник активизирует участие в обсуждении отдельными вопросами, сопоставляет между собой различные мнения и тем самым развивает дискуссию, стремясь направить ее в нужное русло.

- Лекция с применением обратной связи включает в себе то, что в начале и конце каждого раздела лекции задаются вопросы. Первый - для того, чтобы узнать, насколько обучающиеся ориентируются в излагаемом материале, вопрос в конце раздела предназначен для выяснения степени усвоения только что изложенного материала. При

неудовлетворительных результатах контрольного опроса педагогический работник возвращается к уже прочитанному разделу, изменив при этом методику подачи материала.

- Проблемная лекция опирается на логику последовательно моделируемых проблемных ситуаций путем постановки проблемных вопросов или предъявления проблемных задач. Проблемный вопрос - это диалектическое противоречие, требующее для своего решения размышления, сравнения, поиска, приобретения и применения новых знаний. Проблемная задача содержит дополнительную вводную информацию и при необходимости некоторые ориентиры поиска ее решения.

- Программированная лекция - консультация – педагогический работник сам составляет и предлагает обучающимся вопросы. На подготовленные вопросы педагогический работник сначала просит ответить обучающихся, а затем проводит анализ и обсуждение неправильных ответов. В лекциях можно использовать наглядные материалы, а также подготовить презентацию. Что касается презентации, то в качестве визуальной поддержки ее можно органично интегрировать во все вышеупомянутые лекции. В то же время лекцию-презентацию возможно выделить и в качестве самостоятельной формы. Лекция-презентация должна отражать суть основных и (или) проблемных вопросов лекции, на которые особо следует обратить внимание обучающихся. В условиях применения активного метода проведения занятий презентация представляется весьма удачным способом донесения информации до слушателей. Единственное, на что следует обратить внимание при подготовке слайдов, - это их оформление и текст. Слайд не должен быть перегружен картинками и лишней информацией, которая будет отвлекать от основного аспекта того или иного вопроса лекции. Во время лекции можно задавать вопросы аудитории в отношении того или иного слайда, тем самым еще больше вовлекая обучающихся в проблематику.

Краткое содержание лекционных занятий

Наименование разделов и тем	Содержание учебного материала
Раздел 1. Объектно-ориентированная парадигма программирования	
Тема 1.1. Объектно-ориентированная парадигма	Понятие парадигмы программирования. История появления объектно-ориентированной парадигмы программирования. Объектно-ориентированные языки программирования.
Тема 1.2. Этапы разработки объектно-ориентированной программы	Объектно-ориентированный анализ. Объектно-ориентированное проектирование. Объектно-ориентированное программирование.
Тема 1.3. Основные концепции объектно-ориентированной парадигмы	Абстракция, инкапсуляция, наследование, полиморфизм. Понятия класса и объекта, поля и метода, состояния и поведения.
Раздел 2. Объектно-ориентированное программирование C++	
Тема 2.1. Введение в объектно-ориентированное программирование	Введение в объектно-ориентированное программирование. Понятие объекта, его методов и свойств. Классы в языке C++. Описание методов в структурах и классах. Конструкторы и деструкторы. Решение задач на структуры и классы.
Тема 2.2. Наследование	Понятие родительского класса. Понятие производного класса. Виды наследования: private, public, protected. Одиночное и множественное наследование. Решение задач на одиночное и множественное наследование.

Тема 2.3. Шаблоны	Родовые подпрограммы. Шаблоны классов. Виды шаблонов. Описание шаблонов. Создание шаблонов линейных структур данных. Решение задач на создание шаблонов.
Тема 2.4. Перегрузка операторов	Понятие оператора. Операторы языка программирования C++. Понятие перегрузки функции или оператора. Не перегружаемые операторы. Способы перегрузки операторов. Решение задач на перегрузку операторов.

1.2. Методические материалы по подготовке к практическим занятиям по дисциплине (модулю)

Практические (семинарские) занятия - одна из форм учебного занятия, направленная на развитие самостоятельности обучающихся и приобретение умений и навыков. Данные учебные занятия углубляют, расширяют, детализируют полученные ранее знания. Практическое занятие предполагает выполнение обучающимися по заданию и под руководством преподавателей одной или нескольких практических работ.

Цель практических занятий и семинаров состоит в развитии познавательных способностей, самостоятельности мышления и творческой активности обучающихся; углублении, расширении, детализировании знаний, полученных на лекции в обобщенной форме, и содействии выработке навыков профессиональной деятельности. В отдельных случаях на практических занятиях и семинарах руководителем занятия сообщаются дополнительные знания.

Для достижения поставленных целей и решения требуемого перечня задач практические занятия и семинары проводятся традиционными технологиями или с использованием активных и интерактивных образовательных технологий.

Возможные формы проведения практических (семинарских) занятий:

- Деловая игра - это метод группового обучения совместной деятельности в процессе решения общих задач в условиях максимально возможного приближения к реальным проблемным ситуациям. Имитационные игры - на занятиях имитируется деятельность какой-либо организации, предприятия или его подразделения. Имитироваться могут события, конкретная деятельность людей (деловое совещание, обсуждение плана) и обстановка, условия, в которых происходит событие или осуществляется деятельность (кабинет начальника цеха, зал заседаний). Исполнение ролей (ролевые игры) - в этих играх отрабатывается тактика поведения, действий, выполнение функций и обязанностей конкретного лица. Для проведения игр с исполнением роли разрабатывается модель-пьеса ситуации, между студентами распределяются роли с «обязательным содержанием», характеризующиеся различными интересами; в процессе их взаимодействия должно быть найдено компромиссное решение. «Деловой театр» (метод инсценировки) - в нем разыгрывается какая-либо ситуация, поведение человека в этой обстановке, обучающийся должен вжиться в образ определенного лица, понять его действия, оценить обстановку и найти правильную линию поведения. Основная задача метода инсценировки - научить ориентироваться в различных обстоятельствах, давать объективную оценку своему поведению, учитывать возможности других людей, влиять на их интересы, потребности и деятельность, не прибегая к формальным атрибутам власти, к приказу.

- Игровое проектирование - является практическим занятием или циклом занятий, суть которых состоит в разработке инженерного, конструкторского, технологического и других видов проектов в игровых условиях, максимально воссоздающих реальность. Этот метод отличается высокой степенью сочетания индивидуальной и совместной работы обучающихся.

- Познавательно-дидактические игры не относятся к деловым играм. Они предполагают лишь включение изучаемого материала в необычный игровой контекст и иногда содержат лишь элементы ролевых игр. Такие игры могут проводиться в виде копирования научных, культурных, социальных явлений (конкурс знатоков, «Поле чудес», КВН и т.д.) и в виде предметно-содержательных моделей, (например, игры-путешествия, когда надо разработать рациональный маршрут, пользуясь различными картами).

- Анализ конкретных ситуаций. Конкретная ситуация – это любое событие, которое содержит в себе противоречие или вступает в противоречие с окружающей средой. Ситуации могут нести в себе как позитивный, так и отрицательный опыт. Все ситуации делятся на простые, критические и экстремальные.

- Кейс-метод (от английского case – случай, ситуация) – усовершенствованный метод анализа конкретных ситуаций, метод активного проблемно-ситуационного анализа, основанный на обучении путем решения конкретных задач – ситуаций (решение кейсов). Непосредственная цель метода case-study – обучающиеся должны проанализировать ситуацию, разобраться в сути проблем, предложить возможные решения и выбрать лучшее из них. Кейсы делятся на практические (отражающие реальные жизненные ситуации), обучающие (искусственно созданные, содержащие значительные элементы условности при отражении в нем жизни) и исследовательские (ориентированные на проведение исследовательской деятельности посредством применения метода моделирования). Метод конкретных ситуаций (метод case-study) относится к неигровым имитационным активным методам обучения.

- Тренинг (англ. training от train — обучать, воспитывать) – метод активного обучения, направленный на развитие знаний, умений и навыков и социальных установок. Тренинг – форма интерактивного обучения, целью которого является развитие компетентности межличностного и профессионального поведения в общении. Достоинство тренинга заключается в том, что он обеспечивает активное вовлечение всех участников в процесс обучения. Можно выделить основные типы тренингов по критерию направленности воздействия и изменений – навыковый, психотерапевтический, социально-психологический, бизнес-тренинг.

- Метод Сократа (Майевтика) – метод вопросов, предполагающих критическое отношение к догматическим утверждениям, называется еще как метод «сократовской иронии». Это умение извлекать скрытое в человеке знание с помощью искусных наводящих вопросов, подразумевающего короткий, простой и заранее предсказуемый ответ.

- Интерактивная лекция – выступление ведущего обучающего перед большой аудиторией с применением следующих активных форм обучения: дискуссия, беседа, демонстрация слайдов или учебных фильмов, мозговой штурм.

- Групповая, научная дискуссия, диспут. Дискуссия — это целенаправленное обсуждение конкретного вопроса, сопровождающееся обменом мнениями, идеями между двумя и более лицами. Задача дискуссии – обнаружить различия в понимании вопроса и в споре установить истину. Дискуссии могут быть свободными и управляемыми. К технике управляемой дискуссии относятся: четкое определение цели, прогнозирование реакции оппонентов, планирование своего поведения, ограничение времени на выступления и их заданная очередность. Разновидностью свободной дискуссии является форум, где каждому желающему дается неограниченное время на выступление, при условии, что его выступление вызывает интерес аудитории. Каждый конкретный форум имеет свою тематику — достаточно широкую, чтобы в её пределах можно было вести многоплановое обсуждение.

- Дебаты – это четко структурированный и специально организованный публичный обмен мыслями между двумя сторонами по актуальным темам. Это разновидность публичной дискуссии участников дебатов, направляющая на переубеждение в своей правоте третьей стороны, а не друг друга. Поэтому вербальные и невербальные средства, которые используются участниками дебатов, имеют целью получения определённого результата — сформировать у слушателей положительное впечатление от собственной позиции.

- Метод работы в малых группах. Групповое обсуждение кого-либо вопроса направлено на достижение лучшего взаимопонимания и нахождения истины. Групповое обсуждение способствует лучшему усвоению изучаемого материала. Оптимальное количество участников - 5-7 человек. Перед обучающимися ставится проблема, выделяется определенное время, в течение которого они должны подготовить аргументированный обдуманный ответ. Педагогический работник может устанавливать правила проведения группового обсуждения – задавать определенные рамки обсуждения, ввести алгоритм выработки общего мнения, назначить лидера и др.

- Круглый стол - общество, собрание в рамках более крупного мероприятия (съезда, симпозиума, конференции). Мероприятие, как правило, на которое приглашаются эксперты и специалисты из разных сфер деятельности для обсуждения актуальных вопросов. Данная модель обсуждения, основываясь на соглашениях, в качестве итогов даёт результаты, которые, в свою очередь, являются новыми соглашениями.

- Коллоквиум - (лат. colloquium — разговор, беседа) - одна из форм учебных занятий в системе образования, имеющая целью выяснение и повышение знаний обучающихся. На коллоквиумах обсуждаются: отдельные части, разделы, темы, вопросы изучаемого курса (обычно не включаемые в тематику семинарских и других практических учебных занятий), рефераты, проекты и др. работы обучающихся. Это научные собрания, на которых заслушиваются и обсуждаются доклады. Коллоквиум – это и форма контроля, массового опроса, позволяющая преподавателю в сравнительно небольшой срок выяснить уровень знаний студентов по данной теме дисциплины. Коллоквиум проходит обычно в форме дискуссии, в ходе которой обучающимся предоставляется возможность высказать свою точку зрения на рассматриваемую проблему, учиться обосновывать и защищать ее. Аргументируя и отстаивая свое мнение, обучающийся в то же время демонстрирует, насколько глубоко и осознанно он усвоил изученный материал.

- Метод «мозговой штурм» (мозговой штурм, мозговая атака, англ. brainstorming) — оперативный метод решения проблемы на основе стимулирования творческой активности, при котором участникам обсуждения предлагают высказывать как можно большее количество вариантов решения, в том числе самых фантастичных. Затем из общего числа высказанных идей отбирают наиболее удачные, которые могут быть использованы на практике. Является методом экспертного оценивания.

- Метод проектов - это способ достижения дидактической цели через детальную разработку проблемы (технологию), которая должна завершиться вполне реальным, осязаемым практическим результатом, оформленным тем или иным образом; это совокупность приёмов, действий обучающихся в их определённой последовательности для достижения поставленной задачи – решения проблемы, лично значимой для учащихся и оформленной в виде некоего конечного продукта. Основное предназначение метода проектов состоит в предоставлении учащимся возможности самостоятельного приобретения знаний в процессе решения практических задач или проблем, требующего интеграции знаний из различных предметных областей.

- Брифинг - (англ. briefing от англ. brief – короткий, недолгий) – краткая пресс-конференция, посвященная одному вопросу. Основное отличие: отсутствует презентационная часть. То есть практически сразу идут ответы на вопросы журналистов.

- Метод портфолио (итал. portfolio — 'портфель, англ. - папка для документов) - современная образовательная технология, в основе которой используется метод аутентичного оценивания результатов образовательной и профессиональной деятельности. Портфолио как подборка сертифицированных достижений, наиболее значимых работ и отзывов на них.

Вопросы для самоподготовки по разделам (темам) дисциплины (модуля)

РАЗДЕЛ 1. Объектно-ориентированная парадигма программирования.

Тема 1.1. Объектно-ориентированная парадигма.

Вопросы для самоподготовки:

1. Эволюция методологий программирования. Парадигмы программирования.
2. Основные принципы объектного подхода. Абстрагирование.
3. Основные принципы объектного подхода. Инкапсуляция.
4. Основные принципы объектного подхода. Модульность.

Тема 1.2. Этапы разработки объектно-ориентированной программы.

Вопросы для самоподготовки:

1. Основные принципы объектного подхода. Иерархия.
2. Основные принципы объектного подхода. Типизация.
3. Основные принципы объектного подхода. Параллелизм. Сохраняемость.
4. Объект с точки зрения ООП. Состояние. Поведение.
5. Объект с точки зрения ООП. Идентичность и жизненный цикл объектов.

Тема 1.3. Основные концепции объектно-ориентированной парадигмы.

Вопросы для самоподготовки:

1. Объект с точки зрения ООП. Взаимоотношения между объектами.
2. Классы. Природа классов. Мета модель. Инстанцирование.
3. Классы. Структура класса. Абстрактные классы и интерфейсы.
4. Классы. Отношения между классами. Ассоциация и агрегация.
5. Классы. Иерархии классов. Зависимость.

РАЗДЕЛ 2. Объектно-ориентированное программирование C++.

Тема 2.1. Введение в объектно-ориентированное программирование.

Вопросы для самоподготовки:

1. Средства абстракции C++. Структура класса. Статические члены.
2. Средства инкапсуляции C++. Инкапсуляция и наследование. Друзья.
3. Модульность, раздельная компиляция, пространства имен, using директива.
4. Представление иерархических отношений. Наследование.
5. Представление иерархических отношений. Агрегация. Зависимость по времени жизни.

Тема 2.2. Наследование.

Вопросы для самоподготовки:

1. Правила преобразования типов в C++. Параметрический и виртуальный полиморфизм.
2. C++: средства реализации состояния объектов; реализация поведения.
3. Перегрузка операторов.
4. Жизненный цикл объекта. Инициализация массивов. Конструкторы и деструкторы.
5. Порядок вызова конструкторов и деструкторов при наследовании.

Тема 2.3. Шаблоны.

Вопросы для самоподготовки:

1. Шаблоны классов и шаблоны функций.
2. Понятие объекта, его методов и свойств.
3. Классы в языке C++.
4. Описание методов в структурах и классах.
5. Конструкторы и деструкторы.

Тема 2.4. Перегрузка операторов.

Вопросы для самоподготовки:

1. Решение задач на структуры и классы.
2. Применение одиночного и множественного наследования.
3. Применение шаблонов.
4. Операторы языка программирования C++.
5. Перегрузка функций и операторов. Пространство применения.

1.4. Методические материалы по подготовке к лабораторным занятиям по дисциплине (модулю)

Лабораторные занятия - одна из форм учебного занятия, направленная на развитие самостоятельности обучающихся и приобретение умений и навыков. Данные учебные занятия углубляют, расширяют, детализируют полученные ранее знания. Лабораторное занятие предполагает выполнение обучающимися по заданию и под руководством преподавателей одной или нескольких работ.

Цель лабораторных занятий состоит в развитии познавательных способностей, самостоятельности мышления и творческой активности обучающихся; углублении, расширении, детализировании знаний, полученных на лекции в обобщенной форме, и содействии выработке навыков профессиональной деятельности. В отдельных случаях на лабораторных занятиях руководителем занятия сообщаются дополнительные знания.

Для достижения поставленных целей и решения требуемого перечня задач лабораторные занятия проводятся традиционными технологиями или с использованием активных и интерактивных образовательных технологий.

При подготовке и работе во время проведения лабораторных занятий следует обратить внимание на следующие моменты: на процесс предварительной подготовки, на работу во время занятия, обработку полученных результатов, исправление полученных замечаний.

Предварительная подготовка к лабораторному занятию заключается в изучении теоретического материала в отведенное для самостоятельной работы время, ознакомление с инструктивными материалами с целью осознания задач занятия, техники безопасности при работе с приборами, веществами.

Работа во время проведения занятия включает:

- консультирование студентов преподавателями и вспомогательным персоналом с целью предоставления исчерпывающей информации, необходимой для самостоятельного выполнения предложенных преподавателем задач.
- самостоятельное выполнение заданий согласно обозначенной рабочей программой дисциплины (модуля) тематики.

2. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ) И САМОСТОЯТЕЛЬНОЙ РАБОТЕ

Освоение обучающимся дисциплины (модуля) предполагает изучение материалов дисциплины на аудиторных занятиях и в ходе самостоятельной работы. Аудиторные занятия проходят в форме лекций, семинаров, практических и лабораторных занятий. Самостоятельная работа включает разнообразный комплекс видов и форм работы обучающихся.

Для успешного освоения дисциплины (модуля) и достижения поставленных целей необходимо внимательно ознакомиться с рабочей программы дисциплины (модуля), доступной в электронной информационно-образовательной среде РГСУ.

Следует обратить внимание на списки основной и дополнительной литературы, на предлагаемые преподавателем ресурсы информационно-телекоммуникационной сети Интернет. Эта информация необходима для самостоятельной работы обучающегося.

При подготовке к аудиторным занятиям необходимо помнить особенности каждой формы его проведения.

Подготовка к учебному занятию лекционного типа заключается в следующем.

С целью обеспечения успешного обучения обучающийся должен готовиться к лекции, поскольку она является важнейшей формой организации учебного процесса, поскольку:

- знакомит с новым учебным материалом;
- разъясняет учебные элементы, трудные для понимания;
- систематизирует учебный материал;
- ориентирует в учебном процессе.

С этой целью:

- внимательно прочитайте материал предыдущей лекции;
- ознакомьтесь с учебным материалом по учебнику и учебным пособиям с темой прочитанной лекции;
- внесите дополнения к полученным ранее знаниям по теме лекции на полях лекционной тетради;
- запишите возможные вопросы, которые вы зададите лектору на лекции по материалу изученной лекции;
- постарайтесь уяснить место изучаемой темы в своей подготовке;
- узнайте тему предстоящей лекции (по тематическому плану, по информации лектора) и запишите информацию, которой вы владеете по данному вопросу.

Подготовка к занятию семинарского типа.

При подготовке и работе во время проведения занятий семинарского типа следует обратить внимание на следующие моменты: на процесс предварительной подготовки, на работу во время занятия, обработку полученных результатов, исправление полученных замечаний.

Предварительная подготовка к учебному занятию семинарского типа заключается в изучении теоретического материала в отведенное для самостоятельной работы время, ознакомление с инструктивными материалами с целью осознания задач практического занятия, техники безопасности при работе с приборами, веществами.

Работа во время проведения учебного занятия семинарского типа включает:

- консультирование студентов преподавателями и вспомогательным персоналом с целью предоставления исчерпывающей информации, необходимой для самостоятельного выполнения предложенных преподавателем задач.
- самостоятельное выполнение заданий согласно обозначенной рабочей программой дисциплины (модуля) тематики.

Самостоятельная работа.

Самостоятельная работа - планируемая учебная, учебно-исследовательская, научно-исследовательская работа студентов, выполняемая во внеаудиторное (аудиторное) время по заданию и при методическом руководстве преподавателя, но без его непосредственного

участия (при частичном непосредственном участии преподавателя, оставляющем ведущую роль за работой студентов).

Для более углубленного изучения темы задания для самостоятельной работы рекомендуется выполнять параллельно с изучением данной темы. При выполнении заданий по возможности используйте наглядное представление материала.

Самостоятельная работа студентов в ВУЗе является важным видом учебной и научной деятельности студента. Самостоятельная работа студентов играет значительную роль в рейтинговой технологии обучения.

К современному специалисту в области медицины общество предъявляет достаточно широкий перечень требований, среди которых немаловажное значение имеет наличие у выпускников определенных навыков (компетенций) и умения самостоятельно добывать знания из различных источников, систематизировать полученную информацию, давать оценку конкретной профессиональной ситуации. Формирование такого умения происходит в течение всего периода обучения.

В процессе самостоятельной работы студент приобретает необходимые для будущей специальности компетенции, навыки самоорганизации, самоконтроля, самоуправления, саморефлексии и становится активным самостоятельным субъектом учебной деятельности.

Самостоятельная работа студентов должна оказывать важное влияние на формирование личности будущего специалиста, его компетентность. Каждый студент самостоятельно определяет режим своей работы и меру труда, затрачиваемого на овладение учебным содержанием по каждой дисциплине. Он выполняет внеаудиторную работу по личному индивидуальному плану, в зависимости от его подготовки, времени и других условий.

Виды самостоятельной работы.

Работа с литературой.

При работе с книгой необходимо подобрать литературу, научиться правильно ее читать, вести записи. Для подбора литературы в библиотеке используются алфавитный и систематический каталоги. Важно помнить, что рациональные навыки работы с книгой - это всегда большая экономия времени и сил. Правильный подбор учебников рекомендуется преподавателем, читающим лекционный курс. Необходимая литература может быть также указана в методических разработках по данному курсу. Изучая материал по учебнику, следует переходить к следующему вопросу только после правильного уяснения предыдущего, описывая на бумаге все выкладки и вычисления (в том числе те, которые в учебнике опущены или на лекции даны для самостоятельного вывода). При изучении любой дисциплины большую и важную роль играет самостоятельная индивидуальная работа. Особое внимание следует обратить на определение основных понятий курса. Студент должен подробно разбирать примеры, которые поясняют такие определения, и уметь строить аналогичные примеры самостоятельно. Нужно добиваться точного представления о том, что изучаешь. Полезно составлять опорные конспекты. При изучении материала по учебнику полезно в тетради (на специально отведенных полях) дополнять конспект лекций. Там же следует отмечать вопросы, выделенные студентом для консультации с преподавателем. Выводы, полученные в результате изучения, рекомендуется в конспекте выделять, чтобы они при перечитывании записей лучше запоминались. Опыт показывает, что многим студентам помогает составление листа опорных сигналов, содержащего важнейшие и наиболее часто употребляемые формулы и понятия. Такой лист помогает запомнить формулы, основные положения лекции, а также может служить постоянным справочником для студента. Различают два вида чтения: первичное и вторичное. Первичное - это внимательное, неторопливое чтение, при котором можно остановиться на трудных местах.

Задача вторичного чтения - полное усвоение смысла целого (по счету это чтение может быть и не вторым, а третьим или четвертым). Самостоятельная работа с учебниками и книгами (а также самостоятельное теоретическое исследование проблем, обозначенных преподавателем на лекциях) – это важнейшее условие формирования у себя научного способа познания.

Чтение научного текста является частью познавательной деятельности. Ее цель – извлечение из текста необходимой информации. От того, насколько осознанна читающим собственная внутренняя установка при обращении к печатному слову (найти нужные сведения, усвоить информацию полностью или частично, критически проанализировать материал и т.п.) во многом зависит эффективность осуществляемого действия.

Методические рекомендации по составлению конспекта:

1. Внимательно прочитайте текст. Уточните в справочной литературе непонятные слова. При записи не забудьте вынести справочные данные на поля конспекта;
2. Выделите главное, составьте план;
3. Кратко сформулируйте основные положения текста, отметьте аргументацию автора;
4. Законспектируйте материал, четко следуя пунктам плана. При конспектировании старайтесь выразить мысль своими словами. Записи следует вести четко, ясно.
5. Грамотно записывайте цитаты. Цитируя, учитывайте лаконичность, значимость мысли. В тексте конспекта желательно приводить не только тезисные положения, но и их доказательства. При оформлении конспекта необходимо стремиться к емкости каждого предложения. Мысли автора книги следует излагать кратко, заботясь о стиле и выразительности написанного. Число дополнительных элементов конспекта должно быть логически обоснованным, записи должны распределяться в определенной последовательности, отвечающей логической структуре произведения. Для уточнения и дополнения необходимо оставлять поля. Овладение навыками конспектирования требует от студента целеустремленности, повседневной самостоятельной работы.

Методические материалы по самостоятельному решению задач

При самостоятельном решении задач нужно обосновывать каждый этап решения, исходя из теоретических положений курса. Если студент видит несколько путей решения проблемы (задачи), то нужно сравнить их и выбрать самый рациональный. Полезно до начала вычислений составить краткий план решения проблемы (задачи). Решение проблемных задач или примеров следует излагать подробно, вычисления располагать в строгом порядке, отделяя вспомогательные вычисления от основных. Решения при необходимости нужно сопровождать комментариями, схемами, чертежами и рисунками. Следует помнить, что решение каждой учебной задачи должно доводиться до окончательного логического ответа, которого требует условие, и по возможности с выводом. Полученный ответ следует проверить способами, вытекающими из существа данной задачи. Полезно также (если возможно) решать несколькими способами и сравнить полученные результаты. Решение задач данного типа нужно продолжать до приобретения твердых навыков в их решении.

Методические материалы к выполнению реферата

Реферат (от лат. *refere* – сообщать) – краткое изложение в письменном виде или в форме публикации доклада, содержания научного труда (трудов), литературы по теме. Работа над рефератом условно разделяется на выбор темы, подбор литературы, подготовку и защиту плана; написание теоретической части и всего текста с указанием библиографических данных используемых источников, подготовку доклада, выступление с ним. Тематика рефератов полностью связана с основными вопросами изучаемого курса.

Список литературы к темам не дается, и обучающиеся самостоятельно ведут библиографический поиск, причем им не рекомендуется ограничиваться университетской библиотекой.

Важно учитывать, что написание реферата требует от обучающихся определенных усилий. Особое внимание следует уделить подбору литературы, методике ее изучения с целью отбора и обработки собранного материала, обоснованию актуальности темы и теоретического уровня обоснованности используемых в качестве примеров фактов какой-либо деятельности.

Выбрав тему реферата, начав работу над литературой, необходимо составить план. Изучая литературу, продолжается обдумывание темы, осмысливание прочитанного, делаются выписки, сопоставляются точки зрения разных авторов и т.д. Реферативная работа сводится к тому, чтобы в ней выделились две взаимосвязанные стороны: во-первых, ее следует

рассматривать как учебное задание, которое должен выполнить обучаемый, а во-вторых, как форму научной работы, творческого воображения при выполнении учебного задания.

Наличие плана реферата позволяет контролировать ход работы, избежать формального переписывания текстов из первоисточников.

Оформление реферата включает титульный лист, оглавление и краткий список использованной литературы. Список использованной литературы размещается на последней странице рукописи или печатной форме реферата. Реферат выполняется в письменной или печатной форме на белых листах формата А4 (210 х 297 мм). Шрифт Times New Roman, кегель 14, через 1,5 интервала при соблюдении следующих размеров текста: верхнее поле – 25 мм, нижнее – 20 мм, левое – 30 мм, правое – 15 мм. Нумерация страниц производится сверху листа, по центру. Титульный лист нумерации не подлежит.

Рефераты должны быть написаны простым, ясным языком, без претензий на наукообразность. Следует избегать сложных грамматических оборотов, непривычных терминов и символов. Если же такие термины и символы все-таки приводятся, то необходимо разъяснять их значение при первом упоминании в тексте реферата.

Объем реферата предполагает тщательный отбор информации, необходимой для краткого изложения вопроса. Важнейший этап – редактирование готового текста реферата и подготовка к обсуждению. Обсуждение требует хорошей ориентации в материале темы, умения выделить главное, поставить дискуссионный вопрос, привлечь внимание слушателей к интересной литературе, логично и убедительно изложить свои мысли.

Рефераты обязательно подлежат защите. Процедура защиты начинается с определения оппонентов защищающего свою работу. Они стремятся дать основательный анализ работы обучающимся, обращают внимание на положительные моменты и недостатки реферата, дают общую оценку содержанию, форме преподнесения материала, характеру использованной литературы. Иногда они дополняют тот или иной раздел реферата. Последнее особенно ценно, ибо говорит о глубоком знании обучающимся-оппонентом изучаемой проблемы.

Обсуждение не ограничивается заслушиванием оппонентов. Другие обучающиеся имеют право уточнить или опровергнуть какое-либо утверждение. Преподаватель предлагает любому обучающемуся задать вопрос по существу доклада или попытаться подвести итог обсуждению.

Алгоритм работы над рефератом

1. Выбор темы

Тема должна быть сформулирована грамотно (с литературной точки зрения);

В названии реферата следует поставить четкие рамки рассмотрения темы;

Желательно избегать слишком длинных названий;

Следует по возможности воздерживаться от использования в названии спорных с научной точки зрения терминов, излишней наукообразности, а также чрезмерного упрощения формулировок.

2. Реферат следует составлять из пяти основных частей: введения; основной части; заключения; списка литературы; приложений.

3. Основные требования к введению:

Во введении не следует концентрироваться на содержании; введение должно включать краткое обоснование актуальности темы реферата, где требуется показать, почему данный вопрос может представлять научный интерес и есть ли связь представляемого материала с современностью. Таким образом, тема реферата должна быть актуальна либо с научной точки зрения, либо с современных позиций.

Очень важно выделить цель, а также задачи, которые требуется решить для выполнения цели.

Введение должно содержать краткий обзор изученной литературы, в котором указывается взятый из того или иного источника материал, кратко анализируются изученные источники, показываются их сильные и слабые стороны;

Объем введения составляет две страницы текста.

4. Требования к основной части реферата:

Основная часть содержит материал, отобранный для рассмотрения проблемы;

Также основная часть должна включать в себя собственно мнение обучающихся и сформулированные самостоятельные выводы, опирающиеся на приведенные факты;

Материал, представленный в основной части, должен быть логически изложен и распределен по параграфам, имеющим свои названия;

В изложении основной части необходимо использовать сноски (в первую очередь, когда приводятся цифры и чьи-то цитаты);

Основная часть должна содержать иллюстративный материал (графики, таблицы и т. д.);

Объем основной части составляет около 10 страниц.

5. Требования к заключению:

В заключении формулируются выводы по параграфам, обращается внимание на выдвинутые во введении задачи и цели;

Заключение должно быть четким, кратким, вытекающим из содержания основной части.

6. Требования к оформлению списка литературы (по ГОСТу):

Необходимо соблюдать правильность последовательности записи источников: сначала следует писать фамилию, а после инициалы; название работы не ставится в кавычки; после названия сокращенно пишется место издания; затем идет год издания; наконец, называется процитированная страница.

Критерии оценки реферата

Обучающийся, защищающий реферат, должен рассказать о его актуальности, поставленных целях и задачах, изученной литературе, структуре основной части, сделанных в ходе работы выводах.

По окончании выступления ему может быть задано несколько вопросов по представленной проблеме.

Оценка складывается из соблюдения требований к реферату, грамотного раскрытия темы, умения четко рассказывать о представленном реферате, способности понять суть задаваемых по работе вопросов и найти точные ответы на них.

Методические материалы к выполнению эссе

Эссе – литературное произведение небольшого объема, обычно прозаическое, свободной композиции, передающее индивидуальные впечатления, суждения, соображения автора о той или иной проблеме, теме, о том или ином событии или явлении. Это вид самостоятельной исследовательской работы обучающихся, с целью углубления и закрепления теоретических знаний и освоения практических навыков. Цель эссе состоит в развитии самостоятельного творческого мышления и письменного изложения собственных мыслей. При написании эссе обучающийся должен представить развернутый письменный ответ на теоретический или практический актуальный вопрос, объявленный преподавателем в аудитории непосредственно перед ее написанием. В процессе написания эссе разрешается пользоваться нормативно-правовыми актами, конспектом лекций (в печатном виде). Использование интернет-ресурсов не допускается. Темы эссе преподаватель предлагает из числа тех, которые обучающиеся уже рассматривали на лекциях или семинарских занятиях, исходя из содержания заданий в составе оценочных средств. По решению преподавателя, в качестве темы эссе может быть выбрана одна или несколько тем, которые могут быть распределены между обучающимися по желанию.

Требования к выполнению эссе:

1. Проводится письменно.

2. Эссе выполняется на компьютере (гарнитура Times New Roman, шрифт 14) через 1,5 интервала с полями: верхнее, нижнее – 2; правое – 3; левое – 1,5. Отступ первой строки абзаца – 1,25. Сноски – постраничные. Таблицы и рисунки встраиваются в текст работы. При этом обязательный заголовок таблицы надо размещать над табличным полем, а рисунки

сопровождать подрисуночными подписями. При включении в эссе нескольких таблиц и/или рисунков их нумерация обязательна. Обязательна и нумерация страниц. Их целесообразно проставлять внизу страницы – по середине или в правом углу. Номер страницы не ставится на титульном листе, но в общее число страниц он включается. Объем эссе, без учета приложений, не должен превышать 5 страниц. Значительное превышение установленного объема является недостатком работы и указывает на то, что обучающийся не сумел отобрать и переработать необходимый материал.

3. Работа должна содержать собственные умозаключения по сути поставленной проблемы, включать самостоятельно проведенный анализ по сути этой проблемы, выводы, обобщающие авторскую позицию по поставленной проблеме.

Критерии оценки эссе:

«Отлично» – исключительные знания материала, абсолютное понимание сути, безукоризненное знание основных понятий и положений, логически и лексически грамотно изложенный, содержательный, аргументированный, конкретный и исчерпывающий ответ.

«Хорошо» – глубокие знания материала, правильное понимание сути, знание основных понятий и положений, содержательный, полный и конкретный ответ.

«Удовлетворительно» – твердые, но недостаточно полные знания, верное понимание сути, в целом правильный ответ.

«Неудовлетворительно» – непонимание сущности задания, грубые ошибки в ответе.

Методические материалы по выполнению тестирования.

Тестовые задания содержат вопросы и 3-4 варианта ответа по базовым положениям изучаемой темы/раздела, составлены с расчетом на знания, полученные обучающимся в процессе изучения темы/раздела.

Тестовые задания выполняются в письменной форме и сдаются преподавателю, ведущему дисциплину (модуль). На выполнение тестовых заданий обучающимся отводится 45 минут.

При обработке результатов оценочной процедуры используются: критерии оценки по содержанию и качеству полученных ответов, ключи, оценочные листы.

Критерии оценки теста:

«Зачтено» - если обучающийся правильно выполнил не менее 2/3 всей работы или допустил не более одной грубой ошибки и двух недочетов, не более одной грубой и одной негрубой ошибки, не более трех негрубых ошибок, одной негрубой ошибки и трех недочетов, при наличии четырех-пяти недочетов.

«Не зачтено» - если число ошибок и недочетов превысило норму для оценки 3 или правильно выполнено менее 2/3 всей работы.

Методические материалы по выполнению доклада.

Рекомендуется следующая структура доклада:

1. титульный лист, содержание доклада;
2. краткое изложение;
3. цели и задачи;
4. изложение характера исследований и рассмотренных проблем, гипотезы, спорные вопросы;
5. источники информации, методы сбора и анализа данных, степень их полноты и достоверности;
6. анализ и толкование полученных в работе результатов;
7. выводы и оценки;
8. библиография и приложения.

Время выступления докладчика не должно превышать 10 минут.

Основные требования к оформлению доклада:

- титульный лист должен включать название доклада, наименование предметной (цикловой) комиссии, фамилию обучающегося;

- все использованные литературные источники сопровождаются библиографическим описанием;
- приводимая цитата из источника берется в кавычки (оформляются сноски);
- единицы измерения должны применяться в соответствии с действующими стандартами;
- все названия литературных источников следует приводить в соответствии с новейшими изданиями;
- рекомендуется включение таблиц, графиков, схем, если они отражают основное содержание или улучшают ее наглядность;
- названия фирм, учреждений, организаций и предприятий должны именоваться так, как они указываются в источнике;

Критерии оценки доклада

При выполнении доклада обучающийся должен продемонстрировать умение кратко излагать прочитанный материал, а также умение обобщать и анализировать материал по теме доклада.

Презентация

Методические материалы к презентациям

1. Объем презентации 10 -20 слайдов.
2. На титульном слайде должно быть отражено:
 - наименование факультета;
 - тема презентации;
 - фамилия, имя, отчество, направление подготовки/ специальность, направленность (профиль)/ специализация, форма обучения, номер группы автора презентации;
 - фамилия, имя, отчество, степень, звание, должность руководитель работы;
 - год выполнения работы.
3. В презентации должны быть отражены обоснование актуальности представляемого материала, цели и задачи работы.
4. Содержание презентации должно включать наиболее значимый материал доклада, а также, при необходимости, таблицы, диаграммы, рисунки, фотографии, карты, видео – вставки, звуковое сопровождение.
5. Заключительный слайд должен содержать информацию об источниках информации для презентации.

Критерии оценки презентации

1. Объем презентации 10 -20 слайдов.
2. Правильность оформления титульного слайда.
3. Актуальность отобранного материала, обоснованность формулировки цели и задач работы.
4. Наглядность и логичность презентации, обоснованность использования таблиц, диаграмм, рисунков, фотографий, карт, видео – вставок, звукового сопровождения; правильный выбор шрифтов, фона, других элементов дизайна слайда.
5. Объем и качество источников информации (не менее 2-х интернет – источников и не менее 2-х литературных источников).

Методические материалы по подготовке к опросу

Самостоятельная работа обучающихся включает подготовку к опросу на практических занятиях. Для этого обучающийся изучает лекции, основную и дополнительную литературу, публикации, информацию из Интернет-ресурсов.

Тема и вопросы к практическим занятиям, вопросы для самоконтроля содержатся в рабочей программе и доводятся до студентов заранее.

Для подготовки к опросу обучающемуся необходимо ознакомиться с материалом, посвященным теме практического занятия, в учебнике или другой рекомендованной литературе, конспекте лекции, обратить внимание на усвоение основных понятий

дисциплины, выявить неясные вопросы и подобрать дополнительную литературу для их освещения.

Критерии оценки опроса

«Отлично»:

- дан полный, развернутый ответ на поставленный вопрос;
- в ответе прослеживается четкая структура, логическая последовательность, отражающая сущность раскрываемых понятий, теорий, явлений;
- знание по предмету демонстрируется на фоне понимания его в системе данной науки и междисциплинарных связей;
- свободное владение терминологией;
- ответы на дополнительные вопросы четкие, краткие;

«Хорошо»:

- дан полный, развернутый ответ на поставленный вопрос, показано умение выделять существенные и несущественные признаки, причинно-следственные связи;
- ответ недостаточно логичен с единичными ошибками в частностях, исправленные студентом с помощью преподавателя;
- единичные ошибки в терминологии;
- ответы на дополнительные вопросы правильные, недостаточно полные и четкие.

«Удовлетворительно»:

- ответ не полный, с ошибками в деталях, умение раскрыть значение обобщённых знаний не показано, речевое оформление требует поправок, коррекции;
- логика и последовательность изложения имеют нарушения, студент не способен самостоятельно выделить существенные и несущественные признаки и причинно-следственные связи;
- ошибки в раскрываемых понятиях, терминах;
- студент не ориентируется в теме, допускает серьезные ошибки;
- студент не может ответить на большую часть дополнительных вопросов.

«Неудовлетворительно»:

- ответ представляет собой разрозненные знания с существенными ошибками по вопросу;
- присутствуют фрагментарность, нелогичность изложения, студент не осознает связь обсуждаемого вопроса с другими объектами дисциплины, речь неграмотная;
- незнание терминологии;
- ответы на дополнительные вопросы неправильные.

Методические материалы по выполнению практического задания

При выполнении практического задания обучающийся придерживается следующего алгоритма:

1. Записать дату, тему и цель задания;
2. Ознакомиться с правилами и условия выполнения практического задания;
3. Повторить теоретические задания, необходимые для рациональной работы и других практических действий, используя конспекты лекций и рекомендованную литературу, представленной в программе;
4. Выполнить работу по предложенному алгоритму действий;
5. Обобщить результаты работы, сформулировать выводы / дать ответы на контрольные вопросы;

Работа должна быть выполнена грамотно, с соблюдением культуры изложения. При использовании данных из учебных, методических пособий и другой литературы, периодических изданий, Интернет-источников должны иметься ссылки на вышеперечисленные.

Критерии оценки практического задания:

«Отлично» – правильный ответ, дается четкое обоснование принятому решению; рассуждения четкие последовательные логические; используются ссылки на полученные при

изучении дисциплины знания; правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Хорошо» – правильный ответ, дается обоснование принятому решению; но с не существенными ошибками, в рассуждениях отсутствует логическая последовательность; используются ссылки на полученные при изучении дисциплины знания, правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Удовлетворительно» – правильный ответ, допускаются грубые ошибки в обосновании принятого решения; рассуждения не последовательные сумбурные; используются ссылки на полученные при изучении дисциплины знания; используются формулы, процедуры, понятия, имеющие прямое значение для подтверждения принятого решения, однако, при обращении к ним допускаются серьезные ошибки, студент не может правильно ими воспользоваться.

«Неудовлетворительно, не зачтено» – ответ неверный, отсутствует обоснование принятому решению; студент демонстрирует полное непонимание сути вопроса.

Для оценки решения ситуационной задачи (аналитического задания):

Оценка «отлично» выставляется, если задача решена грамотно, ответы на вопросы сформулированы четко. Эталонный ответ полностью соответствует решению студента, которое хорошо обосновано теоретически.

Оценка «хорошо» выставляется, если задача решена, ответы на вопросы сформулированы не достаточно четко. Решение студента в целом соответствует эталонному ответу, но не достаточно хорошо обосновано теоретически.

Оценка «удовлетворительно» выставляется, если задача решена не полностью, ответы не содержат всех необходимых обоснований решения.

Оценка «неудовлетворительно» выставляется, если задача не решена или имеет грубые теоретические ошибки в ответе на поставленные вопросы.

Методические материалы по выполнению лабораторного задания

При выполнении лабораторного задания обучающийся придерживается следующего алгоритма:

1. Записать дату, тему и цель задания;
2. Ознакомиться с правилами и условия выполнения задания;
3. Повторить теоретические задания, необходимые для рациональной работы и других практических действий, используя конспекты лекций и рекомендованную литературу, представленной в программе;
4. Выполнить работу по предложенному алгоритму действий;
5. Обобщить результаты работы, сформулировать выводы / дать ответы на контрольные вопросы;

Работа должна быть выполнена грамотно, с соблюдением культуры изложения. При использовании данных из учебных, методических пособий и другой литературы, периодических изданий, Интернет-источников должны иметься ссылки на вышеперечисленные.

Критерии оценки лабораторного задания:

«Отлично» – правильный ответ, дается четкое обоснование принятому решению; рассуждения четкие последовательные логические; используются ссылки на полученные при изучении дисциплины знания; правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Хорошо» – правильный ответ, дается обоснование принятому решению; но с не существенными ошибками, в рассуждениях отсутствует логическая последовательность; используются ссылки на полученные при изучении дисциплины знания, правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Удовлетворительно» – правильный ответ, допускаются грубые ошибки в обосновании принятого решения; рассуждения не последовательные сумбурные; используются ссылки на полученные при изучении дисциплины знания; используются формулы, процедуры, понятия, имеющие прямое значение для подтверждения принятого решения, однако, при обращении к ним допускаются серьезные ошибки, студент не может правильно ими воспользоваться.

«Неудовлетворительно, не зачтено» – ответ неверный, отсутствует обоснование принятому решению; студент демонстрирует полное непонимание сути вопроса.

Методические указания для подготовки к промежуточной аттестации.

Изучение учебных дисциплин (модулей) завершается зачетом/зачетом с оценкой или экзаменом. Подготовка к промежуточной аттестации способствует закреплению, углублению и обобщению знаний, получаемых, в процессе обучения, а также применению их к решению практических задач. Готовясь к экзамену, студент ликвидирует имеющиеся пробелы в знаниях, углубляет, систематизирует и упорядочивает свои знания. На зачете или экзамене студент демонстрирует то, что он освоил в процессе обучения по дисциплине (модулю).

Вначале следует просмотреть весь материал по дисциплине (модулю), отметить для себя трудные вопросы. Обязательно в них разобраться. В заключение еще раз целесообразно повторить основные положения, используя при этом листы опорных сигналов. Систематическая подготовка к занятиям в течение семестра позволит использовать время промежуточной аттестации для систематизации знаний.

3. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНИВАНИЯ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ И (ИЛИ) ОПЫТА ДЕЯТЕЛЬНОСТИ, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ

3.1. Организационные основы применения балльно-рейтинговой системы оценки успеваемости обучающихся по дисциплине (модулю)

Оценка качества освоения обучающимися дисциплины (модуля) реализуется в формате балльно-рейтинговой системы оценки успеваемости обучающихся (БРСО).

БРСО в ходе текущего контроля успеваемости и промежуточной аттестации осуществляется по 100-балльной шкале.

Академический рейтинг обучающегося по дисциплине (модулю) складывается из результатов:

- текущего контроля успеваемости (максимальный текущий рейтинг обучающегося 80 рейтинговых баллов);
- промежуточной аттестации (максимальный рубежный рейтинг обучающегося 20 рейтинговых баллов).

Условия оценки освоения обучающимся дисциплины (модуля) в формате БРСО доводятся преподавателем до сведения обучающихся на первом учебном занятии, а также размещены в свободном доступе в электронной информационно-образовательной среде Университета.

3.2. Проведение текущего контроля успеваемости обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося

В течение учебного семестра до промежуточной аттестации на основании утвержденной рабочей программы дисциплины (модуля) формируется текущий рейтинг обучающегося. Текущий рейтинг обучающегося складывается как сумма рейтинговых баллов, полученных им в течение учебного семестра по всем видам учебных занятий по учебной дисциплине.

В процессе текущего контроля оцениваются следующие действия обучающегося, направленные на освоение компетенций в рамках изучения учебной дисциплины:

- академическая активность (посещаемость учебных занятий, самостоятельное изучение содержания учебной дисциплины в электронной информационно-образовательной среде, соблюдение сроков сдачи практических заданий и текущих контрольных мероприятий и др.);

- выполнение и сдача текущих и итогового практических заданий (эссе, рефераты, творческие задания, кейс-задания, лабораторные работы, расчетные задания и др., активное участие в групповых интерактивных занятиях (дискуссии, WiKi-проекты и др.), защита проектов и др.);

- прохождение рубежей текущего контроля, включая соблюдение графика их прохождения в электронной информационно-образовательной среде.

Для планирования расчета текущего рейтинга обучающегося используются следующие пропорции:

Вид учебного действия	Максимальная рейтинговая оценка, баллов
академическая активность	10
практические задания	40
<i>из них: текущие практические задания</i>	20
<i>итоговое практическое задание</i>	20
рубежи текущего контроля	30
<i>ИТОГО:</i>	80

В течение учебного семестра по дисциплине (модулю) обучающимся должен быть накоплен текущий рейтинг не менее 52 рейтинговых баллов (65% от максимального значения текущего рейтинга).

Необходимыми условиями допуска обучающегося к промежуточной аттестации по дисциплине являются положительное прохождение обучающимся не менее 65% рубежей текущего контроля с накоплением не менее 65% максимального рейтингового балла за каждый рубеж текущего контроля и положительное выполнение итогового практического задания с накоплением не менее 65% максимального рейтингового балла, установленного за итоговое практическое задание.

Невыполнение вышеуказанных условий является текущей академической задолженностью, которая должна быть ликвидирована обучающимся до контрольного мероприятия промежуточной аттестации.

Сведения о наличии у обучающихся текущей академической задолженности, сроках и порядке добора рейтинговых баллов для её ликвидации доводятся до обучающихся педагогическим работником.

В случае неликвидации текущей академической задолженности, педагогический работник обязан во время контрольного мероприятия промежуточной аттестации поставить обучающемуся 0 рейтинговых баллов. В этом случае ликвидация текущей академической задолженности возможна в периоды проведения повторной промежуточной аттестации.

3.3. Проведение промежуточной аттестации обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося

Промежуточная аттестация по дисциплине (модулю) проводится в соответствии с Положением о промежуточной аттестации обучающихся по основным профессиональным образовательным программам высшего образования – программ специалитета в Российском государственном социальном университете и Положением о балльно-рейтинговой системе оценки успеваемости обучающихся по основным профессиональным образовательным программам высшего образования - программам бакалавриата, программам специалитета, программам магистратуры в Российском государственном социальном университете.

На промежуточную аттестацию отводится 20 рейтинговых баллов.

Ответы обучающегося на контрольном мероприятии промежуточной аттестации оцениваются педагогическим работником по 20 - балльной шкале, а итоговая оценка по

дисциплине (модулю) выставляется по пятибалльной системе для экзамена/дифференцированного зачета и по пятибалльной системе для зачета с оценкой.

Критерии выставления оценки определяются Положением о балльно-рейтинговой системе оценки успеваемости обучающихся по основным профессиональным образовательным программам высшего образования - программам бакалавриата, программам специалитета, программам магистратуры в Российском государственном социальном университете.

В процессе определения рубежного рейтинга обучающегося используется следующая шкала:

Рубежный рейтинг	Критерии оценки освоения обучающимся учебной дисциплины в ходе контрольных мероприятий промежуточной аттестации
19-20 рейтинговых баллов	обучающийся глубоко и прочно усвоил программный материал, исчерпывающе, последовательно, грамотно и логически стройно его излагает, тесно увязывает с задачами и будущей деятельностью, не затрудняется с ответом при видоизменении задания, свободно справляется с задачами и практическими заданиями, правильно обосновывает принятые решения, умеет самостоятельно обобщать и излагать материал, не допуская
16-18 рейтинговых баллов	обучающийся твердо знает программный материал, грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос, может правильно применять теоретические положения и владеет необходимыми умениями и навыками при выполнении практических
13-15 рейтинговых баллов	обучающийся освоил основной материал, но не знает отдельных деталей, допускает неточности, недостаточно правильные формулировки, нарушает последовательность в изложении программного материала и испытывает затруднения в выполнении практических заданий
1-12 рейтинговых баллов	обучающийся не знает значительной части программного материала, допускает существенные ошибки, с большими затруднениями выполняет практические задания
0 рейтинговых баллов	не аттестован

Если результат контроля успеваемости в рамках проведения контрольных мероприятий промежуточной аттестации (рубежный рейтинг обучающегося) неудовлетворительный (получено менее 13 рейтинговых баллов), то промежуточная аттестация по учебной дисциплине (модулю) невозможна даже при наличии высокого текущего рейтинга, полученного по итогам текущего контроля по учебной дисциплине (модулю).

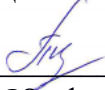
ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

№ п/п	Содержание изменения	Реквизиты документа об утверждении изменения	Дата введения изменения
1.			
2.			
3.			
4.			



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Российский государственный социальный университет»

УТВЕРЖДАЮ
Декана факультета политических и
социальных технологий

 / С.В. Пивнева
«28» февраля 2024 г.

МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ
«WEB-ПРОГРАММИРОВАНИЕ»

Направление подготовки
«Информационные системы и технологии»

Направленность
«Интеллектуальные информационные системы и технологии»

ОБРАЗОВАТЕЛЬНАЯ ПРОГРАММА ВЫСШЕГО ОБРАЗОВАНИЯ - ПРОГРАММА
БАКАЛАВРИАТА

Форма обучения

Очная, заочная

Москва, 2024 г.

Методические материалы по дисциплине (модулю) «Web-программирование» разработаны на основании федерального государственного образовательного стандарта высшего образования – *бакалавриата* по направлению подготовки *09.03.02 Информационные системы и технологии*, утвержденного приказом Министерства образования и науки Российской Федерации от 19.09.2017 № 926, учебного плана по основной профессиональной образовательной программе высшего образования – программе бакалавриата по направлению подготовки *09.03.02 Информационные системы и технологии* (далее – «ОПОП»).

Методические материалы по дисциплине (модулю) разработаны канд. техн. наук, доцентом Е.Ю. Малышевой.

Методические материалы по дисциплине (модулю) обсуждены и утверждены на заседании кафедры информационных технологий, искусственного интеллекта и общественно-социальных технологий цифрового общества факультета социальных и политических технологий.

Протокол № 11 от «28» февраля 2024 г.

Заведующий кафедрой
канд. пед. наук, доцент


(подпись)

С.В. Пивнева

СОДЕРЖАНИЕ

1. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ К ЛЕКЦИОННЫМ И ЛАБОРАТОРНЫМ ЗАНЯТИЯМ	4
1.1. Методические материалы к проведению лекционных занятий по дисциплине (модулю)	4
1.2. Методические материалы по подготовке к лабораторным занятиям по дисциплине (модулю)	6
2. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ) И САМОСТОЯТЕЛЬНОЙ РАБОТЕ	8
3. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНИВАНИЯ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ И (ИЛИ) ОПЫТА ДЕЯТЕЛЬНОСТИ, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ	18
3.1. Организационные основы применения балльно-рейтинговой системы оценки успеваемости обучающихся по дисциплине (модулю)	18
3.2. Проведение текущего контроля успеваемости обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося	18
3.3. Проведение промежуточной аттестации обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося	19
ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ	21

1. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ К ЛЕКЦИОННЫМ И ЛАБОРАТОРНЫМ ЗАНЯТИЯМ

1.1. Методические материалы к проведению лекционных занятий по дисциплине (модулю)

Лекция - один из методов обучения, одна из основных системообразующих форм организации учебного процесса в вузе. Лекционное занятие представляет собой систематическое, последовательное, монологическое изложение педагогическим работником учебного материала, как правило, теоретического характера. Такое занятие представляет собой элемент технологии представления учебного материала путем логически стройного, систематически последовательного и ясного изложения.

Цель лекции – организация целенаправленной познавательной деятельности обучающихся по овладению программным материалом дисциплины (модуля). Чтение курса лекций позволяет дать связанное, последовательное изложение материала в соответствии с новейшими данными науки, сообщить слушателям основное содержание предмета в целостном, систематизированном виде. В ряде случаев лекция выполняет функцию основного источника информации: при отсутствии учебников и учебных пособий, чаще по новым курсам; в случае, когда новые научные данные по той или иной теме не нашли отражения в учебниках; отдельные разделы и темы очень сложны для самостоятельного изучения. В таких случаях только лектор может методически помочь обучающимся в освоении сложного материала.

Возможные формы проведения лекций:

- Вводная лекция – один из наиболее важных и трудных видов лекции при чтении систематических курсов. От успеха этой лекции во многом зависит успех усвоения всего курса. Она может содержать: определение дисциплины (модуля); краткую историческую справку о дисциплине (модуле); цели и задачи дисциплины (модуля), ее роль в общей системе обучения и связь со смежными дисциплинами (модулями); основные проблемы (понятия и определения) данной науки; основную и дополнительную учебную литературу; особенности самостоятельной работы обучающихся над дисциплиной (модулем) и формы участия в научно-исследовательской работе; отчетность по курсу.

- Информационная лекция ориентирована на изложение и объяснение обучающимся научной информации, подлежащей осмыслению и запоминанию. Это самый традиционный тип лекций в практике высшей школы.

- Заключительная лекция предназначена для обобщения полученных знаний и раскрытия перспектив дальнейшего развития данной науки.

- Обзорная лекция – это систематизация научных знаний на высоком уровне, допускающая большое число ассоциативных связей в процессе осмысления информации, излагаемой при раскрытии внутрипредметной и межпредметной связей, исключая детализацию и конкретизацию. Как правило, стержень излагаемых теоретических положений составляет научно-понятийная и концептуальная основа всего курса или крупных его разделов.

- Лекция-беседа - непосредственный контакт педагогического работника с аудиторией - диалог. По ходу лекции педагогический работник задает вопросы для выяснения мнений и уровня осведомленности обучающихся по рассматриваемой проблеме.

- Лекция-дискуссия - свободный обмен мнениями в ходе изложения лекционного материала. Педагогический работник активизирует участие в обсуждении отдельными вопросами, сопоставляет между собой различные мнения и тем самым развивает дискуссию, стремясь направить ее в нужное русло.

- Лекция с применением обратной связи включает в себе то, что в начале и конце каждого раздела лекции задаются вопросы. Первый - для того, чтобы узнать, насколько обучающиеся ориентируются в излагаемом материале, вопрос в конце раздела предназначен для выяснения степени усвоения только что изложенного материала. При

неудовлетворительных результатах контрольного опроса педагогический работник возвращается к уже прочитанному разделу, изменив при этом методику подачи материала.

- Проблемная лекция опирается на логику последовательно моделируемых проблемных ситуаций путем постановки проблемных вопросов или предъявления проблемных задач. Проблемный вопрос - это диалектическое противоречие, требующее для своего решения размышления, сравнения, поиска, приобретения и применения новых знаний. Проблемная задача содержит дополнительную вводную информацию и при необходимости некоторые ориентиры поиска ее решения.

- Программированная лекция - консультация – педагогический работник сам составляет и предлагает обучающимся вопросы. На подготовленные вопросы педагогический работник сначала просит ответить обучающихся, а затем проводит анализ и обсуждение неправильных ответов. В лекциях можно использовать наглядные материалы, а также подготовить презентацию. Что касается презентации, то в качестве визуальной поддержки ее можно органично интегрировать во все вышеупомянутые лекции. В то же время лекцию-презентацию возможно выделить и в качестве самостоятельной формы. Лекция-презентация должна отражать суть основных и (или) проблемных вопросов лекции, на которые особо следует обратить внимание обучающихся. В условиях применения активного метода проведения занятий презентация представляется весьма удачным способом донесения информации до слушателей. Единственное, на что следует обратить внимание при подготовке слайдов, - это их оформление и текст. Слайд не должен быть перегружен картинками и лишней информацией, которая будет отвлекать от основного аспекта того или иного вопроса лекции. Во время лекции можно задавать вопросы аудитории в отношении того или иного слайда, тем самым еще больше вовлекая обучающихся в проблематику.

Краткое содержание лекционных занятий

Наименование разделов	Содержание учебного материала
РАЗДЕЛ 1. ОСНОВНЫЕ ПОНЯТИЯ WEB-ПРОГРАММИРОВАНИЯ. WEB-ДИЗАЙН	Интернет как среда для web-взаимодействия. Основные Интернет-протоколы. Система доменных имен DNS. Структура и принципы организации WWW Протокол HTTP. Безопасность HTTP Назначение и особенности HTML. Структура документа на HTML. Основные теги тела документа HTML. Формы HTML Принцип разделения контента и оформления web-документа. Основы CSS. Особенности применения CSS для указания формы и расположения блоков Понятие и назначение адаптивной верстки. CSS-фреймворки. Использование Bootstrap.
РАЗДЕЛ 2. ОСНОВЫ ПРОГРАММИРОВАНИЯ КЛИЕНТСКОЙ И СЕРВЕРНОЙ ЧАСТИ WEB-ПРИЛОЖЕНИЯ	Назначение и возможности скриптовых языков программирования. Основы создания скриптов на языке JavaScript. Основы языка JavaScript.

	Функции JavaScript. Строки и массивы JavaScript. События JavaScript. Примеры использования JavaScript. Библиотеки JavaScript. JS-фреймворки Серверное программирование. Назначение и возможности PHP. Основы синтаксиса и типы переменных PHP. Условные и циклические операторы в PHP. Массивы в PHP. Функции PHP Основы СУБД MySQL. Основные возможности языка манипулирования данными SQL для работы с базами данных. Функции PHP для работы с MySQL.

1.2. Методические материалы по подготовке к лабораторным занятиям по дисциплине (модулю)

Лабораторные занятия - одна из форм учебного занятия, направленная на развитие самостоятельности обучающихся и приобретение умений и навыков. Данные учебные занятия углубляют, расширяют, детализируют полученные ранее знания. Лабораторное занятие предполагает выполнение обучающимися по заданию и под руководством преподавателей одной или нескольких работ.

Цель лабораторных занятий состоит в развитии познавательных способностей, самостоятельности мышления и творческой активности обучающихся; углублении, расширении, детализировании знаний, полученных на лекции в обобщенной форме, и содействии выработке навыков профессиональной деятельности. В отдельных случаях на лабораторных занятиях руководителем занятия сообщаются дополнительные знания.

Для достижения поставленных целей и решения требуемого перечня задач лабораторные занятия проводятся традиционными технологиями или с использованием активных и интерактивных образовательных технологий.

При подготовке и работе во время проведения лабораторных занятий следует обратить внимание на следующие моменты: на процесс предварительной подготовки, на работу во время занятия, обработку полученных результатов, исправление полученных замечаний.

Предварительная подготовка к лабораторному занятию заключается в изучении теоретического материала в отведенное для самостоятельной работы время, ознакомление с инструктивными материалами с целью осознания задач занятия, техники безопасности при работе с приборами, веществами.

Работа во время проведения занятия включает:

- консультирование студентов преподавателями и вспомогательным персоналом с целью предоставления исчерпывающей информации, необходимой для самостоятельного выполнения предложенных преподавателем задач.
- самостоятельное выполнение заданий согласно обозначенной рабочей программой дисциплины (модуля) тематики.

Задания для самостоятельной работы к Разделу 1

Вопросы для самостоятельной работы к Разделу 1

1. Протокол HTTP
2. Структура документа на HTML.
3. Основные теги тела документа HTML.

4. Формы HTML
5. Основы CSS.
6. Особенности применения CSS
7. Понятие и назначение адаптивной верстки.

Литература для самостоятельного изучения к Разделу 1

Основная литература

1. Полуэктова, Н. Р. Разработка веб-приложений : учебное пособие для вузов / Н. Р. Полуэктова. — Москва : Издательство Юрайт, 2023. — 204 с. — (Высшее образование). — ISBN 978-5-534-13715-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/519714>
2. Сысолетин, Е. Г. Разработка интернет-приложений : учебное пособие для вузов / Е. Г. Сысолетин, С. Д. Ростунцев ; под научной редакцией Л. Г. Доросинского. — Москва : Издательство Юрайт, 2023. — 90 с. — (Высшее образование). — ISBN 978-5-9916-9975-4. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/514303>
3. Тузовский, А. Ф. Проектирование и разработка web-приложений : учебное пособие для вузов / А.Ф. Тузовский. — Москва : Издательство Юрайт, 2023. — 218 с. — (Высшее образование). — ISBN 978-5-534-16300-1: — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/530767>

Дополнительная литература

1. Гаврилов, М. В. Информатика и информационные технологии : учебник для вузов / М. В. Гаврилов, В. А. Климов. — 5-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 355 с. — (Высшее образование). — ISBN 978-5-534-15819-9. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509820>
2. Грекул, В. И. Проектирование информационных систем : учебник и практикум для вузов / В. И. Грекул, Н. Л. Коровкина, Г. А. Левочкина. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-9916-8764-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511889>
3. Маркин А. В. ПРОГРАММИРОВАНИЕ НА SQL В 2 Ч. ЧАСТЬ 1 3-е изд., пер. и доп. Учебник и практикум для вузов. А. В. Маркин — Москва : Издательство Юрайт, 2023. — 429 с. — (Высшее образование). — ISBN 978-5-534-15817-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509818>
4. Проектирование информационных систем : учебник и практикум для вузов / Д. В. Чистов, П. П. Мельников, А. В. Золотарюк, Н. Б. Ничепорук ; под общей редакцией Д. В. Чистова. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 293 с. — (Высшее образование). — ISBN 978-5-534-15923-3. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/510287>

Задания для самостоятельной работы к Разделу 2

Вопросы для самостоятельной работы к Разделу 2

1. Основы языка JavaScript.
2. Функции JavaScript.
3. Строки и массивы JavaScript.
4. События JavaScript.
5. Назначение и возможности PHP.
6. Основы синтаксиса и типы переменных PHP.

7. Условные и циклические операторы в PHP.
8. Массивы в PHP.
9. Функции PHP
10. Основы СУБД MySQL.
11. Функции PHP для работы с MySQL

Литература для самостоятельного изучения к Разделу 2

Основная литература

4. Полуэктова, Н. Р. Разработка веб-приложений : учебное пособие для вузов / Н. Р. Полуэктова. — Москва : Издательство Юрайт, 2023. — 204 с. — (Высшее образование). — ISBN 978-5-534-13715-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/519714>
5. Сысолетин, Е. Г. Разработка интернет-приложений : учебное пособие для вузов / Е. Г. Сысолетин, С. Д. Ростунцев ; под научной редакцией Л. Г. Доросинского. — Москва : Издательство Юрайт, 2023. — 90 с. — (Высшее образование). — ISBN 978-5-9916-9975-4. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/514303>
6. Тузовский, А. Ф. Проектирование и разработка web-приложений : учебное пособие для вузов / А.Ф. Тузовский. — Москва : Издательство Юрайт, 2023. — 218 с. — (Высшее образование). — ISBN 978-5-534-16300-1: — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/530767>

Дополнительная литература

5. Гаврилов, М. В. Информатика и информационные технологии : учебник для вузов / М. В. Гаврилов, В. А. Климов. — 5-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 355 с. — (Высшее образование). — ISBN 978-5-534-15819-9. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509820>
6. Грекул, В. И. Проектирование информационных систем : учебник и практикум для вузов / В. И. Грекул, Н. Л. Коровкина, Г. А. Левочкина. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-9916-8764-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511889>
7. Маркин А. В. ПРОГРАММИРОВАНИЕ НА SQL В 2 Ч. ЧАСТЬ 1 3-е изд., пер. и доп. Учебник и практикум для вузов. А. В. Маркин — Москва : Издательство Юрайт, 2023. — 429 с. — (Высшее образование). — ISBN 978-5-534-15817-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509818>
8. Проектирование информационных систем : учебник и практикум для вузов / Д. В. Чистов, П. П. Мельников, А. В. Золотарюк, Н. Б. Ничепорук ; под общей редакцией Д. В. Чистова. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 293 с. — (Высшее образование). — ISBN 978-5-534-15923-3. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/510287>

2. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ) И САМОСТОЯТЕЛЬНОЙ РАБОТЕ

Освоение обучающимся дисциплины (модуля) «Технологии возможностей и безбарьерной среды» предполагает изучение материалов дисциплины на аудиторных занятиях и в ходе самостоятельной работы. Аудиторные занятия проходят в форме лекций, семинаров, практических и лабораторных занятий. Самостоятельная работа включает разнообразный комплекс видов и форм работы обучающихся.

Для успешного освоения дисциплины (модуля) и достижения поставленных целей необходимо внимательно ознакомиться с рабочей программы дисциплины (модуля), доступной в электронной информационно-образовательной среде РГСУ.

Следует обратить внимание на списки основной и дополнительной литературы, на предлагаемые преподавателем ресурсы информационно-телекоммуникационной сети Интернет. Эта информация необходима для самостоятельной работы обучающегося.

При подготовке к аудиторным занятиям необходимо помнить особенности каждой формы его проведения.

Подготовка к учебному занятию лекционного типа заключается в следующем.

С целью обеспечения успешного обучения обучающийся должен готовиться к лекции, поскольку она является важнейшей формой организации учебного процесса, поскольку:

- знакомит с новым учебным материалом;
- разъясняет учебные элементы, трудные для понимания;
- систематизирует учебный материал;
- ориентирует в учебном процессе.

С этой целью:

- внимательно прочитайте материал предыдущей лекции;
- ознакомьтесь с учебным материалом по учебнику и учебным пособиям с темой прочитанной лекции;
- внесите дополнения к полученным ранее знаниям по теме лекции на полях лекционной тетради;
- запишите возможные вопросы, которые вы зададите лектору на лекции по материалу изученной лекции;
- постарайтесь уяснить место изучаемой темы в своей подготовке;
- узнайте тему предстоящей лекции (по тематическому плану, по информации лектора) и запишите информацию, которой вы владеете по данному вопросу.

Подготовка к занятию семинарского типа.

При подготовке и работе во время проведения занятий семинарского типа следует обратить внимание на следующие моменты: на процесс предварительной подготовки, на работу во время занятия, обработку полученных результатов, исправление полученных замечаний.

Предварительная подготовка к учебному занятию семинарского типа заключается в изучении теоретического материала в отведенное для самостоятельной работы время, ознакомление с инструктивными материалами с целью осознания задач практического занятия, техники безопасности при работе с приборами, веществами.

Работа во время проведения учебного занятия семинарского типа включает:

- консультирование студентов преподавателями и вспомогательным персоналом с целью предоставления исчерпывающей информации, необходимой для самостоятельного выполнения предложенных преподавателем задач.
- самостоятельное выполнение заданий согласно обозначенной рабочей программой дисциплины (модуля) тематики.

Самостоятельная работа.

Самостоятельная работа - планируемая учебная, учебно-исследовательская, научно-исследовательская работа студентов, выполняемая во внеаудиторное (аудиторное) время по заданию и при методическом руководстве преподавателя, но без его непосредственного участия (при частичном непосредственном участии преподавателя, оставляющем ведущую роль за работой студентов).

Для более углубленного изучения темы задания для самостоятельной работы рекомендуется выполнять параллельно с изучением данной темы. При выполнении заданий по возможности используйте наглядное представление материала.

Самостоятельная работа студентов в ВУЗе является важным видом учебной и научной деятельности студента. Самостоятельная работа студентов играет значительную роль в рейтинговой технологии обучения.

К современному специалисту в области медицины общество предъявляет достаточно широкий перечень требований, среди которых немаловажное значение имеет наличие у выпускников определенных навыков (компетенций) и умения самостоятельно добывать знания из различных источников, систематизировать полученную информацию, давать оценку конкретной профессиональной ситуации. Формирование такого умения происходит в течение всего периода обучения.

В процессе самостоятельной работы студент приобретает необходимые для будущей специальности компетенции, навыки самоорганизации, самоконтроля, самоуправления, саморефлексии и становится активным самостоятельным субъектом учебной деятельности.

Самостоятельная работа студентов должна оказывать важное влияние на формирование личности будущего специалиста, его компетентность. Каждый студент самостоятельно определяет режим своей работы и меру труда, затрачиваемого на овладение учебным содержанием по каждой дисциплине. Он выполняет внеаудиторную работу по личному индивидуальному плану, в зависимости от его подготовки, времени и других условий.

Виды самостоятельной работы.

Работа с литературой.

При работе с книгой необходимо подобрать литературу, научиться правильно ее читать, вести записи. Для подбора литературы в библиотеке используются алфавитный и систематический каталоги. Важно помнить, что рациональные навыки работы с книгой - это всегда большая экономия времени и сил. Правильный подбор учебников рекомендуется преподавателем, читающим лекционный курс. Необходимая литература может быть также указана в методических разработках по данному курсу. Изучая материал по учебнику, следует переходить к следующему вопросу только после правильного уяснения предыдущего, описывая на бумаге все выкладки и вычисления (в том числе те, которые в учебнике опущены или на лекции даны для самостоятельного вывода). При изучении любой дисциплины большую и важную роль играет самостоятельная индивидуальная работа. Особое внимание следует обратить на определение основных понятий курса. Студент должен подробно разбирать примеры, которые поясняют такие определения, и уметь строить аналогичные примеры самостоятельно. Нужно добиваться точного представления о том, что изучаешь. Полезно составлять опорные конспекты. При изучении материала по учебнику полезно в тетради (на специально отведенных полях) дополнять конспект лекций. Там же следует отмечать вопросы, выделенные студентом для консультации с преподавателем. Выводы, полученные в результате изучения, рекомендуется в конспекте выделять, чтобы они при перечитывании записей лучше запоминались. Опыт показывает, что многим студентам помогает составление листа опорных сигналов, содержащего важнейшие и наиболее часто употребляемые формулы и понятия. Такой лист помогает запомнить формулы, основные положения лекции, а также может служить постоянным справочником для студента. Различают два вида чтения: первичное и вторичное. Первичное - это внимательное, неторопливое чтение, при котором можно остановиться на трудных местах.

Задача вторичного чтения - полное усвоение смысла целого (по счету это чтение может быть и не вторым, а третьим или четвертым). Самостоятельная работа с учебниками и книгами (а также самостоятельное теоретическое исследование проблем, обозначенных преподавателем на лекциях) – это важнейшее условие формирования у себя научного способа познания.

Чтение научного текста является частью познавательной деятельности. Ее цель – извлечение из текста необходимой информации. От того на сколько осознанна читающим собственная внутренняя установка при обращении к печатному слову (найти нужные сведения, усвоить информацию полностью или частично, критически проанализировать материал и т.п.) во многом зависит эффективность осуществляемого действия.

Методические рекомендации по составлению конспекта:

1. Внимательно прочитайте текст. Уточните в справочной литературе непонятные слова. При записи не забудьте вынести справочные данные на поля конспекта;
2. Выделите главное, составьте план;
3. Кратко сформулируйте основные положения текста, отметьте аргументацию автора;
4. Законспектируйте материал, четко следуя пунктам плана. При конспектировании старайтесь выразить мысль своими словами. Записи следует вести четко, ясно.

5. Грамотно записывайте цитаты. Цитируя, учитывайте лаконичность, значимость мысли. В тексте конспекта желательно приводить не только тезисные положения, но и их доказательства. При оформлении конспекта необходимо стремиться к емкости каждого предложения. Мысли автора книги следует излагать кратко, заботясь о стиле и выразительности написанного. Число дополнительных элементов конспекта должно быть логически обоснованным, записи должны распределяться в определенной последовательности, отвечающей логической структуре произведения. Для уточнения и дополнения необходимо оставлять поля. Овладение навыками конспектирования требует от студента целеустремленности, повседневной самостоятельной работы.

Методические материалы по самостоятельному решению задач

При самостоятельном решении задач нужно обосновывать каждый этап решения, исходя из теоретических положений курса. Если студент видит несколько путей решения проблемы (задачи), то нужно сравнить их и выбрать самый рациональный. Полезно до начала вычислений составить краткий план решения проблемы (задачи). Решение проблемных задач или примеров следует излагать подробно, вычисления располагать в строгом порядке, отделяя вспомогательные вычисления от основных. Решения при необходимости нужно сопровождать комментариями, схемами, чертежами и рисунками. Следует помнить, что решение каждой учебной задачи должно доводиться до окончательного логического ответа, которого требует условие, и по возможности с выводом. Полученный ответ следует проверить способами, вытекающими из существа данной задачи. Полезно также (если возможно) решать несколькими способами и сравнить полученные результаты. Решение задач данного типа нужно продолжать до приобретения твердых навыков в их решении.

Методические материалы к выполнению реферата

Реферат (от лат. *refere* – сообщать) – краткое изложение в письменном виде или в форме публикации доклада, содержания научного труда (трудов), литературы по теме. Работа над рефератом условно разделяется на выбор темы, подбор литературы, подготовку и защиту плана; написание теоретической части и всего текста с указанием библиографических данных используемых источников, подготовку доклада, выступление с ним. Тематика рефератов полностью связана с основными вопросами изучаемого курса.

Список литературы к темам не дается, и обучающиеся самостоятельно ведут библиографический поиск, причем им не рекомендуется ограничиваться университетской библиотекой.

Важно учитывать, что написание реферата требует от обучающихся определенных усилий. Особое внимание следует уделить подбору литературы, методике ее изучения с целью отбора и обработки собранного материала, обоснованию актуальности темы и теоретического уровня обоснованности используемых в качестве примеров фактов какой-либо деятельности.

Выбрав тему реферата, начав работу над литературой, необходимо составить план. Изучая литературу, продолжается обдумывание темы, осмысливание прочитанного, делаются выписки, сопоставляются точки зрения разных авторов и т.д. Реферативная работа сводится к тому, чтобы в ней выделились две взаимосвязанные стороны: во-первых, ее следует рассматривать как учебное задание, которое должен выполнить обучаемый, а во-вторых, как форму научной работы, творческого воображения при выполнении учебного задания.

Наличие плана реферата позволяет контролировать ход работы, избежать формального переписывания текстов из первоисточников.

Оформление реферата включает титульный лист, оглавление и краткий список использованной литературы. Список использованной литературы размещается на последней странице рукописи или печатной форме реферата. Реферат выполняется в письменной или печатной форме на белых листах формата А4 (210 х 297 мм). Шрифт Times New Roman, кегель 14, через 1,5 интервала при соблюдении следующих размеров текста: верхнее поле – 25 мм, нижнее – 20 мм, левое – 30 мм, правое – 15 мм. Нумерация страниц производится вверху листа, по центру. Титульный лист нумерации не подлежит.

Рефераты должны быть написаны простым, ясным языком, без претензий на наукообразность. Следует избегать сложных грамматических оборотов, непривычных терминов и символов. Если же такие термины и символы все-таки приводятся, то необходимо разъяснять их значение при первом упоминании в тексте реферата.

Объем реферата предполагает тщательный отбор информации, необходимой для краткого изложения вопроса. Важнейший этап – редактирование готового текста реферата и подготовка к обсуждению. Обсуждение требует хорошей ориентации в материале темы, умения выделить главное, поставить дискуссионный вопрос, привлечь внимание слушателей к интересной литературе, логично и убедительно изложить свои мысли.

Рефераты обязательно подлежат защите. Процедура защиты начинается с определения оппонентов защищающего свою работу. Они стремятся дать основательный анализ работы обучающимся, обращают внимание на положительные моменты и недостатки реферата, дают общую оценку содержанию, форме преподнесения материала, характеру использованной литературы. Иногда они дополняют тот или иной раздел реферата. Последнее особенно ценно, ибо говорит о глубоком знании обучающимся-оппонентом изучаемой проблемы.

Обсуждение не ограничивается заслушиванием оппонентов. Другие обучающиеся имеют право уточнить или опровергнуть какое-либо утверждение. Преподаватель предлагает любому обучающемуся задать вопрос по существу доклада или попытаться подвести итог обсуждению.

Алгоритм работы над рефератом

1. Выбор темы

Тема должна быть сформулирована грамотно (с литературной точки зрения);

В названии реферата следует поставить четкие рамки рассмотрения темы;

Желательно избегать слишком длинных названий;

Следует по возможности воздерживаться от использования в названии спорных с научной точки зрения терминов, излишней наукообразности, а также чрезмерного упрощения формулировок.

2. Реферат следует составлять из пяти основных частей: введения; основной части; заключения; списка литературы; приложений.

3. Основные требования к введению:

Во введении не следует концентрироваться на содержании; введение должно включать краткое обоснование актуальности темы реферата, где требуется показать, почему данный вопрос может представлять научный интерес и есть ли связь представляемого материала с современностью. Таким образом, тема реферата должна быть актуальна либо с научной точки зрения, либо с современных позиций.

Очень важно выделить цель, а также задачи, которые требуется решить для выполнения цели.

Введение должно содержать краткий обзор изученной литературы, в котором указывается взятый из того или иного источника материал, кратко анализируются изученные источники, показываются их сильные и слабые стороны;

Объем введения составляет две страницы текста.

4. Требования к основной части реферата:

Основная часть содержит материал, отобранный для рассмотрения проблемы;

Также основная часть должна включать в себя собственно мнение обучающихся и сформулированные самостоятельные выводы, опирающиеся на приведенные факты;

Материал, представленный в основной части, должен быть логически изложен и распределен по параграфам, имеющим свои названия;

В изложении основной части необходимо использовать сноски (в первую очередь, когда приводятся цифры и чьи-то цитаты);

Основная часть должна содержать иллюстративный материал (графики, таблицы и т. д.);

Объем основной части составляет около 10 страниц.

5. Требования к заключению:

В заключении формулируются выводы по параграфам, обращается внимание на выдвинутые во введении задачи и цели;

Заключение должно быть четким, кратким, вытекающим из содержания основной части.

6. Требования к оформлению списка литературы (по ГОСТу):

Необходимо соблюдать правильность последовательности записи источников: сначала следует писать фамилию, а после инициалы; название работы не ставится в кавычки; после названия сокращенно пишется место издания; затем идет год издания; наконец, называется процитированная страница.

Критерии оценки реферата

Обучающийся, защищающий реферат, должен рассказать о его актуальности, поставленных целях и задачах, изученной литературе, структуре основной части, сделанных в ходе работы выводах.

По окончанию выступления ему может быть задано несколько вопросов по представленной проблеме.

Оценка складывается из соблюдения требований к реферату, грамотного раскрытия темы, умения четко рассказывать о представленном реферате, способности понять суть задаваемых по работе вопросов и найти точные ответы на них.

Методические материалы к выполнению эссе

Эссе – литературное произведение небольшого объема, обычно прозаическое, свободной композиции, передающее индивидуальные впечатления, суждения, соображения автора о той или иной проблеме, теме, о том или ином событии или явлении. Это вид самостоятельной исследовательской работы обучающихся, с целью углубления и закрепления теоретических знаний и освоения практических навыков. Цель эссе состоит в развитии самостоятельного творческого мышления и письменного изложения собственных мыслей. При написании эссе обучающийся должен представить развернутый письменный ответ на теоретический или практический актуальный вопрос, объявленный преподавателем в аудитории непосредственно перед ее написанием. В процессе написания эссе разрешается пользоваться нормативно-правовыми актами, конспектом лекций (в печатном виде). Использование интернет-ресурсов не допускается. Темы эссе преподаватель предлагает из числа тех, которые обучающиеся уже рассматривали на лекциях или семинарских занятиях, исходя из содержания заданий в составе оценочных средств. По решению преподавателя, в качестве темы эссе может быть выбрана одна или несколько тем, которые могут быть распределены между обучающимися по желанию.

Требования к выполнению эссе:

1. Проводится письменно.

2. Эссе выполняется на компьютере (гарнитура Times New Roman, шрифт 14) через 1,5 интервала с полями: верхнее, нижнее – 2; правое – 3; левое – 1,5. Отступ первой строки абзаца – 1,25. Сноски – постраничные. Таблицы и рисунки встраиваются в текст работы. При этом обязательный заголовок таблицы надо размещать над табличным полем, а рисунки сопровождать подрисуночными подписями. При включении в эссе нескольких таблиц и/или рисунков их нумерация обязательна. Обязательна и нумерация страниц. Их целесообразно проставлять внизу страницы – по середине или в правом углу. Номер страницы не ставится на титульном листе, но в общее число страниц он включается. Объем эссе, без учета

приложений, не должен превышать 5 страниц. Значительное превышение установленного объема является недостатком работы и указывает на то, что обучающийся не сумел отобрать и переработать необходимый материал.

3. Работа должна содержать собственные умозаключения по сути поставленной проблемы, включать самостоятельно проведенный анализ по сути этой проблемы, выводы, обобщающие авторскую позицию по поставленной проблеме.

Критерии оценки эссе:

«Отлично» – исключительные знания материала, абсолютное понимание сути, безукоризненное знание основных понятий и положений, логически и лексически грамотно изложенный, содержательный, аргументированный, конкретный и исчерпывающий ответ.

«Хорошо» – глубокие знания материала, правильное понимание сути, знание основных понятий и положений, содержательный, полный и конкретный ответ.

«Удовлетворительно» – твердые, но недостаточно полные знания, верное понимание сути, в целом правильный ответ.

«Неудовлетворительно» – непонимание сущности задания, грубые ошибки в ответе.

Методические материалы по выполнению тестирования.

Тестовые задания содержат вопросы и 3-4 варианта ответа по базовым положениям изучаемой темы/раздела, составлены с расчетом на знания, полученные обучающимся в процессе изучения темы/раздела.

Тестовые задания выполняются в письменной форме и сдаются преподавателю, ведущему дисциплину (модуль). На выполнение тестовых заданий обучающимся отводится 45 минут.

При обработке результатов оценочной процедуры используются: критерии оценки по содержанию и качеству полученных ответов, ключи, оценочные листы.

Критерии оценки теста:

«Зачтено» - если обучающийся правильно выполнил не менее 2/3 всей работы или допустил не более одной грубой ошибки и двух недочетов, не более одной грубой и одной негрубой ошибки, не более трех негрубых ошибок, одной негрубой ошибки и трех недочетов, при наличии четырех-пяти недочетов.

«Не зачтено» - если число ошибок и недочетов превысило норму для оценки 3 или правильно выполнено менее 2/3 всей работы.

Методические материалы по выполнению доклада.

Рекомендуется следующая структура доклада:

1. титульный лист, содержание доклада;
2. краткое изложение;
3. цели и задачи;
4. изложение характера исследований и рассмотренных проблем, гипотезы, спорные вопросы;
5. источники информации, методы сбора и анализа данных, степень их полноты и достоверности;
6. анализ и толкование полученных в работе результатов;
7. выводы и оценки;
8. библиография и приложения.

Время выступления докладчика не должно превышать 10 минут.

Основные требования к оформлению доклада:

- титульный лист должен включать название доклада, наименование предметной (цикловой) комиссии, фамилию обучающегося;
- все использованные литературные источники сопровождаются библиографическим описанием;
- приводимая цитата из источника берется в кавычки (оформляются сноски);
- единицы измерения должны применяться в соответствии с действующими стандартами;

- все названия литературных источников следует приводить в соответствии с новейшими изданиями;
- рекомендуется включение таблиц, графиков, схем, если они отражают основное содержание или улучшают ее наглядность;
- названия фирм, учреждений, организаций и предприятий должны именоваться так, как они указываются в источнике;

Критерии оценки доклада

При выполнении доклада обучающийся должен продемонстрировать умение кратко излагать прочитанный материал, а также умение обобщать и анализировать материал по теме доклада.

Презентация

Методические материалы к презентациям

1. Объём презентации 10 -20 слайдов.
2. На титульном слайде должно быть отражено:
 - наименование факультета;
 - тема презентации;
 - фамилия, имя, отчество, направление подготовки/ специальность, направленность (профиль)/ специализация, форма обучения, номер группы автора презентации;
 - фамилия, имя, отчество, степень, звание, должность руководитель работы;
 - год выполнения работы.
3. В презентации должны быть отражены обоснование актуальности представляемого материала, цели и задачи работы.
4. Содержание презентации должно включать наиболее значимый материал доклада, а также, при необходимости, таблицы, диаграммы, рисунки, фотографии, карты, видео – вставки, звуковое сопровождение.
5. Заключительный слайд должен содержать информацию об источниках информации для презентации.

Критерии оценки презентации

1. Объём презентации 10 -20 слайдов.
2. Правильность оформления титульного слайда.
3. Актуальность отобранного материала, обоснованность формулировки цели и задач работы.
4. Наглядность и логичность презентации, обоснованность использования таблиц, диаграмм, рисунков, фотографий, карт, видео – вставок, звукового сопровождения; правильный выбор шрифтов, фона, других элементов дизайна слайда.
5. Объём и качество источников информации (не менее 2-х интернет – источников и не менее 2-х литературных источников).

Методические материалы по подготовке к опросу

Самостоятельная работа обучающихся включает подготовку к опросу на практических занятиях. Для этого обучающийся изучает лекции, основную и дополнительную литературу, публикации, информацию из Интернет-ресурсов.

Тема и вопросы к практическим занятиям, вопросы для самоконтроля содержатся в рабочей программе и доводятся до студентов заранее.

Для подготовки к опросу обучающемуся необходимо ознакомиться с материалом, посвященным теме практического занятия, в учебнике или другой рекомендованной литературе, конспекте лекции, обратить внимание на усвоение основных понятий дисциплины, выявить неясные вопросы и подобрать дополнительную литературу для их освещения.

Критерии оценки опроса

«Отлично»:

- дан полный, развернутый ответ на поставленный вопрос;

- в ответе прослеживается четкая структура, логическая последовательность, отражающая сущность раскрываемых понятий, теорий, явлений;
- знание по предмету демонстрируются на фоне понимания его в системе данной науки и междисциплинарных связей;

- свободное владение терминологией;

- ответы на дополнительные вопросы четкие, краткие;

«Хорошо»:

- дан полный, развернутый ответ на поставленный вопрос, показано умение выделять существенные и несущественные признаки, причинно-следственные связи;

- ответ недостаточно логичен с единичными ошибками в частностях, исправленные студентом с помощью преподавателя;

- единичные ошибки в терминологии;

- ответы на дополнительные вопросы правильные, недостаточно полные и четкие.

«Удовлетворительно»:

- ответ не полный, с ошибками в деталях, умение раскрыть значение обобщённых знаний не показано, речевое оформление требует поправок, коррекции;

- логика и последовательность изложения имеют нарушения, студент не способен самостоятельно выделить существенные и несущественные признаки и причинно-следственные связи;

- ошибки в раскрываемых понятиях, терминах;

- студент не ориентируется в теме, допускает серьезные ошибки;

- студент не может ответить на большую часть дополнительных вопросов.

«Неудовлетворительно»:

- ответ представляет собой разрозненные знания с существенными ошибками по вопросу;

- присутствуют фрагментарность, нелогичность изложения, студент не осознает связь обсуждаемого вопроса с другими объектами дисциплины, речь неграмотная;

- незнание терминологии;

- ответы на дополнительные вопросы неправильные.

Методические материалы по выполнению практического задания

При выполнении практического задания обучающийся придерживается следующего алгоритма:

1. Записать дату, тему и цель задания;

2. Ознакомиться с правилами и условия выполнения практического задания;

3. Повторить теоретические задания, необходимые для рациональной работы и других практических действий, используя конспекты лекций и рекомендованную литературу, представленной в программе;

4. Выполнить работу по предложенному алгоритму действий;

5. Обобщить результаты работы, сформулировать выводы / дать ответы на контрольные вопросы;

Работа должна быть выполнена грамотно, с соблюдением культуры изложения. При использовании данных из учебных, методических пособий и другой литературы, периодических изданий, Интернет-источников должны иметься ссылки на вышеперечисленные.

Критерии оценки практического задания:

«Отлично» – правильный ответ, дается четкое обоснование принятому решению; рассуждения четкие последовательные логические; используются ссылки на полученные при изучении дисциплины знания; правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Хорошо» – правильный ответ, дается обоснование принятому решению; но с не существенными ошибками, в рассуждениях отсутствует логическая последовательность; используются ссылки на полученные при изучении дисциплины знания, правильно

используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Удовлетворительно» – правильный ответ, допускаются грубые ошибки в обосновании принятого решения; рассуждения не последовательные сумбурные; используются ссылки на полученные при изучении дисциплины знания; используются формулы, процедуры, понятия, имеющие прямое значение для подтверждения принятого решения, однако, при обращении к ним допускаются серьезные ошибки, студент не может правильно ими воспользоваться.

«Неудовлетворительно, не зачтено» – ответ неверный, отсутствует обоснование принятому решению; студент демонстрирует полное непонимание сути вопроса.

Для оценки решения ситуационной задачи (аналитического задания):

Оценка «отлично» выставляется, если задача решена грамотно, ответы на вопросы сформулированы четко. Эталонный ответ полностью соответствует решению студента, которое хорошо обосновано теоретически.

Оценка «хорошо» выставляется, если задача решена, ответы на вопросы сформулированы не достаточно четко. Решение студента в целом соответствует эталонному ответу, но не достаточно хорошо обосновано теоретически.

Оценка «удовлетворительно» выставляется, если задача решена не полностью, ответы не содержат всех необходимых обоснований решения.

Оценка «неудовлетворительно» выставляется, если задача не решена или имеет грубые теоретические ошибки в ответе на поставленные вопросы.

Методические материалы по выполнению лабораторного задания

При выполнении лабораторного задания обучающийся придерживается следующего алгоритма:

1. Записать дату, тему и цель задания;
2. Ознакомиться с правилами и условия выполнения задания;
3. Повторить теоретические задания, необходимые для рациональной работы и других практических действий, используя конспекты лекций и рекомендованную литературу, представленной в программе;
4. Выполнить работу по предложенному алгоритму действий;
5. Обобщить результаты работы, сформулировать выводы / дать ответы на контрольные вопросы;

Работа должна быть выполнена грамотно, с соблюдением культуры изложения. При использовании данных из учебных, методических пособий и другой литературы, периодических изданий, Интернет-источников должны иметься ссылки на вышеперечисленные.

Критерии оценки лабораторного задания:

«Отлично» – правильный ответ, дается четкое обоснование принятому решению; рассуждения четкие последовательные логические; используются ссылки на полученные при изучении дисциплины знания; правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Хорошо» – правильный ответ, дается обоснование принятому решению; но с не существенными ошибками, в рассуждениях отсутствует логическая последовательность; используются ссылки на полученные при изучении дисциплины знания, правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Удовлетворительно» – правильный ответ, допускаются грубые ошибки в обосновании принятого решения; рассуждения не последовательные сумбурные; используются ссылки на полученные при изучении дисциплины знания; используются формулы, процедуры, понятия, имеющие прямое значение для подтверждения принятого решения, однако, при обращении к ним допускаются серьезные ошибки, студент не может правильно ими воспользоваться.

«Неудовлетворительно, не зачтено» – ответ неверный, отсутствует обоснование принятому решению; студент демонстрирует полное непонимание сути вопроса.

Методические указания для подготовки к промежуточной аттестации.

Изучение учебных дисциплин (модулей) завершается зачетом/зачетом с оценкой или экзаменом. Подготовка к промежуточной аттестации способствует закреплению, углублению

и обобщению знаний, получаемых, в процессе обучения, а также применению их к решению практических задач. Готовясь к экзамену, студент ликвидирует имеющиеся пробелы в знаниях, углубляет, систематизирует и упорядочивает свои знания. На зачете или экзамене студент демонстрирует то, что он освоил в процессе обучения по дисциплине (модулю).

Вначале следует просмотреть весь материал по дисциплине (модулю), отметить для себя трудные вопросы. Обязательно в них разобраться. В заключение еще раз целесообразно повторить основные положения, используя при этом листы опорных сигналов. Систематическая подготовка к занятиям в течение семестра позволит использовать время промежуточной аттестации для систематизации знаний.

3. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНИВАНИЯ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ И (ИЛИ) ОПЫТА ДЕЯТЕЛЬНОСТИ, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ

3.1. Организационные основы применения балльно-рейтинговой системы оценки успеваемости обучающихся по дисциплине (модулю)

Оценка качества освоения обучающимися дисциплины (модуля) реализуется в формате балльно-рейтинговой системы оценки успеваемости обучающихся (БРСО).

БРСО в ходе текущего контроля успеваемости и промежуточной аттестации осуществляется по 100-балльной шкале.

Академический рейтинг обучающегося по дисциплине (модулю) складывается из результатов:

- текущего контроля успеваемости (максимальный текущий рейтинг обучающегося 80 рейтинговых баллов);
- промежуточной аттестации (максимальный рубежный рейтинг обучающегося 20 рейтинговых баллов).

Условия оценки освоения обучающимся дисциплины (модуля) в формате БРСО доводятся преподавателем до сведения обучающихся на первом учебном занятии, а также размещены в свободном доступе в электронной информационно-образовательной среде Университета.

3.2. Проведение текущего контроля успеваемости обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося

В течение учебного семестра до промежуточной аттестации на основании утвержденной рабочей программы дисциплины (модуля) формируется текущий рейтинг обучающегося. Текущий рейтинг обучающегося складывается как сумма рейтинговых баллов, полученных им в течение учебного семестра по всем видам учебных занятий по учебной дисциплине.

В процессе текущего контроля оцениваются следующие действия обучающегося, направленные на освоение компетенций в рамках изучения учебной дисциплины:

- академическая активность (посещаемость учебных занятий, самостоятельное изучение содержания учебной дисциплины в электронной информационно-образовательной среде, соблюдение сроков сдачи практических заданий и текущих контрольных мероприятий и др.);
- выполнение и сдача текущих и итогового практических заданий (эссе, рефераты, творческие задания, кейс-задания, лабораторные работы, расчетные задания и др., активное участие в групповых интерактивных занятиях (дискуссии, Wiki-проекты и др.), защита проектов и др.);
- прохождение рубежей текущего контроля, включая соблюдение графика их прохождения в электронной информационно-образовательной среде.

Для планирования расчета текущего рейтинга обучающегося используются следующие пропорции:

Вид учебного действия	Максимальная рейтинговая оценка, баллов
академическая активность	10
практические задания	40
<i>из них: текущие практические задания</i>	20
<i>итоговое практическое задание</i>	20
рубежи текущего контроля	30
ИТОГО:	80

В течение учебного семестра по дисциплине (модулю) обучающимся должен быть накоплен текущий рейтинг не менее 52 рейтинговых баллов (65% от максимального значения текущего рейтинга).

Необходимыми условиями допуска обучающегося к промежуточной аттестации по дисциплине являются положительное прохождение обучающимся не менее 65% рубежей текущего контроля с накоплением не менее 65% максимального рейтингового балла за каждый рубеж текущего контроля и положительное выполнение итогового практического задания с накоплением не менее 65% максимального рейтингового балла, установленного за итоговое практическое задание.

Невыполнение вышеуказанных условий является текущей академической задолженностью, которая должна быть ликвидирована обучающимся до контрольного мероприятия промежуточной аттестации.

Сведения о наличии у обучающихся текущей академической задолженности, сроках и порядке добора рейтинговых баллов для её ликвидации доводятся до обучающихся педагогическим работником.

В случае неликвидации текущей академической задолженности, педагогический работник обязан во время контрольного мероприятия промежуточной аттестации поставить обучающемуся 0 рейтинговых баллов. В этом случае ликвидация текущей академической задолженности возможна в периоды проведения повторной промежуточной аттестации.

3.3. Проведение промежуточной аттестации обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося

Промежуточная аттестация по дисциплине (модулю) проводится в соответствии с Положением о промежуточной аттестации обучающихся по основным профессиональным образовательным программам высшего образования – программ специалитета в Российском государственном социальном университете и Положением о балльно-рейтинговой системе оценки успеваемости обучающихся по основным профессиональным образовательным программам высшего образования - программам бакалавриата, программам специалитета, программам магистратуры в Российском государственном социальном университете.

На промежуточную аттестацию отводится 20 рейтинговых баллов.

Ответы обучающегося на контрольном мероприятии промежуточной аттестации оцениваются педагогическим работником по 20 - балльной шкале, а итоговая оценка по дисциплине (модулю) выставляется по пятибалльной системе.

Критерии выставления оценки определяются Положением о балльно-рейтинговой системе оценки успеваемости обучающихся по основным профессиональным образовательным программам высшего образования - программам бакалавриата, программам специалитета, программам магистратуры в Российском государственном социальном университете.

В процессе определения рубежного рейтинга обучающегося используется следующая шкала:

Рубежный рейтинг	Критерии оценки освоения обучающимся учебной дисциплины в ходе контрольных мероприятий промежуточной аттестации
19-20 рейтинговых баллов	обучающийся глубоко и прочно усвоил программный материал, исчерпывающе, последовательно, грамотно и логически стройно его излагает, тесно увязывает с задачами и будущей деятельностью, не затрудняется с ответом при видоизменении задания, свободно справляется с задачами и практическими заданиями, правильно обосновывает принятые решения, умеет самостоятельно обобщать и излагать материал, не допуская
16-18 рейтинговых баллов	обучающийся твердо знает программный материал, грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос, может правильно применять теоретические положения и владеет необходимыми умениями и навыками при выполнении практических
13-15 рейтинговых баллов	обучающийся освоил основной материал, но не знает отдельных деталей, допускает неточности, недостаточно правильные формулировки, нарушает последовательность в изложении программного материала и испытывает затруднения в выполнении практических заданий
1-12 рейтинговых баллов	обучающийся не знает значительной части программного материала, допускает существенные ошибки, с большими затруднениями выполняет практические задания
0 рейтинговых баллов	не аттестован

Если результат контроля успеваемости в рамках проведения контрольных мероприятий промежуточной аттестации (рубежный рейтинг обучающегося) неудовлетворительный (получено менее 13 рейтинговых баллов), то промежуточная аттестация по учебной дисциплине (модулю) невозможна даже при наличии высокого текущего рейтинга, полученного по итогам текущего контроля по учебной дисциплине (модулю).

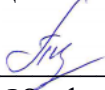
ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

№ п/п	Содержание изменения	Реквизиты документа об утверждении изменения	Дата введения изменения
1.			
2.			
3.			
4.			



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Российский государственный социальный университет»

УТВЕРЖДАЮ
Декана факультета политических и
социальных технологий

 / С.В. Пивнева
«28» февраля 2024 г.

МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ
«ПРОЕКТИРОВАНИЕ И АДМИНИСТРИРОВАНИЕ ИНФОРМАЦИОННЫХ
СИСТЕМ»

Направление подготовки
«Информационные системы и технологии»

Направленность
«Интеллектуальные информационные системы и технологии»

ОБРАЗОВАТЕЛЬНАЯ ПРОГРАММА ВЫСШЕГО ОБРАЗОВАНИЯ - ПРОГРАММА
БАКАЛАВРИАТА

Форма обучения

Очная, заочная

Москва, 2024 г.

Методические материалы по дисциплине (модулю) «Проектирование и администрирование информационных систем» разработаны на основании федерального государственного образовательного стандарта высшего образования – *бакалавриата* по направлению подготовки *09.03.02 Информационные системы и технологии*, утвержденного приказом Министерства образования и науки Российской Федерации от 19.09.2017 № 926, учебного плана по основной профессиональной образовательной программе высшего образования – программе бакалавриата по направлению подготовки *09.03.02 Информационные системы и технологии* (далее – «ОПОП»).

Методические материалы по дисциплине (модулю) разработаны канд. техн. наук, доцентом Симоновым В.Л..

Методические материалы по дисциплине (модулю) обсуждены и утверждены на заседании кафедры информационных технологий, искусственного интеллекта и общественно-социальных технологий цифрового общества факультета социальных и политических технологий.

Протокол № 11 от «28» февраля 2024 г.

Заведующий кафедрой
канд. пед. наук, доцент


_____ С.В. Пивнева
(подпись)

СОДЕРЖАНИЕ

1. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ К ЛЕКЦИОННЫМ И ЛАБОРАТОРНЫМ ЗАНЯТИЯМ	4
1.1. Методические материалы к проведению лекционных занятий по дисциплине (модулю)	4
1.2. Методические материалы по подготовке к лабораторным занятиям по дисциплине (модулю)	6
2. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ) И САМОСТОЯТЕЛЬНОЙ РАБОТЕ	17
3. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНИВАНИЯ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ И (ИЛИ) ОПЫТА ДЕЯТЕЛЬНОСТИ, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ	26
3.1. Организационные основы применения балльно-рейтинговой системы оценки успеваемости обучающихся по дисциплине (модулю)	27
3.2. Проведение текущего контроля успеваемости обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося	27
3.3. Проведение промежуточной аттестации обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося	28
ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ	30

1. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ К ЛЕКЦИОННЫМ И ЛАБОРАТОРНЫМ ЗАНЯТИЯМ

1.1. Методические материалы к проведению лекционных занятий по дисциплине (модулю)

Лекция - один из методов обучения, одна из основных системообразующих форм организации учебного процесса в вузе. Лекционное занятие представляет собой систематическое, последовательное, монологическое изложение педагогическим работником учебного материала, как правило, теоретического характера. Такое занятие представляет собой элемент технологии представления учебного материала путем логически стройного, систематически последовательного и ясного изложения.

Цель лекции – организация целенаправленной познавательной деятельности обучающихся по овладению программным материалом дисциплины (модуля). Чтение курса лекций позволяет дать связанное, последовательное изложение материала в соответствии с новейшими данными науки, сообщить слушателям основное содержание предмета в целостном, систематизированном виде. В ряде случаев лекция выполняет функцию основного источника информации: при отсутствии учебников и учебных пособий, чаще по новым курсам; в случае, когда новые научные данные по той или иной теме не нашли отражения в учебниках; отдельные разделы и темы очень сложны для самостоятельного изучения. В таких случаях только лектор может методически помочь обучающимся в освоении сложного материала.

Возможные формы проведения лекций:

- Вводная лекция – один из наиболее важных и трудных видов лекции при чтении систематических курсов. От успеха этой лекции во многом зависит успех усвоения всего курса. Она может содержать: определение дисциплины (модуля); краткую историческую справку о дисциплине (модуле); цели и задачи дисциплины (модуля), ее роль в общей системе обучения и связь со смежными дисциплинами (модулями); основные проблемы (понятия и определения) данной науки; основную и дополнительную учебную литературу; особенности самостоятельной работы обучающихся над дисциплиной (модулем) и формы участия в научно-исследовательской работе; отчетность по курсу.

- Информационная лекция ориентирована на изложение и объяснение обучающимся научной информации, подлежащей осмыслению и запоминанию. Это самый традиционный тип лекций в практике высшей школы.

- Заключительная лекция предназначена для обобщения полученных знаний и раскрытия перспектив дальнейшего развития данной науки.

- Обзорная лекция – это систематизация научных знаний на высоком уровне, допускающая большое число ассоциативных связей в процессе осмысления информации, излагаемой при раскрытии внутрипредметной и междпредметной связей, исключая детализацию и конкретизацию. Как правило, стержень излагаемых теоретических положений составляет научно-понятийная и концептуальная основа всего курса или крупных его разделов.

- Лекция-беседа - непосредственный контакт педагогического работника с аудиторией - диалог. По ходу лекции педагогический работник задает вопросы для выяснения мнений и уровня осведомленности обучающихся по рассматриваемой проблеме.

- Лекция-дискуссия - свободный обмен мнениями в ходе изложения лекционного материала. Педагогический работник активизирует участие в обсуждении отдельными вопросами, сопоставляет между собой различные мнения и тем самым развивает дискуссию, стремясь направить ее в нужное русло.

- Лекция с применением обратной связи включает в себе то, что в начале и конце каждого раздела лекции задаются вопросы. Первый - для того, чтобы узнать, насколько обучающиеся ориентируются в излагаемом материале, вопрос в конце раздела предназначен для выяснения степени усвоения только что изложенного материала. При

неудовлетворительных результатах контрольного опроса педагогический работник возвращается к уже прочитанному разделу, изменив при этом методику подачи материала.

- Проблемная лекция опирается на логику последовательно моделируемых проблемных ситуаций путем постановки проблемных вопросов или предъявления проблемных задач. Проблемный вопрос - это диалектическое противоречие, требующее для своего решения размышления, сравнения, поиска, приобретения и применения новых знаний. Проблемная задача содержит дополнительную вводную информацию и при необходимости некоторые ориентиры поиска ее решения.

- Программированная лекция - консультация – педагогический работник сам составляет и предлагает обучающимся вопросы. На подготовленные вопросы педагогический работник сначала просит ответить обучающихся, а затем проводит анализ и обсуждение неправильных ответов. В лекциях можно использовать наглядные материалы, а также подготовить презентацию. Что касается презентации, то в качестве визуальной поддержки ее можно органично интегрировать во все вышеупомянутые лекции. В то же время лекцию-презентацию возможно выделить и в качестве самостоятельной формы. Лекция-презентация должна отражать суть основных и (или) проблемных вопросов лекции, на которые особо следует обратить внимание обучающихся. В условиях применения активного метода проведения занятий презентация представляется весьма удачным способом донесения информации до слушателей. Единственное, на что следует обратить внимание при подготовке слайдов, - это их оформление и текст. Слайд не должен быть перегружен картинками и лишней информацией, которая будет отвлекать от основного аспекта того или иного вопроса лекции. Во время лекции можно задавать вопросы аудитории в отношении того или иного слайда, тем самым еще больше вовлекая обучающихся в проблематику.

Краткое содержание лекционных занятий

Наименование разделов	Содержание учебного материала
РАЗДЕЛ 1. Системы; информационные системы; процессы в информационной системе	Понятие системы. Понятие информационной системы. Кибернетическое представление информационной системы. Этапы развития информационных систем. Процессы в информационной системе.
РАЗДЕЛ 2. Роль структуры управления при проектировании информационных систем. Управление, уровни управления	Задачи при создании и использовании информационной системы в любой организации. Структура управления организации. Содержание управленческих функций. Пирамида уровней управления. Прочие элементы организации.
РАЗДЕЛ 3. Обеспечивающие подсистемы информационных систем	Техническое, математическое, программное, алгоритмическое, информационное, организационное, правовое обеспечения.
РАЗДЕЛ 4. Инструментарий проектирования информационных систем	Типы информационных систем. Информационные системы в фирме. Инструментарий для проектирования и администрирования информационных систем.

РАЗДЕЛ 5. Понятие информационно-измерительных систем	Состав и назначение информационно-измерительных систем. Информационные средства. Средства измерений. Вспомогательные технические средства. Преобразование информации.
РАЗДЕЛ 6. Проектирование информационно-измерительных систем	Получение и преобразование измерительной информации; измерительные, вычислительные, иные вспомогательные технические средства; информация в требуемом для потребителя виде; автоматическое осуществление логических функций контроля, диагностики, идентификации.
РАЗДЕЛ 7. Управление объектами	Оценка характеристик объекта для внедрения информационно-измерительной системы. Алгоритмы и законы управления объектами различных типов. Оценка качества управления объектом. Примеры применения различных законов управления для различных объектов.
РАЗДЕЛ 8. Документация	Изучение ГОСТ ЕСКД, а именно: - ГОСТ 15.016-2016 «Система разработки и постановки продукции на производство. Техническое задание. Требования к содержанию и оформлению». - ГОСТ 2.114-2016 «Межгосударственный стандарт. Единая система конструкторской документации. Технические условия». - ГОСТ 2.120-2013 «Единая система конструкторской документации. "Технический проект"».

1.2. Методические материалы по подготовке к лабораторным занятиям по дисциплине (модулю)

Лабораторные занятия - одна из форм учебного занятия, направленная на развитие самостоятельности обучающихся и приобретение умений и навыков. Данные учебные занятия углубляют, расширяют, детализируют полученные ранее знания. Лабораторное занятие предполагает выполнение обучающимися по заданию и под руководством преподавателей одной или нескольких работ.

Цель лабораторных занятий состоит в развитии познавательных способностей, самостоятельности мышления и творческой активности обучающихся; углублении, расширении, детализировании знаний, полученных на лекции в обобщенной форме, и содействии выработке навыков профессиональной деятельности. В отдельных случаях на лабораторных занятиях руководителем занятия сообщаются дополнительные знания.

Для достижения поставленных целей и решения требуемого перечня задач лабораторные занятия проводятся традиционными технологиями или с использованием активных и интерактивных образовательных технологий.

При подготовке и работе во время проведения лабораторных занятий следует обратить внимание на следующие моменты: на процесс предварительной подготовки, на работу во время занятия, обработку полученных результатов, исправление полученных замечаний.

Предварительная подготовка к лабораторному занятию заключается в изучении теоретического материала в отведенное для самостоятельной работы время, ознакомление с инструктивными материалами с целью осознания задач занятия, техники безопасности при работе с приборами, веществами.

Работа во время проведения занятия включает:

- консультирование студентов преподавателями и вспомогательным персоналом с целью предоставления исчерпывающей информации, необходимой для самостоятельного выполнения предложенных преподавателем задач.

- самостоятельное выполнение заданий согласно обозначенной рабочей программой дисциплины (модуля) тематики.

Задания для самостоятельной работы к Разделу 1 (Модуль 1)

Вопросы для самостоятельной работы к Разделу 1 (Модуль 1)

1. Понятие «Информационные системы». Определение, назначение.
2. Понятие «Информационные технологии». Определение, назначение.
3. Основоположники фундаментальных исследований в области информационных систем.
4. Что такое кибернетика? Основоположники создания кибернетики.
5. Что является общим и различным для систем, функционирующих в различных областях - технической, биологической, социальной. Используйте определение «кибернетика».

Перечень тем лабораторных работ к Разделу 1 (Модуль 1):

Темы лабораторных работ соответствуют вопросам для самостоятельной работы (см. выше).

Литература для самостоятельного изучения к Разделу 1 (Модуль 1).

Основная литература

1. Грекул, В. И. Проектирование информационных систем : учебник и практикум для вузов / В. И. Грекул, Н. Л. Коровкина, Г. А. Левочкина. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-9916-8764-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511889> (дата обращения: 08.03.2023).

2. Программирование, тестирование, проектирование, нейросети, технологии аппаратно-программных средств (практические задания и способы их решения) : учебник : [16+] / С. В. Веретехина, К. С. Кармицкий, Д. Д. Лукашин [и др.]. — Москва : Директ-Медиа, 2022. — 144 с. : ил., табл. — Режим доступа: по подписке. — URL: <https://biblioclub.ru/index.php?page=book&id=694782> (дата обращения: 08.03.2023). — Библиогр. в кн. — ISBN 978-5-4499-3321-8. — Текст : электронный.

3. Маркин, А. В. Программирование на SQL в 2 ч. Часть 1 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 429 с. — (Высшее образование). — ISBN 978-5-534-15817-5. — Текст : электронный //

Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509818> (дата обращения: 08.03.2023).

4. Маркин, А. В. Программирование на SQL в 2 ч. Часть 2 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., испр. и доп. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-534-15818-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509819> (дата обращения: 08.03.2023).

Дополнительная литература

1. Зараменских, Е. П. Управление жизненным циклом информационных систем : учебник и практикум для вузов / Е. П. Зараменских. — 2-е изд. — Москва : Издательство Юрайт, 2023. — 497 с. — (Высшее образование). — ISBN 978-5-534-14023-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511960> (дата обращения: 08.03.2023).

2. Стружкин, Н. П. Базы данных: проектирование. Практикум : учебное пособие для вузов / Н. П. Стружкин, В. В. Годин. — Москва : Издательство Юрайт, 2023. — 291 с. — (Высшее образование). — ISBN 978-5-534-00739-8. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/512160> (дата обращения: 08.03.2023).

3. Чистякова, М. А. Проектирование и эксплуатация баз данных : учебно-методическое пособие / М. А. Чистякова, И. А. Иванова, И. Д. Котилевец. — Москва : РТУ МИРЭА, 2021. — 112 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/176572> (дата обращения: 08.03.2023). — Режим доступа: для авториз. пользователей.

Задания для самостоятельной работы к Разделу 2 (Модуль 1)

Вопросы для самостоятельной работы к Разделу 2 (Модуль 1)

1. Что является «системой»? Приведите определение или ваше понимание.
2. Приведите примеры систем. Назовите составные части и охарактеризуйте их связи. Приведите примеры объектов, не являющихся системами, дайте пояснение.
3. Что такое «объект управления» в системе? Приведите примеры в технических, биологических, социальных системах.
4. Что такое «сенсоры» в системе? Какую они играют роль? Приведите примеры в технических, биологических, социальных системах.
5. Что такое «обратные связи» в системе? Какую они играют роль? Что такое положительные и отрицательные обратные связи? Что такое «Обратная связь по перемещению», «Обратная связь по скорости», «Обратная связь по ускорению»? Приведите примеры в технических, биологических, социальных системах.

Перечень тем лабораторных работ к Разделу 2 (Модуль 1):

Темы лабораторных работ соответствуют вопросам для самостоятельной работы (см. выше).

Литература для самостоятельного изучения к Разделу 2 (Модуль 1).

Основная литература

1. Грекул, В. И. Проектирование информационных систем : учебник и практикум для вузов / В. И. Грекул, Н. Л. Коровкина, Г. А. Левочкина. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-9916-8764-5. — Текст : электронный

// Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511889> (дата обращения: 08.03.2023).

2. Программирование, тестирование, проектирование, нейросети, технологии аппаратно-программных средств (практические задания и способы их решения) : учебник : [16+] / С. В. Веретехина, К. С. Кармицкий, Д. Д. Лукашин [и др.]. — Москва : Директ-Медиа, 2022. — 144 с. : ил., табл. — Режим доступа: по подписке. — URL: <https://biblioclub.ru/index.php?page=book&id=694782> (дата обращения: 08.03.2023). — Библиогр. в кн. — ISBN 978-5-4499-3321-8. — Текст : электронный.

3. Маркин, А. В. Программирование на SQL в 2 ч. Часть 1 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 429 с. — (Высшее образование). — ISBN 978-5-534-15817-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509818> (дата обращения: 08.03.2023).

4. Маркин, А. В. Программирование на SQL в 2 ч. Часть 2 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., испр. и доп. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-534-15818-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509819> (дата обращения: 08.03.2023).

Дополнительная литература

1. Зараменских, Е. П. Управление жизненным циклом информационных систем : учебник и практикум для вузов / Е. П. Зараменских. — 2-е изд. — Москва : Издательство Юрайт, 2023. — 497 с. — (Высшее образование). — ISBN 978-5-534-14023-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511960> (дата обращения: 08.03.2023).

2. Стружкин, Н. П. Базы данных: проектирование. Практикум : учебное пособие для вузов / Н. П. Стружкин, В. В. Годин. — Москва : Издательство Юрайт, 2023. — 291 с. — (Высшее образование). — ISBN 978-5-534-00739-8. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/512160> (дата обращения: 08.03.2023).

3. Чистякова, М. А. Проектирование и эксплуатация баз данных : учебно-методическое пособие / М. А. Чистякова, И. А. Иванова, И. Д. Котилевец. — Москва : РТУ МИРЭА, 2021. — 112 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/176572> (дата обращения: 08.03.2023). — Режим доступа: для авториз. пользователей.

Задания для самостоятельной работы к Разделу 3 (Модуль 1)

Вопросы для самостоятельной работы к Разделу 3 (Модуль 1)

1. Что такое «Законы управления» в системе? Какую они играют роль? Приведите примеры в технических, биологических, социальных системах. Приведите примеры различных видов законов регулирования (релейный, ПИД-, и др.). Какие их достоинства и недостатки?
2. Что такое «Исполнительные органы» в системе? Какую они играют роль? Приведите примеры в технических, биологических, социальных системах.
3. Что такое и какими бывают «Состояния системы»? Приведите примеры в технических, биологических, социальных системах. Что такое устойчивые, неустойчивые состояния систем?
4. Что такое «Мозг, орган, принимающий решения» в системе? Какую он играет роль?

Приведите примеры в технических, биологических, социальных системах. Что произойдет, если «Мозг системы» частично или полностью утратит свои функции?

5. Что такое «Управляемость и наблюдаемость» в системе? Приведите примеры в технических, биологических, социальных системах.

Перечень тем лабораторных работ к Разделу 3 (Модуль 1):

Темы лабораторных работ соответствуют вопросам для самостоятельной работы (см. выше).

Литература для самостоятельного изучения к Разделу 3 (Модуль 1).

Основная литература

1. Грекул, В. И. Проектирование информационных систем : учебник и практикум для вузов / В. И. Грекул, Н. Л. Коровкина, Г. А. Левочкина. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-9916-8764-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511889> (дата обращения: 08.03.2023).

2. Программирование, тестирование, проектирование, нейросети, технологии аппаратно-программных средств (практические задания и способы их решения) : учебник : [16+] / С. В. Веретехина, К. С. Кармицкий, Д. Д. Лукашин [и др.]. — Москва : Директ-Медиа, 2022. — 144 с. : ил., табл. — Режим доступа: по подписке. — URL: <https://biblioclub.ru/index.php?page=book&id=694782> (дата обращения: 08.03.2023). — Библиогр. в кн. — ISBN 978-5-4499-3321-8. — Текст : электронный.

3. Маркин, А. В. Программирование на SQL в 2 ч. Часть 1 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 429 с. — (Высшее образование). — ISBN 978-5-534-15817-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509818> (дата обращения: 08.03.2023).

4. Маркин, А. В. Программирование на SQL в 2 ч. Часть 2 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., испр. и доп. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-534-15818-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509819> (дата обращения: 08.03.2023).

Дополнительная литература

1. Зараменских, Е. П. Управление жизненным циклом информационных систем : учебник и практикум для вузов / Е. П. Зараменских. — 2-е изд. — Москва : Издательство Юрайт, 2023. — 497 с. — (Высшее образование). — ISBN 978-5-534-14023-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511960> (дата обращения: 08.03.2023).

2. Стружкин, Н. П. Базы данных: проектирование. Практикум : учебное пособие для вузов / Н. П. Стружкин, В. В. Годин. — Москва : Издательство Юрайт, 2023. — 291 с. — (Высшее образование). — ISBN 978-5-534-00739-8. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/512160> (дата обращения: 08.03.2023).

3. Чистякова, М. А. Проектирование и эксплуатация баз данных : учебно-методическое пособие / М. А. Чистякова, И. А. Иванова, И. Д. Котилевец. — Москва : РТУ МИРЭА, 2021. — 112 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL:

<https://e.lanbook.com/book/176572> (дата обращения: 08.03.2023). — Режим доступа: для авториз. пользователей.

Задания для самостоятельной работы к Разделу 4 (Модуль 1)

Вопросы для самостоятельной работы к Разделу 4 (Модуль 1)

1. Что такое «информационные системы»? Какие процессы имеют место в информационной системе?
2. Этапы развития информационных систем.
3. Приведите определение понятия «организация». Примеры.
4. Охарактеризуйте процессы в информационной системе.
5. Что такое «обратная связь» в определении процессов в информационной системе? Приведите примеры. Какую роль играет обратная связь при функционировании технических систем?
6. Что можно ожидать от внедрения информационных систем?
7. Что понимают под понятием «управление в организации»?
8. Приведите характеристику управленческих функций: организационная; планирование; учетная; анализ; контрольная; стимулирование. Приведите примеры.

Перечень тем лабораторных работ к Разделу 4 (Модуль 1):

Темы лабораторных работ соответствуют вопросам для самостоятельной работы (см. выше).

Литература для самостоятельного изучения к Разделу 4 (Модуль 1).

Основная литература

1. Грекул, В. И. Проектирование информационных систем : учебник и практикум для вузов / В. И. Грекул, Н. Л. Коровкина, Г. А. Левочкина. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-9916-8764-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511889> (дата обращения: 08.03.2023).

2. Программирование, тестирование, проектирование, нейросети, технологии аппаратно-программных средств (практические задания и способы их решения) : учебник : [16+] / С. В. Веретехина, К. С. Кармицкий, Д. Д. Лукашин [и др.]. — Москва : Директ-Медиа, 2022. — 144 с. : ил., табл. — Режим доступа: по подписке. — URL: <https://biblioclub.ru/index.php?page=book&id=694782> (дата обращения: 08.03.2023). — Библиогр. в кн. — ISBN 978-5-4499-3321-8. — Текст : электронный.

3. Маркин, А. В. Программирование на SQL в 2 ч. Часть 1 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 429 с. — (Высшее образование). — ISBN 978-5-534-15817-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509818> (дата обращения: 08.03.2023).

4. Маркин, А. В. Программирование на SQL в 2 ч. Часть 2 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., испр. и доп. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-534-15818-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509819> (дата обращения: 08.03.2023).

Дополнительная литература

1. Зараменских, Е. П. Управление жизненным циклом информационных систем : учебник и практикум для вузов / Е. П. Зараменских. — 2-е изд. — Москва : Издательство Юрайт, 2023. — 497 с. — (Высшее образование). — ISBN 978-5-534-14023-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511960> (дата обращения: 08.03.2023).

2. Стружкин, Н. П. Базы данных: проектирование. Практикум : учебное пособие для вузов / Н. П. Стружкин, В. В. Годин. — Москва : Издательство Юрайт, 2023. — 291 с. — (Высшее образование). — ISBN 978-5-534-00739-8. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/512160> (дата обращения: 08.03.2023).

3. Чистякова, М. А. Проектирование и эксплуатация баз данных : учебно-методическое пособие / М. А. Чистякова, И. А. Иванова, И. Д. Котилевец. — Москва : РТУ МИРЭА, 2021. — 112 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/176572> (дата обращения: 08.03.2023). — Режим доступа: для авториз. пользователей.

Задания для самостоятельной работы к Разделу 5 (Модуль 2)

Вопросы для самостоятельной работы к Разделу 5 (Модуль 2)

1. Информационно-измерительные системы. Выработка информации и сигналов для использования их в информационных системах. Технология и платформа «Arduino».
2. Основные возможности платформы «Arduino» для построения информационных систем. Примеры программ.
3. Основные возможности платформы «IskraJS» для построения информационных систем. Приведите примеры программ.
4. Основные возможности платформы «Raspberry Pi» для построения информационных систем. Примеры программ.

Перечень тем лабораторных работ к Разделу 5 (Модуль 2):

Темы лабораторных работ соответствуют вопросам для самостоятельной работы (см. выше).

Литература для самостоятельного изучения к Разделу 5 (Модуль 2).

Основная литература

1. Грекул, В. И. Проектирование информационных систем : учебник и практикум для вузов / В. И. Грекул, Н. Л. Коровкина, Г. А. Левочкина. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-9916-8764-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511889> (дата обращения: 08.03.2023).

2. Программирование, тестирование, проектирование, нейросети, технологии аппаратно-программных средств (практические задания и способы их решения) : учебник : [16+] / С. В. Веретехина, К. С. Кармицкий, Д. Д. Лукашин [и др.]. — Москва : Директ-Медиа, 2022. — 144 с. : ил., табл. — Режим доступа: по подписке. — URL: <https://biblioclub.ru/index.php?page=book&id=694782> (дата обращения: 08.03.2023). — Библиогр. в кн. — ISBN 978-5-4499-3321-8. — Текст : электронный.

3. Маркин, А. В. Программирование на SQL в 2 ч. Часть 1 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 429 с. — (Высшее образование). — ISBN 978-5-534-15817-5. — Текст : электронный //

Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509818> (дата обращения: 08.03.2023).

4. Маркин, А. В. Программирование на SQL в 2 ч. Часть 2 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., испр. и доп. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-534-15818-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509819> (дата обращения: 08.03.2023).

Дополнительная литература

1. Зараменских, Е. П. Управление жизненным циклом информационных систем : учебник и практикум для вузов / Е. П. Зараменских. — 2-е изд. — Москва : Издательство Юрайт, 2023. — 497 с. — (Высшее образование). — ISBN 978-5-534-14023-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511960> (дата обращения: 08.03.2023).

2. Стружкин, Н. П. Базы данных: проектирование. Практикум : учебное пособие для вузов / Н. П. Стружкин, В. В. Годин. — Москва : Издательство Юрайт, 2023. — 291 с. — (Высшее образование). — ISBN 978-5-534-00739-8. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/512160> (дата обращения: 08.03.2023).

3. Чистякова, М. А. Проектирование и эксплуатация баз данных : учебно-методическое пособие / М. А. Чистякова, И. А. Иванова, И. Д. Котилевец. — Москва : РТУ МИРЭА, 2021. — 112 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/176572> (дата обращения: 08.03.2023). — Режим доступа: для авториз. пользователей.

Задания для самостоятельной работы к Разделу 6 (Модуль 2)

Вопросы для самостоятельной работы к Разделу 6 (Модуль 2)

1. Осуществите проектирование информационно-измерительной системы на основе персептрона – основного элемента искусственных нейронных сетей (ИНС).
2. Проведите аналогию между персептроном и вычислительной платформой «Arduino». Спроектируйте персептрон на базе платформы «Arduino» для минимум трех (можно больше) входных сигналов: освещенности, температуры, влажности. Можете использовать САПР (например, TinkerCAD).
3. Что представляет собой персептрон? Приведите назначение, свойства. Дайте характеристику основным элементам, составляющим персептрон.
4. Объединение в единую систему плат «Arduino» и «Raspberry Pi». Назначение такого объединения, возможности.
5. Онлайн-среды проектирования информационно-измерительных систем на базе платформы «Arduino». Назначение, возможности.

Перечень тем лабораторных работ к Разделу 6 (Модуль 2):

Темы лабораторных работ соответствуют вопросам для самостоятельной работы (см. выше).

Литература для самостоятельного изучения к Разделу 6 (Модуль 2).

Основная литература

1. Грекул, В. И. Проектирование информационных систем : учебник и практикум для вузов / В. И. Грекул, Н. Л. Коровкина, Г. А. Левочкина. — Москва : Издательство Юрайт,

2023. — 385 с. — (Высшее образование). — ISBN 978-5-9916-8764-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511889> (дата обращения: 08.03.2023).

2. Программирование, тестирование, проектирование, нейросети, технологии аппаратно-программных средств (практические задания и способы их решения) : учебник : [16+] / С. В. Веретехина, К. С. Кармицкий, Д. Д. Лукашин [и др.]. — Москва : Директ-Медиа, 2022. — 144 с. : ил., табл. — Режим доступа: по подписке. — URL: <https://biblioclub.ru/index.php?page=book&id=694782> (дата обращения: 08.03.2023). — Библиогр. в кн. — ISBN 978-5-4499-3321-8. — Текст : электронный.

3. Маркин, А. В. Программирование на SQL в 2 ч. Часть 1 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 429 с. — (Высшее образование). — ISBN 978-5-534-15817-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509818> (дата обращения: 08.03.2023).

4. Маркин, А. В. Программирование на SQL в 2 ч. Часть 2 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., испр. и доп. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-534-15818-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509819> (дата обращения: 08.03.2023).

Дополнительная литература

1. Зараменских, Е. П. Управление жизненным циклом информационных систем : учебник и практикум для вузов / Е. П. Зараменских. — 2-е изд. — Москва : Издательство Юрайт, 2023. — 497 с. — (Высшее образование). — ISBN 978-5-534-14023-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511960> (дата обращения: 08.03.2023).

2. Стружкин, Н. П. Базы данных: проектирование. Практикум : учебное пособие для вузов / Н. П. Стружкин, В. В. Годин. — Москва : Издательство Юрайт, 2023. — 291 с. — (Высшее образование). — ISBN 978-5-534-00739-8. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/512160> (дата обращения: 08.03.2023).

3. Чистякова, М. А. Проектирование и эксплуатация баз данных : учебно-методическое пособие / М. А. Чистякова, И. А. Иванова, И. Д. Котилевец. — Москва : РТУ МИРЭА, 2021. — 112 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/176572> (дата обращения: 08.03.2023). — Режим доступа: для авториз. пользователей.

Задания для самостоятельной работы к Разделу 7 (Модуль 2)

Вопросы для самостоятельной работы к Разделу 7 (Модуль 2)

1. Что такое «объект управления» в системе? Приведите примеры в технических, биологических, социальных системах.
2. Что такое «сенсоры» в системе? Какую они играют роль? Приведите примеры в технических, биологических, социальных системах.
3. Что такое «обратные связи» в системе? Какую они играют роль? Что такое положительные и отрицательные обратные связи? Что такое «Обратная связь по перемещению», «Обратная связь по скорости», «Обратная связь по ускорению»? Приведите примеры в технических, биологических, социальных системах.

4. Что такое «Законы управления» в системе? Какую они играют роль? Приведите примеры в технических, биологических, социальных системах. Приведите примеры различных видов законов регулирования (релейный, ПИД-, и др.). Какие их достоинства и недостатки?
5. Что такое «Исполнительные органы» в системе? Какую они играют роль? Приведите примеры в технических, биологических, социальных системах.
6. Что такое и какими бывают «Состояния системы»? Приведите примеры в технических, биологических, социальных системах. Что такое устойчивые, неустойчивые состояния систем?
7. Что такое «Мозг, орган, принимающий решения» в системе? Какую он играет роль? Приведите примеры в технических, биологических, социальных системах. Что произойдет, если «Мозг системы» частично или полностью утратит свои функции?
8. Что такое «Управляемость и наблюдаемость» в системе? Приведите примеры в технических, биологических, социальных системах.

Перечень тем лабораторных работ к Разделу 7 (Модуль 2):

Темы лабораторных работ соответствуют вопросам для самостоятельной работы (см. выше).

Литература для самостоятельного изучения к Разделу 7 (Модуль 2).

Основная литература

1. Грекул, В. И. Проектирование информационных систем : учебник и практикум для вузов / В. И. Грекул, Н. Л. Коровкина, Г. А. Левочкина. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-9916-8764-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511889> (дата обращения: 08.03.2023).

2. Программирование, тестирование, проектирование, нейросети, технологии аппаратно-программных средств (практические задания и способы их решения) : учебник : [16+] / С. В. Веретехина, К. С. Кармицкий, Д. Д. Лукашин [и др.]. — Москва : Директ-Медиа, 2022. — 144 с. : ил., табл. — Режим доступа: по подписке. — URL: <https://biblioclub.ru/index.php?page=book&id=694782> (дата обращения: 08.03.2023). — Библиогр. в кн. — ISBN 978-5-4499-3321-8. — Текст : электронный.

3. Маркин, А. В. Программирование на SQL в 2 ч. Часть 1 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 429 с. — (Высшее образование). — ISBN 978-5-534-15817-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509818> (дата обращения: 08.03.2023).

4. Маркин, А. В. Программирование на SQL в 2 ч. Часть 2 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., испр. и доп. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-534-15818-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509819> (дата обращения: 08.03.2023).

Дополнительная литература

1. Зараменских, Е. П. Управление жизненным циклом информационных систем : учебник и практикум для вузов / Е. П. Зараменских. — 2-е изд. — Москва : Издательство Юрайт, 2023. — 497 с. — (Высшее образование). — ISBN 978-5-534-14023-1. — Текст : электронный

// Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511960> (дата обращения: 08.03.2023).

2. Стружкин, Н. П. Базы данных: проектирование. Практикум : учебное пособие для вузов / Н. П. Стружкин, В. В. Годин. — Москва : Издательство Юрайт, 2023. — 291 с. — (Высшее образование). — ISBN 978-5-534-00739-8. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/512160> (дата обращения: 08.03.2023).

3. Чистякова, М. А. Проектирование и эксплуатация баз данных : учебно-методическое пособие / М. А. Чистякова, И. А. Иванова, И. Д. Котилевец. — Москва : РТУ МИРЭА, 2021. — 112 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/176572> (дата обращения: 08.03.2023). — Режим доступа: для авториз. пользователей.

Задания для самостоятельной работы к Разделу 8 (Модуль 2)

Вопросы для самостоятельной работы к Разделу 8 (Модуль 2)

1. Охарактеризуйте комплект документации, которым должен сопровождаться выпуск продукции (изделия).
2. Охарактеризуйте состав ГОСТ 15.016-2016 «Система разработки и постановки продукции на производство. Техническое задание. Требования к содержанию и оформлению».
3. Охарактеризуйте состав ГОСТ 2.114-2016 «Межгосударственный стандарт. Единая система конструкторской документации. Технические условия».
4. Охарактеризуйте состав ГОСТ 2.120-2013 «ЕСКД. Технический проект». Расскажите, как использовать данный ГОСТ для подготовки «Пояснительной записки технического проекта (ТП)»? Какие разделы будет содержать такая пояснительная записка?

Перечень тем лабораторных работ к Разделу 8 (Модуль 2):

Темы лабораторных работ соответствуют вопросам для самостоятельной работы (см. выше).

Литература для самостоятельного изучения к Разделу 8 (Модуль 2).

Основная литература

1. Грекул, В. И. Проектирование информационных систем : учебник и практикум для вузов / В. И. Грекул, Н. Л. Коровкина, Г. А. Левочкина. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-9916-8764-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511889> (дата обращения: 08.03.2023).

2. Программирование, тестирование, проектирование, нейросети, технологии аппаратно-программных средств (практические задания и способы их решения) : учебник : [16+] / С. В. Веретехина, К. С. Кармицкий, Д. Д. Лукашин [и др.]. — Москва : Директ-Медиа, 2022. — 144 с. : ил., табл. — Режим доступа: по подписке. — URL: <https://biblioclub.ru/index.php?page=book&id=694782> (дата обращения: 08.03.2023). — Библиогр. в кн. — ISBN 978-5-4499-3321-8. — Текст : электронный.

3. Маркин, А. В. Программирование на SQL в 2 ч. Часть 1 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 429 с. — (Высшее образование). — ISBN 978-5-534-15817-5. — Текст : электронный //

Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509818> (дата обращения: 08.03.2023).

4. Маркин, А. В. Программирование на SQL в 2 ч. Часть 2 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., испр. и доп. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-534-15818-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509819> (дата обращения: 08.03.2023).

Дополнительная литература

1. Зараменских, Е. П. Управление жизненным циклом информационных систем : учебник и практикум для вузов / Е. П. Зараменских. — 2-е изд. — Москва : Издательство Юрайт, 2023. — 497 с. — (Высшее образование). — ISBN 978-5-534-14023-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511960> (дата обращения: 08.03.2023).

2. Стружкин, Н. П. Базы данных: проектирование. Практикум : учебное пособие для вузов / Н. П. Стружкин, В. В. Годин. — Москва : Издательство Юрайт, 2023. — 291 с. — (Высшее образование). — ISBN 978-5-534-00739-8. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/512160> (дата обращения: 08.03.2023).

3. Чистякова, М. А. Проектирование и эксплуатация баз данных : учебно-методическое пособие / М. А. Чистякова, И. А. Иванова, И. Д. Котилевец. — Москва : РТУ МИРЭА, 2021. — 112 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/176572> (дата обращения: 08.03.2023). — Режим доступа: для авториз. пользователей.

2. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ) И САМОСТОЯТЕЛЬНОЙ РАБОТЕ

Освоение обучающимся дисциплины (модуля) «Технологии возможностей и безбарьерной среды» предполагает изучение материалов дисциплины на аудиторных занятиях и в ходе самостоятельной работы. Аудиторные занятия проходят в форме лекций, семинаров, практических и лабораторных занятий. Самостоятельная работа включает разнообразный комплекс видов и форм работы обучающихся.

Для успешного освоения дисциплины (модуля) и достижения поставленных целей необходимо внимательно ознакомиться с рабочей программы дисциплины (модуля), доступной в электронной информационно-образовательной среде РГСУ.

Следует обратить внимание на списки основной и дополнительной литературы, на предлагаемые преподавателем ресурсы информационно-телекоммуникационной сети Интернет. Эта информация необходима для самостоятельной работы обучающегося.

При подготовке к аудиторным занятиям необходимо помнить особенности каждой формы его проведения.

Подготовка к учебному занятию лекционного типа заключается в следующем.

С целью обеспечения успешного обучения обучающийся должен готовиться к лекции, поскольку она является важнейшей формой организации учебного процесса, поскольку:

- знакомит с новым учебным материалом;
- разъясняет учебные элементы, трудные для понимания;
- систематизирует учебный материал;
- ориентирует в учебном процессе.

С этой целью:

- внимательно прочитайте материал предыдущей лекции;

- ознакомьтесь с учебным материалом по учебнику и учебным пособиям с темой прочитанной лекции;
- внесите дополнения к полученным ранее знаниям по теме лекции на полях лекционной тетради;
- запишите возможные вопросы, которые вы зададите лектору на лекции по материалу изученной лекции;
- постарайтесь уяснить место изучаемой темы в своей подготовке;
- узнайте тему предстоящей лекции (по тематическому плану, по информации лектора) и запишите информацию, которой вы владеете по данному вопросу.

Подготовка к занятию семинарского типа.

При подготовке и работе во время проведения занятий семинарского типа следует обратить внимание на следующие моменты: на процесс предварительной подготовки, на работу во время занятия, обработку полученных результатов, исправление полученных замечаний.

Предварительная подготовка к учебному занятию семинарского типа заключается в изучении теоретического материала в отведенное для самостоятельной работы время, ознакомление с инструктивными материалами с целью осознания задач практического занятия, техники безопасности при работе с приборами, веществами.

Работа во время проведения учебного занятия семинарского типа включает:

- консультирование студентов преподавателями и вспомогательным персоналом с целью предоставления исчерпывающей информации, необходимой для самостоятельного выполнения предложенных преподавателем задач.
- самостоятельное выполнение заданий согласно обозначенной рабочей программой дисциплины (модуля) тематики.

Самостоятельная работа.

Самостоятельная работа - планируемая учебная, учебно-исследовательская, научно-исследовательская работа студентов, выполняемая во внеаудиторное (аудиторное) время по заданию и при методическом руководстве преподавателя, но без его непосредственного участия (при частичном непосредственном участии преподавателя, оставляющем ведущую роль за работой студентов).

Для более углубленного изучения темы задания для самостоятельной работы рекомендуется выполнять параллельно с изучением данной темы. При выполнении заданий по возможности используйте наглядное представление материала.

Самостоятельная работа студентов в ВУЗе является важным видом учебной и научной деятельности студента. Самостоятельная работа студентов играет значительную роль в рейтинговой технологии обучения.

К современному специалисту в области медицины общество предъявляет достаточно широкий перечень требований, среди которых немаловажное значение имеет наличие у выпускников определенных навыков (компетенций) и умения самостоятельно добывать знания из различных источников, систематизировать полученную информацию, давать оценку конкретной профессиональной ситуации. Формирование такого умения происходит в течение всего периода обучения.

В процессе самостоятельной работы студент приобретает необходимые для будущей специальности компетенции, навыки самоорганизации, самоконтроля, самоуправления, саморефлексии и становится активным самостоятельным субъектом учебной деятельности.

Самостоятельная работа студентов должна оказывать важное влияние на формирование личности будущего специалиста, его компетентность. Каждый студент самостоятельно определяет режим своей работы и меру труда, затрачиваемого на овладение учебным содержанием по каждой дисциплине. Он выполняет внеаудиторную работу по личному индивидуальному плану, в зависимости от его подготовки, времени и других условий.

Виды самостоятельной работы.

Работа с литературой.

При работе с книгой необходимо подобрать литературу, научиться правильно ее читать, вести записи. Для подбора литературы в библиотеке используются алфавитный и систематический каталоги. Важно помнить, что рациональные навыки работы с книгой - это всегда большая экономия времени и сил. Правильный подбор учебников рекомендуется преподавателем, читающим лекционный курс. Необходимая литература может быть также указана в методических разработках по данному курсу. Изучая материал по учебнику, следует переходить к следующему вопросу только после правильного уяснения предыдущего, описывая на бумаге все выкладки и вычисления (в том числе те, которые в учебнике опущены или на лекции даны для самостоятельного вывода). При изучении любой дисциплины большую и важную роль играет самостоятельная индивидуальная работа. Особое внимание следует обратить на определение основных понятий курса. Студент должен подробно разбирать примеры, которые поясняют такие определения, и уметь строить аналогичные примеры самостоятельно. Нужно добиваться точного представления о том, что изучаешь. Полезно составлять опорные конспекты. При изучении материала по учебнику полезно в тетради (на специально отведенных полях) дополнять конспект лекций. Там же следует отмечать вопросы, выделенные студентом для консультации с преподавателем. Выводы, полученные в результате изучения, рекомендуется в конспекте выделять, чтобы они при перечитывании записей лучше запоминались. Опыт показывает, что многим студентам помогает составление листа опорных сигналов, содержащего важнейшие и наиболее часто употребляемые формулы и понятия. Такой лист помогает запомнить формулы, основные положения лекции, а также может служить постоянным справочником для студента. Различают два вида чтения: первичное и вторичное. Первичное - это внимательное, неторопливое чтение, при котором можно остановиться на трудных местах.

Задача вторичного чтения - полное усвоение смысла целого (по счету это чтение может быть и не вторым, а третьим или четвертым). Самостоятельная работа с учебниками и книгами (а также самостоятельное теоретическое исследование проблем, обозначенных преподавателем на лекциях) – это важнейшее условие формирования у себя научного способа познания.

Чтение научного текста является частью познавательной деятельности. Ее цель – извлечение из текста необходимой информации. От того, насколько осознанно читающим собственная внутренняя установка при обращении к печатному слову (найти нужные сведения, усвоить информацию полностью или частично, критически проанализировать материал и т.п.) во многом зависит эффективность осуществляемого действия.

Методические рекомендации по составлению конспекта:

1. Внимательно прочитайте текст. Уточните в справочной литературе непонятные слова. При записи не забудьте вынести справочные данные на поля конспекта;
2. Выделите главное, составьте план;
3. Кратко сформулируйте основные положения текста, отметьте аргументацию автора;
4. Законспектируйте материал, четко следуя пунктам плана. При конспектировании старайтесь выразить мысль своими словами. Записи следует вести четко, ясно.

5. Грамотно записывайте цитаты. Цитируя, учитывайте лаконичность, значимость мысли. В тексте конспекта желательно приводить не только тезисные положения, но и их доказательства. При оформлении конспекта необходимо стремиться к емкости каждого предложения. Мысли автора книги следует излагать кратко, заботясь о стиле и выразительности написанного. Число дополнительных элементов конспекта должно быть логически обоснованным, записи должны распределяться в определенной последовательности, отвечающей логической структуре произведения. Для уточнения и дополнения необходимо оставлять поля. Овладение навыками конспектирования требует от студента целеустремленности, повседневной самостоятельной работы.

Методические материалы по самостоятельному решению задач

При самостоятельном решении задач нужно обосновывать каждый этап решения, исходя из теоретических положений курса. Если студент видит несколько путей решения

проблемы (задачи), то нужно сравнить их и выбрать самый рациональный. Полезно до начала вычислений составить краткий план решения проблемы (задачи). Решение проблемных задач или примеров следует излагать подробно, вычисления располагать в строгом порядке, отделяя вспомогательные вычисления от основных. Решения при необходимости нужно сопровождать комментариями, схемами, чертежами и рисунками. Следует помнить, что решение каждой учебной задачи должно доводиться до окончательного логического ответа, которого требует условие, и по возможности с выводом. Полученный ответ следует проверить способами, вытекающими из существа данной задачи. Полезно также (если возможно) решать несколькими способами и сравнить полученные результаты. Решение задач данного типа нужно продолжать до приобретения твердых навыков в их решении.

Методические материалы к выполнению реферата

Реферат (от лат. *refere* – сообщать) – краткое изложение в письменном виде или в форме публикации доклада, содержания научного труда (трудов), литературы по теме. Работа над рефератом условно разделяется на выбор темы, подбор литературы, подготовку и защиту плана; написание теоретической части и всего текста с указанием библиографических данных используемых источников, подготовку доклада, выступление с ним. Тематика рефератов полностью связана с основными вопросами изучаемого курса.

Список литературы к темам не дается, и обучающиеся самостоятельно ведут библиографический поиск, причем им не рекомендуется ограничиваться университетской библиотекой.

Важно учитывать, что написание реферата требует от обучающихся определенных усилий. Особое внимание следует уделить подбору литературы, методике ее изучения с целью отбора и обработки собранного материала, обоснованию актуальности темы и теоретического уровня обоснованности используемых в качестве примеров фактов какой-либо деятельности.

Выбрав тему реферата, начав работу над литературой, необходимо составить план. Изучая литературу, продолжается обдумывание темы, осмысливание прочитанного, делаются выписки, сопоставляются точки зрения разных авторов и т.д. Реферативная работа сводится к тому, чтобы в ней выделились две взаимосвязанные стороны: во-первых, ее следует рассматривать как учебное задание, которое должен выполнить обучаемый, а во-вторых, как форму научной работы, творческого воображения при выполнении учебного задания.

Наличие плана реферата позволяет контролировать ход работы, избежать формального переписывания текстов из первоисточников.

Оформление реферата включает титульный лист, оглавление и краткий список использованной литературы. Список использованной литературы размещается на последней странице рукописи или печатной форме реферата. Реферат выполняется в письменной или печатной форме на белых листах формата А4 (210 x 297 мм). Шрифт Times New Roman, кегель 14, через 1,5 интервала при соблюдении следующих размеров текста: верхнее поле – 25 мм, нижнее – 20 мм, левое – 30 мм, правое – 15 мм. Нумерация страниц производится сверху листа, по центру. Титульный лист нумерации не подлежит.

Рефераты должны быть написаны простым, ясным языком, без претензий на наукообразность. Следует избегать сложных грамматических оборотов, непривычных терминов и символов. Если же такие термины и символы все-таки приводятся, то необходимо разъяснять их значение при первом упоминании в тексте реферата.

Объем реферата предполагает тщательный отбор информации, необходимой для краткого изложения вопроса. Важнейший этап – редактирование готового текста реферата и подготовка к обсуждению. Обсуждение требует хорошей ориентации в материале темы, умения выделить главное, поставить дискуссионный вопрос, привлечь внимание слушателей к интересной литературе, логично и убедительно изложить свои мысли.

Рефераты обязательно подлежат защите. Процедура защиты начинается с определения оппонентов защищающего свою работу. Они стремятся дать основательный анализ работы обучающимся, обращают внимание на положительные моменты и недостатки реферата, дают общую оценку содержанию, форме преподнесения материала, характеру использованной

литературы. Иногда они дополняют тот или иной раздел реферата. Последнее особенно ценно, ибо говорит о глубоком знании обучающимся-оппонентом изучаемой проблемы.

Обсуждение не ограничивается заслушиванием оппонентов. Другие обучающиеся имеют право уточнить или опровергнуть какое-либо утверждение. Преподаватель предлагает любому обучающемуся задать вопрос по существу доклада или попытаться подвести итог обсуждению.

Алгоритм работы над рефератом

1. Выбор темы

Тема должна быть сформулирована грамотно (с литературной точки зрения);

В названии реферата следует поставить четкие рамки рассмотрения темы;

Желательно избегать слишком длинных названий;

Следует по возможности воздерживаться от использования в названии спорных с научной точки зрения терминов, излишней наукообразности, а также чрезмерного упрощения формулировок.

2. Реферат следует составлять из пяти основных частей: введения; основной части; заключения; списка литературы; приложений.

3. Основные требования к введению:

Во введении не следует концентрироваться на содержании; введение должно включать краткое обоснование актуальности темы реферата, где требуется показать, почему данный вопрос может представлять научный интерес и есть ли связь представляемого материала с современностью. Таким образом, тема реферата должна быть актуальна либо с научной точки зрения, либо с современных позиций.

Очень важно выделить цель, а также задачи, которые требуется решить для выполнения цели.

Введение должно содержать краткий обзор изученной литературы, в котором указывается взятый из того или иного источника материал, кратко анализируются изученные источники, показываются их сильные и слабые стороны;

Объем введения составляет две страницы текста.

4. Требования к основной части реферата:

Основная часть содержит материал, отобранный для рассмотрения проблемы;

Также основная часть должна включать в себя собственно мнение обучающихся и сформулированные самостоятельные выводы, опирающиеся на приведенные факты;

Материал, представленный в основной части, должен быть логически изложен и распределен по параграфам, имеющим свои названия;

В изложении основной части необходимо использовать сноски (в первую очередь, когда приводятся цифры и чьи-то цитаты);

Основная часть должна содержать иллюстративный материал (графики, таблицы и т. д.);

Объем основной части составляет около 10 страниц.

5. Требования к заключению:

В заключении формулируются выводы по параграфам, обращается внимание на выдвинутые во введении задачи и цели;

Заключение должно быть четким, кратким, вытекающим из содержания основной части.

6. Требования к оформлению списка литературы (по ГОСТу):

Необходимо соблюдать правильность последовательности записи источников: сначала следует писать фамилию, а после инициалы; название работы не ставится в кавычки; после названия сокращенно пишется место издания; затем идет год издания; наконец, называется процитированная страница.

Критерии оценки реферата

Обучающийся, защищающий реферат, должен рассказать о его актуальности, поставленных целях и задачах, изученной литературе, структуре основной части, сделанных в ходе работы выводах.

По окончании выступления ему может быть задано несколько вопросов по представленной проблеме.

Оценка складывается из соблюдения требований к реферату, грамотного раскрытия темы, умения четко рассказывать о представленном реферате, способности понять суть задаваемых по работе вопросов и найти точные ответы на них.

Методические материалы к выполнению эссе

Эссе – литературное произведение небольшого объема, обычно прозаическое, свободной композиции, передающее индивидуальные впечатления, суждения, соображения автора о той или иной проблеме, теме, о том или ином событии или явлении. Это вид самостоятельной исследовательской работы обучающихся, с целью углубления и закрепления теоретических знаний и освоения практических навыков. Цель эссе состоит в развитии самостоятельного творческого мышления и письменного изложения собственных мыслей. При написании эссе обучающийся должен представить развернутый письменный ответ на теоретический или практический актуальный вопрос, объявленный преподавателем в аудитории непосредственно перед ее написанием. В процессе написания эссе разрешается пользоваться нормативно-правовыми актами, конспектом лекций (в печатном виде). Использование интернет-ресурсов не допускается. Темы эссе преподаватель предлагает из числа тех, которые обучающиеся уже рассматривали на лекциях или семинарских занятиях, исходя из содержания заданий в составе оценочных средств. По решению преподавателя, в качестве темы эссе может быть выбрана одна или несколько тем, которые могут быть распределены между обучающимися по желанию.

Требования к выполнению эссе:

1. Проводится письменно.
2. Эссе выполняется на компьютере (гарнитура Times New Roman, шрифт 14) через 1,5 интервала с полями: верхнее, нижнее – 2; правое – 3; левое – 1,5. Отступ первой строки абзаца – 1,25. Сноски – постраничные. Таблицы и рисунки встраиваются в текст работы. При этом обязательный заголовок таблицы надо размещать над табличным полем, а рисунки сопровождать подрисовочными подписями. При включении в эссе нескольких таблиц и/или рисунков их нумерация обязательна. Обязательна и нумерация страниц. Их целесообразно проставлять внизу страницы – по середине или в правом углу. Номер страницы не ставится на титульном листе, но в общее число страниц он включается. Объем эссе, без учета приложений, не должен превышать 5 страниц. Значительное превышение установленного объема является недостатком работы и указывает на то, что обучающийся не сумел отобрать и переработать необходимый материал.

3. Работа должна содержать собственные умозаключения по сути поставленной проблемы, включать самостоятельно проведенный анализ по сути этой проблемы, выводы, обобщающие авторскую позицию по поставленной проблеме.

Критерии оценки эссе:

«Отлично» – исключительные знания материала, абсолютное понимание сути, безукоризненное знание основных понятий и положений, логически и лексически грамотно изложенный, содержательный, аргументированный, конкретный и исчерпывающий ответ.

«Хорошо» – глубокие знания материала, правильное понимание сути, знание основных понятий и положений, содержательный, полный и конкретный ответ.

«Удовлетворительно» – твердые, но недостаточно полные знания, верное понимание сути, в целом правильный ответ.

«Неудовлетворительно» – непонимание сущности задания, грубые ошибки в ответе.

Методические материалы по выполнению тестирования.

Тестовые задания содержат вопросы и 3-4 варианта ответа по базовым положениям изучаемой темы/раздела, составлены с расчетом на знания, полученные обучающимся в процессе изучения темы/раздела.

Тестовые задания выполняются в письменной форме и сдаются преподавателю, ведущему дисциплину (модуль). На выполнение тестовых заданий обучающимся отводится 45 минут.

При обработке результатов оценочной процедуры используются: критерии оценки по содержанию и качеству полученных ответов, ключи, оценочные листы.

Критерии оценки теста:

«Зачтено» - если обучающийся правильно выполнил не менее 2/3 всей работы или допустил не более одной грубой ошибки и двух недочетов, не более одной грубой и одной негрубой ошибки, не более трех негрубых ошибок, одной негрубой ошибки и трех недочетов, при наличии четырех-пяти недочетов.

«Не зачтено» - если число ошибок и недочетов превысило норму для оценки 3 или правильно выполнено менее 2/3 всей работы.

Методические материалы по выполнению доклада.

Рекомендуется следующая структура доклада:

1. титульный лист, содержание доклада;
2. краткое изложение;
3. цели и задачи;
4. изложение характера исследований и рассмотренных проблем, гипотезы, спорные вопросы;
5. источники информации, методы сбора и анализа данных, степень их полноты и достоверности;
6. анализ и толкование полученных в работе результатов;
7. выводы и оценки;
8. библиография и приложения.

Время выступления докладчика не должно превышать 10 минут.

Основные требования к оформлению доклада:

- титульный лист должен включать название доклада, наименование предметной (цикловой) комиссии, фамилию обучающегося;
- все использованные литературные источники сопровождаются библиографическим описанием;
- приводимая цитата из источника берется в кавычки (оформляются сноски);
- единицы измерения должны применяться в соответствии с действующими стандартами;
- все названия литературных источников следует приводить в соответствии с новейшими изданиями;
- рекомендуется включение таблиц, графиков, схем, если они отражают основное содержание или улучшают ее наглядность;
- названия фирм, учреждений, организаций и предприятий должны именоваться так, как они указываются в источнике;

Критерии оценки доклада

При выполнении доклада обучающийся должен продемонстрировать умение кратко излагать прочитанный материал, а также умение обобщать и анализировать материал по теме доклада.

Презентация

Методические материалы к презентациям

1. Объем презентации 10 -20 слайдов.
2. На титульном слайде должно быть отражено:
 - наименование факультета;
 - тема презентации;

- фамилия, имя, отчество, направление подготовки/ специальность, направленность (профиль)/ специализация, форма обучения, номер группы автора презентации;
- фамилия, имя, отчество, степень, звание, должность руководитель работы;
- год выполнения работы.

3. В презентации должны быть отражены обоснование актуальности представляемого материала, цели и задачи работы.

4. Содержание презентации должно включать наиболее значимый материал доклада, а также, при необходимости, таблицы, диаграммы, рисунки, фотографии, карты, видео – вставки, звуковое сопровождение.

5. Заключительный слайд должен содержать информацию об источниках информации для презентации.

Критерии оценки презентации

1. Объём презентации 10 -20 слайдов.
2. Правильность оформления титульного слайда.
3. Актуальность отобранного материала, обоснованность формулировки цели и задач работы.

4. Наглядность и логичность презентации, обоснованность использования таблиц, диаграмм, рисунков, фотографий, карт, видео – вставок, звукового сопровождения; правильный выбор шрифтов, фона, других элементов дизайна слайда.

5. Объём и качество источников информации (не менее 2-х интернет – источников и не менее 2-х литературных источников).

Методические материалы по подготовке к опросу

Самостоятельная работа обучающихся включает подготовку к опросу на практических занятиях. Для этого обучающийся изучает лекции, основную и дополнительную литературу, публикации, информацию из Интернет-ресурсов.

Тема и вопросы к практическим занятиям, вопросы для самоконтроля содержатся в рабочей программе и доводятся до студентов заранее.

Для подготовки к опросу обучающемуся необходимо ознакомиться с материалом, посвященным теме практического занятия, в учебнике или другой рекомендованной литературе, конспекте лекции, обратить внимание на усвоение основных понятий дисциплины, выявить неясные вопросы и подобрать дополнительную литературу для их освещения.

Критерии оценки опроса

«Отлично»:

- дан полный, развернутый ответ на поставленный вопрос;
- в ответе прослеживается четкая структура, логическая последовательность, отражающая сущность раскрываемых понятий, теорий, явлений;
- знание по предмету демонстрируются на фоне понимания его в системе данной науки и междисциплинарных связей;
- свободное владение терминологией;
- ответы на дополнительные вопросы четкие, краткие;

«Хорошо»:

- дан полный, развернутый ответ на поставленный вопрос, показано умение выделять существенные и несущественные признаки, причинно-следственные связи;
- ответ недостаточно логичен с единичными ошибками в частностях, исправленные студентом с помощью преподавателя;
- единичные ошибки в терминологии;
- ответы на дополнительные вопросы правильные, недостаточно полные и четкие.

«Удовлетворительно»:

- ответ не полный, с ошибками в деталях, умение раскрыть значение обобщённых знаний не показано, речевое оформление требует поправок, коррекции;

– логика и последовательность изложения имеют нарушения, студент не способен самостоятельно выделить существенные и несущественные признаки и причинно-следственные связи;

- ошибки в раскрываемых понятиях, терминах;
- студент не ориентируется в теме, допускает серьезные ошибки;
- студент не может ответить на большую часть дополнительных вопросов.

«Неудовлетворительно»:

– ответ представляет собой разрозненные знания с существенными ошибками по вопросу;

– присутствуют фрагментарность, нелогичность изложения, студент не осознает связь обсуждаемого вопроса с другими объектами дисциплины, речь неграмотная;

- незнание терминологии;
- ответы на дополнительные вопросы неправильные.

Методические материалы по выполнению практического задания

При выполнении практического задания обучающийся придерживается следующего алгоритма:

1. Записать дату, тему и цель задания;
2. Ознакомиться с правилами и условия выполнения практического задания;
3. Повторить теоретические задания, необходимые для рациональной работы и других практических действий, используя конспекты лекций и рекомендованную литературу, представленной в программе;
4. Выполнить работу по предложенному алгоритму действий;
5. Обобщить результаты работы, сформулировать выводы / дать ответы на контрольные вопросы;

Работа должна быть выполнена грамотно, с соблюдением культуры изложения. При использовании данных из учебных, методических пособий и другой литературы, периодических изданий, Интернет-источников должны иметься ссылки на вышеперечисленные.

Критерии оценки практического задания:

«Отлично» – правильный ответ, дается четкое обоснование принятому решению; рассуждения четкие последовательные логические; используются ссылки на полученные при изучении дисциплины знания; правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Хорошо» – правильный ответ, дается обоснование принятому решению; но с не существенными ошибками, в рассуждениях отсутствует логическая последовательность; используются ссылки на полученные при изучении дисциплины знания, правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Удовлетворительно» – правильный ответ, допускаются грубые ошибки в обосновании принятого решения; рассуждения не последовательные сумбурные; используются ссылки на полученные при изучении дисциплины знания; используются формулы, процедуры, понятия, имеющие прямое значение для подтверждения принятого решения, однако, при обращении к ним допускаются серьезные ошибки, студент не может правильно ими воспользоваться.

«Неудовлетворительно, не зачтено» – ответ неверный, отсутствует обоснование принятому решению; студент демонстрирует полное непонимание сути вопроса.

Для оценки решения ситуационной задачи (аналитического задания):

Оценка «отлично» выставляется, если задача решена грамотно, ответы на вопросы сформулированы четко. Эталонный ответ полностью соответствует решению студента, которое хорошо обосновано теоретически.

Оценка «хорошо» выставляется, если задача решена, ответы на вопросы сформулированы не достаточно четко. Решение студента в целом соответствует эталонному ответу, но не достаточно хорошо обосновано теоретически.

Оценка «удовлетворительно» выставляется, если задача решена не полностью, ответы не содержат всех необходимых обоснований решения.

Оценка «неудовлетворительно» выставляется, если задача не решена или имеет грубые теоретические ошибки в ответе на поставленные вопросы.

Методические материалы по выполнению лабораторного задания

При выполнении лабораторного задания обучающийся придерживается следующего алгоритма:

1. Записать дату, тему и цель задания;
2. Ознакомиться с правилами и условия выполнения задания;
3. Повторить теоретические задания, необходимые для рациональной работы и других практических действий, используя конспекты лекций и рекомендованную литературу, представленной в программе;
4. Выполнить работу по предложенному алгоритму действий;
5. Обобщить результаты работы, сформулировать выводы / дать ответы на контрольные вопросы;

Работа должна быть выполнена грамотно, с соблюдением культуры изложения. При использовании данных из учебных, методических пособий и другой литературы, периодических изданий, Интернет-источников должны иметься ссылки на вышеперечисленные.

Критерии оценки лабораторного задания:

«Отлично» – правильный ответ, дается четкое обоснование принятому решению; рассуждения четкие последовательные логические; используются ссылки на полученные при изучении дисциплины знания; правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Хорошо» – правильный ответ, дается обоснование принятому решению; но с не существенными ошибками, в рассуждениях отсутствует логическая последовательность; используются ссылки на полученные при изучении дисциплины знания, правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Удовлетворительно» – правильный ответ, допускаются грубые ошибки в обосновании принятого решения; рассуждения не последовательные сумбурные; используются ссылки на полученные при изучении дисциплины знания; используются формулы, процедуры, понятия, имеющие прямое значение для подтверждения принятого решения, однако, при обращении к ним допускаются серьезные ошибки, студент не может правильно ими воспользоваться.

«Неудовлетворительно, не зачтено» – ответ неверный, отсутствует обоснование принятому решению; студент демонстрирует полное непонимание сути вопроса.

Методические указания для подготовки к промежуточной аттестации.

Изучение учебных дисциплин (модулей) завершается зачетом/зачетом с оценкой или экзаменом. Подготовка к промежуточной аттестации способствует закреплению, углублению и обобщению знаний, получаемых, в процессе обучения, а также применению их к решению практических задач. Готовясь к экзамену, студент ликвидирует имеющиеся пробелы в знаниях, углубляет, систематизирует и упорядочивает свои знания. На зачете или экзамене студент демонстрирует то, что он освоил в процессе обучения по дисциплине (модулю).

Вначале следует просмотреть весь материал по дисциплине (модулю), отметить для себя трудные вопросы. Обязательно в них разобраться. В заключение еще раз целесообразно повторить основные положения, используя при этом листы опорных сигналов. Систематическая подготовка к занятиям в течение семестра позволит использовать время промежуточной аттестации для систематизации знаний.

3. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНИВАНИЯ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ И (ИЛИ) ОПЫТА ДЕЯТЕЛЬНОСТИ, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ

3.1. Организационные основы применения балльно-рейтинговой системы оценки успеваемости обучающихся по дисциплине (модулю)

Оценка качества освоения обучающимися дисциплины (модуля) реализуется в формате балльно-рейтинговой системы оценки успеваемости обучающихся (БРСО).

БРСО в ходе текущего контроля успеваемости и промежуточной аттестации осуществляется по 100-балльной шкале.

Академический рейтинг обучающегося по дисциплине (модулю) складывается из результатов:

- текущего контроля успеваемости (максимальный текущий рейтинг обучающегося 80 рейтинговых баллов);
- промежуточной аттестации (максимальный рубежный рейтинг обучающегося 20 рейтинговых баллов).

Условия оценки освоения обучающимся дисциплины (модуля) в формате БРСО доводятся преподавателем до сведения обучающихся на первом учебном занятии, а также размещены в свободном доступе в электронной информационно-образовательной среде Университета.

3.2. Проведение текущего контроля успеваемости обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося

В течение учебного семестра до промежуточной аттестации на основании утвержденной рабочей программы дисциплины (модуля) формируется текущий рейтинг обучающегося. Текущий рейтинг обучающегося складывается как сумма рейтинговых баллов, полученных им в течение учебного семестра по всем видам учебных занятий по учебной дисциплине.

В процессе текущего контроля оцениваются следующие действия обучающегося, направленные на освоение компетенций в рамках изучения учебной дисциплины:

- академическая активность (посещаемость учебных занятий, самостоятельное изучение содержания учебной дисциплины в электронной информационно-образовательной среде, соблюдение сроков сдачи практических заданий и текущих контрольных мероприятий и др.);
- выполнение и сдача текущих и итогового практических заданий (эссе, рефераты, творческие задания, кейс-задания, лабораторные работы, расчетные задания и др., активное участие в групповых интерактивных занятиях (дискуссии, WiKi-проекты и др.), защита проектов и др.);
- прохождение рубежей текущего контроля, включая соблюдение графика их прохождения в электронной информационно-образовательной среде.

Для планирования расчета текущего рейтинга обучающегося используются следующие пропорции:

Вид учебного действия	Максимальная рейтинговая оценка, баллов
академическая активность	10
практические задания	40
<i>из них: текущие практические задания</i>	20
<i>итоговое практическое задание</i>	20
рубежи текущего контроля	30
<i>ИТОГО:</i>	80

В течение учебного семестра по дисциплине (модулю) обучающимся должен быть накоплен текущий рейтинг не менее 52 рейтинговых баллов (65% от максимального значения текущего рейтинга).

Необходимыми условиями допуска обучающегося к промежуточной аттестации по дисциплине являются положительное прохождение обучающимся не менее 65% рубежей текущего контроля с накоплением не менее 65% максимального рейтингового балла за каждый рубеж текущего контроля и положительное выполнение итогового практического задания с накоплением не менее 65% максимального рейтингового балла, установленного за итоговое практическое задание.

Невыполнение вышеуказанных условий является текущей академической задолженностью, которая должна быть ликвидирована обучающимся до контрольного мероприятия промежуточной аттестации.

Сведения о наличии у обучающихся текущей академической задолженности, сроках и порядке добора рейтинговых баллов для её ликвидации доводятся до обучающихся педагогическим работником.

В случае неликвидации текущей академической задолженности, педагогический работник обязан во время контрольного мероприятия промежуточной аттестации поставить обучающемуся 0 рейтинговых баллов. В этом случае ликвидация текущей академической задолженности возможна в периоды проведения повторной промежуточной аттестации.

3.3. Проведение промежуточной аттестации обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося

Промежуточная аттестация по дисциплине (модулю) проводится в соответствии с Положением о промежуточной аттестации обучающихся по основным профессиональным образовательным программам высшего образования – программ специалитета в Российском государственном социальном университете и Положением о балльно-рейтинговой системе оценки успеваемости обучающихся по основным профессиональным образовательным программам высшего образования - программам бакалавриата, программам специалитета, программам магистратуры в Российском государственном социальном университете.

На промежуточную аттестацию отводится 20 рейтинговых баллов.

Ответы обучающегося на контрольном мероприятии промежуточной аттестации оцениваются педагогическим работником по 20 - балльной шкале, а итоговая оценка по дисциплине (модулю) выставляется по системе зачтено/не зачтено для зачета и по пятибалльной системе для экзамена..

Критерии выставления оценки определяются Положением о балльно-рейтинговой системе оценки успеваемости обучающихся по основным профессиональным образовательным программам высшего образования - программам бакалавриата, программам специалитета, программам магистратуры в Российском государственном социальном университете.

В процессе определения рубежного рейтинга обучающегося используется следующая шкала:

Рубежный рейтинг	Критерии оценки освоения обучающимся учебной дисциплины в ходе контрольных мероприятий промежуточной аттестации
19-20 рейтинговых баллов	обучающийся глубоко и прочно усвоил программный материал, исчерпывающе, последовательно, грамотно и логически стройно его излагает, тесно увязывает с задачами и будущей деятельностью, не затрудняется с ответом при видоизменении задания, свободно справляется с задачами и практическими заданиями, правильно обосновывает принятые решения, умеет самостоятельно обобщать и излагать материал, не допуская

16-18 рейтинговых баллов	обучающийся твердо знает программный материал, грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос, может правильно применять теоретические положения и владеет необходимыми умениями и навыками при выполнении практических
13-15 рейтинговых баллов	обучающийся освоил основной материал, но не знает отдельных деталей, допускает неточности, недостаточно правильные формулировки, нарушает последовательность в изложении программного материала и испытывает затруднения в выполнении практических заданий
1-12 рейтинговых баллов	обучающийся не знает значительной части программного материала, допускает существенные ошибки, с большими затруднениями выполняет практические задания
0 рейтинговых баллов	не аттестован

Если результат контроля успеваемости в рамках проведения контрольных мероприятий промежуточной аттестации (рубежный рейтинг обучающегося) неудовлетворительный (получено менее 13 рейтинговых баллов), то промежуточная аттестация по учебной дисциплине (модулю) невозможна даже при наличии высокого текущего рейтинга, полученного по итогам текущего контроля по учебной дисциплине (модулю).

ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

№ п/п	Содержание изменения	Реквизиты документа об утверждении изменения	Дата введения изменения
1.			
2.			
3.			
4.			



**Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Российский государственный социальный университет»**

УТВЕРЖДАЮ

Заведующий кафедрой комплекса естественно-
научных дисциплин

/Денисова Д.А./

«27» февраля 2024 г.

МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ

МЕТОДЫ ОПТИМИЗАЦИИ И ТЕОРИЯ ПРИНЯТИЯ РЕШЕНИЙ

Направление подготовки

«Информационные системы и технологии»

Направленность

«Интеллектуальные информационные системы и технологии»

**ОБРАЗОВАТЕЛЬНАЯ ПРОГРАММА ВЫСШЕГО ОБРАЗОВАНИЯ - ПРОГРАММА
БАКАЛАВРИАТА**

Форма обучения

Очная, заочная

Москва, 2024 г.

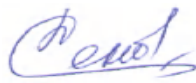
Методические материалы по дисциплине (модулю) «Методы оптимизации и теория принятия решений» разработаны на основании федерального государственного образовательного стандарта высшего образования – *бакалавриата* по направлению подготовки *09.03.02 Информационные системы и технологии*, утвержденного приказом Министерства образования и науки Российской Федерации от 19.09.2017 № 926, учебного плана по основной профессиональной образовательной программе высшего образования – программе бакалавриата по направлению подготовки *09.03.02 Информационные системы и технологии* (далее – «ОПОП»).

Методические материалы по дисциплине (модулю) разработаны рабочей группой в составе: д-р экон. наук, профессор Потехиной Е.В..

Методические материалы по дисциплине (модулю) разработаны на заседании кафедры комплекса естественно-научных дисциплин.

Протокол № 9 от «27» февраля 2024 года

Заведующий кафедрой
канд. техн. наук, доцент



Д.А. Денисова

(подпись)

СОДЕРЖАНИЕ

1. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ К ЛЕКЦИОННЫМ, ЛАБОРАТОРНЫМ ЗАНЯТИЯМ	4
1.1. Методические материалы к проведению лекционных занятий по дисциплине (модулю)	4
1.2. Методические материалы по подготовке к практическим занятиям по дисциплине (модулю)	6
1.4. Методические материалы по подготовке к лабораторным занятиям по дисциплине (модулю)	11
2. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ) И САМОСТОЯТЕЛЬНОЙ РАБОТЕ	12
3. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНИВАНИЯ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ И (ИЛИ) ОПЫТА ДЕЯТЕЛЬНОСТИ, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ.....	21
3.1. Организационные основы применения балльно-рейтинговой системы оценки успеваемости обучающихся по дисциплине (модулю)	21
3.2. Проведение текущего контроля успеваемости обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося.....	21
3.3. Проведение промежуточной аттестации обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося.....	22
ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ	24

1. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ К ЛЕКЦИОННЫМ, ЛАБОРАТОРНЫМ ЗАНЯТИЯМ

1.1. Методические материалы к проведению лекционных занятий по дисциплине (модулю)

Лекция - один из методов обучения, одна из основных системообразующих форм организации учебного процесса в вузе. Лекционное занятие представляет собой систематическое, последовательное, монологическое изложение педагогическим работником учебного материала, как правило, теоретического характера. Такое занятие представляет собой элемент технологии представления учебного материала путем логически стройного, систематически последовательного и ясного изложения.

Цель лекции – организация целенаправленной познавательной деятельности обучающихся по овладению программным материалом дисциплины (модуля). Чтение курса лекций позволяет дать связанное, последовательное изложение материала в соответствии с новейшими данными науки, сообщить слушателям основное содержание предмета в целостном, систематизированном виде. В ряде случаев лекция выполняет функцию основного источника информации: при отсутствии учебников и учебных пособий, чаще по новым курсам; в случае, когда новые научные данные по той или иной теме не нашли отражения в учебниках; отдельные разделы и темы очень сложны для самостоятельного изучения. В таких случаях только лектор может методически помочь обучающимся в освоении сложного материала.

Возможные формы проведения лекций:

- Вводная лекция – один из наиболее важных и трудных видов лекции при чтении систематических курсов. От успеха этой лекции во многом зависит успех усвоения всего курса. Она может содержать: определение дисциплины (модуля); краткую историческую справку о дисциплине (модуле); цели и задачи дисциплины (модуля), ее роль в общей системе обучения и связь со смежными дисциплинами (модулями); основные проблемы (понятия и определения) данной науки; основную и дополнительную учебную литературу; особенности самостоятельной работы обучающихся над дисциплиной (модулем) и формы участия в научно-исследовательской работе; отчетность по курсу.

- Информационная лекция ориентирована на изложение и объяснение обучающимся научной информации, подлежащей осмыслению и запоминанию. Это самый традиционный тип лекций в практике высшей школы.

- Заключительная лекция предназначена для обобщения полученных знаний и раскрытия перспектив дальнейшего развития данной науки.

- Обзорная лекция – это систематизация научных знаний на высоком уровне, допускающая большое число ассоциативных связей в процессе осмысления информации, излагаемой при раскрытии внутрипредметной и межпредметной связей, исключая детализацию и конкретизацию. Как правило, стержень излагаемых теоретических положений составляет научно-понятийная и концептуальная основа всего курса или крупных его разделов.

- Лекция-беседа - непосредственный контакт педагогического работника с аудиторией - диалог. По ходу лекции педагогический работник задает вопросы для выяснения мнений и уровня осведомленности обучающихся по рассматриваемой проблеме.

- Лекция-дискуссия - свободный обмен мнениями в ходе изложения лекционного материала. Педагогический работник активизирует участие в обсуждении отдельными вопросами, сопоставляет между собой различные мнения и тем самым развивает дискуссию, стремясь направить ее в нужное русло.

- Лекция с применением обратной связи включает в себе то, что в начале и конце каждого раздела лекции задаются вопросы. Первый - для того, чтобы узнать, насколько обучающиеся ориентируются в излагаемом материале, вопрос в конце раздела предназначен для выяснения степени усвоения только что изложенного материала. При

неудовлетворительных результатах контрольного опроса педагогический работник возвращается к уже прочитанному разделу, изменив при этом методику подачи материала.

- Проблемная лекция опирается на логику последовательно моделируемых проблемных ситуаций путем постановки проблемных вопросов или предъявления проблемных задач. Проблемный вопрос - это диалектическое противоречие, требующее для своего решения размышления, сравнения, поиска, приобретения и применения новых знаний. Проблемная задача содержит дополнительную вводную информацию и при необходимости некоторые ориентиры поиска ее решения.

- Программированная лекция - консультация – педагогический работник сам составляет и предлагает обучающимся вопросы. На подготовленные вопросы педагогический работник сначала просит ответить обучающихся, а затем проводит анализ и обсуждение неправильных ответов. В лекциях можно использовать наглядные материалы, а также подготовить презентацию. Что касается презентации, то в качестве визуальной поддержки ее можно органично интегрировать во все вышеупомянутые лекции. В то же время лекцию-презентацию возможно выделить и в качестве самостоятельной формы. Лекция-презентация должна отражать суть основных и (или) проблемных вопросов лекции, на которые особо следует обратить внимание обучающихся. В условиях применения активного метода проведения занятий презентация представляется весьма удачным способом донесения информации до слушателей. Единственное, на что следует обратить внимание при подготовке слайдов, - это их оформление и текст. Слайд не должен быть перегружен картинками и лишней информацией, которая будет отвлекать от основного аспекта того или иного вопроса лекции. Во время лекции можно задавать вопросы аудитории в отношении того или иного слайда, тем самым еще больше вовлекая обучающихся в проблематику.

Краткое содержание лекционных занятий

Наименование разделов и тем	Содержание учебного материала
РАЗДЕЛ 1. МОДЕЛИ ЛИНЕЙНОГО ПРОГРАММИРОВАНИЯ	
Тема 1.1. Задачи на построение математической модели задач линейного программирования	Примеры задач линейного программирования. Задача планирования выпуска продукции (планирование производства). Планирование капитальных вложений. Основные определения. Свойства задачи линейного программирования.
Тема 1.2. Графическое решение задач линейного программирования	Геометрическая интерпретация двумерной и многомерной задачи линейного программирования и ее решение. Анализ чувствительности.
Тема 1.3. Графические методы анализа модели на устойчивость	
РАЗДЕЛ 2. АНАЛИТИЧЕСКИЕ МЕТОДЫ РЕШЕНИЯ ЗАДАЧ ЛИНЕЙНОГО ПРОГРАММИРОВАНИЯ	
Тема 2.1. Решение задач линейного программирования симплексным методом	Обоснование симплекс метода. Метод искусственного базиса. Особые случаи симплексного метода. Решение в форме симплекс-таблиц.
Тема 2.2. Двойственная задача	Двойственная задача линейного программирования. Пример прямой и двойственной задачи линейного программирования. Общая формулировка прямой и двойственной задачи. Свойства двойственной задачи. Анализ чувствительности. Экономическая интерпретация двойственной задачи. Объективно обусловленные оценки и их смысл.
Тема 2.3. Транспортная задача	Транспортная задача. Поиск начального опорного плана. Метод северо-западного угла. Метод минимального элемента. Решение транспортной задачи методом потенциалов. Анализ чувствительности. Открытая модель транспортной задачи.
РАЗДЕЛ 2. Наименование раздела	
Тема 3.1. Графический метод	Постановка задачи и основные определения. Геометрическая

решение задачи	интерпретация решения задач нелинейного программирования. Задачи выпуклого программирования.
Тема 3.2. Решение задач нелинейного программирования с ограничениями равенствами	Основные определения и теоремы. Метод неопределенных множителей Лагранжа для решения задач квадратичного программирования.
Тема 3.3. Решение задач нелинейного программирования с ограничениями неравенствами	Метод множителей Лагранжа в задаче нелинейной оптимизации с ограничениями в виде неравенств. Теорема Куна-Таккера.
РАЗДЕЛ 2. Наименование раздела	
Тема 4.1. Задача об оптимальном разделении инвестиций	Задачи динамического программирования. Рекуррентные соотношения Беллмана. Применение алгоритмов динамического программирования к задаче об оптимальном распределении ресурсов. Вычислительные схемы: древовидная, табличная. Задача выбора оптимального пути.
Тема 4.2. Задача выбора стратегии обновления оборудования	Задачи динамического программирования. Рекуррентные соотношения Беллмана. Применение алгоритмов динамического программирования к задаче об оптимальном распределении ресурсов. Вычислительные схемы: древовидная, табличная. Задача о замене оборудования.
Тема 4.3. Выбор оптимального пути в транспортной сети	Задачи динамического программирования. Рекуррентные соотношения Беллмана. Применение алгоритмов динамического программирования к задаче об оптимальном распределении ресурсов. Вычислительные схемы: древовидная, табличная. Задача о распределении средств между предприятиями.
РАЗДЕЛ 2. Наименование раздела	
Тема 5.1. Сетевая модель и ее основные элементы	Сетевая модель и ее основные элементы. Правила построения сетевых графиков.
Тема 5.2. Сетевое планирование в условиях неопределенности	Методы сетевого планирования и управления при планировании сложных комплексных проектах управления. Сетевое планирование в условиях неопределенности.
Тема 5.3. Коэффициент напряженности работы. Анализ и оптимизация сетевого графика	Временные параметры сетевых графиков и коэффициент напряженности. Необходимость оптимизации сетевого графика.

1.2. Методические материалы по подготовке к практическим занятиям по дисциплине (модулю)

Практические (семинарские) занятия - одна из форм учебного занятия, направленная на развитие самостоятельности обучающихся и приобретение умений и навыков. Данные учебные занятия углубляют, расширяют, детализируют полученные ранее знания. Практическое занятие предполагает выполнение обучающимися по заданию и под руководством преподавателей одной или нескольких практических работ.

Цель практических занятий и семинаров состоит в развитии познавательных способностей, самостоятельности мышления и творческой активности обучающихся; углублении, расширении, детализировании знаний, полученных на лекции в обобщенной форме, и содействии выработке навыков профессиональной деятельности. В отдельных случаях на практических занятиях и семинарах руководителем занятия сообщаются дополнительные знания.

Для достижения поставленных целей и решения требуемого перечня задач практические занятия и семинары проводятся традиционными технологиями или с использованием активных и интерактивных образовательных технологий.

Возможные формы проведения практических (семинарских) занятий:

- Деловая игра - это метод группового обучения совместной деятельности в процессе решения общих задач в условиях максимально возможного приближения к реальным

проблемным ситуациям. Имитационные игры - на занятиях имитируется деятельность какой-либо организации, предприятия или его подразделения. Имитироваться могут события, конкретная деятельность людей (деловое совещание, обсуждение плана) и обстановка, условия, в которых происходит событие или осуществляется деятельность (кабинет начальника цеха, зал заседаний). Исполнение ролей (ролевые игры) - в этих играх отрабатывается тактика поведения, действий, выполнение функций и обязанностей конкретного лица. Для проведения игр с исполнением роли разрабатывается модель-пьеса ситуации, между студентами распределяются роли с «обязательным содержанием», характеризующиеся различными интересами; в процессе их взаимодействия должно быть найдено компромиссное решение. «Деловой театр» (метод инсценировки) - в нем разыгрывается какая-либо ситуация, поведение человека в этой обстановке, обучающийся должен вжиться в образ определенного лица, понять его действия, оценить обстановку и найти правильную линию поведения. Основная задача метода инсценировки - научить ориентироваться в различных обстоятельствах, давать объективную оценку своему поведению, учитывать возможности других людей, влиять на их интересы, потребности и деятельность, не прибегая к формальным атрибутам власти, к приказу.

- Игровое проектирование - является практическим занятием или циклом занятий, суть которых состоит в разработке инженерного, конструкторского, технологического и других видов проектов в игровых условиях, максимально воссоздающих реальность. Этот метод отличается высокой степенью сочетания индивидуальной и совместной работы обучающихся.

- Познавательно-дидактические игры не относятся к деловым играм. Они предполагают лишь включение изучаемого материала в необычный игровой контекст и иногда содержат лишь элементы ролевых игр. Такие игры могут проводиться в виде копирования научных, культурных, социальных явлений (конкурс знатоков, «Поле чудес», КВН и т.д.) и в виде предметно-содержательных моделей, (например, игры-путешествия, когда надо разработать рациональный маршрут, пользуясь различными картами).

- Анализ конкретных ситуаций. Конкретная ситуация – это любое событие, которое содержит в себе противоречие или вступает в противоречие с окружающей средой. Ситуации могут нести в себе как позитивный, так и отрицательный опыт. Все ситуации делятся на простые, критические и экстремальные.

- Кейс-метод (от английского case – случай, ситуация) – усовершенствованный метод анализа конкретных ситуаций, метод активного проблемно-ситуационного анализа, основанный на обучении путем решения конкретных задач – ситуаций (решение кейсов). Непосредственная цель метода case-study - обучающиеся должны проанализировать ситуацию, разобраться в сути проблем, предложить возможные решения и выбрать лучшее из них. Кейсы делятся на практические (отражающие реальные жизненные ситуации), обучающие (искусственно созданные, содержащие значительные элементы условности при отражении в нем жизни) и исследовательские (ориентированные на проведение исследовательской деятельности посредством применения метода моделирования). Метод конкретных ситуаций (метод case-study) относится к неигровым имитационным активным методам обучения.

- Тренинг (англ. training от train — обучать, воспитывать) – метод активного обучения, направленный на развитие знаний, умений и навыков и социальных установок. Тренинг – форма интерактивного обучения, целью которого является развитие компетентности межличностного и профессионального поведения в общении. Достоинство тренинга заключается в том, что он обеспечивает активное вовлечение всех участников в процесс обучения. Можно выделить основные типы тренингов по критерию направленности воздействия и изменений – навыковый, психотерапевтический, социально-психологический, бизнес-тренинг.

- Метод Сократа (Майевтика) – метод вопросов, предполагающих критическое отношение к догматическим утверждениям, называется еще как метод «сократовской

иронии». Это умение извлекать скрытое в человеке знание с помощью искусных наводящих вопросов, подразумевающего короткий, простой и заранее предсказуемый ответ.

- Интерактивная лекция – выступление ведущего обучающего перед большой аудиторией с применением следующих активных форм обучения: дискуссия, беседа, демонстрация слайдов или учебных фильмов, мозговой штурм.

- Групповая, научная дискуссия, диспут. Дискуссия — это целенаправленное обсуждение конкретного вопроса, сопровождающееся обменом мнениями, идеями между двумя и более лицами. Задача дискуссии - обнаружить различия в понимании вопроса и в споре установить истину. Дискуссии могут быть свободными и управляемыми. К технике управляемой дискуссии относятся: четкое определение цели, прогнозирование реакции оппонентов, планирование своего поведения, ограничение времени на выступления и их заданная очередность. Разновидностью свободной дискуссии является форум, где каждому желающему дается неограниченное время на выступление, при условии, что его выступление вызывает интерес аудитории. Каждый конкретный форум имеет свою тематику — достаточно широкую, чтобы в её пределах можно было вести многоплановое обсуждение.

- Дебаты – это чётко структурированный и специально организованный публичный обмен мыслями между двумя сторонами по актуальным темам. Это разновидность публичной дискуссии участников дебатов, направляющая на переубеждение в своей правоте третьей стороны, а не друг друга. Поэтому вербальные и невербальные средства, которые используются участниками дебатов, имеют целью получения определённого результата — сформировать у слушателей положительное впечатление от собственной позиции.

- Метод работы в малых группах. Групповое обсуждение кого-либо вопроса направлено на достижение лучшего взаимопонимания и нахождения истины. Групповое обсуждение способствует лучшему усвоению изучаемого материала. Оптимальное количество участников - 5-7 человек. Перед обучающимися ставится проблема, выделяется определенное время, в течение которого они должны подготовить аргументированный обдуманный ответ. Педагогический работник может устанавливать правила проведения группового обсуждения – задавать определенные рамки обсуждения, ввести алгоритм выработки общего мнения, назначить лидера и др.

- Круглый стол - общество, собрание в рамках более крупного мероприятия (съезда, симпозиума, конференции). Мероприятие, как правило, на которое приглашаются эксперты и специалисты из разных сфер деятельности для обсуждения актуальных вопросов. Данная модель обсуждения, основываясь на соглашениях, в качестве итогов даёт результаты, которые, в свою очередь, являются новыми соглашениями.

- Коллоквиум - (лат. colloquium — разговор, беседа) - одна из форм учебных занятий в системе образования, имеющая целью выяснение и повышение знаний обучающихся. На коллоквиумах обсуждаются: отдельные части, разделы, темы, вопросы изучаемого курса (обычно не включаемые в тематику семинарских и других практических учебных занятий), рефераты, проекты и др. работы обучающихся. Это научные собрания, на которых заслушиваются и обсуждаются доклады. Коллоквиум – это и форма контроля, массового опроса, позволяющая преподавателю в сравнительно небольшой срок выяснить уровень знаний студентов по данной теме дисциплины. Коллоквиум проходит обычно в форме дискуссии, в ходе которой обучающимся предоставляется возможность высказать свою точку зрения на рассматриваемую проблему, учиться обосновывать и защищать ее. Аргументируя и отстаивая свое мнение, обучающийся в то же время демонстрирует, насколько глубоко и осознанно он усвоил изученный материал.

- Метод «мозговой штурм» (мозговой штурм, мозговая атака, англ. brainstorming) — оперативный метод решения проблемы на основе стимулирования творческой активности, при котором участникам обсуждения предлагают высказывать как можно большее количество вариантов решения, в том числе самых фантастичных. Затем из общего числа высказанных идей отбирают наиболее удачные, которые могут быть использованы на практике. Является методом экспертного оценивания.

- Метод проектов - это способ достижения дидактической цели через детальную разработку проблемы (технологию), которая должна завершиться вполне реальным, осязаемым практическим результатом, оформленным тем или иным образом; это совокупность приёмов, действий обучающихся в их определённой последовательности для достижения поставленной задачи – решения проблемы, лично значимой для учащихся и оформленной в виде некоего конечного продукта. Основное предназначение метода проектов состоит в предоставлении учащимся возможности самостоятельного приобретения знаний в процессе решения практических задач или проблем, требующего интеграции знаний из различных предметных областей.

- Брифинг - (англ. briefing от англ. brief – короткий, недолгий) – краткая пресс-конференция, посвященная одному вопросу. Основное отличие: отсутствует презентационная часть. То есть практически сразу идут ответы на вопросы журналистов.

- Метод портфолио (итал. portfolio — 'портфель, англ. - папка для документов) - современная образовательная технология, в основе которой используется метод аутентичного оценивания результатов образовательной и профессиональной деятельности. Портфолио как подборка сертифицированных достижений, наиболее значимых работ и отзывов на них.

Вопросы для самоподготовки к практическим (семинарским) занятиям по разделам (темам) дисциплины (модуля)

Задания для самостоятельной работы к Разделу 1

Вопросы для самостоятельной работы к Разделу 1

1. Какие задачи линейного программирования решаются методом искусственного базиса?
2. Как составляется расширенная задача?
3. В каком случае можно сократить количество вводимых искусственных переменных?
4. При каком условии оптимальный план расширенной задачи является оптимальным планом исходной задачи?
5. С какими коэффициентами искусственные переменные вводятся в целевую функцию в задачах, а) «на максимум»; б) «на минимум»?
6. Привести примеры многокритериальных задач линейного программирования и описать методы их решения.

Задания для самостоятельной работы к Разделу 2

Вопросы для самостоятельной работы к Разделу 2

1. Чем отличаются симплексные таблицы для реализации симплексного метода и метода искусственного базиса?
2. Как определяется вектор, вводимый в базис, при использовании искусственного базиса?
3. Когда исходная задача не имеет решения и как это определить, решая расширенную задачу?
4. В чем заключается сущность двойственности в линейном программировании?
5. Какие пары двойственных задач относятся к симметричным и несимметричным?
6. Как формулируются правила построения двойственной задачи?
7. Как формулируются основные теоремы двойственности?
8. Как по решению исходной (двойственной) задачи найти решение двойственной

(исходной) задачи?

9. Какова экономическую интерпретацию двойственной задачи, если исходная задача состоит в оптимальном использовании ресурсов?

10. Как формулируются экономический смысл и значение теорем двойственности?

11. Как определить рентабельность каждого вида продукции, используя двойственные оценки?

12. Как проводится экономический анализ показателей симплексной таблицы с оптимальным планом?

13. Как строится система потенциалов?

14. Как определяются оценки свободных клеток?

15. Как звучит критерий оптимальности плана транспортной задачи?

16. Каково определение цикла по переброске «грузов» в транспортной задаче. Как он строится?

17. Как осуществляется переброска объемов груза по вершинам цикла?

18. В чем заключается суть блокирования перевозок?

19. Как осуществляется блокирование перевозок

20. Почему распределительные задачи называются λ – задачами?

21. Какой экономический смысл имеет параметр λ ?

22. Какова постановка задачи о загрузке оборудования?

23. Как записывается ЭММ задачи о загрузке оборудования?

24. Как λ – задача сводится к транспортной?

25. Какой станок принимается как базовый? Чему равен индекс базового станка?

26. Как вычисляются индексы остальных станков?

27. В каких единицах следует выразить все величины распределительной задачи для сведения ее к транспортной?

Задания для самостоятельной работы к Разделу 3

Вопросы для самостоятельной работы к Разделу 3

1. Постановка и геометрическая интерпретация задачи.
2. Теорема Вейерштрасса о наибольшем и наименьшем экстремуме.
3. Определение выпуклого множества. Примеры.
4. Выпуклые функции. Свойства и признаки выпуклых функций. Примеры.
5. Функция Лагранжа в задаче нелинейной оптимизации.
6. Метод множителей Лагранжа в задаче нелинейной оптимизации с ограничениями в виде равенств. Определение критических точек задачи условного экстремума.
7. Метод множителей Лагранжа в задаче нелинейной оптимизации с ограничениями в виде равенств. Достаточные условия условного экстремума двумерной задачи.
8. Метод множителей Лагранжа в задаче нелинейной оптимизации с ограничениями в виде неравенств. Теорема Куна-Таккера.
9. Метод неопределенных множителей Лагранжа для решения задач квадратичного программирования.

Задания для самостоятельной работы к Разделу 4

Вопросы для самостоятельной работы к Разделу 4

1. Постановка задач динамического программирования. Принцип оптимальности

Беллмана. Уравнения Беллмана.

2. Решение задачи о кратчайшем пути через сеть методом динамического программирования.

3. Решение задачи распределения ресурсов методом динамического программирования.

4. Решение задачи замены оборудования методом динамического программирования.

5. Решение задачи о рюкзаке методом динамического программирования.

Задания для самостоятельной работы к Разделу 5

Вопросы для самостоятельной работы к Разделу 5

1. Структурные таблицы методом сетевого программирования.

2. Временные параметры событий сетевого графика. Нахождение критического пути с помощью временных параметров.

3. Основные элементы сетевой модели.

4. Структурные таблицы методом сетевого программирования.

5. Временные параметры событий сетевого графика. Нахождение критического пути с помощью временных параметров.

6. Сетевое планирование в условиях напряженности. Коэффициент напряженности, что показывает и как определяется?

7. В чем суть оптимизации сетевого графика?

1.4. Методические материалы по подготовке к лабораторным занятиям по дисциплине (модулю)

Лабораторные занятия - одна из форм учебного занятия, направленная на развитие самостоятельности обучающихся и приобретение умений и навыков. Данные учебные занятия углубляют, расширяют, детализируют полученные ранее знания. Лабораторное занятие предполагает выполнение обучающимися по заданию и под руководством преподавателей одной или нескольких работ.

Цель лабораторных занятий состоит в развитии познавательных способностей, самостоятельности мышления и творческой активности обучающихся; углублении, расширении, детализировании знаний, полученных на лекции в обобщенной форме, и содействии выработке навыков профессиональной деятельности. В отдельных случаях на лабораторных занятиях руководителем занятия сообщаются дополнительные знания.

Для достижения поставленных целей и решения требуемого перечня задач лабораторные занятия проводятся традиционными технологиями или с использованием активных и интерактивных образовательных технологий.

При подготовке и работе во время проведения лабораторных занятий следует обратить внимание на следующие моменты: на процесс предварительной подготовки, на работу во время занятия, обработку полученных результатов, исправление полученных замечаний.

Предварительная подготовка к лабораторному занятию заключается в изучении теоретического материала в отведенное для самостоятельной работы время, ознакомление с инструктивными материалами с целью осознания задач занятия, техники безопасности при работе с приборами, веществами.

Работа во время проведения занятия включает:

– консультирование студентов преподавателями и вспомогательным персоналом с целью предоставления исчерпывающей информации, необходимой для самостоятельного выполнения предложенных преподавателем задач.

– самостоятельное выполнение заданий согласно обозначенной рабочей программой дисциплины (модуля) тематики.

2. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ) И САМОСТОЯТЕЛЬНОЙ РАБОТЕ

Освоение обучающимся дисциплины (модуля) «Методы оптимизации и теория принятия решений (модуля)» предполагает изучение материалов дисциплины на аудиторных занятиях и в ходе самостоятельной работы. Аудиторные занятия проходят в форме лекций, семинаров, практических и лабораторных занятий. Самостоятельная работа включает разнообразный комплекс видов и форм работы обучающихся.

Для успешного освоения дисциплины (модуля) и достижения поставленных целей необходимо внимательно ознакомиться с рабочей программы дисциплины (модуля), доступной в электронной информационно-образовательной среде РГСУ.

Следует обратить внимание на списки основной и дополнительной литературы, на предлагаемые преподавателем ресурсы информационно-телекоммуникационной сети Интернет. Эта информация необходима для самостоятельной работы обучающегося.

При подготовке к аудиторным занятиям необходимо помнить особенности каждой формы его проведения.

Подготовка к учебному занятию лекционного типа заключается в следующем.

С целью обеспечения успешного обучения обучающийся должен готовиться к лекции, поскольку она является важнейшей формой организации учебного процесса, поскольку:

- знакомит с новым учебным материалом;
- разъясняет учебные элементы, трудные для понимания;
- систематизирует учебный материал;
- ориентирует в учебном процессе.

С этой целью:

- внимательно прочитайте материал предыдущей лекции;
- ознакомьтесь с учебным материалом по учебнику и учебным пособиям с темой прочитанной лекции;
- внесите дополнения к полученным ранее знаниям по теме лекции на полях лекционной тетради;
- запишите возможные вопросы, которые вы зададите лектору на лекции по материалу изученной лекции;
- постарайтесь уяснить место изучаемой темы в своей подготовке;
- узнайте тему предстоящей лекции (по тематическому плану, по информации лектора) и запишите информацию, которой вы владеете по данному вопросу.

Подготовка к занятию семинарского типа.

При подготовке и работе во время проведения занятий семинарского типа следует обратить внимание на следующие моменты: на процесс предварительной подготовки, на работу во время занятия, обработку полученных результатов, исправление полученных замечаний.

Предварительная подготовка к учебному занятию семинарского типа заключается в изучении теоретического материала в отведенное для самостоятельной работы время, ознакомление с инструктивными материалами с целью осознания задач практического занятия, техники безопасности при работе с приборами, веществами.

Работа во время проведения учебного занятия семинарского типа включает:

- консультирование студентов преподавателями и вспомогательным персоналом с целью предоставления исчерпывающей информации, необходимой для самостоятельного выполнения предложенных преподавателем задач.
- самостоятельное выполнение заданий согласно обозначенной рабочей программой дисциплины (модуля) тематики.

Самостоятельная работа.

Самостоятельная работа - планируемая учебная, учебно-исследовательская, научно-исследовательская работа студентов, выполняемая во внеаудиторное (аудиторное) время по

заданию и при методическом руководстве преподавателя, но без его непосредственного участия (при частичном непосредственном участии преподавателя, оставляющем ведущую роль за работой студентов).

Для более углубленного изучения темы задания для самостоятельной работы рекомендуется выполнять параллельно с изучением данной темы. При выполнении заданий по возможности используйте наглядное представление материала.

Самостоятельная работа студентов в ВУЗе является важным видом учебной и научной деятельности студента. Самостоятельная работа студентов играет значительную роль в рейтинговой технологии обучения.

К современному специалисту в области медицины общество предъявляет достаточно широкий перечень требований, среди которых немаловажное значение имеет наличие у выпускников определенных навыков (компетенций) и умения самостоятельно добывать знания из различных источников, систематизировать полученную информацию, давать оценку конкретной профессиональной ситуации. Формирование такого умения происходит в течение всего периода обучения.

В процессе самостоятельной работы студент приобретает необходимые для будущей специальности компетенции, навыки самоорганизации, самоконтроля, самоуправления, саморефлексии и становится активным самостоятельным субъектом учебной деятельности.

Самостоятельная работа студентов должна оказывать важное влияние на формирование личности будущего специалиста, его компетентность. Каждый студент самостоятельно определяет режим своей работы и меру труда, затрачиваемого на овладение учебным содержанием по каждой дисциплине. Он выполняет внеаудиторную работу по личному индивидуальному плану, в зависимости от его подготовки, времени и других условий.

Виды самостоятельной работы.

Работа с литературой.

При работе с книгой необходимо подобрать литературу, научиться правильно ее читать, вести записи. Для подбора литературы в библиотеке используются алфавитный и систематический каталоги. Важно помнить, что рациональные навыки работы с книгой - это всегда большая экономия времени и сил. Правильный подбор учебников рекомендуется преподавателем, читающим лекционный курс. Необходимая литература может быть также указана в методических разработках по данному курсу. Изучая материал по учебнику, следует переходить к следующему вопросу только после правильного уяснения предыдущего, описывая на бумаге все выкладки и вычисления (в том числе те, которые в учебнике опущены или на лекции даны для самостоятельного вывода). При изучении любой дисциплины большую и важную роль играет самостоятельная индивидуальная работа. Особое внимание следует обратить на определение основных понятий курса. Студент должен подробно разбирать примеры, которые поясняют такие определения, и уметь строить аналогичные примеры самостоятельно. Нужно добиваться точного представления о том, что изучаешь. Полезно составлять опорные конспекты. При изучении материала по учебнику полезно в тетради (на специально отведенных полях) дополнять конспект лекций. Там же следует отмечать вопросы, выделенные студентом для консультации с преподавателем. Выводы, полученные в результате изучения, рекомендуется в конспекте выделять, чтобы они при перечитывании записей лучше запоминались. Опыт показывает, что многим студентам помогает составление листа опорных сигналов, содержащего важнейшие и наиболее часто употребляемые формулы и понятия. Такой лист помогает запомнить формулы, основные положения лекции, а также может служить постоянным справочником для студента. Различают два вида чтения: первичное и вторичное. Первичное - это внимательное, неторопливое чтение, при котором можно остановиться на трудных местах.

Задача вторичного чтения - полное усвоение смысла целого (по счету это чтение может быть и не вторым, а третьим или четвертым). Самостоятельная работа с учебниками и книгами (а также самостоятельное теоретическое исследование проблем, обозначенных

преподавателем на лекциях) – это важнейшее условие формирования у себя научного способа познания.

Чтение научного текста является частью познавательной деятельности. Ее цель – извлечение из текста необходимой информации. От того, насколько осознанна читающим собственная внутренняя установка при обращении к печатному слову (найти нужные сведения, усвоить информацию полностью или частично, критически проанализировать материал и т.п.) во многом зависит эффективность осуществляемого действия.

Методические рекомендации по составлению конспекта:

1. Внимательно прочитайте текст. Уточните в справочной литературе непонятные слова. При записи не забудьте вынести справочные данные на поля конспекта;

2. Выделите главное, составьте план;

3. Кратко сформулируйте основные положения текста, отметьте аргументацию автора;

4. Законспектируйте материал, четко следуя пунктам плана. При конспектировании старайтесь выразить мысль своими словами. Записи следует вести четко, ясно.

5. Грамотно записывайте цитаты. Цитируя, учитывайте лаконичность, значимость мысли. В тексте конспекта желательно приводить не только тезисные положения, но и их доказательства. При оформлении конспекта необходимо стремиться к емкости каждого предложения. Мысли автора книги следует излагать кратко, заботясь о стиле и выразительности написанного. Число дополнительных элементов конспекта должно быть логически обоснованным, записи должны распределяться в определенной последовательности, отвечающей логической структуре произведения. Для уточнения и дополнения необходимо оставлять поля. Овладение навыками конспектирования требует от студента целеустремленности, повседневной самостоятельной работы.

Методические материалы по самостоятельному решению задач

При самостоятельном решении задач нужно обосновывать каждый этап решения, исходя из теоретических положений курса. Если студент видит несколько путей решения проблемы (задачи), то нужно сравнить их и выбрать самый рациональный. Полезно до начала вычислений составить краткий план решения проблемы (задачи). Решение проблемных задач или примеров следует излагать подробно, вычисления располагать в строгом порядке, отделяя вспомогательные вычисления от основных. Решения при необходимости нужно сопровождать комментариями, схемами, чертежами и рисунками. Следует помнить, что решение каждой учебной задачи должно доводиться до окончательного логического ответа, которого требует условие, и по возможности с выводом. Полученный ответ следует проверить способами, вытекающими из существа данной задачи. Полезно также (если возможно) решать несколькими способами и сравнить полученные результаты. Решение задач данного типа нужно продолжать до приобретения твердых навыков в их решении.

Методические материалы к выполнению реферата

Реферат (от лат. *refere* – сообщать) – краткое изложение в письменном виде или в форме публикации доклада, содержания научного труда (трудов), литературы по теме. Работа над рефератом условно разделяется на выбор темы, подбор литературы, подготовку и защиту плана; написание теоретической части и всего текста с указанием библиографических данных используемых источников, подготовку доклада, выступление с ним. Тематика рефератов полностью связана с основными вопросами изучаемого курса.

Список литературы к темам не дается, и обучающиеся самостоятельно ведут библиографический поиск, причем им не рекомендуется ограничиваться университетской библиотекой.

Важно учитывать, что написание реферата требует от обучающихся определенных усилий. Особое внимание следует уделить подбору литературы, методике ее изучения с целью отбора и обработки собранного материала, обоснованию актуальности темы и теоретического уровня обоснованности используемых в качестве примеров фактов какой-либо деятельности.

Выбрав тему реферата, начав работу над литературой, необходимо составить план. Изучая литературу, продолжается обдумывание темы, осмысливание прочитанного, делаются

выписки, сопоставляются точки зрения разных авторов и т.д. Реферативная работа сводится к тому, чтобы в ней выделились две взаимосвязанные стороны: во-первых, ее следует рассматривать как учебное задание, которое должен выполнить обучаемый, а во-вторых, как форму научной работы, творческого воображения при выполнении учебного задания.

Наличие плана реферата позволяет контролировать ход работы, избежать формального переписывания текстов из первоисточников.

Оформление реферата включает титульный лист, оглавление и краткий список использованной литературы. Список использованной литературы размещается на последней странице рукописи или печатной форме реферата. Реферат выполняется в письменной или печатной форме на белых листах формата А4 (210 x 297 мм). Шрифт Times New Roman, кегель 14, через 1,5 интервала при соблюдении следующих размеров текста: верхнее поле – 25 мм, нижнее – 20 мм, левое – 30 мм, правое – 15 мм. Нумерация страниц производится вверху листа, по центру. Титульный лист нумерации не подлежит.

Рефераты должны быть написаны простым, ясным языком, без претензий на наукообразность. Следует избегать сложных грамматических оборотов, непривычных терминов и символов. Если же такие термины и символы все-таки приводятся, то необходимо разъяснять их значение при первом упоминании в тексте реферата.

Объем реферата предполагает тщательный отбор информации, необходимой для краткого изложения вопроса. Важнейший этап – редактирование готового текста реферата и подготовка к обсуждению. Обсуждение требует хорошей ориентации в материале темы, умения выделить главное, поставить дискуссионный вопрос, привлечь внимание слушателей к интересной литературе, логично и убедительно изложить свои мысли.

Рефераты обязательно подлежат защите. Процедура защиты начинается с определения оппонентов защищающего свою работу. Они стремятся дать основательный анализ работы обучающимся, обращают внимание на положительные моменты и недостатки реферата, дают общую оценку содержанию, форме преподнесения материала, характеру использованной литературы. Иногда они дополняют тот или иной раздел реферата. Последнее особенно ценно, ибо говорит о глубоком знании обучающимся-оппонентом изучаемой проблемы.

Обсуждение не ограничивается заслушиванием оппонентов. Другие обучающиеся имеют право уточнить или опровергнуть какое-либо утверждение. Преподаватель предлагает любому обучающемуся задать вопрос по существу доклада или попытаться подвести итог обсуждению.

Алгоритм работы над рефератом

1. Выбор темы

Тема должна быть сформулирована грамотно (с литературной точки зрения);

В названии реферата следует поставить четкие рамки рассмотрения темы;

Желательно избегать слишком длинных названий;

Следует по возможности воздерживаться от использования в названии спорных с научной точки зрения терминов, излишней наукообразности, а также чрезмерного упрощения формулировок.

2. Реферат следует составлять из пяти основных частей: введения; основной части; заключения; списка литературы; приложений.

3. Основные требования к введению:

Во введении не следует концентрироваться на содержании; введение должно включать краткое обоснование актуальности темы реферата, где требуется показать, почему данный вопрос может представлять научный интерес и есть ли связь представляемого материала с современностью. Таким образом, тема реферата должна быть актуальна либо с научной точки зрения, либо с современных позиций.

Очень важно выделить цель, а также задачи, которые требуется решить для выполнения цели.

Введение должно содержать краткий обзор изученной литературы, в котором указывается взятый из того или иного источника материал, кратко анализируются изученные источники, показываются их сильные и слабые стороны;

Объем введения составляет две страницы текста.

4. Требования к основной части реферата:

Основная часть содержит материал, отобранный для рассмотрения проблемы;

Также основная часть должна включать в себя собственно мнение обучающихся и сформулированные самостоятельные выводы, опирающиеся на приведенные факты;

Материал, представленный в основной части, должен быть логически изложен и распределен по параграфам, имеющим свои названия;

В изложении основной части необходимо использовать сноски (в первую очередь, когда приводятся цифры и чьи-то цитаты);

Основная часть должна содержать иллюстративный материал (графики, таблицы и т. д.);

Объем основной части составляет около 10 страниц.

5. Требования к заключению:

В заключении формулируются выводы по параграфам, обращается внимание на выдвинутые во введении задачи и цели;

Заключение должно быть четким, кратким, вытекающим из содержания основной части.

6. Требования к оформлению списка литературы (по ГОСТу):

Необходимо соблюдать правильность последовательности записи источников: сначала следует писать фамилию, а после инициалы; название работы не ставится в кавычки; после названия сокращенно пишется место издания; затем идет год издания; наконец, называется процитированная страница.

Критерии оценки реферата

Обучающийся, защищающий реферат, должен рассказать о его актуальности, поставленных целях и задачах, изученной литературе, структуре основной части, сделанных в ходе работы выводах.

По окончании выступления ему может быть задано несколько вопросов по представленной проблеме.

Оценка складывается из соблюдения требований к реферату, грамотного раскрытия темы, умения четко рассказывать о представленном реферате, способности понять суть задаваемых по работе вопросов и найти точные ответы на них.

Методические материалы к выполнению эссе

Эссе – литературное произведение небольшого объема, обычно прозаическое, свободной композиции, передающее индивидуальные впечатления, суждения, соображения автора о той или иной проблеме, теме, о том или ином событии или явлении. Это вид самостоятельной исследовательской работы обучающихся, с целью углубления и закрепления теоретических знаний и освоения практических навыков. Цель эссе состоит в развитии самостоятельного творческого мышления и письменного изложения собственных мыслей. При написании эссе обучающийся должен представить развернутый письменный ответ на теоретический или практический актуальный вопрос, объявленный преподавателем в аудитории непосредственно перед ее написанием. В процессе написания эссе разрешается пользоваться нормативно-правовыми актами, конспектом лекций (в печатном виде). Использование интернет-ресурсов не допускается. Темы эссе преподаватель предлагает из числа тех, которые обучающиеся уже рассматривали на лекциях или семинарских занятиях, исходя из содержания заданий в составе оценочных средств. По решению преподавателя, в качестве темы эссе может быть выбрана одна или несколько тем, которые могут быть распределены между обучающимися по желанию.

Требования к выполнению эссе:

1. Проводится письменно.

2. Эссе выполняется на компьютере (гарнитура Times New Roman, шрифт 14) через 1,5 интервала с полями: верхнее, нижнее – 2; правое – 3; левое – 1,5. Отступ первой строки абзаца – 1,25. Сноски – постраничные. Таблицы и рисунки встраиваются в текст работы. При этом обязательный заголовок таблицы надо размещать над табличным полем, а рисунки сопровождать подрисуночными подписями. При включении в эссе нескольких таблиц и/или рисунков их нумерация обязательна. Обязательна и нумерация страниц. Их целесообразно проставлять внизу страницы – по середине или в правом углу. Номер страницы не ставится на титульном листе, но в общее число страниц он включается. Объем эссе, без учета приложений, не должен превышать 5 страниц. Значительное превышение установленного объема является недостатком работы и указывает на то, что обучающийся не сумел отобрать и переработать необходимый материал.

3. Работа должна содержать собственные умозаключения по сути поставленной проблемы, включать самостоятельно проведенный анализ по сути этой проблемы, выводы, обобщающие авторскую позицию по поставленной проблеме.

Критерии оценки эссе:

«Отлично» – исключительные знания материала, абсолютное понимание сути, безукоризненное знание основных понятий и положений, логически и лексически грамотно изложенный, содержательный, аргументированный, конкретный и исчерпывающий ответ.

«Хорошо» – глубокие знания материала, правильное понимание сути, знание основных понятий и положений, содержательный, полный и конкретный ответ.

«Удовлетворительно» – твердые, но недостаточно полные знания, верное понимание сути, в целом правильный ответ.

«Неудовлетворительно» – непонимание сущности задания, грубые ошибки в ответе.

Методические материалы по выполнению тестирования.

Тестовые задания содержат вопросы и 3-4 варианта ответа по базовым положениям изучаемой темы/раздела, составлены с расчетом на знания, полученные обучающимся в процессе изучения темы/раздела.

Тестовые задания выполняются в письменной форме и сдаются преподавателю, ведущему дисциплину (модуль). На выполнение тестовых заданий обучающимся отводится 45 минут.

При обработке результатов оценочной процедуры используются: критерии оценки по содержанию и качеству полученных ответов, ключи, оценочные листы.

Критерии оценки теста:

«Зачтено» - если обучающийся правильно выполнил не менее 2/3 всей работы или допустил не более одной грубой ошибки и двух недочетов, не более одной грубой и одной негрубой ошибки, не более трех негрубых ошибок, одной негрубой ошибки и трех недочетов, при наличии четырех-пяти недочетов.

«Не зачтено» - если число ошибок и недочетов превысило норму для оценки 3 или правильно выполнено менее 2/3 всей работы.

Методические материалы по выполнению доклада.

Рекомендуется следующая структура доклада:

1. титульный лист, содержание доклада;
2. краткое изложение;
3. цели и задачи;
4. изложение характера исследований и рассмотренных проблем, гипотезы, спорные вопросы;
5. источники информации, методы сбора и анализа данных, степень их полноты и достоверности;
6. анализ и толкование полученных в работе результатов;
7. выводы и оценки;
8. библиография и приложения.

Время выступления докладчика не должно превышать 10 минут.

Основные требования к оформлению доклада:

- титульный лист должен включать название доклада, наименование предметной (цикловой) комиссии, фамилию обучающегося;
- все использованные литературные источники сопровождаются библиографическим описанием;
- приводимая цитата из источника берется в кавычки (оформляются сноски);
- единицы измерения должны применяться в соответствии с действующими стандартами;
- все названия литературных источников следует приводить в соответствии с новейшими изданиями;
- рекомендуется включение таблиц, графиков, схем, если они отражают основное содержание или улучшают ее наглядность;
- названия фирм, учреждений, организаций и предприятий должны именоваться так, как они указываются в источнике;

Критерии оценки доклада

При выполнении доклада обучающийся должен продемонстрировать умение кратко излагать прочитанный материал, а также умение обобщать и анализировать материал по теме доклада.

Презентация

Методические материалы к презентациям

1. Объем презентации 10 -20 слайдов.
2. На титульном слайде должно быть отражено:
 - наименование факультета;
 - тема презентации;
 - фамилия, имя, отчество, направление подготовки/ специальность, направленность (профиль)/ специализация, форма обучения, номер группы автора презентации;
 - фамилия, имя, отчество, степень, звание, должность руководитель работы;
 - год выполнения работы.
3. В презентации должны быть отражены обоснование актуальности представляемого материала, цели и задачи работы.
4. Содержание презентации должно включать наиболее значимый материал доклада, а также, при необходимости, таблицы, диаграммы, рисунки, фотографии, карты, видео – вставки, звуковое сопровождение.
5. Заключительный слайд должен содержать информацию об источниках информации для презентации.

Критерии оценки презентации

1. Объем презентации 10 -20 слайдов.
2. Правильность оформления титульного слайда.
3. Актуальность отобранного материала, обоснованность формулировки цели и задач работы.
4. Наглядность и логичность презентации, обоснованность использования таблиц, диаграмм, рисунков, фотографий, карт, видео – вставок, звукового сопровождения; правильный выбор шрифтов, фона, других элементов дизайна слайда.
5. Объем и качество источников информации (не менее 2-х интернет – источников и не менее 2-х литературных источников).

Методические материалы по подготовке к опросу

Самостоятельная работа обучающихся включает подготовку к опросу на практических занятиях. Для этого обучающийся изучает лекции, основную и дополнительную литературу, публикации, информацию из Интернет-ресурсов.

Тема и вопросы к практическим занятиям, вопросы для самоконтроля содержатся в рабочей программе и доводятся до студентов заранее.

Для подготовки к опросу обучающемуся необходимо ознакомиться с материалом, посвященным теме практического занятия, в учебнике или другой рекомендованной литературе, конспекте лекции, обратить внимание на усвоение основных понятий дисциплины, выявить неясные вопросы и подобрать дополнительную литературу для их освещения.

Критерии оценки опроса

«Отлично»:

- дан полный, развернутый ответ на поставленный вопрос;
- в ответе прослеживается четкая структура, логическая последовательность, отражающая сущность раскрываемых понятий, теорий, явлений;
- знание по предмету демонстрируется на фоне понимания его в системе данной науки и междисциплинарных связей;
- свободное владение терминологией;
- ответы на дополнительные вопросы четкие, краткие;

«Хорошо»:

- дан полный, развернутый ответ на поставленный вопрос, показано умение выделять существенные и несущественные признаки, причинно-следственные связи;
- ответ недостаточно логичен с единичными ошибками в частностях, исправленные студентом с помощью преподавателя;
- единичные ошибки в терминологии;
- ответы на дополнительные вопросы правильные, недостаточно полные и четкие.

«Удовлетворительно»:

- ответ не полный, с ошибками в деталях, умение раскрыть значение обобщенных знаний не показано, речевое оформление требует поправок, коррекции;
- логика и последовательность изложения имеют нарушения, студент не способен самостоятельно выделить существенные и несущественные признаки и причинно-следственные связи;
- ошибки в раскрываемых понятиях, терминах;
- студент не ориентируется в теме, допускает серьезные ошибки;
- студент не может ответить на большую часть дополнительных вопросов.

«Неудовлетворительно»:

- ответ представляет собой разрозненные знания с существенными ошибками по вопросу;
- присутствуют фрагментарность, нелогичность изложения, студент не осознает связь обсуждаемого вопроса с другими объектами дисциплины, речь неграмотная;
- незнание терминологии;
- ответы на дополнительные вопросы неправильные.

Методические материалы по выполнению практического задания

При выполнении практического задания обучающийся придерживается следующего алгоритма:

1. Записать дату, тему и цель задания;
2. Ознакомиться с правилами и условия выполнения практического задания;
3. Повторить теоретические задания, необходимые для рациональной работы и других практических действий, используя конспекты лекций и рекомендованную литературу, представленной в программе;
4. Выполнить работу по предложенному алгоритму действий;
5. Обобщить результаты работы, сформулировать выводы / дать ответы на контрольные вопросы;

Работа должна быть выполнена грамотно, с соблюдением культуры изложения. При использовании данных из учебных, методических пособий и другой литературы, периодических изданий, Интернет-источников должны иметься ссылки на вышеперечисленные.

Критерии оценки практического задания:

«Отлично» – правильный ответ, дается четкое обоснование принятому решению; рассуждения четкие последовательные логические; используются ссылки на полученные при изучении дисциплины знания; правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Хорошо» – правильный ответ, дается обоснование принятому решению; но с не существенными ошибками, в рассуждениях отсутствует логическая последовательность; используются ссылки на полученные при изучении дисциплины знания, правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Удовлетворительно» – правильный ответ, допускаются грубые ошибки в обосновании принятого решения; рассуждения не последовательные сумбурные; используются ссылки на полученные при изучении дисциплины знания; используются формулы, процедуры, понятия, имеющие прямое значение для подтверждения принятого решения, однако, при обращении к ним допускаются серьезные ошибки, студент не может правильно ими воспользоваться.

«Неудовлетворительно, не зачтено» – ответ неверный, отсутствует обоснование принятому решению; студент демонстрирует полное непонимание сути вопроса.

Для оценки решения ситуационной задачи (аналитического задания):

Оценка «отлично» выставляется, если задача решена грамотно, ответы на вопросы сформулированы четко. Эталонный ответ полностью соответствует решению студента, которое хорошо обосновано теоретически.

Оценка «хорошо» выставляется, если задача решена, ответы на вопросы сформулированы не достаточно четко. Решение студента в целом соответствует эталонному ответу, но не достаточно хорошо обосновано теоретически.

Оценка «удовлетворительно» выставляется, если задача решена не полностью, ответы не содержат всех необходимых обоснований решения.

Оценка «неудовлетворительно» выставляется, если задача не решена или имеет грубые теоретические ошибки в ответе на поставленные вопросы.

Методические материалы по выполнению лабораторного задания

При выполнении лабораторного задания обучающийся придерживается следующего алгоритма:

1. Записать дату, тему и цель задания;
2. Ознакомиться с правилами и условия выполнения задания;
3. Повторить теоретические задания, необходимые для рациональной работы и других практических действий, используя конспекты лекций и рекомендованную литературу, представленной в программе;
4. Выполнить работу по предложенному алгоритму действий;
5. Обобщить результаты работы, сформулировать выводы / дать ответы на контрольные вопросы;

Работа должна быть выполнена грамотно, с соблюдением культуры изложения. При использовании данных из учебных, методических пособий и другой литературы, периодических изданий, Интернет-источников должны иметься ссылки на вышеперечисленные.

Критерии оценки лабораторного задания:

«Отлично» – правильный ответ, дается четкое обоснование принятому решению; рассуждения четкие последовательные логические; используются ссылки на полученные при изучении дисциплины знания; правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Хорошо» – правильный ответ, дается обоснование принятому решению; но с не существенными ошибками, в рассуждениях отсутствует логическая последовательность; используются ссылки на полученные при изучении дисциплины знания, правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Удовлетворительно» – правильный ответ, допускаются грубые ошибки в обосновании принятого решения; рассуждения не последовательные сумбурные; используются ссылки на полученные при изучении дисциплины знания; используются формулы, процедуры, понятия, имеющие

прямое значение для подтверждения принятого решения, однако, при обращении к ним допускаются серьезные ошибки, студент не может правильно ими воспользоваться.

«Неудовлетворительно, не зачтено» – ответ неверный, отсутствует обоснование принятому решению; студент демонстрирует полное непонимание сути вопроса.

Методические указания для подготовки к промежуточной аттестации.

Изучение учебных дисциплин (модулей) завершается зачетом/зачетом с оценкой или экзаменом. Подготовка к промежуточной аттестации способствует закреплению, углублению и обобщению знаний, получаемых, в процессе обучения, а также применению их к решению практических задач. Готовясь к экзамену, студент ликвидирует имеющиеся пробелы в знаниях, углубляет, систематизирует и упорядочивает свои знания. На зачете или экзамене студент демонстрирует то, что он освоил в процессе обучения по дисциплине (модулю).

Вначале следует просмотреть весь материал по дисциплине (модулю), отметить для себя трудные вопросы. Обязательно в них разобраться. В заключение еще раз целесообразно повторить основные положения, используя при этом листы опорных сигналов. Систематическая подготовка к занятиям в течение семестра позволит использовать время промежуточной аттестации для систематизации знаний.

3. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНИВАНИЯ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ И (ИЛИ) ОПЫТА ДЕЯТЕЛЬНОСТИ, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ

3.1. Организационные основы применения балльно-рейтинговой системы оценки успеваемости обучающихся по дисциплине (модулю)

Оценка качества освоения обучающимися дисциплины (модуля) реализуется в формате балльно-рейтинговой системы оценки успеваемости обучающихся (БРСО).

БРСО в ходе текущего контроля успеваемости и промежуточной аттестации осуществляется по 100-балльной шкале.

Академический рейтинг обучающегося по дисциплине (модулю) складывается из результатов:

- текущего контроля успеваемости (максимальный текущий рейтинг обучающегося 80 рейтинговых баллов);
- промежуточной аттестации (максимальный рубежный рейтинг обучающегося 20 рейтинговых баллов).

Условия оценки освоения обучающимся дисциплины (модуля) в формате БРСО доводятся преподавателем до сведения обучающихся на первом учебном занятии, а также размещены в свободном доступе в электронной информационно-образовательной среде Университета.

3.2. Проведение текущего контроля успеваемости обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося

В течение учебного семестра до промежуточной аттестации на основании утвержденной рабочей программы дисциплины (модуля) формируется текущий рейтинг обучающегося. Текущий рейтинг обучающегося складывается как сумма рейтинговых баллов, полученных им в течение учебного семестра по всем видам учебных занятий по учебной дисциплине.

В процессе текущего контроля оцениваются следующие действия обучающегося, направленные на освоение компетенций в рамках изучения учебной дисциплины:

- академическая активность (посещаемость учебных занятий, самостоятельное изучение содержания учебной дисциплины в электронной информационно-образовательной среде, соблюдение сроков сдачи практических заданий и текущих контрольных мероприятий и др.);

- выполнение и сдача текущих и итогового практических заданий (эссе, рефераты, творческие задания, кейс-задания, лабораторные работы, расчетные задания и др., активное участие в групповых интерактивных занятиях (дискуссии, WiKi-проекты и др.), защита проектов и др.);

- прохождение рубежей текущего контроля, включая соблюдение графика их прохождения в электронной информационно-образовательной среде.

Для планирования расчета текущего рейтинга обучающегося используются следующие пропорции:

Вид учебного действия	Максимальная рейтинговая оценка, баллов
академическая активность	10
практические задания	40
<i>из них: текущие практические задания</i>	20
<i>итоговое практическое задание</i>	20
рубежи текущего контроля	30
ИТОГО:	80

В течение учебного семестра по дисциплине (модулю) обучающимся должен быть накоплен текущий рейтинг не менее 52 рейтинговых баллов (65% от максимального значения текущего рейтинга).

Необходимыми условиями допуска обучающегося к промежуточной аттестации по дисциплине являются положительное прохождение обучающимся не менее 65% рубежей текущего контроля с накоплением не менее 65% максимального рейтингового балла за каждый рубеж текущего контроля и положительное выполнение итогового практического задания с накоплением не менее 65% максимального рейтингового балла, установленного за итоговое практическое задание.

Невыполнение вышеуказанных условий является текущей академической задолженностью, которая должна быть ликвидирована обучающимся до контрольного мероприятия промежуточной аттестации.

Сведения о наличии у обучающихся текущей академической задолженности, сроках и порядке добора рейтинговых баллов для её ликвидации доводятся до обучающихся педагогическим работником.

В случае неликвидации текущей академической задолженности, педагогический работник обязан во время контрольного мероприятия промежуточной аттестации поставить обучающемуся 0 рейтинговых баллов. В этом случае ликвидация текущей академической задолженности возможна в периоды проведения повторной промежуточной аттестации.

3.3. Проведение промежуточной аттестации обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося

Промежуточная аттестация по дисциплине (модулю) проводится в соответствии с Положением о промежуточной аттестации обучающихся по основным профессиональным образовательным программам высшего образования – программ специалитета в Российском государственном социальном университете и Положением о балльно-рейтинговой системе оценки успеваемости обучающихся по основным профессиональным образовательным программам высшего образования - программам бакалавриата, программам специалитета, программам магистратуры в Российском государственном социальном университете.

На промежуточную аттестацию отводится 20 рейтинговых баллов.

Ответы обучающегося на контрольном мероприятии промежуточной аттестации оцениваются педагогическим работником по 20 - балльной шкале, а итоговая оценка по дисциплине (модулю) выставляется по системе зачтено/не зачтено для зачета.

Критерии выставления оценки определяются Положением о балльно-рейтинговой системе оценки успеваемости обучающихся по основным профессиональным образовательным программам высшего образования - программам бакалавриата, программам специалитета, программам магистратуры в Российском государственном социальном университете.

В процессе определения рубежного рейтинга обучающегося используется следующая шкала:

Рубежный рейтинг	Критерии оценки освоения обучающимся учебной дисциплины в ходе контрольных мероприятий промежуточной аттестации
19-20 рейтинговых баллов	обучающийся глубоко и прочно усвоил программный материал, исчерпывающе, последовательно, грамотно и логически стройно его излагает, тесно увязывает с задачами и будущей деятельностью, не затрудняется с ответом при видоизменении задания, свободно справляется с задачами и практическими заданиями, правильно обосновывает принятые решения, умеет самостоятельно обобщать и излагать материал, не допуская
16-18 рейтинговых баллов	обучающийся твердо знает программный материал, грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос, может правильно применять теоретические положения и владеет необходимыми умениями и навыками при выполнении практических
13-15 рейтинговых баллов	обучающийся освоил основной материал, но не знает отдельных деталей, допускает неточности, недостаточно правильные формулировки, нарушает последовательность в изложении программного материала и испытывает затруднения в выполнении практических заданий
1-12 рейтинговых баллов	обучающийся не знает значительной части программного материала, допускает существенные ошибки, с большими затруднениями выполняет практические задания
0 рейтинговых баллов	не аттестован

Если результат контроля успеваемости в рамках проведения контрольных мероприятий промежуточной аттестации (рубежный рейтинг обучающегося) неудовлетворительный (получено менее 13 рейтинговых баллов), то промежуточная аттестация по учебной дисциплине (модулю) невозможна даже при наличии высокого текущего рейтинга, полученного по итогам текущего контроля по учебной дисциплине (модулю).

ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

№ п/п	Содержание изменения	Реквизиты документа об утверждении изменения	Дата введения изменения
1.			
2.			
3.			
4.			



**Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Российский государственный социальный университет»**

УТВЕРЖДАЮ

декан факультета политических и социальных

технологий  /Пивнева С.В./

«28» февраля 2024 года

**МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ
УПРАВЛЕНИЕ ПРОЕКТАМИ ИНФОРМАЦИОННЫХ СИСТЕМ**

**Направление подготовки
«Информационные системы и технологии»**

**Направленность
«Интеллектуальные информационные системы и технологии»**

**ОБРАЗОВАТЕЛЬНАЯ ПРОГРАММА ВЫСШЕГО ОБРАЗОВАНИЯ - ПРОГРАММА
БАКАЛАВРИАТА**

Форма обучения

Очная, заочная

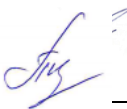
Москва, 2024 г.

Методические материалы по дисциплине (модулю) «Управление проектами информационных систем» разработаны на основании федерального государственного образовательного стандарта высшего образования – *бакалавриата* по направлению подготовки 09.03.02 Информационные системы и технологии, утвержденного приказом Министерства образования и науки Российской Федерации от 19.09.2017 № 926, учебного плана по основной профессиональной образовательной программе высшего образования – программе бакалавриата по направлению подготовки 09.03.02 Информационные системы и технологии (далее – «ОПОП»).

Методические материалы по дисциплине (модулю) разработаны рабочей группой в составе: канд. техн. наук А.О. Блинов

Методические материалы по дисциплине (модулю) обсуждены и утверждены на заседании кафедры информационных технологий, искусственного интеллекта и общественно-социальных технологий цифрового общества факультета социальных и политических технологий. Протокол № 11 от «28» февраля 2024 года.

Заведующий кафедрой
канд. пед. наук, доцент



(подпись)

С.В. Пивнева

СОДЕРЖАНИЕ

1. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ К ЛЕКЦИОННЫМ И ПРАКТИЧЕСКИМ ЗАНЯТИЯМ	4
1.1. Методические материалы к проведению лекционных занятий по дисциплине (модулю)	4
1.2. Методические материалы по подготовке к практическим занятиям по дисциплине (модулю)	7
2. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ) И САМОСТОЯТЕЛЬНОЙ РАБОТЕ	12
3. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНИВАНИЯ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ И (ИЛИ) ОПЫТА ДЕЯТЕЛЬНОСТИ, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ	21
3.1. Организационные основы применения балльно-рейтинговой системы оценки успеваемости обучающихся по дисциплине (модулю)	21
3.2. Проведение текущего контроля успеваемости обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося	22
3.3. Проведение промежуточной аттестации обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося	23
ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ	25

1. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ К ЛЕКЦИОННЫМ И ПРАКТИЧЕСКИМ ЗАНЯТИЯМ

1.1. Методические материалы к проведению лекционных занятий по дисциплине (модулю)

Лекция - один из методов обучения, одна из основных системообразующих форм организации учебного процесса в вузе. Лекционное занятие представляет собой систематическое, последовательное, монологическое изложение педагогическим работником учебного материала, как правило, теоретического характера. Такое занятие представляет собой элемент технологии представления учебного материала путем логически стройного, систематически последовательного и ясного изложения.

Цель лекции – организация целенаправленной познавательной деятельности обучающихся по овладению программным материалом дисциплины (модуля). Чтение курса лекций позволяет дать связанное, последовательное изложение материала в соответствии с новейшими данными науки, сообщить слушателям основное содержание предмета в целостном, систематизированном виде. В ряде случаев лекция выполняет функцию основного источника информации: при отсутствии учебников и учебных пособий, чаще по новым курсам; в случае, когда новые научные данные по той или иной теме не нашли отражения в учебниках; отдельные разделы и темы очень сложны для самостоятельного изучения. В таких случаях только лектор может методически помочь обучающимся в освоении сложного материала.

Возможные формы проведения лекций:

- Вводная лекция – один из наиболее важных и трудных видов лекции при чтении систематических курсов. От успеха этой лекции во многом зависит успех усвоения всего курса. Она может содержать: определение дисциплины (модуля); краткую историческую справку о дисциплине (модуле); цели и задачи дисциплины (модуля), ее роль в общей системе обучения и связь со смежными дисциплинами (модулями); основные проблемы (понятия и определения) данной науки; основную и дополнительную учебную литературу; особенности самостоятельной работы обучающихся над дисциплиной (модулем) и формы участия в научно-исследовательской работе; отчетность по курсу.

- Информационная лекция ориентирована на изложение и объяснение обучающимся научной информации, подлежащей осмыслению и запоминанию. Это самый традиционный тип лекций в практике высшей школы.

- Заключительная лекция предназначена для обобщения полученных знаний и раскрытия перспектив дальнейшего развития данной науки.

- Обзорная лекция – это систематизация научных знаний на высоком уровне, допускающая большое число ассоциативных связей в процессе осмысления информации, излагаемой при раскрытии внутрипредметной и межпредметной связей, исключая детализацию и конкретизацию. Как правило, стержень излагаемых теоретических положений составляет научно-понятийная и концептуальная основа всего курса или крупных его разделов.

- Лекция-беседа - непосредственный контакт педагогического работника с аудиторией - диалог. По ходу лекции педагогический работник задает вопросы для выяснения мнений и уровня осведомленности обучающихся по рассматриваемой проблеме.

- Лекция-дискуссия - свободный обмен мнениями в ходе изложения лекционного материала. Педагогический работник активизирует участие в обсуждении отдельными вопросами, сопоставляет между собой различные мнения и тем самым развивает дискуссию, стремясь направить ее в нужное русло.

- Лекция с применением обратной связи включает в себе то, что в начале и конце каждого раздела лекции задаются вопросы. Первый - для того, чтобы узнать, насколько обучающиеся ориентируются в излагаемом материале, вопрос в конце раздела предназначен для выяснения степени усвоения только что изложенного материала. При неудовлетворительных результатах контрольного опроса педагогический работник

возвращается к уже прочитанному разделу, изменив при этом методику подачи материала.

- Проблемная лекция опирается на логику последовательно моделируемых проблемных ситуаций путем постановки проблемных вопросов или предъявления проблемных задач. Проблемный вопрос - это диалектическое противоречие, требующее для своего решения размышления, сравнения, поиска, приобретения и применения новых знаний. Проблемная задача содержит дополнительную вводную информацию и при необходимости некоторые ориентиры поиска ее решения.

- Программированная лекция - консультация – педагогический работник сам составляет и предлагает обучающимся вопросы. На подготовленные вопросы педагогический работник сначала просит ответить обучающихся, а затем проводит анализ и обсуждение неправильных ответов. В лекциях можно использовать наглядные материалы, а также подготовить презентацию. Что касается презентации, то в качестве визуальной поддержки ее можно органично интегрировать во все вышеупомянутые лекции. В то же время лекцию-презентацию возможно выделить и в качестве самостоятельной формы. Лекция-презентация должна отражать суть основных и (или) проблемных вопросов лекции, на которые особо следует обратить внимание обучающихся. В условиях применения активного метода проведения занятий презентация представляется весьма удачным способом донесения информации до слушателей. Единственное, на что следует обратить внимание при подготовке слайдов, - это их оформление и текст. Слайд не должен быть перегружен картинками и лишней информацией, которая будет отвлекать от основного аспекта того или иного вопроса лекции. Во время лекции можно задавать вопросы аудитории в отношении того или иного слайда, тем самым еще больше вовлекая обучающихся в проблематику.

Краткое содержание лекционных занятий

Наименование разделов и тем	Содержание учебного материала
РАЗДЕЛ 1. ИНФОРМАЦИОННЫЕ СИСТЕМЫ И ИХ РОЛЬ В ДЕЯТЕЛЬНОСТИ СОВРЕМЕННЫХ ПРЕДПРИЯТИЙ	
Тема 1.1. Основные функции управления проектом ИС	Современные и эффективные методы управления производственно-хозяйственной и финансовой деятельностью предприятия. Основные функции управления проектом информационных систем. Планирование. Организация. Координация. Мотивация. Контроль. Нормативно-законодательное регулирование управления ИТ-инфраструктурой. Современные технологии управления проектами ИС. Информационные процессы управления проектом ИС.
Тема 1.2. Информационные системы и их роль в деятельности современных предприятий	Организационная структура предприятия. Производственная структура предприятия. Организационная структура предприятия. Аппарат управления. Подразделения предприятий.
РАЗДЕЛ 2. КОМПЬЮТЕРНО-ОРИЕНТИРОВАННЫЕ ТЕХНОЛОГИИ УПРАВЛЕНИЯ	

Тема 2.1. Развитие методов управления проектами ИС	Основные компьютерно-ориентированные технологии управления современным предприятием, а также роль компьютерных систем в организации процессно-ориентированного управления. Совокупность компьютерных программ календарного планирования. Количественные методы. Методы планирования по Генри Гантту (Henry L. Gantt). Упорядочение работ. Метод планирования взаимосвязанных работ. Метод многовариантного планирования. Метод планирования и учета критического ресурса (время).
Тема 2.2. Планирование потребностей в системах планирование MRP/ERP II	Методология планирования. Единицы измерения учетных единиц. Планирование цепочек управления. Оптимизация планирования управления ИС.
РАЗДЕЛ 3. СОЗДАНИЕ ИНФОРМАЦИОННЫХ СИСТЕМ УПРАВЛЕНИЯ ПРЕДПРИЯТИЕМ	
Тема 3.1. Методы и организация создания информационных систем и информационных технологий в управлении организацией	Процесс создания и развития информационных систем, внедрение которых позволит осуществить качественное совершенствование управления предприятием. Задачи оперативного управления. Экономико-математические методы и модели технических, программных средств для управления проектами ИС. Функции оперативного управления (определение потребности, централизация управления, определение затрат на приобретение, техническую поддержку и использование технических средств и программного обеспечения ИС). Обеспечение системы хранения информации и документации в ИС.
Тема 3.2. Классификация ИС по типу пользовательского интерфейса	Командный интерфейс. WIMP-интерфейс расшифровывается как Windows (окно) Image (образ) Menu (меню) Pointer (указатель). На экране высвечивается окно, содержащее образы программ и меню действий. SILK-интерфейс – расшифровка Speech (речь) Image (образ) Language (язык) Knowledge (знание). При использовании такого интерфейса на экране по речевой команде происходит перемещение от одних поисковых запросов к другим (Рисунок 1). Автоматизированное рабочее место руководителя проектов. Формирование команды проектного управления. Управление проектами информационных систем. Модель жизненного цикла (ЖЦ) информационной системы управления (ИС). Каскадная модель ЖЦ ИС. Поэтапная модель с промежуточным контролем. Спиральная модель. Прототип информационной модели. Макетирование. Проектирование информационной системы. Формирование перечня документации для управления проектом жизненного цикла информационной системы управления.
РАЗДЕЛ 4. РАЗРАБОТКА ПРОЕКТА. УПРАВЛЕНИЕ ПРОЕКТОМ	

Тема 4.1. Управление проектом информационных систем «Материально-техническое обеспечение»	Процесс создания и развития проектной деятельности. Разработка проекта, провести планирование этапов разработки проекта в программном обеспечении. Провести управление проектом. Концептуальный уровень описания информационной системы во времени и пространстве. Априорными данными являются: общая схема структуры предприятия или организации. Составьте перечень выполняемых работ по одному проекту. Проведите проектирование информационной системы, включающей: планирование календарных работ; финансовый калькулятор; данные из систем планирования материально-техническим обеспечением ERP/MRP II; классификатор единиц изменения; затраты по трудоемкости работ. Используйте синхронизацию работ. Примените интеграцию данных из одной системы в другую. Разработайте перечень систем и описание данных (см. образец выполнения «Кейс 1»).
Тема 4.2. Управление проектом информационных систем «Оцифровка исторических документов»	Априорными данными являются: общая схема структуры предприятия или организации. Составьте перечень выполняемых работ по одному проекту по оцифровке исторических документов. Проведите проектирование информационной системы, включающей: планирование календарных работ; финансовый калькулятор; данные по типам документов исторического архива; классификатор единиц измерения работ по проекту; затраты трудоемкости всех видов работ по проекту. Используйте синхронизацию работ. Примените интеграцию данных из одной системы в другую. Разработайте перечень систем и описание данных по образцу (см. образец выполнения «Кейс 2»).

1.2. Методические материалы по подготовке к практическим занятиям по дисциплине (модулю)

Практические (семинарские) занятия - одна из форм учебного занятия, направленная на развитие самостоятельности обучающихся и приобретение умений и навыков. Данные учебные занятия углубляют, расширяют, детализируют полученные ранее знания. Практическое занятие предполагает выполнение обучающимися по заданию и под руководством преподавателей одной или нескольких практических работ.

Цель практических занятий и семинаров состоит в развитии познавательных способностей, самостоятельности мышления и творческой активности обучающихся; углублении, расширении, детализировании знаний, полученных на лекции в обобщенной форме, и содействии выработке навыков профессиональной деятельности. В отдельных случаях на практических занятиях и семинарах руководителем занятия сообщаются дополнительные знания.

Для достижения поставленных целей и решения требуемого перечня задач практические занятия и семинары проводятся традиционными технологиями или с использованием активных и интерактивных образовательных технологий.

Возможные формы проведения практических (семинарских) занятий:

- Деловая игра - это метод группового обучения совместной деятельности в процессе решения общих задач в условиях максимально возможного приближения к реальным проблемным ситуациям. Имитационные игры - на занятиях имитируется деятельность какой-либо организации, предприятия или его подразделения. Имитироваться могут события,

конкретная деятельность людей (деловое совещание, обсуждение плана) и обстановка, условия, в которых происходит событие или осуществляется деятельность (кабинет начальника цеха, зал заседаний). Исполнение ролей (ролевые игры) - в этих играх отрабатывается тактика поведения, действий, выполнение функций и обязанностей конкретного лица. Для проведения игр с исполнением роли разрабатывается модель-пьеса ситуации, между студентами распределяются роли с «обязательным содержанием», характеризующиеся различными интересами; в процессе их взаимодействия должно быть найдено компромиссное решение. «Деловой театр» (метод инсценировки) - в нем разыгрывается какая-либо ситуация, поведение человека в этой обстановке, обучающийся должен вжиться в образ определенного лица, понять его действия, оценить обстановку и найти правильную линию поведения. Основная задача метода инсценировки - научить ориентироваться в различных обстоятельствах, давать объективную оценку своему поведению, учитывать возможности других людей, влиять на их интересы, потребности и деятельность, не прибегая к формальным атрибутам власти, к приказу.

- Игровое проектирование - является практическим занятием или циклом занятий, суть которых состоит в разработке инженерного, конструкторского, технологического и других видов проектов в игровых условиях, максимально воссоздающих реальность. Этот метод отличается высокой степенью сочетания индивидуальной и совместной работы обучающихся.

- Познавательные-дидактические игры не относятся к деловым играм. Они предполагают лишь включение изучаемого материала в необычный игровой контекст и иногда содержат лишь элементы ролевых игр. Такие игры могут проводиться в виде копирования научных, культурных, социальных явлений (конкурс знатоков, «Поле чудес», КВН и т.д.) и в виде предметно-содержательных моделей, (например, игры-путешествия, когда надо разработать рациональный маршрут, пользуясь различными картами).

- Анализ конкретных ситуаций. Конкретная ситуация – это любое событие, которое содержит в себе противоречие или вступает в противоречие с окружающей средой. Ситуации могут нести в себе как позитивный, так и отрицательный опыт. Все ситуации делятся на простые, критические и экстремальные.

- Кейс-метод (от английского case – случай, ситуация) – усовершенствованный метод анализа конкретных ситуаций, метод активного проблемно-ситуационного анализа, основанный на обучении путем решения конкретных задач – ситуаций (решение кейсов). Непосредственная цель метода case-study - обучающиеся должны проанализировать ситуацию, разобраться в сути проблем, предложить возможные решения и выбрать лучшее из них. Кейсы делятся на практические (отражающие реальные жизненные ситуации), обучающие (искусственно созданные, содержащие значительные элементы условности при отражении в нем жизни) и исследовательские (ориентированные на проведение исследовательской деятельности посредством применения метода моделирования). Метод конкретных ситуаций (метод case-study) относится к неигровым имитационным активным методам обучения.

- Тренинг (англ. training от train — обучать, воспитывать) – метод активного обучения, направленный на развитие знаний, умений и навыков и социальных установок. Тренинг – форма интерактивного обучения, целью которого является развитие компетентности межличностного и профессионального поведения в общении. Достоинство тренинга заключается в том, что он обеспечивает активное вовлечение всех участников в процесс обучения. Можно выделить основные типы тренингов по критерию направленности воздействия и изменений – навыковый, психотерапевтический, социально-психологический, бизнес-тренинг.

- Метод Сократа (Майевтика) – метод вопросов, предполагающих критическое отношение к догматическим утверждениям, называется еще как метод «сократовской иронии». Это умение извлекать скрытое в человеке знание с помощью искусных наводящих вопросов, подразумевающего короткий, простой и заранее предсказуемый ответ.

- Интерактивная лекция – выступление ведущего обучающего перед большой

аудиторией с применением следующих активных форм обучения: дискуссия, беседа, демонстрация слайдов или учебных фильмов, мозговой штурм.

- Групповая, научная дискуссия, диспут. Дискуссия — это целенаправленное обсуждение конкретного вопроса, сопровождающееся обменом мнениями, идеями между двумя и более лицами. Задача дискуссии - обнаружить различия в понимании вопроса и в споре установить истину. Дискуссии могут быть свободными и управляемыми. К технике управляемой дискуссии относятся: четкое определение цели, прогнозирование реакции оппонентов, планирование своего поведения, ограничение времени на выступления и их заданная очередность. Разновидностью свободной дискуссии является форум, где каждому желающему дается неограниченное время на выступление, при условии, что его выступление вызывает интерес аудитории. Каждый конкретный форум имеет свою тематику — достаточно широкую, чтобы в её пределах можно было вести многоплановое обсуждение.

- Дебаты – это чётко структурированный и специально организованный публичный обмен мыслями между двумя сторонами по актуальным темам. Это разновидность публичной дискуссии участников дебатов, направляющая на переубеждение в своей правоте третьей стороны, а не друг друга. Поэтому вербальные и невербальные средства, которые используются участниками дебатов, имеют целью получения определённого результата — сформировать у слушателей положительное впечатление от собственной позиции.

- Метод работы в малых группах. Групповое обсуждение кого-либо вопроса направлено на достижение лучшего взаимопонимания и нахождения истины. Групповое обсуждение способствует лучшему усвоению изучаемого материала. Оптимальное количество участников - 5-7 человек. Перед обучающимися ставится проблема, выделяется определенное время, в течение которого они должны подготовить аргументированный обдуманный ответ. Педагогический работник может устанавливать правила проведения группового обсуждения – задавать определенные рамки обсуждения, ввести алгоритм выработки общего мнения, назначить лидера и др.

- Круглый стол - общество, собрание в рамках более крупного мероприятия (съезда, симпозиума, конференции). Мероприятие, как правило, на которое приглашаются эксперты и специалисты из разных сфер деятельности для обсуждения актуальных вопросов. Данная модель обсуждения, основываясь на соглашениях, в качестве итогов даёт результаты, которые, в свою очередь, являются новыми соглашениями.

- Коллоквиум - (лат. colloquium — разговор, беседа) - одна из форм учебных занятий в системе образования, имеющая целью выяснение и повышение знаний обучающихся. На коллоквиумах обсуждаются: отдельные части, разделы, темы, вопросы изучаемого курса (обычно не включаемые в тематику семинарских и других практических учебных занятий), рефераты, проекты и др. работы обучающихся. Это научные собрания, на которых заслушиваются и обсуждаются доклады. Коллоквиум – это и форма контроля, массового опроса, позволяющая преподавателю в сравнительно небольшой срок выяснить уровень знаний студентов по данной теме дисциплины. Коллоквиум проходит обычно в форме дискуссии, в ходе которой обучающимся предоставляется возможность высказать свою точку зрения на рассматриваемую проблему, учиться обосновывать и защищать ее. Аргументируя и отстаивая свое мнение, обучающийся в то же время демонстрирует, насколько глубоко и осознанно он усвоил изученный материал.

- Метод «мозговой штурм» (мозговой штурм, мозговая атака, англ. brainstorming) — оперативный метод решения проблемы на основе стимулирования творческой активности, при котором участникам обсуждения предлагают высказывать как можно большее количество вариантов решения, в том числе самых фантастичных. Затем из общего числа высказанных идей отбирают наиболее удачные, которые могут быть использованы на практике. Является методом экспертного оценивания.

- Метод проектов - это способ достижения дидактической цели через детальную разработку проблемы (технологию), которая должна завершиться вполне реальным, осязаемым практическим результатом, оформленным тем или иным образом; это

совокупность приёмов, действий обучающихся в их определённой последовательности для достижения поставленной задачи – решения проблемы, лично значимой для учащихся и оформленной в виде некоего конечного продукта. Основное предназначение метода проектов состоит в предоставлении учащимся возможности самостоятельного приобретения знаний в процессе решения практических задач или проблем, требующего интеграции знаний из различных предметных областей.

- Брифинг - (англ. briefing от англ. brief – короткий, недолгий) – краткая пресс-конференция, посвященная одному вопросу. Основное отличие: отсутствует презентационная часть. То есть практически сразу идут ответы на вопросы журналистов.

- Метод портфолио (итал. portfolio — 'портфель, англ. - папка для документов) - современная образовательная технология, в основе которой используется метод аутентичного оценивания результатов образовательной и профессиональной деятельности. Портфолио как подборка сертифицированных достижений, наиболее значимых работ и отзывов на них.

Вопросы для самоподготовки к практическим (семинарским) занятиям по разделам (темам) дисциплины (модуля)

РАЗДЕЛ 1. Информационные системы и их роль в деятельности современных предприятий

Тема 1.1. Основные функции управления проектом ИС

Вопросы для самоподготовки:

1. Система управления проектом ИС.
2. Функции управления проектом ИС.
3. Информационная технология. Технологии управления.
4. Информационные процессы управления проектом ИС.
5. Организационная структура предприятия.
6. Производственная структура предприятия.
7. Организационно-штатная структура предприятия.
8. Аппарат управления.
9. Подразделения предприятий.
10. Каким требованиям должна отвечать организационная структура предприятия (гибкость, устойчивость, экономичность)?

Тема 1.2. Информационные системы и их роль в деятельности современных предприятий

Вопросы для самоподготовки:

1. Какое планирование является долгосрочным и составляется на срок от года до пяти лет?
2. Какой спрос может быть прогнозом?
3. Какая задача не относится к разряду важных, которая стоит перед промышленными предприятиями в современных условиях?
4. Дать определение «Набор проверенных на практике разумных принципов, моделей и процедур управления и контроля, служащих повышению показателей экономической деятельности предприятия».
5. Сколько этапов в своем развитии прошел стандарт MRP II?
6. Что является задачей информационных систем класса MRP II?
7. На какие два уровня стандарт MRP II делит сферы отдельных функций?
8. Что положено в основу стандарта MRP II?
9. Какая компьютерная программа работает по алгоритму, регламентированному MRP методологией?
10. Что является первым стандартом управления бизнесом?

РАЗДЕЛ 2. Компьютерно-ориентированные технологии управления

Тема 2.1. Развитие методов управления проектами ИС

Вопросы для самоподготовки:

1. Автоматизированное рабочее место.
2. Мировой и российский опыт управления проектами ИС.
3. Определение требований к разработке проекта.
4. Разработка технического задания на проектирование информационной системы.
5. Совокупность компьютерных программ календарного планирования.
6. Количественные методы.
7. Методы планирования по Генри Гантту (Henry L. Gantt).
8. Упорядочение работ.
9. Метод планирования взаимосвязанных работ.
10. Метод многовариантного планирования.

Тема 2.2. Планирование потребностей в системах планирование MRP/ERP II

Вопросы для самоподготовки:

1. Методология планирования управления проектами ИС.
2. Единицы измерения учетных единиц.
3. Планирование цепочек управления.
4. Оптимизация планирования управления ИС.
5. Какая компьютерная программа работает по алгоритму, регламентированному MRP методологией?
6. Что является первым стандартом управления бизнесом?
7. Что относится к основным входным данным в MRP системе?
8. Какие из процессов включает в себя функция планирования потребности в MRP II системе?
9. Как необходимо управлять изменениями, чтобы они были востребованными?
10. Чем отличается автоматизация структурированных и неструктурированных бизнес-процессов?

РАЗДЕЛ 3. Создание информационных систем управления предприятием

Тема 3.1. Методы и организация создания информационных систем и информационных технологий в управлении организацией

Вопросы для самоподготовки:

1. Автоматизированное рабочее место руководителя управления проектами ИС.
2. Формирование и состав команды проекта.
3. Задачи оперативного управления.
4. Экономико-математические методы управления проектами ИС.
5. Модели технических, программных средств для управления проектами и программами информационных систем.
6. Функции оперативного управления.
7. Определение потребностей.
8. Централизованная функция управления.
9. Определение затрат на приобретение, техническую поддержку и использование технических средств и программного обеспечения ИС.
10. Обеспечение системы хранения информации и документации в ИС.

Тема 3.2. Классификация ИС по типу пользовательского интерфейса

Вопросы для самоподготовки:

1. Командный интерфейс. WIMP-интерфейс. SILK-интерфейс.
2. Модель жизненного цикла (ЖЦ) информационной системы управления (ИС).
3. Каскадная модель ЖЦ ИС.
4. Поэтапная модель жизненного цикла информационной системы управления с промежуточным контролем.
5. Спиральная модель жизненного цикла информационной системы управления.
6. Прототип информационной модели.
7. Макетирование информационной модели.
8. Проектирование информационной модели. Разработка проекта.
9. Формирование перечня документации для управления ИС.
10. Управление проектами в информационных системах управления.

РАЗДЕЛ 4. Разработка проекта. Управление проектом

Тема 4.1. Управление проектом информационных систем «Материально-техническое обеспечение»

Вопросы для самоподготовки:

1. Концептуальный уровень описания системы во времени и пространстве.
2. Архитектура проекта.
3. Жизненный цикл проекта.
4. Стандарты проектирования информационных систем.
5. Проектирование информационных систем.
6. Преимущества автоматизированных систем и программного обеспечения управления проектной деятельностью.
7. Понятие всех составляющих конфигурации системного управления.
8. ИТ-инфраструктура компании.
9. Сквозное управление службами ИТ-инфраструктуры.
10. Эффективное управление конфигурациями, серверами, настольными ПК и мобильными устройствам.

Тема 4.2. Управление проектом информационных систем «Оцифровка исторических документов»

Вопросы для самоподготовки:

1. Защита данных и приложений ИС.
2. Управление хранением, резервным копированием и восстановлением данных.
3. Основные функции современных систем управления ИТ-инфраструктурой.
4. Автоматическая установка программного обеспечения ИС.
5. Снижения расходов на ИТ-обслуживание.
6. Повышение доступности инфраструктуры путем высокой скорости отклика при возникновении проблем.
7. Современные системы автоматизированного управления.
8. Выполнение однотипных повторяющихся операций – способы решения поставленных задач.
9. Управление лицензиями программного обеспечения.
10. Обработка событий.

2. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ ОБУЧАЮЩИХСЯ ПО

ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ) И САМОСТОЯТЕЛЬНОЙ РАБОТЕ

Освоение обучающимся дисциплины (модуля) предполагает изучение материалов дисциплины на аудиторных занятиях и в ходе самостоятельной работы. Аудиторные занятия проходят в форме лекций и практических занятий. Самостоятельная работа включает разнообразный комплекс видов и форм работы обучающихся.

Для успешного освоения дисциплины (модуля) и достижения поставленных целей необходимо внимательно ознакомиться с рабочей программы дисциплины (модуля), доступной в электронной информационно-образовательной среде РГСУ.

Следует обратить внимание на списки основной и дополнительной литературы, на предлагаемые преподавателем ресурсы информационно-телекоммуникационной сети Интернет. Эта информация необходима для самостоятельной работы обучающегося.

При подготовке к аудиторным занятиям необходимо помнить особенности каждой формы его проведения.

Подготовка к учебному занятию лекционного типа заключается в следующем.

С целью обеспечения успешного обучения обучающийся должен готовиться к лекции, поскольку она является важнейшей формой организации учебного процесса, поскольку:

- знакомит с новым учебным материалом;
- разъясняет учебные элементы, трудные для понимания;
- систематизирует учебный материал;
- ориентирует в учебном процессе.

С этой целью:

- внимательно прочитайте материал предыдущей лекции;
- ознакомьтесь с учебным материалом по учебнику и учебным пособиям с темой прочитанной лекции;
- внесите дополнения к полученным ранее знаниям по теме лекции на полях лекционной тетради;
- запишите возможные вопросы, которые вы зададите лектору на лекции по материалу изученной лекции;
- постарайтесь уяснить место изучаемой темы в своей подготовке;
- узнайте тему предстоящей лекции (по тематическому плану, по информации лектора) и запишите информацию, которой вы владеете по данному вопросу.

Подготовка к занятию семинарского типа.

При подготовке и работе во время проведения занятий семинарского типа следует обратить внимание на следующие моменты: на процесс предварительной подготовки, на работу во время занятия, обработку полученных результатов, исправление полученных замечаний.

Предварительная подготовка к учебному занятию семинарского типа заключается в изучении теоретического материала в отведенное для самостоятельной работы время, ознакомление с инструктивными материалами с целью осознания задач практического занятия, техники безопасности при работе с приборами, веществами.

Работа во время проведения учебного занятия семинарского типа включает:

- консультирование студентов преподавателями и вспомогательным персоналом с целью предоставления исчерпывающей информации, необходимой для самостоятельного выполнения предложенных преподавателем задач.
- самостоятельное выполнение заданий согласно обозначенной рабочей программой дисциплины (модуля) тематики.

Самостоятельная работа.

Самостоятельная работа - планируемая учебная, учебно-исследовательская, научно-исследовательская работа студентов, выполняемая во внеаудиторное (аудиторное) время по заданию и при методическом руководстве преподавателя, но без его непосредственного участия (при частичном непосредственном участии преподавателя, оставляющем ведущую роль за работой студентов).

Для более углубленного изучения темы задания для самостоятельной работы рекомендуется выполнять параллельно с изучением данной темы. При выполнении заданий по возможности используйте наглядное представление материала.

Самостоятельная работа студентов в ВУЗе является важным видом учебной и научной деятельности студента. Самостоятельная работа студентов играет значительную роль в рейтинговой технологии обучения.

К современному специалисту в области медицины общество предъявляет достаточно широкий перечень требований, среди которых немаловажное значение имеет наличие у выпускников определенных навыков (компетенций) и умения самостоятельно добывать знания из различных источников, систематизировать полученную информацию, давать оценку конкретной профессиональной ситуации. Формирование такого умения происходит в течение всего периода обучения.

В процессе самостоятельной работы студент приобретает необходимые для будущей специальности компетенции, навыки самоорганизации, самоконтроля, самоуправления, саморефлексии и становится активным самостоятельным субъектом учебной деятельности.

Самостоятельная работа студентов должна оказывать важное влияние на формирование личности будущего специалиста, его компетентность. Каждый студент самостоятельно определяет режим своей работы и меру труда, затрачиваемого на овладение учебным содержанием по каждой дисциплине. Он выполняет внеаудиторную работу по личному индивидуальному плану, в зависимости от его подготовки, времени и других условий.

Виды самостоятельной работы.

Работа с литературой.

При работе с книгой необходимо подобрать литературу, научиться правильно ее читать, вести записи. Для подбора литературы в библиотеке используются алфавитный и систематический каталоги. Важно помнить, что рациональные навыки работы с книгой - это всегда большая экономия времени и сил. Правильный подбор учебников рекомендуется преподавателем, читающим лекционный курс. Необходимая литература может быть также указана в методических разработках по данному курсу. Изучая материал по учебнику, следует переходить к следующему вопросу только после правильного уяснения предыдущего, описывая на бумаге все выкладки и вычисления (в том числе те, которые в учебнике опущены или на лекции даны для самостоятельного вывода). При изучении любой дисциплины большую и важную роль играет самостоятельная индивидуальная работа. Особое внимание следует обратить на определение основных понятий курса. Студент должен подробно разбирать примеры, которые поясняют такие определения, и уметь строить аналогичные примеры самостоятельно. Нужно добиваться точного представления о том, что изучаешь. Полезно составлять опорные конспекты. При изучении материала по учебнику полезно в тетради (на специально отведенных полях) дополнять конспект лекций. Там же следует отмечать вопросы, выделенные студентом для консультации с преподавателем. Выводы, полученные в результате изучения, рекомендуется в конспекте выделять, чтобы они при перечитывании записей лучше запоминались. Опыт показывает, что многим студентам помогает составление листа опорных сигналов, содержащего важнейшие и наиболее часто употребляемые формулы и понятия. Такой лист помогает запомнить формулы, основные положения лекции, а также может служить постоянным справочником для студента. Различают два вида чтения: первичное и вторичное. Первичное - это внимательное, неторопливое чтение, при котором можно остановиться на трудных местах.

Задача вторичного чтения - полное усвоение смысла целого (по счету это чтение может быть и не вторым, а третьим или четвертым). Самостоятельная работа с учебниками и книгами (а также самостоятельное теоретическое исследование проблем, обозначенных преподавателем на лекциях) – это важнейшее условие формирования у себя научного способа познания.

Чтение научного текста является частью познавательной деятельности. Ее цель – извлечение из текста необходимой информации. От того насколько осознанно читающим

собственная внутренняя установка при обращении к печатному слову (найти нужные сведения, усвоить информацию полностью или частично, критически проанализировать материал и т.п.) во многом зависит эффективность осуществляемого действия.

Методические рекомендации по составлению конспекта:

1. Внимательно прочитайте текст. Уточните в справочной литературе непонятные слова. При записи не забудьте вынести справочные данные на поля конспекта;
2. Выделите главное, составьте план;
3. Кратко сформулируйте основные положения текста, отметьте аргументацию автора;
4. Законспектируйте материал, четко следуя пунктам плана. При конспектировании старайтесь выразить мысль своими словами. Записи следует вести четко, ясно.

5. Грамотно записывайте цитаты. Цитируя, учитывайте лаконичность, значимость мысли. В тексте конспекта желательно приводить не только тезисные положения, но и их доказательства. При оформлении конспекта необходимо стремиться к емкости каждого предложения. Мысли автора книги следует излагать кратко, заботясь о стиле и выразительности написанного. Число дополнительных элементов конспекта должно быть логически обоснованным, записи должны распределяться в определенной последовательности, отвечающей логической структуре произведения. Для уточнения и дополнения необходимо оставлять поля. Овладение навыками конспектирования требует от студента целеустремленности, повседневной самостоятельной работы.

Методические материалы по самостоятельному решению задач

При самостоятельном решении задач нужно обосновывать каждый этап решения, исходя из теоретических положений курса. Если студент видит несколько путей решения проблемы (задачи), то нужно сравнить их и выбрать самый рациональный. Полезно до начала вычислений составить краткий план решения проблемы (задачи). Решение проблемных задач или примеров следует излагать подробно, вычисления располагать в строгом порядке, отделяя вспомогательные вычисления от основных. Решения при необходимости нужно сопровождать комментариями, схемами, чертежами и рисунками. Следует помнить, что решение каждой учебной задачи должно доводиться до окончательного логического ответа, которого требует условие, и по возможности с выводом. Полученный ответ следует проверить способами, вытекающими из существа данной задачи. Полезно также (если возможно) решать несколькими способами и сравнить полученные результаты. Решение задач данного типа нужно продолжать до приобретения твердых навыков в их решении.

Методические материалы к выполнению реферата

Реферат (от лат. *refere* – сообщать) – краткое изложение в письменном виде или в форме публикации доклада, содержания научного труда (трудов), литературы по теме. Работа над рефератом условно разделяется на выбор темы, подбор литературы, подготовку и защиту плана; написание теоретической части и всего текста с указанием библиографических данных используемых источников, подготовку доклада, выступление с ним. Тематика рефератов полностью связана с основными вопросами изучаемого курса.

Список литературы к темам не дается, и обучающиеся самостоятельно ведут библиографический поиск, причем им не рекомендуется ограничиваться университетской библиотекой.

Важно учитывать, что написание реферата требует от обучающихся определенных усилий. Особое внимание следует уделить подбору литературы, методике ее изучения с целью отбора и обработки собранного материала, обоснованию актуальности темы и теоретического уровня обоснованности используемых в качестве примеров фактов какой-либо деятельности.

Выбрав тему реферата, начав работу над литературой, необходимо составить план. Изучая литературу, продолжается обдумывание темы, осмысливание прочитанного, делаются выписки, сопоставляются точки зрения разных авторов и т.д. Реферативная работа сводится к тому, чтобы в ней выделились две взаимосвязанные стороны: во-первых, ее следует рассматривать как учебное задание, которое должен выполнить обучаемый, а во-вторых, как форму научной работы, творческого воображения при выполнении учебного задания.

Наличие плана реферата позволяет контролировать ход работы, избежать формального переписывания текстов из первоисточников.

Оформление реферата включает титульный лист, оглавление и краткий список использованной литературы. Список использованной литературы размещается на последней странице рукописи или печатной форме реферата. Реферат выполняется в письменной или печатной форме на белых листах формата А4 (210 x 297 мм). Шрифт Times New Roman, кегель 14, через 1,5 интервала при соблюдении следующих размеров текста: верхнее поле – 25 мм, нижнее – 20 мм, левое – 30 мм, правое – 15 мм. Нумерация страниц производится сверху листа, по центру. Титульный лист нумерации не подлежит.

Рефераты должны быть написаны простым, ясным языком, без претензий на наукообразность. Следует избегать сложных грамматических оборотов, непривычных терминов и символов. Если же такие термины и символы все-таки приводятся, то необходимо разъяснять их значение при первом упоминании в тексте реферата.

Объем реферата предполагает тщательный отбор информации, необходимой для краткого изложения вопроса. Важнейший этап – редактирование готового текста реферата и подготовка к обсуждению. Обсуждение требует хорошей ориентации в материале темы, умения выделить главное, поставить дискуссионный вопрос, привлечь внимание слушателей к интересной литературе, логично и убедительно изложить свои мысли.

Рефераты обязательно подлежат защите. Процедура защиты начинается с определения оппонентов защищающего свою работу. Они стремятся дать основательный анализ работы обучающимся, обращают внимание на положительные моменты и недостатки реферата, дают общую оценку содержанию, форме преподнесения материала, характеру использованной литературы. Иногда они дополняют тот или иной раздел реферата. Последнее особенно ценно, ибо говорит о глубоком знании обучающимся-оппонентом изучаемой проблемы.

Обсуждение не ограничивается заслушиванием оппонентов. Другие обучающиеся имеют право уточнить или опровергнуть какое-либо утверждение. Преподаватель предлагает любому обучающемуся задать вопрос по существу доклада или попытаться подвести итог обсуждению.

Алгоритм работы над рефератом

1. Выбор темы

Тема должна быть сформулирована грамотно (с литературной точки зрения);

В названии реферата следует поставить четкие рамки рассмотрения темы;

Желательно избегать слишком длинных названий;

Следует по возможности воздерживаться от использования в названии спорных с научной точки зрения терминов, излишней наукообразности, а также чрезмерного упрощения формулировок.

2. Реферат следует составлять из пяти основных частей: введения; основной части; заключения; списка литературы; приложений.

3. Основные требования к введению:

Во введении не следует концентрироваться на содержании; введение должно включать краткое обоснование актуальности темы реферата, где требуется показать, почему данный вопрос может представлять научный интерес и есть ли связь представляемого материала с современностью. Таким образом, тема реферата должна быть актуальна либо с научной точки зрения, либо с современных позиций.

Очень важно выделить цель, а также задачи, которые требуется решить для выполнения цели.

Введение должно содержать краткий обзор изученной литературы, в котором указывается взятый из того или иного источника материал, кратко анализируются изученные источники, показываются их сильные и слабые стороны;

Объем введения составляет две страницы текста.

4. Требования к основной части реферата:

Основная часть содержит материал, отобранный для рассмотрения проблемы;

Также основная часть должна включать в себя собственно мнение обучающихся и сформулированные самостоятельные выводы, опирающиеся на приведенные факты;

Материал, представленный в основной части, должен быть логически изложен и распределен по параграфам, имеющим свои названия;

В изложении основной части необходимо использовать сноски (в первую очередь, когда приводятся цифры и чьи-то цитаты);

Основная часть должна содержать иллюстративный материал (графики, таблицы и т. д.);

Объем основной части составляет около 10 страниц.

5. Требования к заключению:

В заключении формулируются выводы по параграфам, обращается внимание на выдвинутые во введении задачи и цели;

Заключение должно быть четким, кратким, вытекающим из содержания основной части.

6. Требования к оформлению списка литературы (по ГОСТу):

Необходимо соблюдать правильность последовательности записи источников: сначала следует писать фамилию, а после инициалы; название работы не ставится в кавычки; после названия сокращенно пишется место издания; затем идет год издания; наконец, называется процитированная страница.

Критерии оценки реферата

Обучающийся, защищающий реферат, должен рассказать о его актуальности, поставленных целях и задачах, изученной литературе, структуре основной части, сделанных в ходе работы выводах.

По окончании выступления ему может быть задано несколько вопросов по представленной проблеме.

Оценка складывается из соблюдения требований к реферату, грамотного раскрытия темы, умения четко рассказывать о представленном реферате, способности понять суть задаваемых по работе вопросов и найти точные ответы на них.

Методические материалы к выполнению эссе

Эссе – литературное произведение небольшого объема, обычно прозаическое, свободной композиции, передающее индивидуальные впечатления, суждения, соображения автора о той или иной проблеме, теме, о том или ином событии или явлении. Это вид самостоятельной исследовательской работы обучающихся, с целью углубления и закрепления теоретических знаний и освоения практических навыков. Цель эссе состоит в развитии самостоятельного творческого мышления и письменного изложения собственных мыслей. При написании эссе обучающийся должен представить развернутый письменный ответ на теоретический или практический актуальный вопрос, объявленный преподавателем в аудитории непосредственно перед ее написанием. В процессе написания эссе разрешается пользоваться нормативно-правовыми актами, конспектом лекций (в печатном виде). Использование интернет-ресурсов не допускается. Темы эссе преподаватель предлагает из числа тех, которые обучающиеся уже рассматривали на лекциях или семинарских занятиях, исходя из содержания заданий в составе оценочных средств. По решению преподавателя, в качестве темы эссе может быть выбрана одна или несколько тем, которые могут быть распределены между обучающимися по желанию.

Требования к выполнению эссе:

1. Проводится письменно.

2. Эссе выполняется на компьютере (гарнитура Times New Roman, шрифт 14) через 1,5 интервала с полями: верхнее, нижнее – 2; правое – 3; левое – 1,5. Отступ первой строки абзаца – 1,25. Сноски – постраничные. Таблицы и рисунки встраиваются в текст работы. При этом обязательный заголовок таблицы надо размещать над табличным полем, а рисунки сопровождать подрисуночными подписями. При включении в эссе нескольких таблиц и/или рисунков их нумерация обязательна. Обязательна и нумерация страниц. Их целесообразно

проставлять внизу страницы – по середине или в правом углу. Номер страницы не ставится на титульном листе, но в общее число страниц он включается. Объем эссе, без учета приложений, не должен превышать 5 страниц. Значительное превышение установленного объема является недостатком работы и указывает на то, что обучающийся не сумел отобрать и переработать необходимый материал.

3. Работа должна содержать собственные умозаключения по сути поставленной проблемы, включать самостоятельно проведенный анализ по сути этой проблемы, выводы, обобщающие авторскую позицию по поставленной проблеме.

Критерии оценки эссе:

«Отлично» – исключительные знания материала, абсолютное понимание сути, безукоризненное знание основных понятий и положений, логически и лексически грамотно изложенный, содержательный, аргументированный, конкретный и исчерпывающий ответ.

«Хорошо» – глубокие знания материала, правильное понимание сути, знание основных понятий и положений, содержательный, полный и конкретный ответ.

«Удовлетворительно» – твердые, но недостаточно полные знания, верное понимание сути, в целом правильный ответ.

«Неудовлетворительно» – непонимание сущности задания, грубые ошибки в ответе.

Методические материалы по выполнению тестирования.

Тестовые задания содержат вопросы и 3-4 варианта ответа по базовым положениям изучаемой темы/раздела, составлены с расчетом на знания, полученные обучающимся в процессе изучения темы/раздела.

Тестовые задания выполняются в письменной форме и сдаются преподавателю, ведущему дисциплину (модуль). На выполнение тестовых заданий обучающимся отводится 45 минут.

При обработке результатов оценочной процедуры используются: критерии оценки по содержанию и качеству полученных ответов, ключи, оценочные листы.

Критерии оценки теста:

«Зачтено» - если обучающийся правильно выполнил не менее 2/3 всей работы или допустил не более одной грубой ошибки и двух недочетов, не более одной грубой и одной негрубой ошибки, не более трех негрубых ошибок, одной негрубой ошибки и трех недочетов, при наличии четырех-пяти недочетов.

«Не зачтено» - если число ошибок и недочетов превысило норму для оценки 3 или правильно выполнено менее 2/3 всей работы.

Методические материалы по выполнению доклада.

Рекомендуется следующая структура доклада:

1. титульный лист, содержание доклада;
2. краткое изложение;
3. цели и задачи;
4. изложение характера исследований и рассмотренных проблем, гипотезы, спорные вопросы;
5. источники информации, методы сбора и анализа данных, степень их полноты и достоверности;
6. анализ и толкование полученных в работе результатов;
7. выводы и оценки;
8. библиография и приложения.

Время выступления докладчика не должно превышать 10 минут.

Основные требования к оформлению доклада:

- титульный лист должен включать название доклада, наименование предметной (цикловой) комиссии, фамилию обучающегося;
- все использованные литературные источники сопровождаются библиографическим описанием;
- приводимая цитата из источника берется в кавычки (оформляются сноски);

- единицы измерения должны применяться в соответствии с действующими стандартами;
- все названия литературных источников следует приводить в соответствии с новейшими изданиями;
- рекомендуется включение таблиц, графиков, схем, если они отражают основное содержание или улучшают ее наглядность;
- названия фирм, учреждений, организаций и предприятий должны именоваться так, как они указываются в источнике;

Критерии оценки доклада

При выполнении доклада обучающийся должен продемонстрировать умение кратко излагать прочитанный материал, а также умение обобщать и анализировать материал по теме доклада.

Презентация

Методические материалы к презентациям

1. Объем презентации 10 -20 слайдов.
2. На титульном слайде должно быть отражено:
 - наименование факультета;
 - тема презентации;
 - фамилия, имя, отчество, направление подготовки/ специальность, направленность (профиль)/ специализация, форма обучения, номер группы автора презентации;
 - фамилия, имя, отчество, степень, звание, должность руководитель работы;
 - год выполнения работы.
3. В презентации должны быть отражены обоснование актуальности представляемого материала, цели и задачи работы.
4. Содержание презентации должно включать наиболее значимый материал доклада, а также, при необходимости, таблицы, диаграммы, рисунки, фотографии, карты, видео – вставки, звуковое сопровождение.
5. Заключительный слайд должен содержать информацию об источниках информации для презентации.

Критерии оценки презентации

1. Объем презентации 10 -20 слайдов.
2. Правильность оформления титульного слайда.
3. Актуальность отобранного материала, обоснованность формулировки цели и задач работы.
4. Наглядность и логичность презентации, обоснованность использования таблиц, диаграмм, рисунков, фотографий, карт, видео – вставок, звукового сопровождения; правильный выбор шрифтов, фона, других элементов дизайна слайда.
5. Объем и качество источников информации (не менее 2-х интернет – источников и не менее 2-х литературных источников).

Методические материалы по подготовке к опросу

Самостоятельная работа обучающихся включает подготовку к опросу на практических занятиях. Для этого обучающийся изучает лекции, основную и дополнительную литературу, публикации, информацию из Интернет-ресурсов.

Тема и вопросы к практическим занятиям, вопросы для самоконтроля содержатся в рабочей программе и доводятся до студентов заранее.

Для подготовки к опросу обучающемуся необходимо ознакомиться с материалом, посвященным теме практического занятия, в учебнике или другой рекомендованной литературе, конспекте лекции, обратить внимание на усвоение основных понятий дисциплины, выявить неясные вопросы и подобрать дополнительную литературу для их освещения.

Критерии оценки опроса

«Отлично»:

- дан полный, развернутый ответ на поставленный вопрос;
- в ответе прослеживается четкая структура, логическая последовательность, отражающая сущность раскрываемых понятий, теорий, явлений;
- знание по предмету демонстрируются на фоне понимания его в системе данной науки и междисциплинарных связей;

- свободное владение терминологией;

- ответы на дополнительные вопросы четкие, краткие;

«Хорошо»:

- дан полный, развернутый ответ на поставленный вопрос, показано умение выделять существенные и несущественные признаки, причинно-следственные связи;

- ответ недостаточно логичен с единичными ошибками в частностях, исправленные студентом с помощью преподавателя;

- единичные ошибки в терминологии;

- ответы на дополнительные вопросы правильные, недостаточно полные и четкие.

«Удовлетворительно»:

- ответ не полный, с ошибками в деталях, умение раскрыть значение обобщённых знаний не показано, речевое оформление требует поправок, коррекции;

- логика и последовательность изложения имеют нарушения, студент не способен самостоятельно выделить существенные и несущественные признаки и причинно-следственные связи;

- ошибки в раскрываемых понятиях, терминах;

- студент не ориентируется в теме, допускает серьезные ошибки;

- студент не может ответить на большую часть дополнительных вопросов.

«Неудовлетворительно»:

- ответ представляет собой разрозненные знания с существенными ошибками по вопросу;

- присутствуют фрагментарность, нелогичность изложения, студент не осознает связь обсуждаемого вопроса с другими объектами дисциплины, речь неграмотная;

- незнание терминологии;

- ответы на дополнительные вопросы неправильные.

Методические материалы по выполнению практического задания

При выполнении практического задания обучающийся придерживается следующего алгоритма:

1. Записать дату, тему и цель задания;

2. Ознакомиться с правилами и условия выполнения практического задания;

3. Повторить теоретические задания, необходимые для рациональной работы и других практических действий, используя конспекты лекций и рекомендованную литературу, представленной в программе;

4. Выполнить работу по предложенному алгоритму действий;

5. Обобщить результаты работы, сформулировать выводы / дать ответы на контрольные вопросы;

Работа должна быть выполнена грамотно, с соблюдением культуры изложения. При использовании данных из учебных, методических пособий и другой литературы, периодических изданий, Интернет-источников должны иметься ссылки на вышеперечисленные.

Критерии оценки практического задания:

«Отлично» – правильный ответ, дается четкое обоснование принятому решению; рассуждения четкие последовательные логические; используются ссылки на полученные при изучении дисциплины знания; правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Хорошо» – правильный ответ, дается обоснование принятому решению; но с не существенными ошибками, в рассуждениях отсутствует логическая последовательность;

используются ссылки на полученные при изучении дисциплины знания, правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Удовлетворительно» – правильный ответ, допускаются грубые ошибки в обосновании принятого решения; рассуждения не последовательные сумбурные; используются ссылки на полученные при изучении дисциплины знания; используются формулы, процедуры, понятия, имеющие прямое значение для подтверждения принятого решения, однако, при обращении к ним допускаются серьезные ошибки, студент не может правильно ими воспользоваться.

«Неудовлетворительно, не зачтено» – ответ неверный, отсутствует обоснование принятому решению; студент демонстрирует полное непонимание сути вопроса.

Для оценки решения ситуационной задачи (аналитического задания):

Оценка «отлично» выставляется, если задача решена грамотно, ответы на вопросы сформулированы четко. Эталонный ответ полностью соответствует решению студента, которое хорошо обосновано теоретически.

Оценка «хорошо» выставляется, если задача решена, ответы на вопросы сформулированы не достаточно четко. Решение студента в целом соответствует эталонному ответу, но не достаточно хорошо обосновано теоретически.

Оценка «удовлетворительно» выставляется, если задача решена не полностью, ответы не содержат всех необходимых обоснований решения.

Оценка «неудовлетворительно» выставляется, если задача не решена или имеет грубые теоретические ошибки в ответе на поставленные вопросы.

Методические указания для подготовки к промежуточной аттестации.

Изучение учебных дисциплин (модулей) завершается зачетом/зачетом с оценкой или экзаменом. Подготовка к промежуточной аттестации способствует закреплению, углублению и обобщению знаний, получаемых, в процессе обучения, а также применению их к решению практических задач. Готовясь к экзамену, студент ликвидирует имеющиеся пробелы в знаниях, углубляет, систематизирует и упорядочивает свои знания. На зачете или экзамене студент демонстрирует то, что он освоил в процессе обучения по дисциплине (модулю).

Вначале следует просмотреть весь материал по дисциплине (модулю), отметить для себя трудные вопросы. Обязательно в них разобраться. В заключение еще раз целесообразно повторить основные положения, используя при этом листы опорных сигналов. Систематическая подготовка к занятиям в течение семестра позволит использовать время промежуточной аттестации для систематизации знаний.

3. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНИВАНИЯ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ И (ИЛИ) ОПЫТА ДЕЯТЕЛЬНОСТИ, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ

3.1. Организационные основы применения балльно-рейтинговой системы оценки успеваемости обучающихся по дисциплине (модулю)

Оценка качества освоения обучающимися дисциплины (модуля) реализуется в формате балльно-рейтинговой системы оценки успеваемости обучающихся (БРСО).

БРСО в ходе текущего контроля успеваемости и промежуточной аттестации осуществляется по 100-балльной шкале.

Академический рейтинг обучающегося по дисциплине (модулю) складывается из результатов:

- текущего контроля успеваемости (максимальный текущий рейтинг обучающегося 80 рейтинговых баллов);

- промежуточной аттестации (максимальный рубежный рейтинг обучающегося 20 рейтинговых баллов).

Условия оценки освоения обучающимся дисциплины (модуля) в формате БРСО

доводятся преподавателем до сведения обучающихся на первом учебном занятии, а также размещены в свободном доступе в электронной информационно-образовательной среде Университета.

3.2. Проведение текущего контроля успеваемости обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося

В течение учебного семестра до промежуточной аттестации на основании утвержденной рабочей программы дисциплины (модуля) формируется текущий рейтинг обучающегося. Текущий рейтинг обучающегося складывается как сумма рейтинговых баллов, полученных им в течение учебного семестра по всем видам учебных занятий по учебной дисциплине.

В процессе текущего контроля оцениваются следующие действия обучающегося, направленные на освоение компетенций в рамках изучения учебной дисциплины:

- академическая активность (посещаемость учебных занятий, самостоятельное изучение содержания учебной дисциплины в электронной информационно-образовательной среде, соблюдение сроков сдачи практических заданий и текущих контрольных мероприятий и др.);

- выполнение и сдача текущих и итогового практических заданий (эссе, рефераты, творческие задания, кейс-задания, лабораторные работы, расчетные задания и др., активное участие в групповых интерактивных занятиях (дискуссии, WiKi-проекты и др.), защита проектов и др.);

- прохождение рубежей текущего контроля, включая соблюдение графика их прохождения в электронной информационно-образовательной среде.

Для планирования расчета текущего рейтинга обучающегося используются следующие пропорции:

Вид учебного действия	Максимальная рейтинговая оценка, баллов
академическая активность	10
практические задания	40
из них: текущие практические задания	20
итоговое практическое задание	20
рубежи текущего контроля	30
ИТОГО:	80

В течение учебного семестра по дисциплине (модулю) обучающимся должен быть накоплен текущий рейтинг не менее 52 рейтинговых баллов (65% от максимального значения текущего рейтинга).

Необходимыми условиями допуска обучающегося к промежуточной аттестации по дисциплине являются положительное прохождение обучающимся не менее 65% рубежей текущего контроля с накоплением не менее 65% максимального рейтингового балла за каждый рубеж текущего контроля и положительное выполнение итогового практического задания с накоплением не менее 65% максимального рейтингового балла, установленного за итоговое практическое задание.

Невыполнение вышеуказанных условий является текущей академической задолженностью, которая должна быть ликвидирована обучающимся до контрольного мероприятия промежуточной аттестации.

Сведения о наличии у обучающихся текущей академической задолженности, сроках и порядке добора рейтинговых баллов для её ликвидации доводятся до обучающихся

педагогическим работником.

В случае неликвидации текущей академической задолженности, педагогический работник обязан во время контрольного мероприятия промежуточной аттестации поставить обучающемуся 0 рейтинговых баллов. В этом случае ликвидация текущей академической задолженности возможна в периоды проведения повторной промежуточной аттестации.

3.3. Проведение промежуточной аттестации обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося

Промежуточная аттестация по дисциплине (модулю) проводится в соответствии с Положением о промежуточной аттестации обучающихся по основным профессиональным образовательным программам высшего образования – программ магистратуры в Российском государственном социальном университете и Положением о балльно-рейтинговой системе оценки успеваемости обучающихся по основным профессиональным образовательным программам высшего образования - программам магистратуры в Российском государственном социальном университете.

На промежуточную аттестацию отводится 20 рейтинговых баллов.

Ответы обучающегося на контрольном мероприятии промежуточной аттестации оцениваются педагогическим работником по 20 - балльной шкале, а итоговая оценка по дисциплине (модулю) выставляется по системе зачтено/не зачтено для зачета.

Критерии выставления оценки определяются Положением о балльно-рейтинговой системе оценки успеваемости обучающихся по основным профессиональным образовательным программам высшего образования – программам магистратуры в Российском государственном социальном университете.

В процессе определения рубежного рейтинга обучающегося используется следующая шкала:

Рубежный рейтинг	Критерии оценки освоения обучающимся учебной дисциплины в ходе контрольных мероприятий промежуточной аттестации
19-20 рейтинговых баллов	обучающийся глубоко и прочно усвоил программный материал, исчерпывающе, последовательно, грамотно и логически стройно его излагает, тесно увязывает с задачами и будущей деятельностью, не затрудняется с ответом при видоизменении задания, свободно справляется с задачами и практическими заданиями, правильно обосновывает принятые решения, умеет самостоятельно обобщать и излагать материал, не допуская ошибок
16-18 рейтинговых баллов	обучающийся твердо знает программный материал, грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос, может правильно применять теоретические положения и владеет необходимыми умениями и навыками при выполнении практических заданий
13-15 рейтинговых баллов	обучающийся освоил основной материал, но не знает отдельных деталей, допускает неточности, недостаточно правильные формулировки, нарушает последовательность в изложении программного материала и испытывает затруднения в выполнении практических заданий
1-12 рейтинговых баллов	обучающийся не знает значительной части программного материала, допускает существенные ошибки, с большими затруднениями выполняет практические задания
0 рейтинговых баллов	не аттестован

Если результат контроля успеваемости в рамках проведения контрольных мероприятий промежуточной аттестации (рубежный рейтинг обучающегося) неудовлетворительный (получено менее 13 рейтинговых баллов), то промежуточная аттестация по учебной дисциплине (модулю) невозможна даже при наличии высокого текущего рейтинга, полученного по итогам текущего контроля по учебной дисциплине (модулю).

ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

№ п/п	Содержание изменения	Реквизиты документа об утверждении изменения	Дата введения изменения
1.			
2.			
3.			
4.			



**Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Российский государственный социальный университет»**

УТВЕРЖДАЮ

декан факультета политических и социальных

технологий  /Пивнева С.В./

«28» февраля 2024 года

**МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ
ЯЗЫКИ И СИСТЕМЫ ПРОГРАММИРОВАНИЯ БАЗ ДАННЫХ**

**Направление подготовки
«Информационные системы и технологии»**

**Направленность
«Интеллектуальные информационные системы и технологии»**

**ОБРАЗОВАТЕЛЬНАЯ ПРОГРАММА ВЫСШЕГО ОБРАЗОВАНИЯ - ПРОГРАММА
БАКАЛАВРИАТА**

Форма обучения

Очная, заочная

Методические материалы по дисциплине (модулю) «Языки и системы программирования баз данных» разработаны на основании федерального государственного образовательного стандарта высшего образования – *бакалавриата* по направлению подготовки *09.03.02 Информационные системы и технологии*, утвержденного приказом Министерства образования и науки Российской Федерации от 19.09.2017 № 926, учебного плана по основной профессиональной образовательной программе высшего образования – программе бакалавриата по направлению подготовки *09.03.02 Информационные системы и технологии* (далее – «ОПОП»).

Методические материалы по дисциплине (модулю) разработаны рабочей группой в составе: Д.Ю. Елисеева.

Методические материалы по дисциплине (модулю) обсуждены и утверждены на заседании кафедры информационных технологий, искусственного интеллекта и общественно-социальных технологий цифрового общества факультета социальных и политических технологий.

Протокол № 11 от «28» февраля 2024 года

Заведующий кафедрой
канд. пед. наук, доцент

_____ 

_____ С.В. Пивнева

СОДЕРЖАНИЕ

1. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ К ЛЕКЦИОННЫМ, ПРАКТИЧЕСКИМ, ЛАБОРАТОРНЫМ ЗАНЯТИЯМ	4
1.1. Методические материалы к проведению лекционных занятий по дисциплине (модулю)	4
1.2. Методические материалы по подготовке к практическим занятиям по дисциплине (модулю)	6
1.3. Учебно-наглядные пособия по разделам (темам) дисциплины (модуля)	10
1.4. Методические материалы по подготовке к лабораторным занятиям по дисциплине (модулю)	13
2. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ) И САМОСТОЯТЕЛЬНОЙ РАБОТЕ.....	13
3. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНИВАНИЯ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ И (ИЛИ) ОПЫТА ДЕЯТЕЛЬНОСТИ, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ.....	23
3.1. Организационные основы применения балльно-рейтинговой системы оценки успеваемости обучающихся по дисциплине (модулю)	23
3.2. Проведение текущего контроля успеваемости обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося.....	23
3.3. Проведение промежуточной аттестации обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося.....	24
Приложение № 1 к методическим материалам по дисциплине (модулю). Конспекты лекционных занятий по дисциплине (модулю).....	26
КОНСПЕКТЫ ЛЕКЦИОННЫХ ЗАНЯТИЙ ПО ДИСЦИПЛИНЕ (МОДУЛЮ).....	26
Приложение № 2 к методическим материалам по дисциплине (модулю). Конспекты практических (семинарских) занятий по дисциплине (модулю)	142
КОНСПЕКТЫ ПРАКТИЧЕСКИХ (СЕМИНАРСКИХ) ЗАНЯТИЙ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)...	142
Приложение № 3 к методическим материалам по дисциплине (модулю). Конспекты лабораторных занятий по дисциплине (модулю).....	143
КОНСПЕКТЫ ЛАБОРАТОРНЫХ ЗАНЯТИЙ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)	143
Приложение № 4 к методическим материалам по дисциплине (модулю). Учебно-наглядные пособия по дисциплине (модулю).....	148
УЧЕБНО-НАГЛЯДНЫЕ ПОСОБИЯ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)	148
ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ	148

1. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ К ЛЕКЦИОННЫМ, ЛАБОРАТОРНЫМ ЗАНЯТИЯМ

1.1. Методические материалы к проведению лекционных занятий по дисциплине (модулю)

Лекция - один из методов обучения, одна из основных системообразующих форм организации учебного процесса в вузе. Лекционное занятие представляет собой систематическое, последовательное, монологическое изложение педагогическим работником учебного материала, как правило, теоретического характера. Такое занятие представляет собой элемент технологии представления учебного материала путем логически стройного, систематически последовательного и ясного изложения.

Цель лекции – организация целенаправленной познавательной деятельности обучающихся по овладению программным материалом дисциплины (модуля). Чтение курса лекций позволяет дать связанное, последовательное изложение материала в соответствии с новейшими данными науки, сообщить слушателям основное содержание предмета в целостном, систематизированном виде. В ряде случаев лекция выполняет функцию основного источника информации: при отсутствии учебников и учебных пособий, чаще по новым курсам; в случае, когда новые научные данные по той или иной теме не нашли отражения в учебниках; отдельные разделы и темы очень сложны для самостоятельного изучения. В таких случаях только лектор может методически помочь обучающимся в освоении сложного материала.

Возможные формы проведения лекций:

- Вводная лекция – один из наиболее важных и трудных видов лекции при чтении систематических курсов. От успеха этой лекции во многом зависит успех усвоения всего курса. Она может содержать: определение дисциплины (модуля); краткую историческую справку о дисциплине (модуле); цели и задачи дисциплины (модуля), ее роль в общей системе обучения и связь со смежными дисциплинами (модулями); основные проблемы (понятия и определения) данной науки; основную и дополнительную учебную литературу; особенности самостоятельной работы обучающихся над дисциплиной (модулем) и формы участия в научно-исследовательской работе; отчетность по курсу.

- Информационная лекция ориентирована на изложение и объяснение обучающимся научной информации, подлежащей осмыслению и запоминанию. Это самый традиционный тип лекций в практике высшей школы.

- Заключительная лекция предназначена для обобщения полученных знаний и раскрытия перспектив дальнейшего развития данной науки.

- Обзорная лекция – это систематизация научных знаний на высоком уровне, допускающая большое число ассоциативных связей в процессе осмысления информации, излагаемой при раскрытии внутрипредметной и междпредметной связей, исключая детализацию и конкретизацию. Как правило, стержень излагаемых теоретических положений составляет научно-понятийная и концептуальная основа всего курса или крупных его разделов.

- Лекция-беседа - непосредственный контакт педагогического работника с аудиторией - диалог. По ходу лекции педагогический работник задает вопросы для выяснения мнений и уровня осведомленности обучающихся по рассматриваемой проблеме.

- Лекция-дискуссия - свободный обмен мнениями в ходе изложения лекционного материала. Педагогический работник активизирует участие в обсуждении отдельными вопросами, сопоставляет между собой различные мнения и тем самым развивает дискуссию, стремясь направить ее в нужное русло.

- Лекция с применением обратной связи включает в себе то, что в начале и конце каждого раздела лекции задаются вопросы. Первый - для того, чтобы узнать, насколько обучающиеся ориентируются в излагаемом материале, вопрос в конце раздела предназначен для выяснения степени усвоения только что изложенного материала. При

неудовлетворительных результатах контрольного опроса педагогический работник возвращается к уже прочитанному разделу, изменив при этом методику подачи материала.

- Проблемная лекция опирается на логику последовательно моделируемых проблемных ситуаций путем постановки проблемных вопросов или предъявления проблемных задач. Проблемный вопрос - это диалектическое противоречие, требующее для своего решения размышления, сравнения, поиска, приобретения и применения новых знаний. Проблемная задача содержит дополнительную вводную информацию и при необходимости некоторые ориентиры поиска ее решения.

- Программированная лекция - консультация – педагогический работник сам составляет и предлагает обучающимся вопросы. На подготовленные вопросы педагогический работник сначала просит ответить обучающихся, а затем проводит анализ и обсуждение неправильных ответов. В лекциях можно использовать наглядные материалы, а также подготовить презентацию. Что касается презентации, то в качестве визуальной поддержки ее можно органично интегрировать во все вышеупомянутые лекции. В то же время лекцию-презентацию возможно выделить и в качестве самостоятельной формы. Лекция-презентация должна отражать суть основных и (или) проблемных вопросов лекции, на которые особо следует обратить внимание обучающихся. В условиях применения активного метода проведения занятий презентация представляется весьма удачным способом донесения информации до слушателей. Единственное, на что следует обратить внимание при подготовке слайдов, - это их оформление и текст. Слайд не должен быть перегружен картинками и лишней информацией, которая будет отвлекать от основного аспекта того или иного вопроса лекции. Во время лекции можно задавать вопросы аудитории в отношении того или иного слайда, тем самым еще больше вовлекая обучающихся в проблематику.

Краткое содержание лекционных занятий

Наименование разделов и тем	Содержание учебного материала
РАЗДЕЛ 1. Теория баз данных	
Тема 1.1. Введение в базы данных. Общая характеристика основных понятий	Основные понятия теории баз данных
Тема 1.2. Системы управления базами данных	Описание целей и задач СУБД.
РАЗДЕЛ 2. Проектирование приложения базы данных	
Тема 2.1. Различные архитектурные решения, используемые при реализации многопользовательских СУБД. Краткий обзор СУБД	Типы архитектур СУБД
Тема 2.2. Различные представления о данных в базах данных. Основные этапы проектирования баз данных	Проектирование баз данных. Первая стадия концептуального проектирования базы данных (концептуальное моделирование). Вторая стадия концептуального проектирования (Модели данных СУБД. Представление концептуальной модели средствами модели данных СУБД).
РАЗДЕЛ 3. Введение в реляционные базы данных.	
Тема 3.1. Формализация реляционной модели	Правила применения и принципы организации РБД. Использование формального аппарата для оптимизации схем отношений.
Тема 3.2. Физические модели	Организация обмена между оперативной и внешней

данных (внутренний уровень). Структура современной СУБД. Программное обеспечение работы с современными базами данных	памятью. Структуры хранения данных во внешней памяти.
РАЗДЕЛ 4. SQL — язык структурированных запросов	
Тема 4.1. Основные операторы языка SQL. Интерактивный SQL	Основные операторы языка SQL.
Тема 4.2. Использование языка SQL в прикладных программах. Направления развития баз данных	Программный (встроенный) SQL. Статический SQL. Динамический SQL.

1.2. Методические материалы по подготовке к практическим занятиям по дисциплине (модулю)

Практические (семинарские) занятия - одна из форм учебного занятия, направленная на развитие самостоятельности обучающихся и приобретение умений и навыков. Данные учебные занятия углубляют, расширяют, детализируют полученные ранее знания. Практическое занятие предполагает выполнение обучающимися по заданию и под руководством преподавателей одной или нескольких практических работ.

Цель практических занятий и семинаров состоит в развитии познавательных способностей, самостоятельности мышления и творческой активности обучающихся; углублении, расширении, детализировании знаний, полученных на лекции в обобщенной форме, и содействии выработке навыков профессиональной деятельности. В отдельных случаях на практических занятиях и семинарах руководителем занятия сообщаются дополнительные знания.

Для достижения поставленных целей и решения требуемого перечня задач практические занятия и семинары проводятся традиционными технологиями или с использованием активных и интерактивных образовательных технологий.

Возможные формы проведения практических (семинарских) занятий:

- Деловая игра - это метод группового обучения совместной деятельности в процессе решения общих задач в условиях максимально возможного приближения к реальным проблемным ситуациям. Имитационные игры - на занятиях имитируется деятельность какой-либо организации, предприятия или его подразделения. Имитироваться могут события, конкретная деятельность людей (деловое совещание, обсуждение плана) и обстановка, условия, в которых происходит событие или осуществляется деятельность (кабинет начальника цеха, зал заседаний). Исполнение ролей (ролевые игры) - в этих играх отрабатывается тактика поведения, действий, выполнение функций и обязанностей конкретного лица. Для проведения игр с исполнением роли разрабатывается модель-пьеса ситуации, между студентами распределяются роли с «обязательным содержанием», характеризующиеся различными интересами; в процессе их взаимодействия должно быть найдено компромиссное решение. «Деловой театр» (метод инсценировки) - в нем разыгрывается какая-либо ситуация, поведение человека в этой обстановке, обучающийся должен вжиться в образ определенного лица, понять его действия, оценить обстановку и найти правильную линию поведения. Основная задача метода инсценировки - научить ориентироваться в различных обстоятельствах, давать объективную оценку своему поведению, учитывать возможности других людей, влиять на их интересы, потребности и деятельность, не прибегая к формальным атрибутам власти, к приказу.

- Игровое проектирование - является практическим занятием или циклом занятий, суть которых состоит в разработке инженерного, конструкторского, технологического и других видов проектов в игровых условиях, максимально воссоздающих реальность. Этот метод отличается высокой степенью сочетания индивидуальной и совместной работы обучающихся.

- Познавательно-дидактические игры не относятся к деловым играм. Они предполагают лишь включение изучаемого материала в необычный игровой контекст и иногда содержат лишь элементы ролевых игр. Такие игры могут проводиться в виде копирования научных, культурных, социальных явлений (конкурс знатоков, «Поле чудес», КВН и т.д.) и в виде предметно-содержательных моделей, (например, игры-путешествия, когда надо разработать рациональный маршрут, пользуясь различными картами).

- Анализ конкретных ситуаций. Конкретная ситуация – это любое событие, которое содержит в себе противоречие или вступает в противоречие с окружающей средой. Ситуации могут нести в себе как позитивный, так и отрицательный опыт. Все ситуации делятся на простые, критические и экстремальные.

- Кейс-метод (от английского case – случай, ситуация) – усовершенствованный метод анализа конкретных ситуаций, метод активного проблемно-ситуационного анализа, основанный на обучении путем решения конкретных задач – ситуаций (решение кейсов). Непосредственная цель метода case-study - обучающиеся должны проанализировать ситуацию, разобраться в сути проблем, предложить возможные решения и выбрать лучшее из них. Кейсы делятся на практические (отражающие реальные жизненные ситуации), обучающие (искусственно созданные, содержащие значительные элементы условности при отражении в нем жизни) и исследовательские (ориентированные на проведение исследовательской деятельности посредством применения метода моделирования). Метод конкретных ситуаций (метод case-study) относится к неигровым имитационным активным методам обучения.

- Тренинг (англ. training от train — обучать, воспитывать) – метод активного обучения, направленный на развитие знаний, умений и навыков и социальных установок. Тренинг – форма интерактивного обучения, целью которого является развитие компетентности межличностного и профессионального поведения в общении. Достоинство тренинга заключается в том, что он обеспечивает активное вовлечение всех участников в процесс обучения. Можно выделить основные типы тренингов по критерию направленности воздействия и изменений – навыковый, психотерапевтический, социально-психологический, бизнес-тренинг.

- Метод Сократа (Майевтика) – метод вопросов, предполагающих критическое отношение к догматическим утверждениям, называется еще как метод «сократовской иронии». Это умение извлекать скрытое в человеке знание с помощью искусных наводящих вопросов, подразумевающего короткий, простой и заранее предсказуемый ответ.

- Интерактивная лекция – выступление ведущего обучающего перед большой аудиторией с применением следующих активных форм обучения: дискуссия, беседа, демонстрация слайдов или учебных фильмов, мозговой штурм.

- Групповая, научная дискуссия, диспут. Дискуссия — это целенаправленное обсуждение конкретного вопроса, сопровождающееся обменом мнениями, идеями между двумя и более лицами. Задача дискуссии - обнаружить различия в понимании вопроса и в споре установить истину. Дискуссии могут быть свободными и управляемыми. К технике управляемой дискуссии относятся: четкое определение цели, прогнозирование реакции оппонентов, планирование своего поведения, ограничение времени на выступления и их заданная очередность. Разновидностью свободной дискуссии является форум, где каждому желающему дается неограниченное время на выступление, при условии, что его выступление вызывает интерес аудитории. Каждый конкретный форум имеет свою тематику — достаточно широкую, чтобы в её пределах можно было вести многоплановое обсуждение.

- Дебаты – это четко структурированный и специально организованный публичный обмен мыслями между двумя сторонами по актуальным темам. Это разновидность публичной

дискуссии участников дебатов, направляющая на переубеждение в своей правоте третьей стороны, а не друг друга. Поэтому вербальные и невербальные средства, которые используются участниками дебатов, имеют целью получения определённого результата — сформировать у слушателей положительное впечатление от собственной позиции.

- Метод работы в малых группах. Групповое обсуждение кого-либо вопроса направлено на достижение лучшего взаимопонимания и нахождения истины. Групповое обсуждение способствует лучшему усвоению изучаемого материала. Оптимальное количество участников - 5-7 человек. Перед обучающимися ставится проблема, выделяется определенное время, в течение которого они должны подготовить аргументированный обдуманный ответ. Педагогический работник может устанавливать правила проведения группового обсуждения – задавать определенные рамки обсуждения, ввести алгоритм выработки общего мнения, назначить лидера и др.

- Круглый стол - общество, собрание в рамках более крупного мероприятия (съезда, симпозиума, конференции). Мероприятие, как правило, на которое приглашаются эксперты и специалисты из разных сфер деятельности для обсуждения актуальных вопросов. Данная модель обсуждения, основываясь на соглашениях, в качестве итогов даёт результаты, которые, в свою очередь, являются новыми соглашениями.

- Коллоквиум - (лат. colloquium — разговор, беседа) - одна из форм учебных занятий в системе образования, имеющая целью выяснение и повышение знаний обучающихся. На коллоквиумах обсуждаются: отдельные части, разделы, темы, вопросы изучаемого курса (обычно не включаемые в тематику семинарских и других практических учебных занятий), рефераты, проекты и др. работы обучающихся. Это научные собрания, на которых заслушиваются и обсуждаются доклады. Коллоквиум – это и форма контроля, массового опроса, позволяющая преподавателю в сравнительно небольшой срок выяснить уровень знаний студентов по данной теме дисциплины. Коллоквиум проходит обычно в форме дискуссии, в ходе которой обучающимся предоставляется возможность высказать свою точку зрения на рассматриваемую проблему, учиться обосновывать и защищать ее. Аргументируя и отстаивая свое мнение, обучающийся в то же время демонстрирует, насколько глубоко и осознанно он усвоил изученный материал.

- Метод «мозговой штурм» (мозговой штурм, мозговая атака, англ. brainstorming) — оперативный метод решения проблемы на основе стимулирования творческой активности, при котором участникам обсуждения предлагают высказывать как можно большее количество вариантов решения, в том числе самых фантастичных. Затем из общего числа высказанных идей отбирают наиболее удачные, которые могут быть использованы на практике. Является методом экспертного оценивания.

- Метод проектов - это способ достижения дидактической цели через детальную разработку проблемы (технологию), которая должна завершиться вполне реальным, осязаемым практическим результатом, оформленным тем или иным образом; это совокупность приёмов, действий обучающихся в их определённой последовательности для достижения поставленной задачи – решения проблемы, лично значимой для учащихся и оформленной в виде некоего конечного продукта. Основное предназначение метода проектов состоит в предоставлении учащимся возможности самостоятельного приобретения знаний в процессе решения практических задач или проблем, требующего интеграции знаний из различных предметных областей.

- Брифинг - (англ. briefing от англ. brief – короткий, недолгий) – краткая пресс-конференция, посвященная одному вопросу. Основное отличие: отсутствует презентационная часть. То есть практически сразу идут ответы на вопросы журналистов.

- Метод портфолио (итал. portfolio — 'портфель, англ. - папка для документов) - современная образовательная технология, в основе которой используется метод аутентичного оценивания результатов образовательной и профессиональной деятельности. Портфолио как подборка сертифицированных достижений, наиболее значимых работ и отзывов на них.

**Вопросы для самоподготовки к практическим (семинарским) занятиям по
разделам (темам) дисциплины (модуля)**

РАЗДЕЛ 1. Теория баз данных

Тема 1.1. Введение в базы данных. Общая характеристика основных понятий

Вопросы для самоподготовки:

1. Основные параметры команды утилиты psql для подключения к базе данных.
2. Основные команды утилиты psql для работы с буфером запросов.
3. Основные команды утилиты psql для работы со списком объектов.

Тема 1.2. Системы управления базами данных

Вопросы для самоподготовки:

1. Основные параметры команды утилиты psql для подключения к базе данных.
2. Основные команды утилиты psql для работы с буфером запросов.
3. Основные команды утилиты psql для работы со списком объектов.

РАЗДЕЛ 2. Проектирование приложения базы данных

Тема 2.1. Различные архитектурные решения, используемые при реализации многопользовательских СУБД. Краткий обзор СУБД

Вопросы для самоподготовки:

1. Последовательности

Тема 2.2. С Различные представления о данных в базах данных. Основные этапы проектирования баз данных

Вопросы для самоподготовки:

1. Индексы

РАЗДЕЛ 3. Введение в реляционные базы данных

Тема 3.1. Формализация реляционной модели

Вопросы для самоподготовки:

1. Оператор CREATE TABLE.
2. Определение ограничений CONSTRAINT.
3. Оператор вставки данных INSERT.
4. Оператор обновления UPDATE.
5. Оператор удаления строк DELETE.
6. Чтение строк и столбцов.

Тема 3.2. Физические модели данных (внутренний уровень). Структура современной СУБД. Программное обеспечение работы с современными базами данных

Вопросы для самоподготовки:

1. Шаблоны поиска в SQL.
2. Сортировка результатов запроса.
3. Встроенные функции SQL.
4. Группировка в запросах.
5. Вложенные запросы.
6. Внешние и внутренние соединения.

РАЗДЕЛ 4. SQL — язык структурированных запросов

Тема 4.1. Основные операторы языка SQL. Интерактивный SQL

Вопросы для самоподготовки:

1. Среда выполнения PL/SQL.
2. Неименованный блок PL/SQL.
3. Курсоры в PL/SQL.
4. Атрибуты курсора.
5. Неявные курсоры.
6. Курсоры-циклы.
7. Курсоры с параметрами.
8. Курсоры с обновлением.
9. Процедуры и функции PL/SQL.

Тема 4.2. Использование языка SQL в прикладных программах. Направления развития баз данных

Вопросы для самоподготовки:

1. Строковые и операторные триггеры.
2. Псевдозаписи триггера.
3. Триггерные предикаты.
4. Перечислите основные функции PostgreSQL для работы с числами.
5. Перечислите основные тригонометрические функции PostgreSQL.
6. Перечислите основные функции PostgreSQL для работы со строками.
7. Перечислите основные функции PostgreSQL для работы с датой и временем.

1.3. Учебно-наглядные пособия по разделам (темам) дисциплины (модуля)¹

РАЗДЕЛ 1. Теория баз данных

Тема 1.1. Введение в базы данных. Общая характеристика основных понятий

1.Маркин, А. В. Программирование на SQL в 2 ч. Часть 1 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 429 с. — (Высшее образование). — ISBN 978-5-534-15817-5. — Текст : электронный //

Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509818> (дата обращения: 08.03.2023).

2.Маркин, А. В. Программирование на SQL в 2 ч. Часть 2 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., испр. и доп. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-534-15818-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509819> (дата обращения: 08.03.2023).

Тема 1.2. Системы управления базами данных

1.Маркин, А. В. Программирование на SQL в 2 ч. Часть 1 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 429 с. — (Высшее образование). — ISBN 978-5-534-15817-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509818> (дата обращения: 08.03.2023).

2.Маркин, А. В. Программирование на SQL в 2 ч. Часть 2 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., испр. и доп. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-534-15818-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509819> (дата обращения: 08.03.2023).

РАЗДЕЛ 2. Проектирование приложения базы данных

Тема 2.1. Различные архитектурные решения, используемые при реализации многопользовательских СУБД. Краткий обзор СУБД

1.Маркин, А. В. Программирование на SQL в 2 ч. Часть 1 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 429 с. — (Высшее образование). — ISBN 978-5-534-15817-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509818> (дата обращения: 08.03.2023).

2.Маркин, А. В. Программирование на SQL в 2 ч. Часть 2 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., испр. и доп. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-534-15818-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509819> (дата обращения: 08.03.2023).

Тема 2.2. С Различные представления о данных в базах данных. Основные этапы проектирования баз данных

1.Маркин, А. В. Программирование на SQL в 2 ч. Часть 1 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 429 с. — (Высшее образование). — ISBN 978-5-534-15817-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509818> (дата обращения: 08.03.2023).

2.Маркин, А. В. Программирование на SQL в 2 ч. Часть 2 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., испр. и доп. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-534-15818-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509819> (дата обращения: 08.03.2023).

РАЗДЕЛ 3. Введение в реляционные базы данных

Тема 3.1. Формализация реляционной модели

1.Маркин, А. В. Программирование на SQL в 2 ч. Часть 1 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 429 с. — (Высшее образование). — ISBN 978-5-534-15817-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509818> (дата обращения: 08.03.2023).

2.Маркин, А. В. Программирование на SQL в 2 ч. Часть 2 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., испр. и доп. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-534-15818-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509819> (дата обращения: 08.03.2023).

Тема 3.2. Физические модели данных (внутренний уровень). Структура современной СУБД. Программное обеспечение работы с современными базами данных

1.Маркин, А. В. Программирование на SQL в 2 ч. Часть 1 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 429 с. — (Высшее образование). — ISBN 978-5-534-15817-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509818> (дата обращения: 08.03.2023).

2.Маркин, А. В. Программирование на SQL в 2 ч. Часть 2 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., испр. и доп. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-534-15818-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509819> (дата обращения: 08.03.2023).

РАЗДЕЛ 4. SQL — язык структурированных запросов

Тема 4.1. Основные операторы языка SQL. Интерактивный SQL

1.Маркин, А. В. Программирование на SQL в 2 ч. Часть 1 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 429 с. — (Высшее образование). — ISBN 978-5-534-15817-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509818> (дата обращения: 08.03.2023).

2.Маркин, А. В. Программирование на SQL в 2 ч. Часть 2 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., испр. и доп. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-534-15818-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509819> (дата обращения: 08.03.2023).

Тема 4.2. Использование языка SQL в прикладных программах. Направления развития баз данных

1.Маркин, А. В. Программирование на SQL в 2 ч. Часть 1 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 429 с. — (Высшее образование). — ISBN 978-5-534-15817-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509818> (дата обращения: 08.03.2023).

2.Маркин, А. В. Программирование на SQL в 2 ч. Часть 2 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., испр. и доп. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-534-15818-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509819> (дата обращения: 08.03.2023).

1.4. Методические материалы по подготовке к лабораторным занятиям по дисциплине (модулю)

Лабораторные занятия - одна из форм учебного занятия, направленная на развитие самостоятельности обучающихся и приобретение умений и навыков. Данные учебные занятия углубляют, расширяют, детализируют полученные ранее знания. Лабораторное занятие предполагает выполнение обучающимися по заданию и под руководством преподавателей одной или нескольких работ.

Цель лабораторных занятий состоит в развитии познавательных способностей, самостоятельности мышления и творческой активности обучающихся; углублении, расширении, детализировании знаний, полученных на лекции в обобщенной форме, и содействии выработке навыков профессиональной деятельности. В отдельных случаях на лабораторных занятиях руководителем занятия сообщаются дополнительные знания.

Для достижения поставленных целей и решения требуемого перечня задач лабораторные занятия проводятся традиционными технологиями или с использованием активных и интерактивных образовательных технологий.

При подготовке и работе во время проведения лабораторных занятий следует обратить внимание на следующие моменты: на процесс предварительной подготовки, на работу во время занятия, обработку полученных результатов, исправление полученных замечаний.

Предварительная подготовка к лабораторному занятию заключается в изучении теоретического материала в отведенное для самостоятельной работы время, ознакомление с инструктивными материалами с целью осознания задач занятия, техники безопасности при работе с приборами, веществами.

Работа во время проведения занятия включает:

- консультирование студентов преподавателями и вспомогательным персоналом с целью предоставления исчерпывающей информации, необходимой для самостоятельного выполнения предложенных преподавателем задач.
- самостоятельное выполнение заданий согласно обозначенной рабочей программой дисциплины (модуля) тематики.

2. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ) И САМОСТОЯТЕЛЬНОЙ РАБОТЕ

Освоение обучающимся дисциплины (модуля) *«Наименование дисциплины (модуля)»* предполагает изучение материалов дисциплины на аудиторных занятиях и в ходе самостоятельной работы. Аудиторные занятия проходят в форме лекций, семинаров, практических и лабораторных занятий. Самостоятельная работа включает разнообразный комплекс видов и форм работы обучающихся.

Для успешного освоения дисциплины (модуля) и достижения поставленных целей необходимо внимательно ознакомиться с рабочей программы дисциплины (модуля), доступной в электронной информационно-образовательной среде РГСУ.

Следует обратить внимание на списки основной и дополнительной литературы, на предлагаемые преподавателем ресурсы информационно-телекоммуникационной сети Интернет. Эта информация необходима для самостоятельной работы обучающегося.

При подготовке к аудиторным занятиям необходимо помнить особенности каждой формы его проведения.

Подготовка к учебному занятию лекционного типа заключается в следующем.

С целью обеспечения успешного обучения обучающийся должен готовиться к лекции, поскольку она является важнейшей формой организации учебного процесса, поскольку:

- знакомит с новым учебным материалом;
- разъясняет учебные элементы, трудные для понимания;
- систематизирует учебный материал;
- ориентирует в учебном процессе.

С этой целью:

- внимательно прочитайте материал предыдущей лекции;
- ознакомьтесь с учебным материалом по учебнику и учебным пособиям с темой прочитанной лекции;
- внесите дополнения к полученным ранее знаниям по теме лекции на полях лекционной тетради;
- запишите возможные вопросы, которые вы зададите лектору на лекции по материалу изученной лекции;
- постарайтесь уяснить место изучаемой темы в своей подготовке;
- узнайте тему предстоящей лекции (по тематическому плану, по информации лектора) и запишите информацию, которой вы владеете по данному вопросу.

Подготовка к занятию семинарского типа.

При подготовке и работе во время проведения занятий семинарского типа следует обратить внимание на следующие моменты: на процесс предварительной подготовки, на работу во время занятия, обработку полученных результатов, исправление полученных замечаний.

Предварительная подготовка к учебному занятию семинарского типа заключается в изучении теоретического материала в отведенное для самостоятельной работы время, ознакомление с инструктивными материалами с целью осознания задач практического занятия, техники безопасности при работе с приборами, веществами.

Работа во время проведения учебного занятия семинарского типа включает:

- консультирование студентов преподавателями и вспомогательным персоналом с целью предоставления исчерпывающей информации, необходимой для самостоятельного выполнения предложенных преподавателем задач.
- самостоятельное выполнение заданий согласно обозначенной рабочей программой дисциплины (модуля) тематики.

Самостоятельная работа.

Самостоятельная работа - планируемая учебная, учебно-исследовательская, научно-исследовательская работа студентов, выполняемая во внеаудиторное (аудиторное) время по заданию и при методическом руководстве преподавателя, но без его непосредственного участия (при частичном непосредственном участии преподавателя, оставляющем ведущую роль за работой студентов).

Для более углубленного изучения темы задания для самостоятельной работы рекомендуется выполнять параллельно с изучением данной темы. При выполнении заданий по возможности используйте наглядное представление материала.

Самостоятельная работа студентов в ВУЗе является важным видом учебной и научной деятельности студента. Самостоятельная работа студентов играет значительную роль в рейтинговой технологии обучения.

К современному специалисту в области медицины общество предъявляет достаточно широкий перечень требований, среди которых немаловажное значение имеет наличие у выпускников определенных навыков (компетенций) и умения самостоятельно добывать знания из различных источников, систематизировать полученную информацию, давать оценку

конкретной профессиональной ситуации. Формирование такого умения происходит в течение всего периода обучения.

В процессе самостоятельной работы студент приобретает необходимые для будущей специальности компетенции, навыки самоорганизации, самоконтроля, самоуправления, саморефлексии и становится активным самостоятельным субъектом учебной деятельности.

Самостоятельная работа студентов должна оказывать важное влияние на формирование личности будущего специалиста, его компетентность. Каждый студент самостоятельно определяет режим своей работы и меру труда, затрачиваемого на овладение учебным содержанием по каждой дисциплине. Он выполняет внеаудиторную работу по личному индивидуальному плану, в зависимости от его подготовки, времени и других условий.

Виды самостоятельной работы.

Работа с литературой.

При работе с книгой необходимо подобрать литературу, научиться правильно ее читать, вести записи. Для подбора литературы в библиотеке используются алфавитный и систематический каталоги. Важно помнить, что рациональные навыки работы с книгой - это всегда большая экономия времени и сил. Правильный подбор учебников рекомендуется преподавателем, читающим лекционный курс. Необходимая литература может быть также указана в методических разработках по данному курсу. Изучая материал по учебнику, следует переходить к следующему вопросу только после правильного уяснения предыдущего, описывая на бумаге все выкладки и вычисления (в том числе те, которые в учебнике опущены или на лекции даны для самостоятельного вывода). При изучении любой дисциплины большую и важную роль играет самостоятельная индивидуальная работа. Особое внимание следует обратить на определение основных понятий курса. Студент должен подробно разбирать примеры, которые поясняют такие определения, и уметь строить аналогичные примеры самостоятельно. Нужно добиваться точного представления о том, что изучаешь. Полезно составлять опорные конспекты. При изучении материала по учебнику полезно в тетради (на специально отведенных полях) дополнять конспект лекций. Там же следует отмечать вопросы, выделенные студентом для консультации с преподавателем. Выводы, полученные в результате изучения, рекомендуется в конспекте выделять, чтобы они при перечитывании записей лучше запоминались. Опыт показывает, что многим студентам помогает составление листа опорных сигналов, содержащего важнейшие и наиболее часто употребляемые формулы и понятия. Такой лист помогает запомнить формулы, основные положения лекции, а также может служить постоянным справочником для студента. Различают два вида чтения: первичное и вторичное. Первичное - это внимательное, неторопливое чтение, при котором можно остановиться на трудных местах.

Задача вторичного чтения - полное усвоение смысла целого (по счету это чтение может быть и не вторым, а третьим или четвертым). Самостоятельная работа с учебниками и книгами (а также самостоятельное теоретическое исследование проблем, обозначенных преподавателем на лекциях) – это важнейшее условие формирования у себя научного способа познания.

Чтение научного текста является частью познавательной деятельности. Ее цель – извлечение из текста необходимой информации. От того, насколько осознанно читающим собственная внутренняя установка при обращении к печатному слову (найти нужные сведения, усвоить информацию полностью или частично, критически проанализировать материал и т.п.) во многом зависит эффективность осуществляемого действия.

Методические рекомендации по составлению конспекта:

1. Внимательно прочитайте текст. Уточните в справочной литературе непонятные слова. При записи не забудьте вынести справочные данные на поля конспекта;
2. Выделите главное, составьте план;
3. Кратко сформулируйте основные положения текста, отметьте аргументацию автора;
4. Законспектируйте материал, четко следуя пунктам плана. При конспектировании старайтесь выразить мысль своими словами. Записи следует вести четко, ясно.

5. Грамотно записывайте цитаты. Цитируя, учитывайте лаконичность, значимость мысли. В тексте конспекта желательно приводить не только тезисные положения, но и их доказательства. При оформлении конспекта необходимо стремиться к емкости каждого предложения. Мысли автора книги следует излагать кратко, заботясь о стиле и выразительности написанного. Число дополнительных элементов конспекта должно быть логически обоснованным, записи должны распределяться в определенной последовательности, отвечающей логической структуре произведения. Для уточнения и дополнения необходимо оставлять поля. Овладение навыками конспектирования требует от студента целеустремленности, повседневной самостоятельной работы.

Методические материалы по самостоятельному решению задач

При самостоятельном решении задач нужно обосновывать каждый этап решения, исходя из теоретических положений курса. Если студент видит несколько путей решения проблемы (задачи), то нужно сравнить их и выбрать самый рациональный. Полезно до начала вычислений составить краткий план решения проблемы (задачи). Решение проблемных задач или примеров следует излагать подробно, вычисления располагать в строгом порядке, отделяя вспомогательные вычисления от основных. Решения при необходимости нужно сопровождать комментариями, схемами, чертежами и рисунками. Следует помнить, что решение каждой учебной задачи должно доводиться до окончательного логического ответа, которого требует условие, и по возможности с выводом. Полученный ответ следует проверить способами, вытекающими из существа данной задачи. Полезно также (если возможно) решать несколькими способами и сравнить полученные результаты. Решение задач данного типа нужно продолжать до приобретения твердых навыков в их решении.

Методические материалы к выполнению реферата

Реферат (от лат. *refere* – сообщать) – краткое изложение в письменном виде или в форме публикации доклада, содержания научного труда (трудов), литературы по теме. Работа над рефератом условно разделяется на выбор темы, подбор литературы, подготовку и защиту плана; написание теоретической части и всего текста с указанием библиографических данных используемых источников, подготовку доклада, выступление с ним. Тематика рефератов полностью связана с основными вопросами изучаемого курса.

Список литературы к темам не дается, и обучающиеся самостоятельно ведут библиографический поиск, причем им не рекомендуется ограничиваться университетской библиотекой.

Важно учитывать, что написание реферата требует от обучающихся определенных усилий. Особое внимание следует уделить подбору литературы, методике ее изучения с целью отбора и обработки собранного материала, обоснованию актуальности темы и теоретического уровня обоснованности используемых в качестве примеров фактов какой-либо деятельности.

Выбрав тему реферата, начав работу над литературой, необходимо составить план. Изучая литературу, продолжается обдумывание темы, осмысливание прочитанного, делаются выписки, сопоставляются точки зрения разных авторов и т.д. Реферативная работа сводится к тому, чтобы в ней выделились две взаимосвязанные стороны: во-первых, ее следует рассматривать как учебное задание, которое должен выполнить обучаемый, а во-вторых, как форму научной работы, творческого воображения при выполнении учебного задания.

Наличие плана реферата позволяет контролировать ход работы, избежать формального переписывания текстов из первоисточников.

Оформление реферата включает титульный лист, оглавление и краткий список использованной литературы. Список использованной литературы размещается на последней странице рукописи или печатной форме реферата. Реферат выполняется в письменной или печатной форме на белых листах формата А4 (210 x 297 мм). Шрифт Times New Roman, кегель 14, через 1,5 интервала при соблюдении следующих размеров текста: верхнее поле – 25 мм, нижнее – 20 мм, левое – 30 мм, правое – 15 мм. Нумерация страниц производится сверху листа, по центру. Титульный лист нумерации не подлежит.

Рефераты должны быть написаны простым, ясным языком, без претензий на наукообразность. Следует избегать сложных грамматических оборотов, непривычных терминов и символов. Если же такие термины и символы все-таки приводятся, то необходимо разъяснять их значение при первом упоминании в тексте реферата.

Объем реферата предполагает тщательный отбор информации, необходимой для краткого изложения вопроса. Важнейший этап – редактирование готового текста реферата и подготовка к обсуждению. Обсуждение требует хорошей ориентации в материале темы, умения выделить главное, поставить дискуссионный вопрос, привлечь внимание слушателей к интересной литературе, логично и убедительно изложить свои мысли.

Рефераты обязательно подлежат защите. Процедура защиты начинается с определения оппонентов защищающего свою работу. Они стремятся дать основательный анализ работы обучающимся, обращают внимание на положительные моменты и недостатки реферата, дают общую оценку содержанию, форме преподнесения материала, характеру использованной литературы. Иногда они дополняют тот или иной раздел реферата. Последнее особенно ценно, ибо говорит о глубоком знании обучающимся-оппонентом изучаемой проблемы.

Обсуждение не ограничивается заслушиванием оппонентов. Другие обучающиеся имеют право уточнить или опровергнуть какое-либо утверждение. Преподаватель предлагает любому обучающемуся задать вопрос по существу доклада или попытаться подвести итог обсуждению.

Алгоритм работы над рефератом

1. Выбор темы

Тема должна быть сформулирована грамотно (с литературной точки зрения);

В названии реферата следует поставить четкие рамки рассмотрения темы;

Желательно избегать слишком длинных названий;

Следует по возможности воздерживаться от использования в названии спорных с научной точки зрения терминов, излишней наукообразности, а также чрезмерного упрощения формулировок.

2. Реферат следует составлять из пяти основных частей: введения; основной части; заключения; списка литературы; приложений.

3. Основные требования к введению:

Во введении не следует концентрироваться на содержании; введение должно включать краткое обоснование актуальности темы реферата, где требуется показать, почему данный вопрос может представлять научный интерес и есть ли связь представляемого материала с современностью. Таким образом, тема реферата должна быть актуальна либо с научной точки зрения, либо с современных позиций.

Очень важно выделить цель, а также задачи, которые требуется решить для выполнения цели.

Введение должно содержать краткий обзор изученной литературы, в котором указывается взятый из того или иного источника материал, кратко анализируются изученные источники, показываются их сильные и слабые стороны;

Объем введения составляет две страницы текста.

4. Требования к основной части реферата:

Основная часть содержит материал, отобранный для рассмотрения проблемы;

Также основная часть должна включать в себя собственно мнение обучающихся и сформулированные самостоятельные выводы, опирающиеся на приведенные факты;

Материал, представленный в основной части, должен быть логически изложен и распределен по параграфам, имеющим свои названия;

В изложении основной части необходимо использовать сноски (в первую очередь, когда приводятся цифры и чьи-то цитаты);

Основная часть должна содержать иллюстративный материал (графики, таблицы и т. д.);

Объем основной части составляет около 10 страниц.

5. Требования к заключению:

В заключении формулируются выводы по параграфам, обращается внимание на выдвинутые во введении задачи и цели;

Заключение должно быть четким, кратким, вытекающим из содержания основной части.

6. Требования к оформлению списка литературы (по ГОСТу):

Необходимо соблюдать правильность последовательности записи источников: сначала следует писать фамилию, а после инициалы; название работы не ставится в кавычки; после названия сокращенно пишется место издания; затем идет год издания; наконец, называется процитированная страница.

Критерии оценки реферата

Обучающийся, защищающий реферат, должен рассказать о его актуальности, поставленных целях и задачах, изученной литературе, структуре основной части, сделанных в ходе работы выводах.

По окончанию выступления ему может быть задано несколько вопросов по представленной проблеме.

Оценка складывается из соблюдения требований к реферату, грамотного раскрытия темы, умения четко рассказывать о представленном реферате, способности понять суть задаваемых по работе вопросов и найти точные ответы на них.

Методические материалы к выполнению эссе

Эссе – литературное произведение небольшого объема, обычно прозаическое, свободной композиции, передающее индивидуальные впечатления, суждения, соображения автора о той или иной проблеме, теме, о том или ином событии или явлении. Это вид самостоятельной исследовательской работы обучающихся, с целью углубления и закрепления теоретических знаний и освоения практических навыков. Цель эссе состоит в развитии самостоятельного творческого мышления и письменного изложения собственных мыслей. При написании эссе обучающийся должен представить развернутый письменный ответ на теоретический или практический актуальный вопрос, объявленный преподавателем в аудитории непосредственно перед ее написанием. В процессе написания эссе разрешается пользоваться нормативно-правовыми актами, конспектом лекций (в печатном виде). Использование интернет-ресурсов не допускается. Темы эссе преподаватель предлагает из числа тех, которые обучающиеся уже рассматривали на лекциях или семинарских занятиях, исходя из содержания заданий в составе оценочных средств. По решению преподавателя, в качестве темы эссе может быть выбрана одна или несколько тем, которые могут быть распределены между обучающимися по желанию.

Требования к выполнению эссе:

1. Проводится письменно.

2. Эссе выполняется на компьютере (гарнитура Times New Roman, шрифт 14) через 1,5 интервала с полями: верхнее, нижнее – 2; правое – 3; левое – 1,5. Отступ первой строки абзаца – 1,25. Сноски – постраничные. Таблицы и рисунки встраиваются в текст работы. При этом обязательный заголовок таблицы надо размещать над табличным полем, а рисунки сопровождать подрисуночными подписями. При включении в эссе нескольких таблиц и/или рисунков их нумерация обязательна. Обязательна и нумерация страниц. Их целесообразно проставлять внизу страницы – по середине или в правом углу. Номер страницы не ставится на титульном листе, но в общее число страниц он включается. Объем эссе, без учета приложений, не должен превышать 5 страниц. Значительное превышение установленного объема является недостатком работы и указывает на то, что обучающийся не сумел отобрать и переработать необходимый материал.

3. Работа должна содержать собственные умозаключения по сути поставленной проблемы, включать самостоятельно проведенный анализ по сути этой проблемы, выводы, обобщающие авторскую позицию по поставленной проблеме.

Критерии оценки эссе:

«Отлично» – исключительные знания материала, абсолютное понимание сути, безукоризненное знание основных понятий и положений, логически и лексически грамотно изложенный, содержательный, аргументированный, конкретный и исчерпывающий ответ.

«Хорошо» – глубокие знания материала, правильное понимание сути, знание основных понятий и положений, содержательный, полный и конкретный ответ.

«Удовлетворительно» – твердые, но недостаточно полные знания, верное понимание сути, в целом правильный ответ.

«Неудовлетворительно» – непонимание сущности задания, грубые ошибки в ответе.

Методические материалы по выполнению тестирования.

Тестовые задания содержат вопросы и 3-4 варианта ответа по базовым положениям изучаемой темы/раздела, составлены с расчетом на знания, полученные обучающимся в процессе изучения темы/раздела.

Тестовые задания выполняются в письменной форме и сдаются преподавателю, ведущему дисциплину (модуль). На выполнение тестовых заданий обучающимся отводится 45 минут.

При обработке результатов оценочной процедуры используются: критерии оценки по содержанию и качеству полученных ответов, ключи, оценочные листы.

Критерии оценки теста:

«Зачтено» - если обучающийся правильно выполнил не менее 2/3 всей работы или допустил не более одной грубой ошибки и двух недочетов, не более одной грубой и одной негрубой ошибки, не более трех негрубых ошибок, одной негрубой ошибки и трех недочетов, при наличии четырех-пяти недочетов.

«Не зачтено» - если число ошибок и недочетов превысило норму для оценки 3 или правильно выполнено менее 2/3 всей работы.

Методические материалы по выполнению доклада.

Рекомендуется следующая структура доклада:

1. титульный лист, содержание доклада;
2. краткое изложение;
3. цели и задачи;
4. изложение характера исследований и рассмотренных проблем, гипотезы, спорные вопросы;
5. источники информации, методы сбора и анализа данных, степень их полноты и достоверности;
6. анализ и толкование полученных в работе результатов;
7. выводы и оценки;
8. библиография и приложения.

Время выступления докладчика не должно превышать 10 минут.

Основные требования к оформлению доклада:

- титульный лист должен включать название доклада, наименование предметной (цикловой) комиссии, фамилию обучающегося;
- все использованные литературные источники сопровождаются библиографическим описанием;
- приводимая цитата из источника берется в кавычки (оформляются сноски);
- единицы измерения должны применяться в соответствии с действующими стандартами;
- все названия литературных источников следует приводить в соответствии с новейшими изданиями;
- рекомендуется включение таблиц, графиков, схем, если они отражают основное содержание или улучшают ее наглядность;
- названия фирм, учреждений, организаций и предприятий должны именоваться так, как они указываются в источнике;

Критерии оценки доклада

При выполнении доклада обучающийся должен продемонстрировать умение кратко излагать прочитанный материал, а также умение обобщать и анализировать материал по теме доклада.

Презентация

Методические материалы к презентациям

1. Объём презентации 10 -20 слайдов.
2. На титульном слайде должно быть отражено:
 - наименование факультета;
 - тема презентации;
 - фамилия, имя, отчество, направление подготовки/ специальность, направленность (профиль)/ специализация, форма обучения, номер группы автора презентации;
 - фамилия, имя, отчество, степень, звание, должность руководитель работы;
 - год выполнения работы.
3. В презентации должны быть отражены обоснование актуальности представляемого материала, цели и задачи работы.
4. Содержание презентации должно включать наиболее значимый материал доклада, а также, при необходимости, таблицы, диаграммы, рисунки, фотографии, карты, видео – вставки, звуковое сопровождение.
5. Заключительный слайд должен содержать информацию об источниках информации для презентации.

Критерии оценки презентации

1. Объём презентации 10 -20 слайдов.
2. Правильность оформления титульного слайда.
3. Актуальность отобранного материала, обоснованность формулировки цели и задач работы.
4. Наглядность и логичность презентации, обоснованность использования таблиц, диаграмм, рисунков, фотографий, карт, видео – вставок, звукового сопровождения; правильный выбор шрифтов, фона, других элементов дизайна слайда.
5. Объём и качество источников информации (не менее 2-х интернет – источников и не менее 2-х литературных источников).

Методические материалы по подготовке к опросу

Самостоятельная работа обучающихся включает подготовку к опросу на практических занятиях. Для этого обучающийся изучает лекции, основную и дополнительную литературу, публикации, информацию из Интернет-ресурсов.

Тема и вопросы к практическим занятиям, вопросы для самоконтроля содержатся в рабочей программе и доводятся до студентов заранее.

Для подготовки к опросу обучающемуся необходимо ознакомиться с материалом, посвященным теме практического занятия, в учебнике или другой рекомендованной литературе, конспекте лекции, обратить внимание на усвоение основных понятий дисциплины, выявить неясные вопросы и подобрать дополнительную литературу для их освещения.

Критерии оценки опроса

- «Отлично»:
- дан полный, развернутый ответ на поставленный вопрос;
 - в ответе прослеживается четкая структура, логическая последовательность, отражающая сущность раскрываемых понятий, теорий, явлений;
 - знание по предмету демонстрируется на фоне понимания его в системе данной науки и междисциплинарных связей;
 - свободное владение терминологией;
 - ответы на дополнительные вопросы четкие, краткие;
- «Хорошо»:

– дан полный, развернутый ответ на поставленный вопрос, показано умение выделять существенные и несущественные признаки, причинно-следственные связи;

– ответ недостаточно логичен с единичными ошибками в частностях, исправленные студентом с помощью преподавателя;

– единичные ошибки в терминологии;

– ответы на дополнительные вопросы правильные, недостаточно полные и четкие.

«Удовлетворительно»:

– ответ не полный, с ошибками в деталях, умение раскрыть значение обобщённых знаний не показано, речевое оформление требует поправок, коррекции;

– логика и последовательность изложения имеют нарушения, студент не способен самостоятельно выделить существенные и несущественные признаки и причинно-следственные связи;

– ошибки в раскрываемых понятиях, терминах;

– студент не ориентируется в теме, допускает серьезные ошибки;

– студент не может ответить на большую часть дополнительных вопросов.

«Неудовлетворительно»:

– ответ представляет собой разрозненные знания с существенными ошибками по вопросу;

– присутствуют фрагментарность, нелогичность изложения, студент не осознает связь обсуждаемого вопроса с другими объектами дисциплины, речь неграмотная;

– незнание терминологии;

– ответы на дополнительные вопросы неправильные.

Методические материалы по выполнению практического задания

При выполнении практического задания обучающийся придерживается следующего алгоритма:

1. Записать дату, тему и цель задания;

2. Ознакомиться с правилами и условия выполнения практического задания;

3. Повторить теоретические задания, необходимые для рациональной работы и других практических действий, используя конспекты лекций и рекомендованную литературу, представленной в программе;

4. Выполнить работу по предложенному алгоритму действий;

5. Обобщить результаты работы, сформулировать выводы / дать ответы на контрольные вопросы;

Работа должна быть выполнена грамотно, с соблюдением культуры изложения. При использовании данных из учебных, методических пособий и другой литературы, периодических изданий, Интернет-источников должны иметься ссылки на вышеперечисленные.

Критерии оценки практического задания:

«Отлично» – правильный ответ, дается четкое обоснование принятому решению; рассуждения четкие последовательные логические; используются ссылки на полученные при изучении дисциплины знания; правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Хорошо» – правильный ответ, дается обоснование принятому решению; но с не существенными ошибками, в рассуждениях отсутствует логическая последовательность; используются ссылки на полученные при изучении дисциплины знания, правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Удовлетворительно» – правильный ответ, допускаются грубые ошибки в обосновании принятого решения; рассуждения не последовательные сумбурные; используются ссылки на полученные при изучении дисциплины знания; используются формулы, процедуры, понятия, имеющие прямое значение для подтверждения принятого решения, однако, при обращении к ним допускаются серьезные ошибки, студент не может правильно ими воспользоваться.

«Неудовлетворительно, не зачтено» – ответ неверный, отсутствует обоснование принятому решению; студент демонстрирует полное непонимание сути вопроса.

Для оценки решения ситуационной задачи (аналитического задания):

Оценка «отлично» выставляется, если задача решена грамотно, ответы на вопросы сформулированы четко. Эталонный ответ полностью соответствует решению студента, которое хорошо обосновано теоретически.

Оценка «хорошо» выставляется, если задача решена, ответы на вопросы сформулированы не достаточно четко. Решение студента в целом соответствует эталонному ответу, но не достаточно хорошо обосновано теоретически.

Оценка «удовлетворительно» выставляется, если задача решена не полностью, ответы не содержат всех необходимых обоснований решения.

Оценка «неудовлетворительно» выставляется, если задача не решена или имеет грубые теоретические ошибки в ответе на поставленные вопросы.

Методические материалы по выполнению лабораторного задания

При выполнении лабораторного задания обучающийся придерживается следующего алгоритма:

1. Записать дату, тему и цель задания;
2. Ознакомиться с правилами и условия выполнения задания;
3. Повторить теоретические задания, необходимые для рациональной работы и других практических действий, используя конспекты лекций и рекомендованную литературу, представленной в программе;
4. Выполнить работу по предложенному алгоритму действий;
5. Обобщить результаты работы, сформулировать выводы / дать ответы на контрольные вопросы;

Работа должна быть выполнена грамотно, с соблюдением культуры изложения. При использовании данных из учебных, методических пособий и другой литературы, периодических изданий, Интернет-источников должны иметься ссылки на вышеперечисленные.

Критерии оценки лабораторного задания:

«Отлично» – правильный ответ, дается четкое обоснование принятому решению; рассуждения четкие последовательные логические; используются ссылки на полученные при изучении дисциплины знания; правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Хорошо» – правильный ответ, дается обоснование принятому решению; но с не существенными ошибками, в рассуждениях отсутствует логическая последовательность; используются ссылки на полученные при изучении дисциплины знания, правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Удовлетворительно» – правильный ответ, допускаются грубые ошибки в обосновании принятого решения; рассуждения не последовательные сумбурные; используются ссылки на полученные при изучении дисциплины знания; используются формулы, процедуры, понятия, имеющие прямое значение для подтверждения принятого решения, однако, при обращении к ним допускаются серьезные ошибки, студент не может правильно ими воспользоваться.

«Неудовлетворительно, не зачтено» – ответ неверный, отсутствует обоснование принятому решению; студент демонстрирует полное непонимание сути вопроса.

Методические указания для подготовки к промежуточной аттестации.

Изучение учебных дисциплин (модулей) завершается зачетом/зачетом с оценкой или экзаменом. Подготовка к промежуточной аттестации способствует закреплению, углублению и обобщению знаний, получаемых, в процессе обучения, а также применению их к решению практических задач. Готовясь к экзамену, студент ликвидирует имеющиеся пробелы в знаниях, углубляет, систематизирует и упорядочивает свои знания. На зачете или экзамене студент демонстрирует то, что он освоил в процессе обучения по дисциплине (модулю).

Вначале следует просмотреть весь материал по дисциплине (модулю), отметить для себя трудные вопросы. Обязательно в них разобраться. В заключение еще раз целесообразно повторить основные положения, используя при этом листы опорных сигналов. Систематическая подготовка к занятиям в течение семестра позволит использовать время промежуточной аттестации для систематизации знаний.

3. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНИВАНИЯ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ И (ИЛИ) ОПЫТА ДЕЯТЕЛЬНОСТИ, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ

3.1. Организационные основы применения балльно-рейтинговой системы оценки успеваемости обучающихся по дисциплине (модулю)

Оценка качества освоения обучающимися дисциплины (модуля) реализуется в формате балльно-рейтинговой системы оценки успеваемости обучающихся (БРСО).

БРСО в ходе текущего контроля успеваемости и промежуточной аттестации осуществляется по 100-балльной шкале.

Академический рейтинг обучающегося по дисциплине (модулю) складывается из результатов:

- текущего контроля успеваемости (максимальный текущий рейтинг обучающегося 80 рейтинговых баллов);
- промежуточной аттестации (максимальный рубежный рейтинг обучающегося 20 рейтинговых баллов).

Условия оценки освоения обучающимся дисциплины (модуля) в формате БРСО доводятся преподавателем до сведения обучающихся на первом учебном занятии, а также размещены в свободном доступе в электронной информационно-образовательной среде Университета.

3.2. Проведение текущего контроля успеваемости обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося

В течение учебного семестра до промежуточной аттестации на основании утвержденной рабочей программы дисциплины (модуля) формируется текущий рейтинг обучающегося. Текущий рейтинг обучающегося складывается как сумма рейтинговых баллов, полученных им в течение учебного семестра по всем видам учебных занятий по учебной дисциплине.

В процессе текущего контроля оцениваются следующие действия обучающегося, направленные на освоение компетенций в рамках изучения учебной дисциплины:

- академическая активность (посещаемость учебных занятий, самостоятельное изучение содержания учебной дисциплины в электронной информационно-образовательной среде, соблюдение сроков сдачи практических заданий и текущих контрольных мероприятий и др.);
- выполнение и сдача текущих и итогового практических заданий (эссе, рефераты, творческие задания, кейс-задания, лабораторные работы, расчетные задания и др., активное участие в групповых интерактивных занятиях (дискуссии, WiKi-проекты и др.), защита проектов и др.);
- прохождение рубежей текущего контроля, включая соблюдение графика их прохождения в электронной информационно-образовательной среде.

Для планирования расчета текущего рейтинга обучающегося используются следующие пропорции:

Вид учебного действия	Максимальная рейтинговая оценка, баллов
академическая активность	10
практические задания	40
<i>из них: текущие практические задания</i>	20
<i>итоговое практическое задание</i>	20
рубежи текущего контроля	30

ИТОГО:	80

В течение учебного семестра по дисциплине (модулю) обучающимся должен быть накоплен текущий рейтинг не менее 52 рейтинговых баллов (65% от максимального значения текущего рейтинга).

Необходимыми условиями допуска обучающегося к промежуточной аттестации по дисциплине являются положительное прохождение обучающимся не менее 65% рубежей текущего контроля с накоплением не менее 65% максимального рейтингового балла за каждый рубеж текущего контроля и положительное выполнение итогового практического задания с накоплением не менее 65% максимального рейтингового балла, установленного за итоговое практическое задание.

Невыполнение вышеуказанных условий является текущей академической задолженностью, которая должна быть ликвидирована обучающимся до контрольного мероприятия промежуточной аттестации.

Сведения о наличии у обучающихся текущей академической задолженности, сроках и порядке добора рейтинговых баллов для её ликвидации доводятся до обучающихся педагогическим работником.

В случае неликвидации текущей академической задолженности, педагогический работник обязан во время контрольного мероприятия промежуточной аттестации поставить обучающемуся 0 рейтинговых баллов. В этом случае ликвидация текущей академической задолженности возможна в периоды проведения повторной промежуточной аттестации.

3.3. Проведение промежуточной аттестации обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося

Промежуточная аттестация по дисциплине (модулю) проводится в соответствии с Положением о промежуточной аттестации обучающихся по основным профессиональным образовательным программам высшего образования – программ специалитета в Российском государственном социальном университете и Положением о балльно-рейтинговой системе оценки успеваемости обучающихся по основным профессиональным образовательным программам высшего образования - программам бакалавриата, программам специалитета, программам магистратуры в Российском государственном социальном университете.

На промежуточную аттестацию отводится 20 рейтинговых баллов.

Ответы обучающегося на контрольном мероприятии промежуточной аттестации оцениваются педагогическим работником по 20 - балльной шкале, а итоговая оценка по дисциплине (модулю) выставляется по пятибалльной системе для дифференцированного зачета.

Критерии выставления оценки определяются Положением о балльно-рейтинговой системе оценки успеваемости обучающихся по основным профессиональным образовательным программам высшего образования - программам бакалавриата, программам специалитета, программам магистратуры в Российском государственном социальном университете.

В процессе определения рубежного рейтинга обучающегося используется следующая шкала:

Рубежный рейтинг	Критерии оценки освоения обучающимся учебной дисциплины в ходе контрольных мероприятий промежуточной аттестации

19-20 рейтинговых баллов	обучающийся глубоко и прочно усвоил программный материал, исчерпывающе, последовательно, грамотно и логически стройно его излагает, тесно увязывает с задачами и будущей деятельностью, не затрудняется с ответом при видоизменении задания, свободно справляется с задачами и практическими заданиями, правильно обосновывает принятые решения, умеет самостоятельно обобщать и излагать материал, не допуская
16-18 рейтинговых баллов	обучающийся твердо знает программный материал, грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос, может правильно применять теоретические положения и владеет необходимыми умениями и навыками при выполнении практических
13-15 рейтинговых баллов	обучающийся освоил основной материал, но не знает отдельных деталей, допускает неточности, недостаточно правильные формулировки, нарушает последовательность в изложении программного материала и испытывает затруднения в выполнении практических заданий
1-12 рейтинговых баллов	обучающийся не знает значительной части программного материала, допускает существенные ошибки, с большими затруднениями выполняет практические задания
0 рейтинговых баллов	не аттестован

Если результат контроля успеваемости в рамках проведения контрольных мероприятий промежуточной аттестации (рубежный рейтинг обучающегося) неудовлетворительный (получено менее 13 рейтинговых баллов), то промежуточная аттестация по учебной дисциплине (модулю) невозможна даже при наличии высокого текущего рейтинга, полученного по итогам текущего контроля по учебной дисциплине (модулю).

КОНСПЕКТЫ ЛЕКЦИОННЫХ ЗАНЯТИЙ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)

Проектирование баз данных

РАЗДЕЛ 1. Теория баз данных

Цели занятия.

Сформировать представление об основных понятиях объект, сущность, параметр, атрибут, таблица, запись, поле, домен, кортеж, первичный ключ и т.д. Три вида архитектур: локальная, «файл-сервер» и «клиент-сервер» (два варианта). Способы физического хранения данных и обмена данными. Сравнительный анализ и рекомендации по использованию каждого вида архитектур. Подробно рассматривается базовая технология СОМ. В краткой форме с указанием преимуществ и особенностей работы рассматриваются технологии ADO, CORBA и проч. Необходимость создания журнала аудита. Доступность и приемы заполнения журнала аудита. Назначение и использование просмотров. Просмотры редактируемые и не редактируемые. Область применения и преимущества каждого вида просмотра. Подробно рассматривается пять этапов проектирования базы данных. Состав работ, выполняемых на каждом этапе.

4. Структура лекционного занятия.

№ п/п	Содержание (кратко)	Методы и средства обучения
1	Основные понятия теории баз данных	1. Персональные компьютеры;
2	Описание целей и задач СУБД.	2. Средства доступа в Интернет; 3. Проектор.

5. Содержание лекционного занятия и взаимодействие с аудиторией.

Введение.

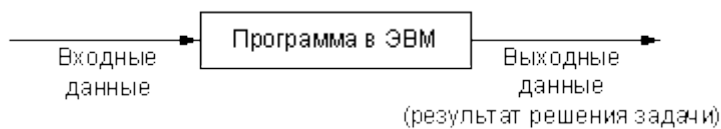
Формулирование темы занятия. Использование вводного примера, описание ситуации, демонстрация какого-либо процесса и т. д. (в зависимости от вида, замысла лекции и её структуры)

Тема 1.1. Введение в базы данных. Общая характеристика основных понятий **Текст лекции.**

Цель 1 лекции: показать, что с изменением вида решаемых на ЭВМ задач в программировании возникают новые виды представления данных, в том числе такой вид, как *базы данных*.

1.1. Развитие основных понятий представления данных

Любой вычислительный процесс представляет собой *отображение* (по определенному алгоритму) некоторых входных данных в выходные.



Соотношение сложности *представления* обрабатываемых данных и алгоритма вычислений определяет два класса задач:

- вычислительные задачи – достаточно простое представление данных и сложный, многооперационный процесс вычислений;
- задачи обработки данных (невывислительные задачи) – простой алгоритм обработки данных и сложное *представление обрабатываемых данных*.

На начальной стадии обучения программированию основное внимание уделяется разработке алгоритма решения задачи. Однако часто оказывается, что возможность (или невозможность) решения конкретной задачи зависит не только от выбранного алгоритма, но и от того, какие понятия используются для представления обрабатываемых данных.

Рассмотрим простейший пример вычисления по формуле:

$$Y = X^2 + 5X,$$

где X и Y – определенные числа, которые являются здесь элементарными единицами данных (*элементами данных*).

При программировании алгоритма решения этой задачи (*программирование формулы*) используется простейший вид данных – простая *переменная* (X и Y представляются в программе простыми переменными). Заметим, что простая *переменная* в системах программирования характеризуется определенным типом ее значений, которые должны выбираться при программировании. Даже в этом простейшем случае необходимо правильно выбрать *тип переменной*, причем от этого выбора может зависеть возможность или невозможность решения конкретной прикладной задачи (например, для представления конкретных данных не хватит отведенных разрядов).

Рассмотрим другой пример:

$$S = a_1 + a_2 + \dots + a_N.$$

Решение этой задачи в общем случае невозможно получить используя только простые переменные. Здесь обрабатывается не отдельное число, а последовательность чисел. В этом случае при программировании используется такой вид данных, как *массив* – совокупность элементов, с каждым из которых связан упорядоченный набор целых чисел, называемых индексами. Все элементы должны иметь одинаковый тип их значений, который и будет *типом массива*. В этом случае числа $a_1, a_2, \dots, a_N$ представляются в программе массивом $A(1), A(2), \dots, A(N)$. Приведенные примеры показывают, что изменение вида задач обуславливает необходимость использования других видов данных.

Ранние языки программирования (ФОРТРАН, АЛГОЛ-60) были предназначены для решения научно-технических вычислительных задач. В этих языках использовались только вышеуказанные виды данных (простые переменные и массивы) что было вполне достаточно.

Начиная с конца 60-х годов компьютеры начинают интенсивно использоваться для решения так называемых невывислительных задач, связанных с обработкой различного рода документов. Рассмотрим появление новых видов данных на примере упрощенных задач обработки данных.

Задача 1. Начисление заработной платы.

Рассматриваем задачу при двух упрощающих предположениях:

- сотруднику начисляется заработная плата на основе его оклада;
- никакие налоги и вычеты не учитываются.

Необходимые для решения этой задачи сведения о сотруднике представлены в следующей карточке НАЧИСЛЕНИЕ:

Фамилия, имя, отчество	Оклад	Количество отработанных дней в месяц	Начисленная сумма
ФИО	O	K_o	S

Для каждого работника начисленная сумма за определенный месяц рассчитывается по следующей формуле:

$$S = K_o O / K_r,$$

где K_r – количество рабочих дней в данном месяце.

Для каждого сотрудника соответствующие данные имеют конкретное значение, например:

Иванов Иван Иванович	1800	24	1800

Эти значения имеют смысл только во взаимосвязи друг с другом. Отдельно выбранное число 1800 теряет свой содержательный смысл, поэтому использовать такой вид данных, как простая *переменная*, здесь нельзя. В то же время набор соответствующих значений, характеризующих конкретного сотрудника, имеет разные типы (символьный и числовой), т.е. использовать для его представления такой вид данных, как *массив*, также нельзя. Таким образом, понятий "простая *переменная*" и "*массив*" недостаточно, чтобы представить соответствующую карточку.

Для описания аналогичных представлений данных в *предметной области* невычислительных задач вводится ряд новых понятий [[1.1]].

Элемент данных (поле) – наименьшая единица поименованных данных.

Для данного примера элементами данных являются $ФИО$, O , K_o , S .

Для описания карточки сотрудника используется понятие " *Логическая запись* ".

Логическая запись – поименованная совокупность элементов данных (полей).

Экземпляр логической записи – текущее значение элементов записи.

Для представления всего набора карточек сотрудников используется понятие " *Логический файл* "

Логический файл - поименованная совокупность всех экземпляров записей заданного типа.

Пример логического файла НАЧИСЛЕНИЕ:

Иванов Иван Иванович	1800	24	1800

Петров Петр Петрович	2200	20	1830

Сидоров Сидор Сидорович	2500	24	2500

Таким образом, с помощью введенных понятий можно описывать соответствующие данные. Для отображения этих понятий в современных языках программирования, предназначенных как для вычислительных задач, так и для задач обработки данных, введены новые виды данных.

В алгоритмическом языке *Паскаль* вводится такой вид данных, как *запись (RECORD)* – сложная *переменная* с несколькими компонентами, которые могут иметь разные типы. Кроме того, *доступ* к компонентам записи (полям) осуществляется не по индексу, а по имени. При программировании задачи 1 на языке *Паскаль* логическая запись НАЧИСЛЕНИЕ представляется видом данных *RECORD*, набор экземпляров логических записей сотрудников (*логический файл*) представляется "физическим" файлом, формируемым средствами языка *Паскаль* и операционной системы.

Salary = RECORD

FIO: string;

O: real;

Ko: real;

S: real;

END;

Отметим важную специфику таких невычислительных задач. Для этих задач характерны большие объемы данных (большое количество сотрудников, большое количество производимых изделий и т. п.). Указанные данные, как правило, используются для решения задачи многократно (зарплата начисляется постоянно каждый месяц), поэтому данные должны достаточно долго храниться в памяти ЭВМ. Для длительного хранения всегда используется внешняя *память*.

В связи с этим решение задачи 1 состоит из двух этапов.

1. Ввод исходных данных и занесение их во внешнюю *память*.


```

type
Salary = RECORD
FIO:      string;
O:        real;
Ko:        real;
S:real;
END;
FSalary = File of Salary;

var
    F: FSalary;
...
{ Ввод исходных данных }
repeat
    write('Введите количество сотрудников (не более',
        MaxN,' ): ');
    readln(N);
until (N>0) AND (N<=MaxN);
For I := 1 to N do
Begin
    Write('Введите фамилию сотрудника с номером ',I,' ');
    ReadLn(Sotr[i].FIO);
    Write('Введите оклад сотрудника с номером ', I, ' ');
    ReadLn(Sotr[i].O);
    Write('Введите кол-во отработанных дней сотрудника с
        номером ', I, ' ');
    ReadLn(Sotr[i].Ko);
End;

```

{ Занесение данных во внешнюю память }

Assign(F, 'MyFile.fsf');

Rewrite(F);

For I := 1 to N do

 Write(F, Sotr[i]);

Close(F);

...

2. Чтение исходных данных из внешней памяти, расчет начисленных сумм и вывод на печать.

...

{ Чтение данных из внешней памяти }

Assign(F, 'MyFile.fsf');

Reset(F);

For I := 1 to N do

 Read(F, Sotr[i]);

Close(F);

{ Расчет и печать начисленных сумм }

For I := 1 to N do

Begin

$Sotr[i].S := Sotr[i].O * Sotr[i].Ko / Kr;$

 WriteLn(Sotr[i].FIO, ': ', Sotr[i].S);

End;

...

Представленные программы решают поставленную задачу при сделанных предположениях. Необходимые для этого данные хранятся в файле *MyFile.fsf*, предназначенном только для решения этой задачи. Отметим, что в этом случае описание данных включено в прикладную программу. При изменении формата записей файла необходимо изменение прикладной программы. Таким образом, программная система, решающая поставленную задачу, определяет свои собственные данные и управляет ими. Такие программные системы называются файловыми системами [[1.2](#)], [[1.3](#)].

Задача 2. Учет кадрового состава.

Здесь обрабатываются сведения о сотруднике, представленные в карточке СОТРУДНИК:

Фамилия, имя, отчество	Должность	Год рождения	Оклад	Место жительства
ФИО	Д	Г	О	М

Решение задачи состоит из следующих этапов:

Ввод исходных данных и занесение их во внешнюю *память*.

Чтение исходных данных из внешней памяти с целью удаления, корректировки или добавления записи.

...

{ Чтение данных из внешней памяти }

Assign(F, 'MyFile.fsf');

Reset(F);

IsFound := False;

For I := 1 to N do

Begin

Read(F, Sotr);

If Sotr.FIO = KeyFio Then

Begin

IsFound := True;

Sotr.D := 'Начальник отдела';

Seek(F, FilePos(F)-1);

Write(F, Sotr);

Break;

End;

If IsFound Then

WriteLn('Корректировка успешно произведена')

Else WriteLn('Сотрудника ', KeyFio, ' не обнаружено');

Close(F);

...

В рассматриваемом случае задача 2 решается независимо от задачи 1.

Задача 3. Учет экономии фонда оплаты труда (ФОТ) в связи с болезнью сотрудников.

Обрабатываются сведения, представленные записями ЭКОНОМИЯ ФОТ:

Фамилия, отчество	имя, Оклад	Количество дней на больничном листе	Невыплаченная сумма
FIO	O	K_{дв}	SN

$$SN = K_{дв} O / K_r.$$

Программа решения задачи 3 аналогична программе решения задачи 1.

Рассмотрим типичный случай, когда все три вышеуказанные программные системы функционируют в одной организации. Отметим следующие принципиальные эксплуатационные недостатки:

Информация дублируется. В трех файлах присутствуют поля **FIO**, **O**, что приводит к существенному перерасходу памяти. При внесении изменений (например, изменении фамилии) приходится вносить одно и то же значение несколько раз в разные файлы, что приводит к увеличению затрат машинного времени. Существует потенциальная возможность противоречивости данных (в один файл изменения внесены, в другой – нет).

Устранить перечисленные недостатки можно, объединив соответствующие записи и создав единую информационную базу для всех вышеназванных задач. На первый взгляд наиболее естественно объединить все записи в одну, убрав дублирующие поля. Получаем возможный вариант объединения:

FIO	D	O	G	K_о	M	K_{дв}	S	SN

Дублирование информации полностью убрано. Расход памяти минимален. Недостатки устранены. Рассмотрим, как в этом случае изменится время решения задач 1–3. Время решения задачи прямо пропорционально объему считываемых из внешней памяти данных.

Обозначим **T_i**, **l_i**, **N_i** соответственно время решения, длину записи, число записей **i**-й задачи (**i** = 1, 2, 3) при использовании отдельных файлов для каждой задачи:

$$T_i \approx C \times l_i \times N_i$$

где **C** – некоторый коэффициент пропорциональности.

Обозначим **R_i**, **d**, **N** соответственно время решения **i**-й задачи (**i** = 1, 2, 3) при использовании файла объединенных записей, длину записи, число записей:

$$R_i \approx C \times d \times N$$

Заметим, что **N1 = N2 = N**, **N3 << N**.

Тогда время решения i -й задачи ($i = 1, 2$) при использовании объединенного файла увеличится в $R_i/T_i \approx d/l_i$ раз.

Для нашего примера время решения задач в зависимости от выбранной длины полей может изменяться в 2–3 раза. Таким образом, платой за *исключение* дублирования информации является увеличение времени решаемых задач. Заметим, что такое увеличение, как правило, допустимо.

Время решения задачи 3 увеличится в $R_3/T_3 \approx d \times N/l_3 \times N_3$ раз. Так как для данного примера $N_3 \ll N$, то $R_3 \gg T_3$. Время решения задачи 3 может увеличиться на несколько порядков, что совершенно недопустимо.

Рассмотрим другой вариант построения единой информационной базы. Объединим записи задач 1 и 2, запись задачи 3 оставим отдельно. Получим два типа записей:

FIO	D	O	G	K _o	S	M
FIO	O	K _{дв}	SN			

В этом случае дублирование остается (дублируются поля **FIO**, **O**). Но так как $N_3 \ll N$, то общий объем дублирования незначителен.

Время решения задачи 1 и 2 в этом случае незначительно возрастет по сравнению с вариантом отдельных файловых систем, время решения задачи 3 такое же, как и в начальном варианте отдельного файла. Такое *объединение* позволяет значительно уменьшить влияние недостатков и в то же время существенно увеличивает время решения всех задач. Все три задачи можно решать, используя общую информационную базу из двух типов записей. Отметим, что два приведенных типа записей связаны друг с другом по полю **FIO** (находятся в некотором отношении). Отметим, что приведенные варианты интеграции не исчерпывают все возможные способы *интеграции данных* для приведенных задач и к вопросу выбора наилучшего варианта вернемся в последующих лекциях.

Здесь очень важно, что в этом случае для решения вышеуказанных задач используется некоторый новый вид данных, формируемый на основе интеграции записей.

Для описания этого вида данных вводится новое понятие "База данных" [[1.1]].

База данных – совокупность экземпляров различных типов записей и отношений между записями и элементами.

Базу данных можно определить как совокупность взаимосвязанных хранящихся вместе данных при наличии такой минимальной избыточности, которая допускает их использование оптимальным образом для одного или нескольких приложений.

Таким образом, появление понятия "Базы данных" обусловлено возникновением нового класса невычислительных задач, при решении которых используются общие данные. В качестве основного критерия оптимальности функционирования *базы данных*, как правило, используются временные характеристики реализации запросов пользователей прикладными программами.

Краткие итоги. Рассмотрено развитие основных понятий представления данных. Описаны классические понятия программирования, связанные с данными (*переменная, массив*) и появление новых понятий программирования (*поле, запись, файл*) как следствие расширения круга решаемых задач и их отражения в системах программирования. Поставлена задача интегрирования данных при использовании несколькими задачами общих данных. Определено понятие *базы данных*.

Тема 1.2. Системы управления базами данных

Цель 2 лекции: показать необходимость создания программного интерфейса между прикладными программами и базой данных, определить понятие системы управления базами данных и сформулировать основные *функции СУБД*, вытекающие из задачи взаимодействия многих пользователей с базой данных.

В прикладной программе, использующей при решении задачи один или несколько отдельных файлов, за сохранность и *достоверность* данных отвечал программист, работающий с этой задачей. *Использование базы данных* предполагает работу с ней нескольких прикладных программ, решающих задачи разных пользователей.

Естественно, что за сохранность и *достоверность* интегрированных данных программист, решающий одну из прикладных задач, отвечать уже не может. Кроме того, расширение круга решаемых с использованием *базы данных* задач может приводить к появлению новых типов записей и отношений между ними. Такое изменение структуры *базы данных* не должно вести к изменению *множества* ранее разработанных и успешно функционирующих прикладных программных систем, работающих с базой данных. С другой стороны, возможное изменение любой из прикладных программ, в свою *очередь*, не должно приводить к изменению структуры данных. Все вышесказанное обуславливает необходимость отделения данных от прикладных программ.

Роль интерфейса между прикладными программами и базой данных, обеспечивающего их независимость, играет программный комплекс – *система управления базами данных (СУБД)* (рис. 2.1).

СУБД – программный комплекс поддержки интегрированной совокупности данных, предназначенный для создания, ведения и использования базы данных многими пользователями (прикладными программами).

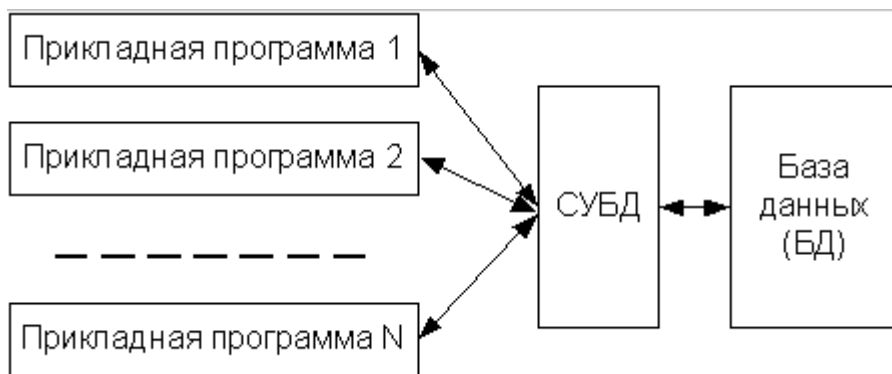


Рис. 2.1. Обеспечение независимости прикладных программ и базы данных. Определим еще одно понятие.

Банк данных – система языковых, алгоритмических, программных, технических и организационных средств поддержки интегрированной совокупности данных, а также сами эти данные, представленные в виде баз данных.

Перечислим основные функции системы управления базами данных.

1. *Определение* структуры создаваемой базы данных, ее инициализация и проведение начальной загрузки.

Как правило, создание структуры **базы данных** происходит в режиме диалога. СУБД последовательно запрашивает у пользователя необходимые данные. В большинстве современных СУБД база данных представляется в виде совокупности таблиц. Рассматриваемая функция позволяет описать и создать в памяти структуру таблицы, провести начальную загрузку данных в таблицы. Примеры таких действий для СУБД MS Access приведены на [рисунке 2.2](#).

Имя поля	Тип данных	Описание
Номер	Счетчик	
Фамилия	Текстовый	
Имя	Текстовый	
Отчество	Текстовый	
Номер_аттестата	Текстовый	
Дата_аттестата	Дата/время	

Общие	
Размер поля	Длинное целое
Новые значения	Последовательные
Формат поля	
Подпись	
Индексированное поле	Да (Совпадения не допускаются)

Имя поля может состоять из 64 знаков с учетом пробелов. Для справки по именам полей нажмите клавишу F1.

Рис. 2.2. Формирование структуры базы данных в СУБД Access

2. Предоставление пользователям возможности манипулирования данными (*выборка* необходимых данных, выполнение вычислений, разработка интерфейса ввода/вывода, *визуализация*).

Такие возможности в СУБД представляются либо на основе использования специального языка программирования, входящего в состав СУБД, либо с помощью графического интерфейса.

В MS Access реализация данной функции может быть реализована созданием запросов и форм ввода с помощью графического интерфейса ([рис. 2.3](#)).

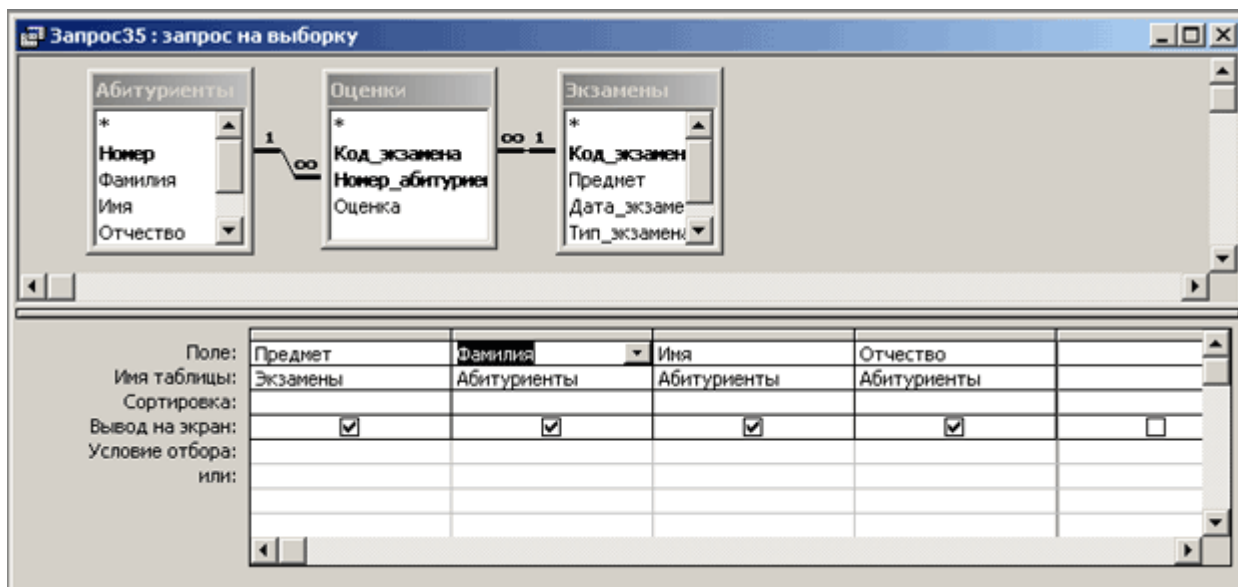


Рис. 2.3. Формирование запроса на выборку в СУБД Access

Для клиент-серверных *СУБД* существуют средства, позволяющие выполнять запросы, и *программные средства*, позволяющие создавать графический *интерфейс* пользователя.

3. Обеспечение независимости прикладных программ и данных (логической и физической независимости).

Важнейшим свойством *СУБД* является возможность поддерживать два независимых взгляда на базу данных – "взгляд пользователя", воплощаемый в логическом представлении данных, и его отражения в прикладных программах; и "взгляд системы" – физическое *представление* данных в памяти ЭВМ. Обеспечение логической независимости данных предоставляет возможность изменения (в определенных пределах) логического представления *базы данных* без необходимости изменения физических структур хранения данных. Таким образом, изменение логического представления данных в прикладных программах не приводит к изменению структур хранения данных. Обеспечение физической независимости данных предоставляет возможность изменять (в определенных пределах) способы организации *базы данных* в памяти ЭВМ не вызывая необходимости изменения "логического" представления данных. Таким образом, изменение способов организации *базы данных* не приводит к изменению прикладных программ.

4. Защита логической целостности **базы данных**.

Основной целью реализации этой функции является повышение достоверности данных в базе данных. *Достоверность* данных может быть нарушена при их вводе в *БД* или при неправомерных действиях процедур обработки данных, получающих и заносящих в *БД* неправильные данные. Для повышения достоверности данных в системе объявляются так называемые ограничения целостности, которые в определенных случаях "отлавливают" неверные данные. Так, во всех современных *СУБД* проверяется соответствие вводимых данных их типу, описанному при создании структуры. Система не позволит ввести символ в *поле* числового типа, не позволит ввести недопустимую дату и т.п. В развитых системах ограничения целостности описывает программист, исходя из содержательного смысла задачи, и их проверка осуществляется при каждом обновлении данных. Более подробно разные аспекты логической целостности *базы данных* будут рассматриваться в последующих разделах.

5. Защита физической целостности.

При работе ЭВМ возможны сбои в работе (например, из-за отключения электропитания), повреждение машинных носителей данных. При этом могут быть нарушены связи между данными, что приводит к невозможности дальнейшей работы. Развитые СУБД имеют средства восстановления *базы данных*. Важнейшим используемым понятием является понятие "*транзакции*". *Транзакция* – это единица действий, производимых с базой данных. В состав транзакции может входить несколько операторов изменения *базы данных*, но либо выполняются все эти *операторы*, либо не выполняется ни один. СУБД, кроме ведения собственно *базы данных*, ведет также журнал *транзакций*.

Необходимость использования *транзакций* в базах данных проиллюстрируем на упрощенном примере. Предположим, что *база данных* используется в некотором банке и один из клиентов желает перевести деньги на счет другого клиента банка. В базе данных хранится информация о количестве денег у каждого из клиентов. Нам нужно сделать два изменения в базе данных – уменьшить сумму денег на счете одного из клиентов и, соответственно, увеличить сумму денег на другом счете. Конечно, реальный перевод денег в банке представляет собой гораздо более сложный процесс, затрагивающий много таблиц, а возможно, и много баз данных. Однако суть остается та же – нужно либо совершить все действия (увеличить счет одного клиента и уменьшить счет другого), либо не выполнить ни одно из этих действий. Нельзя уменьшить сумму денег на одном счете, но не увеличить сумму денег на другом. Предположим также, что после выполнения первого из действий (уменьшения суммы денег на счете первого клиента) произошел сбой. Например, могла прерваться связь клиентского компьютера с базой данных или на клиентском компьютере мог произойти системный сбой, что привело к перезагрузке операционной системы. Что в этом случае стало с базой данных? Команда на уменьшение денег на счете первого клиента была выполнена, а вторая команда – на увеличение денег на другом счете – нет, что привело бы к противоречивому, неактуальному состоянию *базы данных*.

Использование механизма *транзакций* позволяет находить решение в этом и подобных случаях. Перед выполнением первого действия выдается команда начала транзакции. В транзакцию включается операция снятия денег на одном счете и увеличения суммы на другом счете. Оператор завершения транзакций обычно называется COMMIT. Поскольку после выполнения первого действия *транзакция* не была завершена, изменения не будут внесены в базу данных. Изменения вносятся (фиксируются) только после завершения транзакции. До выдачи данного оператора сохранения данных в базе не произойдет.

В нашем примере, поскольку оператор фиксации транзакции не был выдан, *база данных* "откатится" в первоначальное состояние – иными словами, суммы на счетах клиентов останутся те же, что и были до начала транзакции. Администратор базы данных может отслеживать состояние транзакций и в необходимых случаях вручную "откатывать" транзакции. Кроме того, в очевидных случаях СУБД самостоятельно принимает решение об "откате" транзакции.

Транзакции не обязательно могут быть короткими. Бывают транзакции, которые длятся несколько часов или даже несколько дней. Увеличение количества действий в рамках одной транзакции требует увеличения занимаемых системных ресурсов. Поэтому желательно делать транзакции по возможности короткими. В журнал *транзакций* заносятся все транзакции – и зафиксированные, и завершившиеся "откатом". Ведение журнала транзакций совместно с созданием резервных копий *базы данных* позволяет достичь высокой надежности *базы данных*.

Предположим, что *база данных* была испорчена в результате аппаратного сбоя компьютера, на котором был установлен *сервер СУБД*. В этом случае нужно использовать последнюю сделанную резервную копию *базы данных* и *журнал транзакций*. Причем применить к базе данных нужно только те транзакции, которые были зафиксированы после создания резервной копии. Большинство современных *СУБД* позволяют администратору воссоздать базу данных исходя из резервной копии и журнала транзакций. В таких системах в определенный момент *БД* копируется на резервные носители. Все обращения к *БД* записываются программно в *журнал изменений*. Если *база данных* разрушена, запускается процедура восстановления, в процессе которой в резервную копию из журнала изменений вносятся все произведенные изменения.

6. Управление полномочиями пользователей на *доступ* к базе данных.

Разные пользователи могут иметь разные полномочия по работе с данными (некоторые данные должны быть недоступны; определенным пользователям не разрешается обновлять данные и т.п.). В *СУБД* предусматриваются *механизмы* разграничения полномочий доступа, основанные либо на принципах паролей, либо на описании полномочий.

7. Синхронизация работы нескольких пользователей.

Достаточно часто может иметь место ситуация, когда несколько пользователей одновременно выполняют операцию обновления одних и тех же данных. Такие коллизии могут привести к нарушению логической целостности данных, поэтому система должна предусматривать меры, не допускающие обновление данных другим пользователям, пока работающий с этими данными *пользователь* полностью не закончит с ними работать. Основным используемым здесь понятием является понятие "*блокировка*". *Блокировки* необходимы для того, чтобы запретить различным пользователям возможность одновременно работать с базой данных, поскольку это может привести к ошибкам.

Для реализации этого запрета *СУБД* устанавливает блокировку на объекты, которые использует *транзакция*. Существуют разные типы блокировок – табличные, страничные, строчные и другие, которые отличаются друг от друга количеством заблокированных записей. Чаще других используется строчная *блокировка* – при обращении транзакции к одной строке блокируется только эта строка, остальные строки остаются доступными для изменения.

Таким образом, процесс внесения изменений в базу данных состоит из следующей последовательности действий: выдается оператор начала транзакции, выдается оператор изменения данных, *СУБД* анализирует оператор и пытается установить блокировки, необходимые для его выполнения, в случае успешной блокировки оператор выполняется, затем процесс повторяется для следующего оператора транзакции. После успешного выполнения всех операторов внутри транзакции выполняется оператор фиксации транзакции. *СУБД* фиксирует изменения, сделанные транзакцией, и снимает блокировки. В случае неуспеха выполнения какого-либо из операторов *транзакция* "откатывается", данные получают прежние значения, блокировки снимаются.

8. Управление ресурсами среды хранения.

БД располагается во внешней памяти ЭВМ. При работе в *БД* заносятся новые данные (занимается *память*) и удаляются данные (освобождается *память*). *СУБД* выделяет ресурсы памяти для новых данных, перераспределяет освободившуюся *память*, организует ведение очереди запросов к внешней памяти и т.п.

9. Поддержка деятельности системного персонала.

При эксплуатации *базы данных* может возникать необходимость изменения параметров *СУБД*, выбора новых методов доступа, изменения (в определенных пределах) структуры хранимых данных, а также выполнения ряда других общесистемных действий. *СУБД* предоставляет возможность выполнения этих и других действий для поддержки деятельности *БД* обслуживающему *БД* системному персоналу, называемому администратором *БД*.

Краткие итоги. Рассмотрено понятие системы управления базами данных как интерфейса между прикладными программами и базами данных. Введено понятие банка данных. Дана характеристика основных функций систем управления базами данных, вытекающих из задачи взаимодействия многих пользователей с базой данных:

- Определение структуры создаваемой базы данных, ее инициализация и проведение начальной загрузки
- Предоставление пользователям возможности манипулирования данными (выборка необходимых данных, выполнение вычислений, разработка интерфейса ввода/вывода, визуализация).
- Обеспечение независимости прикладных программ (логической и физической независимости).
 - Защита логической целостности базы данных.
 - Защита физической целостности.
 - Управление полномочиями пользователей на доступ к базе данных.
 - Синхронизация работы нескольких пользователей.
 - Управление ресурсами среды хранения.
 - Поддержка деятельности системного персонала.

РАЗДЕЛ 2. Проектирование приложения базы данных

Цели занятия.

Сформировать представление о способах создания файла базы данных: на уровне команд и диалоговый режим работы с утилитой *IBConsol*. Приемы модификации структуры базы данных на уровне команд, с рассмотрением последствий не корректных действий при выполнении данной операции. Команда удаления файла базы данных. Команды по созданию, модификации и удалению таблиц, с указанием особенностей работы каждой из команд. Допустимые типы полей и способы их задания и приемы указания дополнительных характеристик полей. Понятие целостности и достоверности данных. Опция *CHECK* в командах создания и модификации таблицы. Приемы создания вычисляемых (виртуальных) полей, установка и удаление условий ссылочной целостности данных. Понятия индекса и ключа, с указанием общих свойств и различий. Типы и виды ключей и два способа создания ключей. Приемы создания именованных и не именованных условий ссылочной целостности и особенностей их работы. Понятие набора данных, его размера. Методы перемещения курсора по записям набора данных. Четыре метода доступа к полям и их значениям. Методы поиска данных на стороне клиента, как по произвольному полю (полям), так и по индексному полю (полям), с указанием особенностей каждого вида поиска. Поиск на частичное совпадение и поиск без учета регистра. Поиск по индексному полю: на точное совпадение и пошаговое приближение (инкрементальный локатор). Варианты формирования запросов на добавление, редактирование и удаление записи: Особенности и преимущества каждого из вариантов. Методы сортировки данных на стороне сервера и клиента. Сортировка по-простому и сложному индексу. Запросы на извлечение записей с группировкой. Запросов с фильтрацией записей. Комбинированные запросы с сортировкой, группировкой и фильтрацией. Понятие отчета. Процедура формирования отчета. Отчеты с вычислениями, группировкой и сортировкой. Вывод отчета на экран и печать. Понятие и назначение генератора. Команды

создания, запуска, модификации, управления и удаления генератора. Понятие хранимой процедуры, её назначение. Создание различных типов хранимых процедур и их модификации. Назначение хранимых процедур, преимущества использования хранимых процедур. Вызов хранимых процедур. Понятие и назначение триггера. Создание, изменение и удаление триггера. Понятие отчета и его назначение. Компоненты, методы и свойства, с помощью которых формируется отчет. Редактирование отчета. Предварительный просмотр и вывод отчета на печать. Понятие транзакции. Излагаются методы создания и управления транзакциями (два способа). Сравнительная характеристика двух способов управления транзакциями. Состояния, в которых могут находиться транзакции. Конкурирующие транзакции. Понятие изоляции транзакций. Конфликты и приемы их разрешения.

4. Структура лекционного занятия.

№ п/п	Содержание (кратко)	Методы и средства обучения
1	Типы архитектур СУБД	1. Персональные компьютеры;
2	Проектирование баз данных. Первая стадия концептуального проектирования базы данных (концептуальное моделирование). Вторая стадия концептуального проектирования (Модели данных СУБД. Представление концептуальной модели средствами модели данных СУБД).	2. Средства доступа в Интернет; 3. Проектор.

5. Содержание лекционного занятия и взаимодействие с аудиторией.

Введение.

Формулирование темы занятия. Использование вводного примера, описание ситуации, демонстрация какого-либо процесса и т. д. (в зависимости от вида, замысла лекции и её структуры)

Тема 2.1. Различные архитектурные решения, используемые при реализации многопользовательских СУБД. Краткий обзор СУБД

Цель 3 лекции: показать основные варианты технологии работы нескольких пользователей с одной базой данных, связанные как с основными свойствами вычислительной техники, так и с развитием программного обеспечения.

Как уже отмечалось, понятие *базы данных* изначально предполагало возможность решения многих задач несколькими пользователями. В связи с этим, важнейшей характеристикой *современных СУБД* является наличие многопользовательской технологии работы. Разная реализация таких технологий в разное время была связана как с основными свойствами вычислительной техники, так и с развитием программного обеспечения. Дадим краткую характеристику этих технологий в хронологическом порядке.

3.1. Централизованная архитектура

При использовании этой технологии *база данных, СУБД* и *прикладная программа (приложение)* располагаются на одном компьютере (мэйнфрейме или персональном компьютере) ([рис.3.1.](#)). Для такого способа организации не требуется поддержки сети и все сводится к автономной работе. Работа построена следующим образом:

- База данных в виде набора файлов находится на жестком диске компьютера.

- На том же компьютере установлены СУБД и приложение для работы с БД .
- Пользователь запускает приложение. Используя предоставляемый приложением пользовательский интерфейс, он инициирует обращение к БД на выборку/обновление информации.
 - Все обращения к БД идут через СУБД, которая инкапсулирует внутри себя все сведения о физической структуре БД.
 - СУБД инициирует обращения к данным, обеспечивая выполнение запросов пользователя (осуществляя необходимые операции над данными).
 - Результат СУБД возвращает в приложение.
 - Приложение, используя пользовательский интерфейс, отображает результат выполнения запросов.

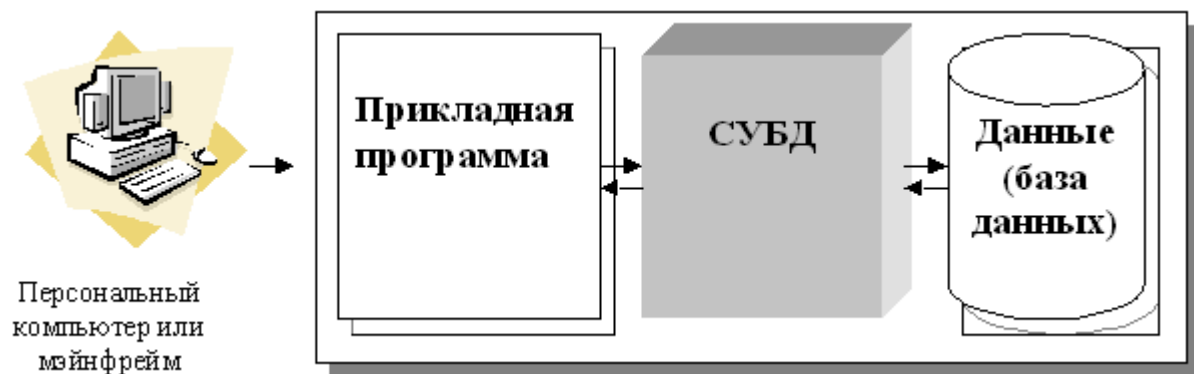


Рис. 3.1. Централизованная архитектура

Подобная архитектура использовалась в первых версиях СУБД DB2, Oracle, Ingres [[3.1]].

Многопользовательская технология работы обеспечивалась либо режимом мультипрограммирования (одновременно могли работать процессор и внешние устройства – например, пока в прикладной программе одного пользователя шло считывание данных из внешней памяти, программа другого пользователя обрабатывалась процессором), либо режимом разделения времени (пользователям по очереди выделялись кванты времени на выполнение их программ). Такая технология была распространена в период "господства" больших ЭВМ (IBM-370, ЕС-1045, ЕС-1060). Основным недостатком этой модели является резкое снижение производительности при увеличении числа пользователей.

3.2. Технология с сетью и файловым сервером (архитектура "файл-сервер")

Увеличение сложности задач, появление персональных компьютеров и локальных вычислительных сетей явились предпосылками появления новой архитектуры *файл-сервер*. Эта архитектура баз данных с сетевым доступом предполагает назначение одного из компьютеров сети в качестве выделенного сервера, на котором будут храниться файлы базы данных [[3.2]]. В соответствии с запросами пользователей файлы с *файл-сервера* передаются на рабочие станции пользователей, где и осуществляется основная часть обработки данных. Центральный сервер выполняет в основном только роль хранилища файлов, не участвуя в обработке самих данных (рис. 3.2.).

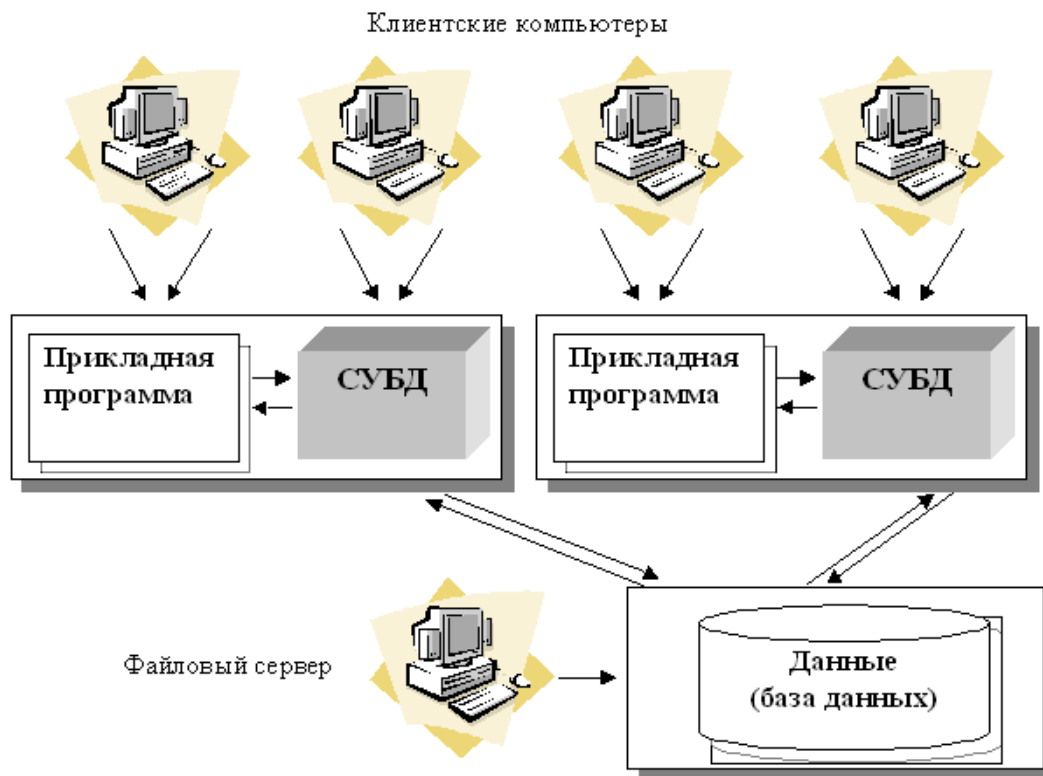


Рис. 3.2. Архитектура "файл-сервер"

Работа построена следующим образом:

- База данных в виде набора файлов находится на жестком диске специально выделенного компьютера (файлового сервера).
- Существует локальная сеть, состоящая из клиентских компьютеров, на каждом из которых установлены СУБД и приложение для работы с БД.
- На каждом из клиентских компьютеров пользователи имеют возможность запустить приложение. Используя предоставляемый приложением пользовательский интерфейс, он инициирует обращение к БД на выборку/обновление информации.
- Все обращения к БД идут через СУБД, которая инкапсулирует внутри себя все сведения о физической структуре БД, расположенной на файловом сервере.
- СУБД инициирует обращения к данным, находящимся на файловом сервере, в результате которых часть файлов БД копируется на клиентский компьютер и обрабатывается, что обеспечивает выполнение запросов пользователя (осуществляются необходимые операции над данными).
- При необходимости (в случае изменения данных) данные отправляются назад на файловый сервер с целью обновления БД.
- Результат СУБД возвращает в приложение.
- Приложение, используя пользовательский интерфейс, отображает результат выполнения запросов.

В рамках архитектуры "файл-сервер" были выполнены первые версии популярных так называемых настольных СУБД, таких, как dBase и Microsoft Access.

В литературе [[3.2]] указываются следующие основные недостатки данной архитектуры:

- При одновременном обращении множества пользователей к одним и тем же данным производительность работы резко падает, т.к. необходимо дожидаться пока пользователь, работающий с данными, завершит свою работу. В противном случае возможно затирание исправлений, сделанных одними пользователями, изменениями других пользователей.

- Вся тяжесть вычислительной нагрузки при доступе к БД ложится на приложение клиента, так как при выдаче запроса на выборку информации из таблицы вся таблица БД копируется на клиентскую машину и выборка осуществляется на клиенте. Таким образом, неоптимально расходуются ресурсы клиентского компьютера и сети. В результате возрастает сетевой трафик и увеличиваются требования к аппаратным мощностям пользовательского компьютера.

- Как правило, используется навигационный подход, ориентированный на работу с отдельными записями.

- В БД на *файл-сервере* гораздо проще вносить изменения в отдельные таблицы, минуя приложения, непосредственно из инструментальных средств (например, из утилиты Database Desktop фирмы Borland для файлов Paradox и dBase); подобная возможность облегчается тем обстоятельством, что фактически у таких СУБД база данных – понятие более логическое, чем физическое, поскольку под БД понимается набор отдельных таблиц, сосуществующих в отдельном каталоге на диске. Все это позволяет говорить о низком уровне безопасности – как с точки зрения хищения и нанесения вреда, так и с точки зрения внесения ошибочных изменений.

- Недостаточно развитый аппарат транзакций служит потенциальным источником ошибок в плане нарушения смысловой и ссылочной целостности информации при одновременном внесении изменений в одну и ту же запись.

3.3. Технология "клиент – сервер"

Использование технологии "*клиент – сервер*" предполагает наличие некоторого количества компьютеров, объединенных в *сеть*, один из которых выполняет особые *управляющие* функции (является сервером сети).

Так, *архитектура "клиент – сервер"* разделяет функции приложения пользователя (называемого клиентом) и сервера. *Приложение-клиент* формирует *запрос* к серверу, на котором расположена *БД*, на структурном языке *запросов SQL* (Structured Query Language), являющемся промышленным стандартом в мире реляционных *БД*. Удаленный *сервер* принимает *запрос* и переадресует его *SQL-серверу БД*. *SQL-сервер* – специальная *программа*, управляющая удаленной базой данных. *SQL-сервер* обеспечивает интерпретацию запроса, его выполнение в базе данных, формирование результата выполнения запроса и выдачу его приложению-клиенту. При этом ресурсы клиентского компьютера не участвуют в физическом выполнении запроса; клиентский *компьютер* лишь отправляет *запрос* к серверной *БД* и получает результат, после чего интерпретирует его необходимым образом и представляет пользователю. Так как клиентскому приложению посылается результат выполнения запроса, по сети "путешествуют" только те данные, которые необходимы клиенту. В итоге снижается нагрузка на *сеть*. Поскольку выполнение запроса происходит там же, где хранятся данные (на сервере), нет необходимости в пересылке больших пакетов данных. Кроме того, *SQL-сервер*, если это возможно, оптимизирует полученный *запрос* таким образом, чтобы он был выполнен в минимальное время с наименьшими накладными расходами [[3.2] , [3.3]]. *Архитектура* системы представлена на [рис. 3.3](#).

Все это повышает *быстродействие* системы и снижает *время ожидания* результата запроса. При выполнении запросов сервером существенно повышается степень безопасности данных, поскольку правила целостности данных определяются в базе данных на сервере и являются едиными для всех приложений, использующих эту *БД*. Таким образом, исключается возможность определения противоречивых правил поддержания целостности. Мощный аппарат транзакций, поддерживаемый *SQL-серверами*, позволяет исключить одновременное изменение одних и тех же данных различными пользователями и предоставляет возможность откатов к первоначальным значениям при внесении в *БД* изменений, закончившихся аварийно [[3.2] , [3.3]].

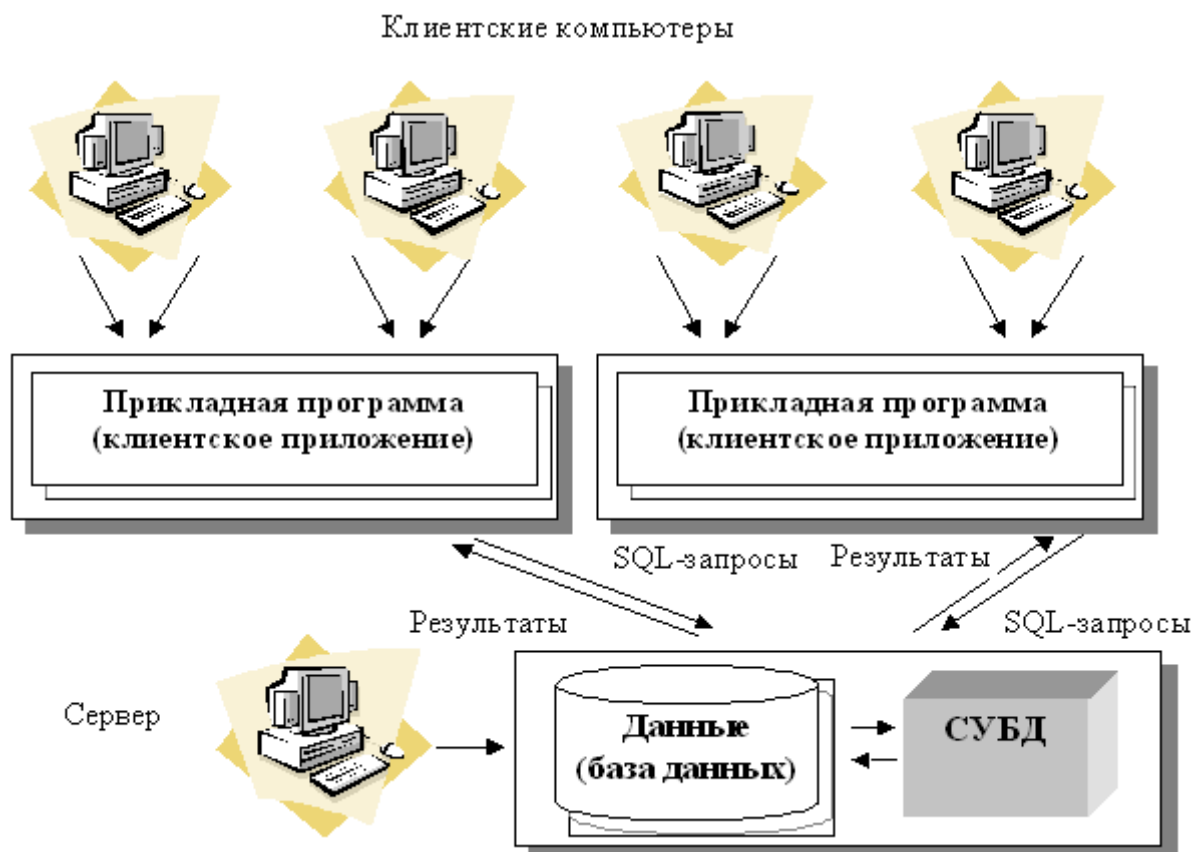


Рис. 3.3. Архитектура "клиент – сервер"

Итак, в результате работа построена следующим образом:

- База данных в виде набора файлов находится на жестком диске специально выделенного компьютера (сервера сети).
- СУБД располагается также на сервере сети.
- Существует локальная сеть, состоящая из клиентских компьютеров, на каждом из которых установлено клиентское приложение для работы с БД.
- На каждом из клиентских компьютеров пользователи имеют возможность запустить приложение. Используя предоставляемый приложением пользовательский интерфейс, он инициирует обращение к СУБД, расположенной на сервере, на выборку/обновление информации. Для общения используется специальный язык *запросов SQL*, т.е. по сети от клиента к серверу передается лишь текст запроса.
- СУБД инкапсулирует внутри себя все сведения о физической структуре БД, расположенной на сервере.
- СУБД инициирует обращения к данным, находящимся на сервере, в результате которых на сервере осуществляется вся обработка данных и лишь результат выполнения запроса копируется на клиентский компьютер. Таким образом СУБД возвращает результат в приложение.
- Приложение, используя пользовательский интерфейс, отображает результат выполнения запросов.

Рассмотрим, как выглядит разграничение функций между сервером и клиентом.

- Функции приложения-клиента:
 - Посылка запросов серверу.
 - Интерпретация результатов запросов, полученных от сервера.
 - Представление результатов пользователю в некоторой форме (интерфейс пользователя).
- Функции серверной части:
 - Прием запросов от приложений-клиентов.

- Интерпретация запросов.
- Оптимизация и выполнение запросов к БД.
- Отправка результатов приложению-клиенту.
- Обеспечение системы безопасности и разграничение доступа.
- Управление целостностью БД.
- Реализация стабильности многопользовательского режима работы.

В архитектуре "*клиент – сервер*" работают так называемые "промышленные" СУБД. Промышленными они называются из-за того, что именно СУБД этого класса могут обеспечить работу информационных систем масштаба среднего и крупного предприятия, организации, банка. К разряду промышленных СУБД принадлежат MS SQL Server, Oracle, Gupta, Informix, Sybase, DB2, InterBase и ряд других [[3.2]].

Как правило, SQL-сервер обслуживается отдельным сотрудником или группой сотрудников (администраторы SQL-сервера). Они управляют физическими характеристиками баз данных, производят оптимизацию, настройку и *переопределение* различных компонентов БД, создают новые БД, изменяют существующие и т.д., а также выдают привилегии (разрешения на *доступ* определенного уровня к конкретным БД, SQL-серверу) различным пользователям [[3.2]].

Рассмотрим основные достоинства данной архитектуры по сравнению с архитектурой "*файл-сервер*":

- Существенно уменьшается сетевой трафик.
- Уменьшается сложность клиентских приложений (большая часть нагрузки ложится на серверную часть), а, следовательно, снижаются требования к аппаратным мощностям клиентских компьютеров.
- Наличие специального программного средства – SQL-сервера – приводит к тому, что существенная часть проектных и программистских задач становится уже решенной.
- Существенно повышается целостность и безопасность БД.

К числу недостатков можно отнести более высокие финансовые *затраты* на аппаратное и *программное обеспечение*, а также то, что большое количество клиентских компьютеров, расположенных в разных местах, вызывает определенные трудности со своевременным обновлением клиентских приложений на всех компьютерах-клиентах. Тем не менее, архитектура "*клиент – сервер*" хорошо зарекомендовала себя на практике, в настоящий момент существует и функционирует большое количество БД, построенных в соответствии с данной архитектурой.

3.4. Трехзвенная (многозвенная) архитектура "клиент – сервер".

Трехзвенная (в некоторых случаях *многозвенная*) архитектура (N-tier или multi-tier). представляет собой дальнейшее совершенствование технологии "*клиент – сервер*". Рассмотрев архитектуру "*клиент – сервер*", можно заключить, что она является 2-звенной: первое звено – клиентское *приложение*, второе звено – *сервер БД* + сама БД. В *трехзвенной архитектуре* вся бизнес-логика (деловая логика), ранее входившая в клиентские приложения, выделяется в отдельное звено, называемое сервером приложений. При этом клиентским приложениям остается лишь пользовательский *интерфейс*. Так, в качестве клиентского приложения в описанном выше примере выступает Web-браузер.

Что улучшается при использовании *трехзвенной архитектуры*? Теперь при изменении бизнес-логики более нет необходимости изменять клиентские приложения и обновлять их у всех пользователей. Кроме того, максимально снижаются требования к аппаратуре пользователей.

Итак, в результате работа построена следующим образом:

- База данных в виде набора файлов находится на жестком диске специально выделенного компьютера (сервера сети).
- СУБД располагается также на сервере сети.

- Существует специально выделенный сервер приложений, на котором располагается программное обеспечение (ПО) делового анализа (бизнес-логика) [[3.1]].
- Существует множество клиентских компьютеров, на каждом из которых установлен так называемый "тонкий клиент" – клиентское приложение, реализующее интерфейс пользователя.
- На каждом из клиентских компьютеров пользователи имеют возможность запустить приложение – тонкий клиент. Используя предоставляемый приложением пользовательский интерфейс, он инициирует обращение к ПО делового анализа, расположенному на сервере приложений.
- Сервер приложений анализирует требования пользователя и формирует запросы к БД. Для общения используется специальный *язык запросов SQL*, т.е. по сети от сервера приложений к серверу БД передается лишь текст запроса.
- СУБД инкапсулирует внутри себя все сведения о физической структуре БД, расположенной на сервере.
- СУБД инициирует обращения к данным, находящимся на сервере, в результате которых результат выполнения запроса копируется на сервер приложений.
- Сервер приложений возвращает результат в клиентское приложение (пользователю).
- Приложение, используя пользовательский интерфейс, отображает результат выполнения запросов.

3.5. Краткий обзор СУБД

Многие авторы классифицируют *СУБД* на две большие категории: так называемые "настольные" и "серверные".

3.5.1. Настольные СУБД

Настольные СУБД используются для сравнительно небольших задач (небольшой объем обрабатываемых данных, малое количество пользователей). С учетом этого, указанные СУБД имеют относительно упрощенную архитектуру, в частности, функционируют в режиме файл-сервер, поддерживают не все возможные *функции СУБД* (например, не ведется журнал транзакций, отсутствует возможность автоматического восстановления базы данных после сбоев и т. п.). Тем не менее, такие системы имеют достаточно обширную область применения. Прежде всего, это государственные (муниципальные) учреждения, сфера образования, сфера обслуживания, малый и средний бизнес. Специфика возникающих там задач заключается в том, что объемы данных не являются катастрофически большими, частота обновлений не бывает слишком высокой, организация территориально обычно расположена в одном небольшом здании, количество пользователей колеблется от одного до 10–15 человек. В подобных условиях использование настольных СУБД для управления информационными системами является вполне оправданным, и они с успехом применяются.

Одними из первых СУБД были так называемые dBase-совместимые программные системы, разработанные разными фирмами. Первой широко распространенной системой такого рода была система dBase III – PLUS (фирма Achton-Tate). Развитый язык программирования, удобный интерфейс, доступный для массового пользователя, способствовали широкому распространению системы. В то же время работа системы в режиме интерпретации обуславливала низкую производительность на стадии выполнения. Это привело к появлению новых систем-компиляторов, близких к системе dBase III – PLUS: *Clipper* (фирма Nantucket Inc.), *FoxPro* (фирма Fox Software), *FoxBase+* (фирма Fox Software), *Visual FoxPro* (фирма Microsoft). Одно время достаточно широко использовалась СУБД PARADOX (фирма Borland International).

В последние годы очень широкое распространение получила система управления базами данных Microsoft Access, которая входит в целый ряд версий пакета Microsoft Office (фирма Microsoft).

3.5.2. Серверные СУБД

Для крупных организаций ситуация принципиально меняется. Там использование *файл-серверных* технологий является неудовлетворительным по описанным выше причинам. Поэтому на передний край борьбы за автоматизацию выходят так называемые серверные СУБД.

Основными производителями таких систем обработки и хранения данных являются 3 корпорации: Oracle, Microsoft и IBM. Диаграмма соотношения объемов продаж соответствующих систем (источник: *IDC Report*, Май 2006) приводится на [рис. 3.4](#).

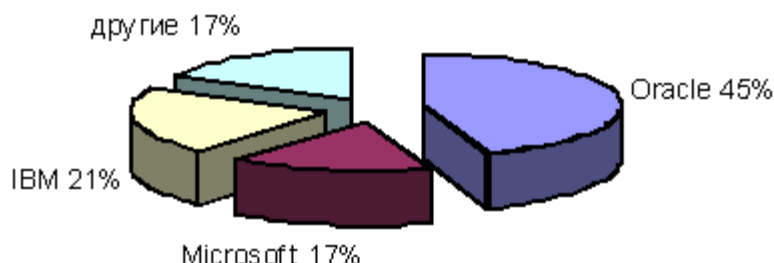


Рис. 3.4. Продажи ПО систем хранения данных в мире

Наиболее распространенными *клиент-серверными* системами здесь соответственно являются системы Oracle (разработчик *компания Oracle*), MS SQL Server (разработчик *компания Microsoft*), DB2, *Informix Dynamic Server* (*компания IBM*).

Дадим краткую характеристику этим системам.

MS SQL Server

К настоящему времени разработано несколько версий систем: MS SQL Server-2000, MS SQL Server -2005, MS SQL Server-2008. Приведем информацию о системе MS SQL Server-2008 с сервера Microsoft (<http://www.microsoft.com/rus/SQL/2008/default.mspx>)

Microsoft [®] SQL Server [™] 2008 - это законченное предложение в области баз данных и анализа данных для быстрого создания *масштабируемых решений* электронной коммерции, бизнес-приложений и хранилищ данных. Оно позволяет значительно сократить время выхода этих решений на рынок, одновременно обеспечивая масштабируемость, отвечающую самым высоким требованиям. В SQL Server включена поддержка языка XML и протокола HTTP, средства повышения быстродействия и доступности, позволяющие распределить нагрузку и обеспечить бесперебойную работу, функции для улучшения управления и настройки, снижающие совокупную стоимость владения.

Платформа бизнес-анализа SQL Server 2008, тесно интегрированная с Microsoft Office, предоставляет развитую масштабируемую инфраструктуру для внедрения мощных возможностей бизнес-анализа в рабочий процесс всех бизнес-подразделений вашей компании, открывая доступ к нужной бизнес-информации через знакомый интерфейс MS Excel и MS Word.

MS SQL Server-2008 поддерживает создание и работу с корпоративным хранилищем данных, объединяющим информацию со всех систем и приложений, позволяющим получить единую комплексную картину бизнеса вашей компании.

MS SQL Server-2008 предоставляет масштабируемый и высокопроизводительный "процессор данных" - для самых ответственных и требовательных бизнес-приложений, тем, кому необходим высочайший уровень надежности и защиты, позволяя при этом снизить совокупную стоимость владения за счет расширенных возможностей по управлению серверной инфраструктурой.

MS SQL Server-2008 предлагает разработчикам развитую, удобную и функциональную среду программирования, включая средства работы с веб службами, инновационные технологии доступа к данным – все, что необходимо для эффективной работы с данными любых типов и форматов.

Отдельные аспекты MS SQL Server – 2008 будут описаны в лекциях "[Структура современной СУБД на примере Microsoft SQL Server 2008](#)" и "[Направления развития баз данных](#)" 14.

Oracle

К настоящему времени разработано несколько версий систем, каждая из которых включает целую линейку продуктов, например Oracle 8, Oracle 9i, Oracle 10g.

Соответствующие линейки продуктов включают как собственно СУБД (например Oracle Database 10g, Oracle Database 11g), так и средства разработки и анализа данных.

Приведем информацию о системе с сервера Oracle (http://www.oracle.com/global/ru/mid/oracle_products/database.html).

Oracle предлагает комплексные, открытые, доступные и удобные в использовании технологические решения. Готовые пакетизируемые решения автоматически включают в свою стоимость базу данных, сервер приложений, интеграционную платформу, инструменты аналитики и управления неструктурированными данными. Масштабируемые бизнес-приложения Oracle могут быть легко интегрированы с ИТ-инфраструктурой предприятия без потери уже вложенных в ИТ инвестиций.

СУБД Oracle Database 11g обеспечивает улучшенные характеристики за счет автоматизации задач администрирования и обеспечения лучших в отрасли возможностей по безопасности и соответствию нормативно-правовым актам в области защиты информации. Появилось больше функций автоматизации, самодиагностики и управления. Среди характеристик системы можно отметить управление большими объемами данных с использованием распределенных таблиц и компрессии, эффективную защиту данных, возможность полного восстановления, возможность интеграции геофизических данных медиа-контента в бизнес-процесс и т.д.

Серверы баз данных компании IBM

К настоящему времени разработаны линейки продуктов DB2 и *Informix*, включающие как собственно СУБД так и средства разработки и анализа данных (DB2 Universal Database DB2 Personal Edition, DB2 Enterprise 9 и др., а также *Informix* Dynamic Server, *Informix* Dynamic Server Express, *Informix* Extended Parallel Server и др.

Приведем информацию о части таких систем с сервера (<http://www-01.ibm.com/software/ru/data/?pgel=ibmhzn>)

Универсальный сервер баз данных DB2 Universal Database - это масштабируемая, объектно-реляционная система управления базами данных с интегрированной поддержкой мультимедиа и Web, работающая на системах от персональных компьютеров и серверов на процессорах Intel до Unix, от однопроцессорных систем до симметричных многопроцессорных систем (SMP) и систем с массовым параллелизмом (MPP), на хостах AS/400 и мейнфреймах. DB2 Universal Database объединяет в себе высокую производительность систем обработки транзакций в режиме on-line, объектно-реляционные расширения, усовершенствованные средства оптимизации с возможностями параллельной обработки и поддержкой очень больших баз данных. DB2 Universal Database также имеет новые встроенные средства для облегчения переноса на свою базу приложений, разработанных на других системах управления базами данных, таких как Oracle, Microsoft, Sybase и *Informix*. Помимо этого, DB2 Universal Database включает в себя дополнительные средства поддержки систем аналитической обработки в реальном времени (OLAP) и систем поддержки принятия решений, множество простых в использовании расширений (DB2 extenders). DB2 Universal Database доступна на абсолютном большинстве ключевых платформ, что дает заказчикам ту гибкость, которая им необходима.

Кроме вышеуказанных зарубежных систем отметим и отечественную разработку – СУБД НИКА, преемницу широко распространенной в Советском Союзе СУБД ИНЕС для ЕС ЭВМ.

Краткие итоги. В лекции рассмотрены различные архитектурные решения, используемые при реализации *многопользовательских СУБД. Централизованная*

архитектура. Технология с сетью и файловым сервером (архитектура "файл-сервер"). Архитектура "клиент – сервер" (распределенная модель вычислений). Трехзвенная (многозвенная) архитектура клиент – сервер. Дан обзор современных СУБД (настольные СУБД, серверные СУБД).

Тема 2.2. Различные представления о данных в базах данных. Основные этапы проектирования баз данных

Цель 4 лекции: Показать существование различных представлений о данных (различных моделей) у разных групп лиц, работающих с данными. Рассмотреть отражение этих представлений в трехуровневой архитектуре базы данных (внешний уровень, концептуальный уровень, внутренний уровень), сформулировать достоинство трехуровневой архитектуры. Выделить основные этапы проектирования базы данных как процесса построения вышеуказанных моделей.

4.1. Различные представления о данных в базах данных

Создание базы данных предполагает интеграцию данных, предназначенных для решения нескольких прикладных задач разных пользователей. Соответственно, при интеграции данных должны учитываться требования к данным каждого пользователя, основанные на его представлении о данных и связях между ними. Далее эти требования должны обобщаться в единое представление, которое и будет служить основой для построения единой базы данных (рис. 4.1).

Обобщение представлений всех пользователей о данных называется **концептуальной моделью** (схемой) БД. **Концептуальная модель** представляет информационное описание предметной области с учетом логических взаимосвязей, поэтому её еще называют инфологической (информационно-логической) моделью. В модели отсутствуют какие-либо понятия, связанные с ЭВМ, памятью ЭВМ, способами размещения данных в памяти ЭВМ, и, по сути, это модель только предметной области.

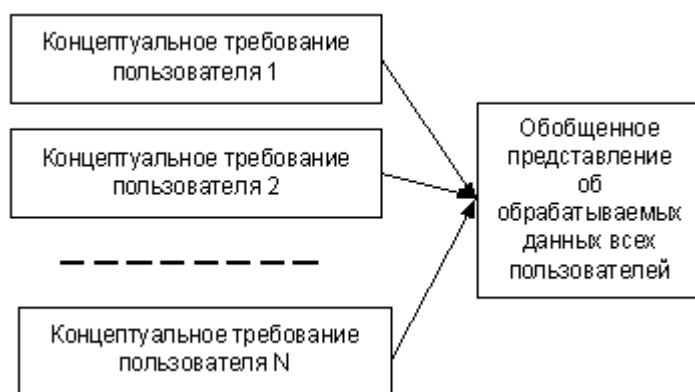


Рис. 4.1. Обобщение представления пользователей о данных

Как уже отмечалось, для создания базы данных и работы с ней используется система управления базами данных. Каждая конкретная СУБД поддерживает определенный вид данных (форматов записей и отношений), называемый *моделью данных СУБД*.

Следующий этап разработки базы данных предполагает выбор представления концептуальной модели с помощью модели данных конкретной СУБД. Полученное таким образом представление концептуальной модели называется логической моделью БД. Или другими словами, логическая модель – это концептуальная схема, специфицированная в языке конкретной СУБД. Логическая модель представляет данные и элементы данных вне зависимости от их содержания и среды хранения. Далее разработчик системы средствами СУБД отображает полученную логическую модель БД в память ЭВМ и определяет методы доступа. Полученное представление данных в памяти ЭВМ называется внутренним представлением или структурой хранения. Прикладные программы работают с логической моделью, причем каждому пользователю представляется подмножество этой логической модели (подсхема), отражающее его представление о предметной области. Каждая прикладная программа "видит" и обрабатывает только те данные, которые необходимы именно ей.

Соответствующее "видение" данных прикладными программами (пользователями) представляет собой внешние представления. Взаимосвязь вышеуказанных моделей изображена на рис. 4.2.

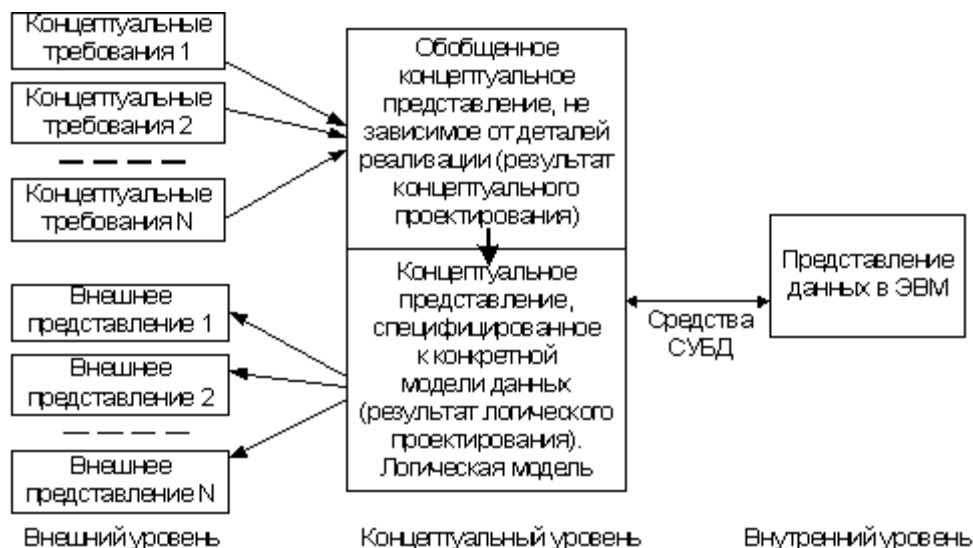


Рис. 4.2. Различные представления о данных в БД

На данной схеме выделены три различных уровня описания данных (внешний, концептуальный, внутренний). Эти уровни формируют так называемую *трехуровневую архитектуру ANSI/SPARC*, предложенную в 1975 г. Комитетом планирования стандартов и норм SPARC (Standards Planning and Requirements Committee) Национального института стандартизации США (American National Standards Institute – ANSI). Основная цель этой архитектуры состоит в отделении пользовательского представления о данных в базе данных от их физического представления. Использование таких представлений о данных позволяет обеспечить выполнение основного требования к БД – независимости программ и данных. При изменении прикладных программ может измениться соответствующее внешнее представление, логическая модель данных не изменяется и, соответственно, не будут изменяться другие прикладные программы. При изменении внутреннего представления (структур хранения) логическая модель не изменяется, соответственно, не изменяются прикладные программы.

Использование соответствующих представлений также позволяет четко разграничить полномочия различных лиц, работающих с базой данных.

Соответствующие представления позволяют описать "видение" *базы данных* разными лицами, работающими с ней:

- *внешнее представление* – представление специалиста предметной области (пользователя);
- *внешнее представление* и *логическая модель* – представление прикладного программиста, разрабатывающего конкретное приложение для пользователя;
- *логическая модель* и *внутреннее представление* – представление системного программиста, администрирующего базу данных.

4.2. Основные этапы проектирования базы данных

Проектирование данных (базы данных) представляет собой процесс последовательного отображения исследуемых явлений реального мира в виде данных в памяти ЭВМ (рис. 4.3).

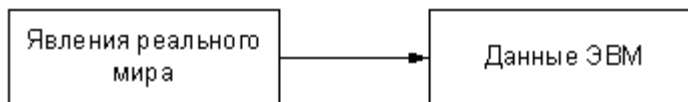


Рис. 4.3. Общая схема проектирования

Конкретные явления реального мира, представляющие интерес для проводимого исследования, будем называть предметной областью.

Проектирование (моделирование) базы данных представляет собой многоэтапный процесс. Основные этапы этого процесса приведены нарис. 4.4.).

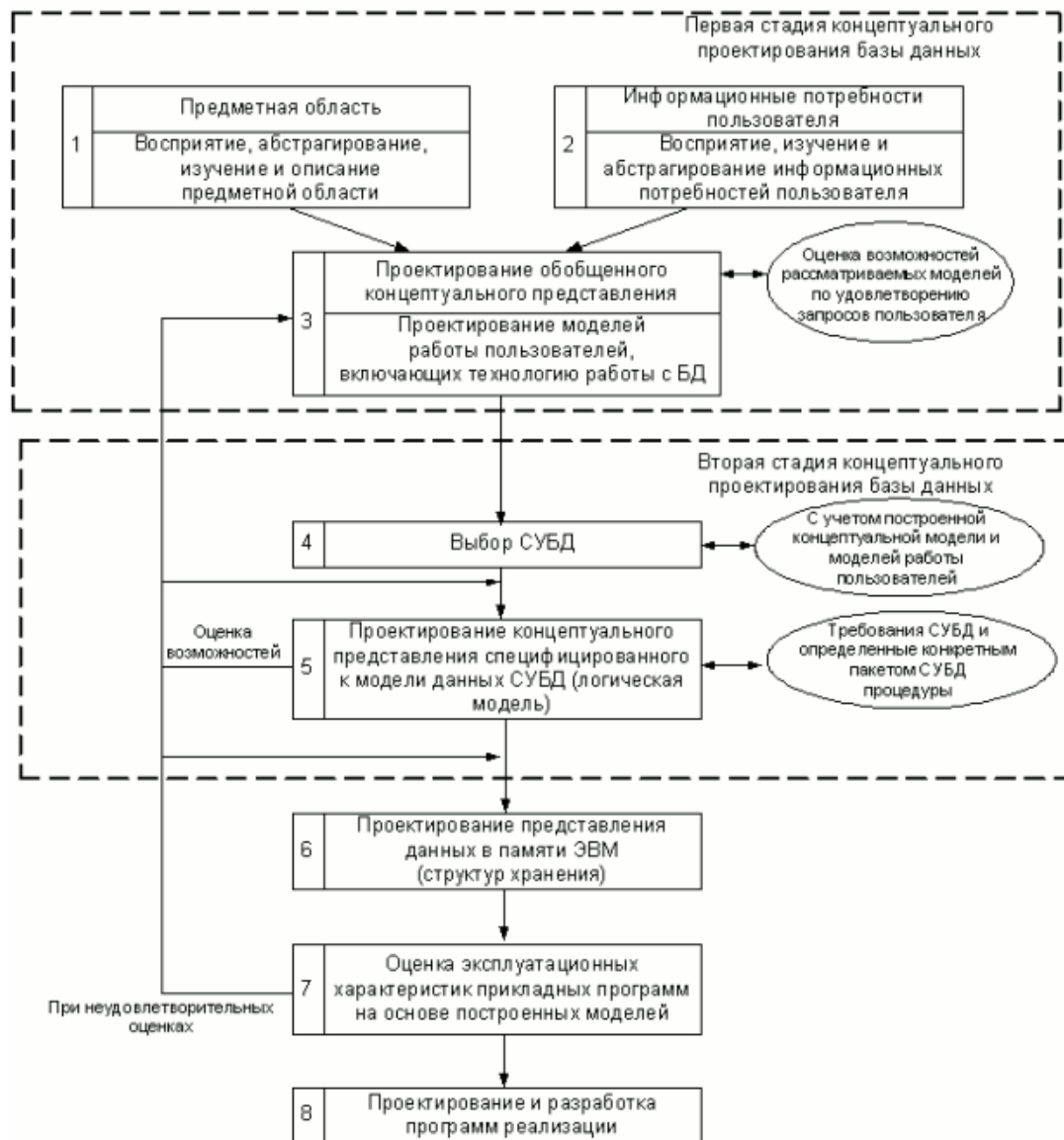


Рис. 4.4. Этапы проектирования базы данных

Подробно действия, отраженные на приведенном рисунке, будут рассмотрены в следующих лекциях. Здесь дадим лишь краткие комментарии к соответствующим блокам.

В блоках 1,2 необходимо особое внимание обратить на слово "абстрагирование". Имеется ввиду, что *проектирование базы данных* нужно вести не под конкретный документ, обрабатываемый пользователем, и не под конкретные действия пользователя с этим документом, а под обобщенный (абстрактный) образ документов и обобщенные (абстрактные) действия пользователей. Например, рассматривать документ не с конкретными числами строк и столбцов, а с абстрактными числами n и m ; вместо требуемого пользователем поиска по конкретному полю (например, фамилии) рассматривать *поиск* по любому полю и т.д. Это очень важно, так как конкретные формы документов и действия пользователей при работе с ними достаточно часто изменяются. В этом случае при *проектировании базы данных* под конкретные формы документов и конкретные действия придется перепроектировать базу данных, что связано с существенными временными и стоимостными затратами.

Очень важным является выбор *СУБД* (блок 4), от которого в значительной степени зависит работоспособность построенной *базы данных*. Проблема выбора *СУБД* уже обсуждалась в лекции 3. Заметим здесь, что выбор *СУБД* зависит от количества форм

документов, от сложности связей между данными, от объема обрабатываемых данных, от количества пользователей, работающих с БД и т.д.

Ранее отмечалось, что *отображение* логической модели базы данных в структуру хранения (*представление* данных в памяти компьютера) осуществляется системой управления базой данных. Тем не менее, во многих СУБД для повышения эффективности функционирования базы данных представляется возможность выбора ряда параметров, управляющих представлением данных в памяти компьютера. Выбор таких параметров и подразумевается в блоке 6.

Заметим, что очень важно при *проектировании* базы данных делать оценки ее возможной работоспособности. Так, по завершении проектирования обобщенного концептуального представления нужно попытаться оценить необходимое число производимых операций с элементами моделей при реализации возможных запросов пользователей. При невозможности в рамках построенной модели ответить на какой-то *запрос* пользователя или при значительном числе производимых при этом операций (что приведет к невозможности реализации соответствующего запроса в реальном масштабе времени) необходим возврат по схеме [рис. 4.4.](#) на шаг назад (построение более эффективного обобщенного концептуального представления). Аналогичные оценки необходимо делать и при завершении других этапов проектирования (блоки 5, 7). При этом возможен возврат назад на один или несколько шагов. Так, например, при проектировании логической модели (блок 5) не удастся достичь адекватного представления *концептуальной модели* средствами *модели данных СУБД*. В этом случае необходимо либо вернуться на шаг назад и выбрать другую СУБД, либо вернуться к блоку 3 и изменить вид *концептуальной модели*. Так же, если полученные при реализации блока 7 оценки эксплуатационных характеристик не отвечают требованиям пользователя, возможны пересмотры всех ранее полученных решений (блоки 7, 6, 5, 4, 3). Кроме этого, необходим возврат на проектирование обобщенного концептуального представления при изменении внешних требований пользователей, а также при выявленных ошибках проектирования.

Краткие итоги: Рассмотрены различные представления о данных в базах данных - модели обрабатываемых данных (*внешнее представление, концептуальная модель*, структура хранения). Представлено отражение этих представлений в трехуровневой архитектуре *базы данных* (*внешний уровень, концептуальный уровень, внутренний уровень*), сформулировано достоинство трехуровневой архитектуры. Описаны основные *этапы проектирования базы данных* как процесса построения вышеуказанных моделей и *жизненный цикл* проектирования *базы данных* (создание, апробация, исправление ошибок и улучшение характеристик, *опытная эксплуатация*).

Цель 5 лекции: показать, как описывается предметная область при концептуальном моделировании (с помощью каких понятий, средств представления и приемов построения) и как обеспечивается *достоверность* информации в базе данных за счет ограничений целостности концептуальной модели.

5.1. Описание информационного представления предметной области. ER-диаграмма

Иллюстрацию вводимых понятий и этапов проектирования *базы данных* будем проводить на примере близкой для читателя конкретной *предметной области*: *представление* данных о студентах вуза. Дадим краткое описание рассматриваемой *предметной области*. В вузе имеется несколько факультетов, на каждом из которых ведется подготовка по нескольким специальностям или направлениям. Для каждой специальности на факультете есть свой учебный план, в котором приводится перечень изучаемых учебных курсов с указанием количества часов занятий. Студенты изучают соответствующие дисциплины, сдают экзамены и зачеты, получают оценки.

Чаще всего концептуальная модель представляется в виде диаграммы сущностей – связей (entity – relationship) или ER-диаграммы. Процесс построения ER-диаграммы называется ER-моделированием.

Введем основные понятия, с помощью которых описывается предметная область.

Сущность (Entity) или объект – то, о чем будет накапливаться информация в информационной системе (нечто такое, за чем пользователь хотел бы наблюдать).

Если в системе обрабатывается информация о факультетах, *сущностью* будет являться факультет, если о студентах, сущность – студент и т.п.

Имя сущности при ER-моделировании, как правило, записывается заглавными буквами. Каждая *сущность* обладает определенным набором свойств (рассматриваем только свойства, представляющие интерес для пользователей в рамках проводимого исследования), которые запоминаются в информационной системе. Так, например, в качестве свойств сущности ФАКУЛЬТЕТ можно указать номер факультета, название факультета, в качестве свойств сущности СТУДЕНТ можно указать фамилию, дату рождения, *место* рождения, в качестве свойств сущности ЭКЗАМЕН – предмет, дату проведения экзамена, экзаменаторов.

Для информационного описания сущности вводится понятие атрибута.

Атрибут – поименованное свойство (характеристика) сущности. *Атрибут* представляет собой информационное *отображение* свойства сущности и принимает конкретное значение из множества допустимых значений. Так, например, для сущности ФАКУЛЬТЕТ атрибут "название" у конкретного экземпляра сущности принимает конкретное значение "вычислительной математики и кибернетики". Таким образом, атрибут представляет информационное описание количественных или качественных свойств сущности, описывает состояние сущности, позволяет идентифицировать сущность. Информация о сущности представляется совокупностью атрибутов. **Такую совокупность атрибутов часто называют записью об объекте.**

Совокупность сущностей, характеризующихся в информационной системе одним и тем же перечнем свойств, называется *классом сущностей* (набором объектов). Так, например, совокупность всех сущностей СТУДЕНТ составляет *класс сущностей* СТУДЕНТ, совокупность всех сущностей ФАКУЛЬТЕТ составляет *класс сущностей* ФАКУЛЬТЕТ. *Класс сущностей* описывается перечнем свойств сущностей, составляющих этот класс.

Экземпляр сущности будем называть конкретной сущностью (сущность с конкретными значениями соответствующих свойств). Выше мы определили сущность как то, о чем будет накапливаться информация в информационной системе. Это только одна сторона. Информация должна не просто храниться сама по себе, а использоваться для удовлетворения информационных потребностей пользователя. Для реализации подавляющего числа запросов пользователю прежде всего необходимо найти интересующий его экземпляр сущности (с целью обработки, корректировки, удаления). **Поэтому важнейшим свойством сущности является однозначная идентификация ее экземпляров по одному или группе атрибутов (уникальному идентификатору).** У сущности ФАКУЛЬТЕТ это, например, номер факультета, у сущности СТУДЕНТ это может быть атрибут "фамилия", если у всех студентов разные фамилии, группа атрибутов "фамилия", "имя", "отчество", или специально введенный уникальный идентификатор, например дополнительно введенный атрибут "код студента".

Наиболее распространенным способом представления концептуальной модели является так называемая *ER-диаграмма*. В разных источниках используются разные системы обозначений в *ER-диаграммах*. На практике использование различных способов записи *ER-диаграмм* не представляет особой сложности – беглое ознакомление с соответствующим разделом документации позволяет быстро освоить используемую систему обозначений. В данном пособии в *ER-диаграмме* класс сущностей будем представлять в виде четырехугольника. В четырехугольнике записано уникальное имя класса сущности (прописными буквами) и имена атрибутов строчными буквами.

Пример класса сущностей СТУДЕНТ и конкретного экземпляра сущности показан на [рис. 5.1](#)

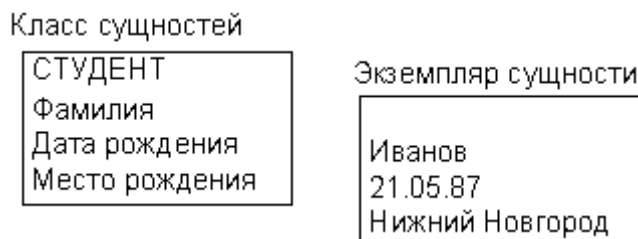


Рис. 5.1. Класс сущностей и экземпляр сущности

Для реализации информационных потребностей пользователя недостаточно найти интересующий его *экземпляр сущности*. Информационные потребности тесно связаны с функциональными взаимоотношениями, существующими в организации (например, необходимо определить, на каком факультете учиться конкретный студент). Для реализации таких запросов (информационных потребностей пользователя) используются существующие в *предметной области* взаимоотношения между сущностями. **Соответствующие взаимоотношения сущностей выражаются связями (Relationships)**. Различают классы связей и экземпляры связей. **Классы связей – это взаимоотношения между классами сущностей, а экземпляры связи – взаимоотношения между экземплярами сущностей.**

Класс связей может затрагивать несколько классов сущностей. Число классов сущностей, участвующих в связи, называется *степенью связи* $n = 2, 3, \dots$. Так, например, класс сущностей **СТУДЕНТ** связан с классом сущностей **ФАКУЛЬТЕТ** связью "учится на факультете". Степень этой связи равна двум. При $n=2$ связь называется бинарной. Заметим, что связь нужно рассматривать как двустороннюю: "студент учится на факультете" и "на факультете учатся студенты". Рассмотрим классификацию *бинарных связей*. В зависимости от того, сколько экземпляров сущности одного класса связаны со сколькими экземплярами сущности другого класса, различают следующие *типы связей*:

- **Связь 1:1.** Одиночный экземпляр сущности одного класса связан с одиночным экземпляром сущности другого класса. Примером является связь между классами сущностей **ФАКУЛЬТЕТ** и **УЧЕБНЫЙ ПЛАН ПО СПЕЦИАЛЬНОСТИ ДЛЯ ФАКУЛЬТЕТА** (каждому факультету соответствует свой учебный план по специальности или направлению).
- **Связь 1:M.** Единый экземпляр сущности одного класса связан со многими экземплярами сущности другого класса. Примером является связь между классами сущностей **ФАКУЛЬТЕТ** и **СТУДЕНТ** (на одном факультете учатся много студентов).
- **Связь M:N.** Несколько экземпляров сущности одного класса связаны с несколькими экземплярами сущности другого класса. Примером является связь между классами сущностей **ФАКУЛЬТЕТ** и **СПЕЦИАЛЬНОСТЬ** (на факультете может быть несколько специальностей и одна и та же специальность может быть на нескольких факультетах).

Числа, описывающие типы *бинарных связей* (**1:1**, **1:M**, **M:N**), обозначают максимальное количество сущностей на каждой стороне связи. Эти числа называются *максимальными кардинальными числами*, а соответствующая пара чисел называется *максимальной кардинальностью*.

В данном пособии на *ER-диаграммах* связи между сущностями будем обозначать стрелками, рядом со стрелками указываем *имя связи*, а также *тип связи*. Пример *ER-диаграммы*, представляющей сущности **СТУДЕНТ**, **ФАКУЛЬТЕТ**, **СПЕЦИАЛЬНОСТЬ** и их взаимосвязи приводится на [рис. 5.2](#).

Напомним, что каждый *экземпляр сущности* должен уникально идентифицироваться (иметь *уникальный идентификатор*). Так как могут быть несколько студентов с одинаковой фамилией, введем дополнительный *атрибут* "код студента". У сущностей **ФАКУЛЬТЕТ** и **СПЕЦИАЛЬНОСТЬ** *атрибут* "номер" является уникальным идентификатором.



Рис. 5.2. Пример фрагмента ER-диаграммы

Заметим, что по этой ER-диаграмме можно указать последовательность действий, производимых при реализации запроса пользователей. Например, для реализации запроса "на каком факультете учится студент Иванов" необходимо выполнить следующие действия: найти среди экземпляров сущности **СТУДЕНТ** экземпляр с фамилией Иванов, перейти по связи "Студент учится на факультете" к экземпляру сущности **ФАКУЛЬТЕТ**, значение атрибута "Название" этого экземпляра и есть искомое название факультета. Отметим также, что иногда на ER-диаграммах две связи между сущностями изображают одной двухсторонней стрелкой или просто линией. Заметим, что на приведенной ER-диаграмме не представлены какие-либо способы реализации этих связей (на логическом и, тем более, на физическом уровнях). Соответствующие способы реализации связей зависят от возможностей модели данных конкретной СУБД и будут рассмотрены в следующей лекции ("Вторая стадия концептуального проектирования (Модели данных СУБД. Представление концептуальной модели средствами модели данных СУБД)") на второй стадии концептуального проектирования при представлении концептуальной модели средствами модели данных СУБД.

5.2. Построение концептуальной модели в виде ER-диаграммы

5.2.1 Основные этапы построения

Как уже отмечалось, концептуальная модель представляет собой обобщение представлений разных пользователей о данных. В связи с этим построение концептуальной модели, как правило, происходит в два этапа. На первом этапе производится сбор и анализ характеристик данных и строятся так называемые модели локальных представлений (локальные модели). Чаще всего локальная модель отражает представление отдельного пользователя (отдельной функциональной задачи). Иногда такая модель может описывать и некоторую независимую область данных нескольких функциональных задач (нескольких приложений). Здесь необходимо отметить, что моделирование представлений отдельных пользователей приводит к снижению уровня интеграции данных, а моделирование совместных представлений группы пользователей – к повышению сложности проектирования. В связи с этим при выборе области данных для локального моделирования приходится выбирать компромиссное решение между вышеуказанными вариантами.

При разработке концептуальной модели, прежде всего, следует определить сущности. С этой целью нужно сделать следующее:

- необходимо понять, какая информация должна храниться и обрабатываться и можно ли это определить как сущность;
- присвоить этой сущности имя;

- выявить *атрибуты сущности* и присвоить им имя;
- определить *уникальный идентификатор сущности*.

Выявив сущности, необходимо определить, какие связи имеются между ними.

При определении связей (естественно, рассматриваем только те связи, которые имеют отношение к решаемым задачам обработки данных) необходимо учитывать следующее:

- то, как экземпляр одной сущности связан с экземпляром другой сущности;
- то, как должны быть установлены связи, чтобы была возможность ответа на все запросы пользователей (исходя из их информационных потребностей).

Далее необходимо присвоить связям имена и определить тип связей.

На втором этапе построенные локальные модели объединяются в обобщенную концептуальную модель.

5.2.2. Моделирование локальных представлений

Прежде всего, необходимо отметить, что построенная модель должна удовлетворять ряду требований:

- адекватно отражать представление пользователя о данных;
- давать возможность ответа на возможные запросы пользователя, причем делать это с минимальными затратами по количеству просматриваемых сущностей;
- представлять данные с минимальным дублированием.

Процесс построения модели, удовлетворяющей указанным требованиям, является творческим, и формализовать его, как правило, невозможно. Тем не менее можно указать некоторые способы порождения вариантов при моделировании. Выбор одного из таких вариантов на основе оценок объемов дублирования и числа просматриваемых объектов при ответах на запросы пользователей позволяет улучшить эксплуатационные характеристики проектируемой базы данных.

Вариативность моделирования обуславливается неоднозначностью выбора сущностей, атрибутов и связей. В одном варианте можно что-то взять за сущность, в другом варианте это же можно взять за атрибут (несколько атрибутов), в третьем варианте это можно определить как *связь*. Так, например, ранее мы определили сущность ФАКУЛЬТЕТ с атрибутами "номер факультета", "название факультета". Введем сущность КАФЕДРА с атрибутами "номер кафедры", "название кафедры". Между этими сущностями есть *связь* "факультет состоит из кафедр". Возможен другой вариант, в котором вышеуказанная *связь* представляется через *атрибуты сущности* (у сущности ФАКУЛЬТЕТ введем дополнительные атрибуты, представляющие номера и названия всех кафедр этого факультета).

После того как выбран рациональный вариант локальной модели, производится редактирование введенных наименований сущностей, атрибутов и связей. Здесь выполняются следующие действия:

- устраняются расплывчатые наименования (все наименования должны однозначно пониматься каждым пользователем);
- устраняются синонимы (различные наименования одного и того же понятия);
- устраняются омонимы (одно и то же наименование разных понятий).

Эти действия, вообще говоря, носят итерационный характер, т.к. после их выполнения вновь могут возникать и расплывчатые наименования, и синонимы, и омонимы.

5.2.3. Объединение локальных моделей

На этом этапе ранее построенные модели *локальных представлений* отдельных пользователей (или групп пользователей) объединяются в единую концептуальную модель. Объединение локальных моделей производится следующими путями:

- слияние идентичных элементов;
- установление связей между наборами сущностей разных моделей;
- введение новых агрегированных элементов для представления связей между элементами разных моделей;
- обобщение различных подобных *типов сущностей*, позволяющее трактовать эти сущности как одну обобщенную сущность.

Рассмотрим каждый из этих путей.

Слияние идентичных элементов

Два или более элементов модели идентичны, если они имеют одинаковое смысловое значение.

Объединение моделей с идентичными элементами осуществляется путем "слияния" этих элементов в один. Два набора сущностей СПЕЦИАЛЬНОСТЬ в модели 1 и 2 имеют одинаковое смысловое значение ([рис. 5.3.](#)), и могут быть заменены одним набором сущностей ([рис. 5.4.](#)).

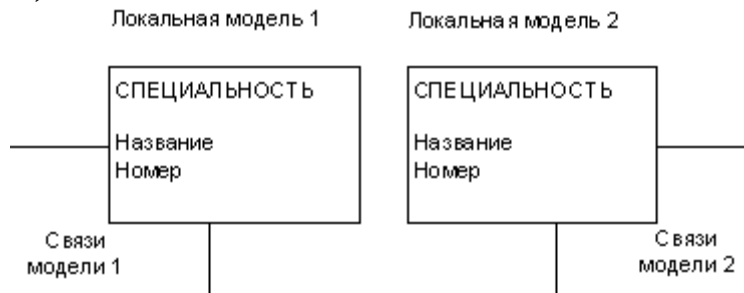


Рис. 5.3. Модели с идентичным элементом

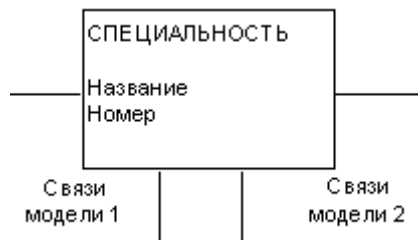


Рис. 5.4. Объединенная модель

Установление связей между наборами сущностей разных моделей

При рассмотрении наборов сущностей объединяемых моделей необходимо выявление связей между ними, т.к. именно эти связи и определяют в конечном итоге интегрированную базу данных.

Введение агрегированных элементов

При объединении моделей связь между элементами разных моделей может рассматриваться как новый элемент.

Рассмотрим в качестве примера моделирование информационного представления сдачи студентом экзаменов. Можно выделить ряд *локальных представлений* ([рис. 5.5.](#)).

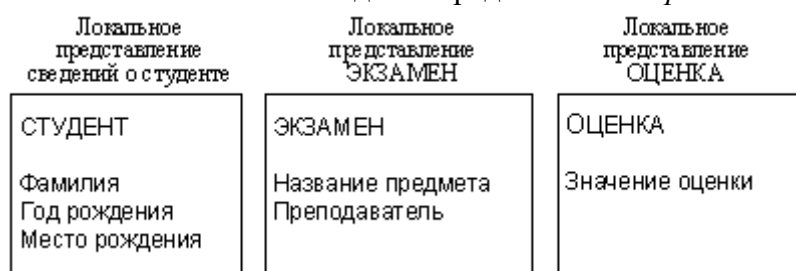


Рис. 5.5. Локальные представления

Объединяя *локальные представления*, устанавливаем новые связи ([рис. 5.6.](#)).

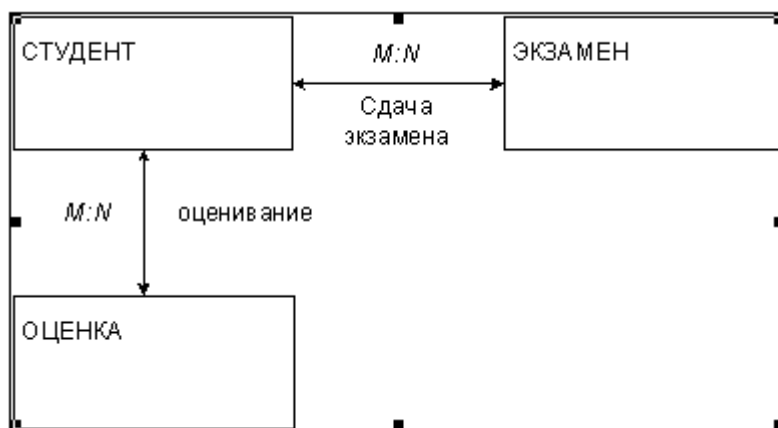


Рис. 5.6. Объединение локальных представлений

Как уже отмечалось, одним из показателей "зрелости" модели является возможность ответа на запросы пользователей, и установление связей преследует именно эту цель. Нетрудно видеть, что какие бы связи в рассматриваемой модели ни вводились, невозможно ответить на запрос "какую оценку получил студент А по дисциплине В". В таком случае необходимо использовать принцип агрегации – необходимую *связь* между элементами модели ввести как некоторый новый элемент. В данном примере можно определить этот новый агрегированный элемент как ЭКЗАМЕН СТУДЕНТА (рис. 5.7.).

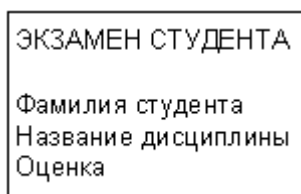


Рис. 5.7. Агрегированный элемент

Далее процесс объединения локальных моделей продолжается обычным образом. Обобщение подобных типов сущностей

Рассмотрим локальные модели разных факультетов, например – модель факультета вычислительной математики и кибернетики (ВМК), модель экономического факультета и так далее. В локальную модель факультета ВМК входят сущности СПЕЦИАЛЬНОСТИ ФАКУЛЬТЕТА ВМК и СТУДЕНТЫ ФАКУЛЬТЕТА ВМК, в локальную модель экономического факультета входят, соответственно, сущности СПЕЦИАЛЬНОСТИ ЭКОНОМИЧЕСКОГО ФАКУЛЬТЕТА и СТУДЕНТЫ ЭКОНОМИЧЕСКОГО ФАКУЛЬТЕТА (рис. 5.8.).

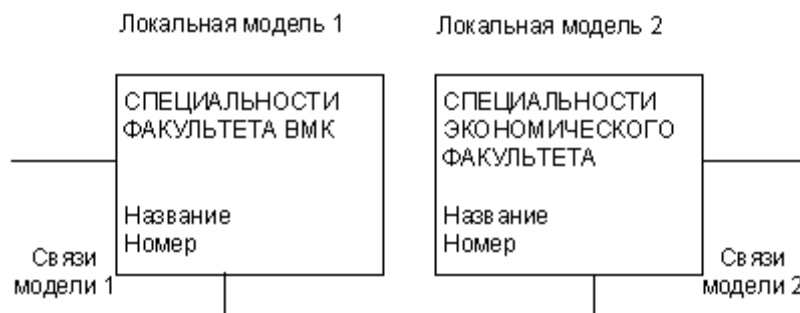


Рис. 5.8. Модели с подобным элементом

Два набора сущностей СПЕЦИАЛЬНОСТИ ФАКУЛЬТЕТА ВМК и СПЕЦИАЛЬНОСТИ ЭКОНОМИЧЕСКОГО ФАКУЛЬТЕТА в моделях 1 и 2 имеют одинаковое смысловое значение и могут быть заменены одним набором сущностей с добавлением нового атрибута – название факультета (рис. 5.9.).

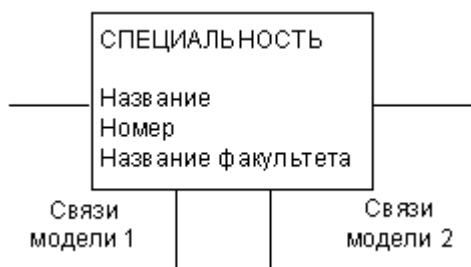


Рис. 5.9. Пример обобщенной сущности

Отметим, что в данном случае подобным образом можно слить и все остальные сущности локальных моделей факультетов, так как сущности **СТУДЕНТЫ ЭКОНОМИЧЕСКОГО ФАКУЛЬТЕТА** и **СТУДЕНТЫ ВМК** также имеют одинаковое смысловое значение и их также можно объединить. Однако в общем случае каждая локальная модель может содержать сущности и связи, которых нет в других локальных моделях.

Рассмотрим другой пример. Предположим, что мы храним данные о студентах (фамилия, имя, отчество, курс, группа) и о преподавателях (фамилия, имя, отчество, кафедра, должность). Соответственно, в предметной области выделяем две сущности – **СТУДЕНТ** и **ПРЕПОДАВАТЕЛЬ**.

Эти разные сущности можно в некоторых случаях трактовать как подобные. Для обобщения соответствующих сущностей необходимо, прежде всего, обобщить их атрибуты. Заметим, что атрибуты "Фамилия, Имя, Отчество" у обеих сущностей совпадают, атрибуты "Кафедра" и "Курс", "Группа" показывают место работы (учебы) и их можно заменить обобщенным атрибутом "Место работы (учебы)". Атрибут "Должность" можно использовать и у сущности **СТУДЕНТ**, если в качестве значения соответствующего атрибута использовать значение "студент". Тогда две сущности **ПРЕПОДАВАТЕЛЬ** и **СТУДЕНТ** можно трактовать как подобные и заменить их на обобщенную сущность. Дадим этой обобщенной сущности название **КАДРОВАЯ ЕДИНИЦА** (рис. 5.10.).

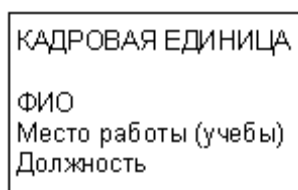


Рис. 5.10. Пример обобщенной сущности

У студента атрибут "Место работы (учебы)" будет принимать значение соответствующее атрибутам "Курс. Группа", у преподавателя – название кафедры. Обобщенная модель представлена на рис. 5.11.

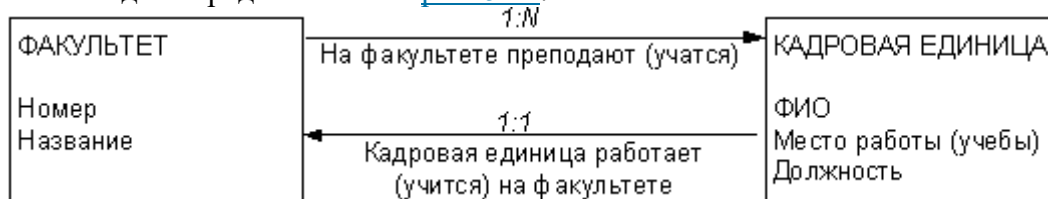


Рис. 5.11. Обобщенная модель

В этом случае почти в два раза упрощается структура концептуальной модели, и соответственно, структура базы данных. Для работы с данными о преподавателях и студентах достаточно одного набора программ. Таким образом, обобщение подобных типов объектов может существенно сократить последующие затраты на программирование.

В процессе объединения локальных представлений, как и при локальном моделировании, производится редактирование наименований (т.к. здесь появляются новые наименования). Процесс объединения также носит итерационный характер и продолжается до тех пор, пока не будут интегрированы все представления, согласованы и устранены все противоречия, отредактированы все наименования. Полученное в результате

объединения *локальных представлений* обобщенное представление и является концептуальной моделью.

5.3. Ограничения целостности

Под целостностью *базы данных* понимается то, что в ней содержится полная, непротиворечивая и адекватно отражающая предметную область (правильная) *информация*.

Огромный объем данных, вводимых в базу данных (причем разные данные могут вводиться разными пользователями), обуславливает большое число ошибок ввода (занесения). Заметим, что при традиционной "бумажной" обработке информации также достаточно часто встречаются данные, записанные неверно. Но человек, работая с определенными данными, неявно использует для контроля имеющиеся у него представления об этих данных. Например, сотрудник отдела кадров, увидев в карточке работника год рождения 1693, сразу заметит эту ошибку и предположит, что просто переставлены две цифры и реальный год рождения 1963. То есть в представлениях сотрудника заключены некоторые логические ограничения на данные. Очевидно, что для контроля правильности вводимых данных при работе с базой данных целесообразно сформировать и использовать ограничения.

Соответствующие ограничения обычно разделяют на 3 группы: внешние, специально конструируемые и внутренние. К *предметной области* относятся первые две группы, которые мы кратко охарактеризуем в этом подразделе. Внутренние ограничения относятся уже к модели данных и будут рассматриваться в разделе, посвященном модели данных.

Внешние ограничения

Эти ограничения связаны с адекватностью отражения предметной области. Например, сотрудник организации не может быть моложе 17 и старше 90 лет. Соответствующее ограничение на год рождения (GR) можно записать следующим образом:

Текущий год – 17 > GR > Текущий год – 90.

Одним из способов задания таких ограничений является перечисление конечного множества допустимых значений какого-либо атрибута (так называемый "перечислимый" тип данных). Например, должность преподавателя в вузе может принимать одно из следующих значений: профессор, доцент, старший преподаватель, преподаватель, ассистент. Вводимое значение должности для конкретного экземпляра, не совпадающее с одним из перечисленных значений, является ошибкой.

Ограничения, описанные с помощью специальных конструкций

Например, в базу данных вуза вводятся данные о числе студентов и преподавателей. По нормативным документам задано конкретное *значение отношения* числа студентов к числу преподавателей. Проверку этого отношения можно использовать для контроля достоверности данных. Такие конструкции строятся исходя из специфики данных рассматриваемой предметной области. Можно, например, построить много конструкций следующего вида: сумма значений по заданному атрибуту по всем *экземплярам сущностей* должна совпадать со значением определенного атрибута в экземпляре другой сущности.

Таким образом, на стадии ER-моделирования для повышения достоверности данных необходимо сформулировать соответствующие ограничения на данные. В идеальном случае каждое значение атрибута должно каким-то образом контролироваться. Использование этих ограничений позволяет существенно повысить достоверность данных в базе данных.

Краткие итоги. Рассмотрен процесс моделирования *предметной области*. Определены используемые при этом основные понятия (сущность, *атрибут*, *идентификатор*, *связь*, *типы связей*, *ER-диаграмма*).

Рассмотрены основные этапы моделирования сущностей и связей (*моделирование локальных представлений*, объединение локальных моделей с использованием понятий *идентичность*, *агрегация*, *обобщение*).

Дано понятие *ограничений целостности*, имеющих непосредственное *отношение к предметной области* (*внешние ограничения*; ограничения, описанные с помощью специальных конструкций).

Более подробно с материалом этой лекции можно познакомиться в [[1.1](#)], - [[5.10](#)].

Цель 6 лекции: дать общее представление о модели данных СУБД как средства для представления концептуальной модели при создании базы данных, рассмотреть типовые модели данных (*сетевая модель, иерархическая модель, реляционная модель, многомерная модель*), показать как представляется концептуальная модель в разных СУБД, рассмотреть основные принципы работы средств автоматизированного проектирования баз данных.

6.1. Представление концептуальной модели средствами модели данных СУБД

Общие представления о моделях данных СУБД

В соответствии с основными этапами проектирования базы данных после построения концептуальной модели выбирается система управления базой данных, с помощью которой будет организована база данных и работа с ней. Каждая СУБД поддерживает определенные виды и типы данных, а также средства представления связей между данными, составляющими модель данных СУБД. Вторая стадия проектирования базы данных состоит в представлении построенной на предыдущей стадии концептуальной модели средствами модели данных СУБД или в отображении концептуальной модели в модель данных СУБД. Этот этап часто называют *логическим проектированием* базы данных. Полученная при этом модель часто также называется концептуальной моделью или схемой (но специфицированной к понятиям модели данных СУБД). В некоторых источниках полученную модель называют логической структурой данных или моделью данных базы данных.

Можно по-разному характеризовать понятие модели данных СУБД. С одной стороны, модель данных СУБД – это способ структурирования данных, которые рассматриваются как некоторая абстракция в отрыве от предметной области. С другой стороны, модель данных СУБД – это инструмент представления концептуальной модели предметной области и динамики ее изменения в виде базы данных.

Учитывая обе вышеуказанные стороны, определим основные структуры моделей данных СУБД, используемые для представления концептуальной модели предметной области (сущностей, атрибутов, связей).

Элемент данных (поле) – наименьшая поименованная единица данных. Используется для представления значения атрибута.

С элементом данных неразрывно связано понятие "*тип данных*", который может принимать соответствующее поле. В разных СУБД могут использоваться разные типы данных, наиболее распространенными из которых (используемые во многих СУБД) являются следующие: числовой (*numeric*), символьный (*char*), дата (*date*) и т.д.

Запись – поименованная совокупность полей. Используется для представления совокупности атрибутов сущности (записи о сущности).

Экземпляр записи – запись с конкретными значениями полей.

Первичный ключ – минимальный набор полей записи, однозначно идентифицирующий экземпляр записи файла.

Файл – поименованная совокупность экземпляров записей одного типа. Используется для представления однородного набора сущностей.

Набор файлов – поименованная совокупность файлов, обрабатываемых в системе. Используется для представления нескольких наборов сущностей.

Введем понятие "*группа*", обобщающее понятия "*файл*" и "*запись*".

Группа – это поименованная совокупность элементов данных и других групп.

Важнейшим понятием концептуальной модели является понятие связи между сущностями (наборами сущностей). В моделях данных СУБД соответствующее понятие отражается понятием "**групповое отношение**".

Групповое отношение – поименованное бинарное отношение, заданное на двух множествах экземпляров рассматриваемых групп. По характеру бинарных связей различают групповые отношения вида 1:1, 1:M, M:1, M:N. Пары чисел называют коэффициентами группового отношения. В групповом отношении один член группы назначается владельцем отношения, другой – членом.

База данных – поименованная совокупность экземпляров групп и групповых отношений.

Для представления группового отношения используется две формы:

а) **Графовая.** Группы изображаются вершинами графа, связи между группами – дугами, направленными от группы-владельца к группе-члену с указанием имени отношения и коэффициента.

По типу графов различают:

- *иерархическую модель (граф без циклов – дерево) ;*
- *сетевую модель (ориентированный граф общего вида).*

б) **Табличная.** Связь между группами изображается таблицей, столбцы которой представляют ключи соответствующих групп. Для формального описания таблицы используется математическое (теоретико-множественное) понятие отношения. Соответствующая *модель данных* называется реляционной моделью.

Модель данных СУБД описывается следующим образом:

- определены возможные типы и характеристики логических структур данных (полей, записей, файлов);
- заданы правила составления структур более общего типа из структур более простых типов (например, записей из полей, файлов из записей и т.д.);
- определен способ представления связей (отношений) между файлами и записями с помощью дополнительных полей ;
- определены возможные действия над структурами и правила их выполнения, включающие:
 - основные элементарные операции над данными;
 - обобщенные операции (процедуры);
 - средства контроля относительно простых условий корректности операций добавления, обновления или удаления данных (ограничения), реализуемые автоматически запускаемыми при выполнении вышеуказанных операций специальными процедурами (триггерами);
 - средства контроля сколь угодно сложных условий корректности выполнения определенных действий (правила);
 - специальный класс процедур (триггеры).

В качестве основных элементарных операций обычно рассматриваются следующие: *поиск* записи с заданным значением ключа, чтение нужной записи, добавление записи, корректировка, удаление. В *моделях данных СУБД* также предусматриваются специальные *операции* для установления групповых отношений.

Обобщенные *операции* или процедуры – последовательность операций, реализующая определенный *алгоритм* обработки данных. Процедуры могут инициироваться *СУБД* автоматически, а также могут запускаться пользователем. Примерами процедур являются процедуры копирования *БД*, восстановления *БД*, процедуры, вычисляющие значения определенных атрибутов в *БД* по значениям других атрибутов, и т.п.

Средства контроля используются для реализации ограничений целостности концептуальной модели. Простейшие средства контроля – ограничения – используются для реализации как *внешних ограничений* концептуальной модели, так и внутренних ограничений модели данных. В качестве последних ограничений, в частности, реализованы ограничения на ввод данных *несоответствующего типа*, несоответствующей характеристики (*по числу битов, по числу полей, по количеству записей и т.п.*). Более сложные средства контроля (правила) позволяют вызывать выполнение определенной последовательности операций (сколь угодно сложной) при изменении или добавлении данных в *БД* и тем самым реализовывать ограничения целостности, описанные с помощью специальных конструкций.

Построение модели данных базы данных (отображение концептуальной модели в модель данных СУБД)

После первой стадии концептуального проектирования у нас сформировано обобщенное *представление* пользователей о данных, как правило, представленное в виде *ER-диаграммы*. На следующей стадии (после того, как выбрана определенная *СУБД* с конкретной моделью данных) необходимо записать концептуальную схему в терминах и понятиях выбранной *СУБД*. На этой стадии каждая сущность концептуальной модели описывается как *запись*, состоящая из полей. Каждый *атрибут* описывается как *поле* с типом и характеристиками, возможными в выбранной *СУБД*. Описываются связи концептуальной модели в понятиях, соответствующих выбранной *СУБД*, определяется порядок реализации запросов пользователей к базе данных с помощью типовых операций *СУБД* и т. д.

Результатом этой стадии проектирования будет *концептуальная модель*, специфицированная к конкретной *СУБД*.

6.2 Типовые модели данных СУБД и представление концептуальной модели

6.2.1. Сетевая модель данных

Это одна из наиболее ранних моделей данных СУБД. Типовая *сетевая модель* данных была предложена рабочей группой по базам данных (Data Base Task Group – DBTG) системного комитета CODASYL (*Conference of Data System Languages*), основными функциями которого были анализ известных фирменных систем обработки управленческих данных с единых позиций и в единой терминологии, обобщение опыта организации таких систем и разработка рекомендаций по созданию соответствующих систем. Структура данных *сетевой модели* определяется в терминах раздела 6.1 (элемент, запись, группа, групповое отношение, файл, база данных).

Реализация групповых отношений в *сетевой модели* осуществляется с использованием специально вводимых дополнительных полей - указателей (адресов связи или ссылок), которые устанавливают связь между владельцем и членом группового отношения. Запись может состоять в отношениях разных типов (1:1, 1:N, M:N). Заметим, что если один из вариантов установления связи 1:1 очевиден (в запись – владелец отношения, поля которой соответствуют *атрибутам сущности*, включается дополнительное поле – указатель на запись – член отношения), то возможность представления связей 1:N и M:N таким же образом весьма проблематична. Поэтому наиболее распространенным способом организации связей в сетевых СУБД является введение дополнительного типа записей (и соответственно, дополнительного файла), полями которых являются указатели.

Рассмотрим для примера представление группового отношения M:N. В модель вводится *дополнительная группа* (дополнительный вид записей). Элементы этой записи представляют собой указатели на две исходные группы и указатели на экземпляры рассматриваемой дополнительной записи, связывающие их в список (цепь), соответствующий M и (или) N членам группового отношения ([рис. 6.1.](#)).

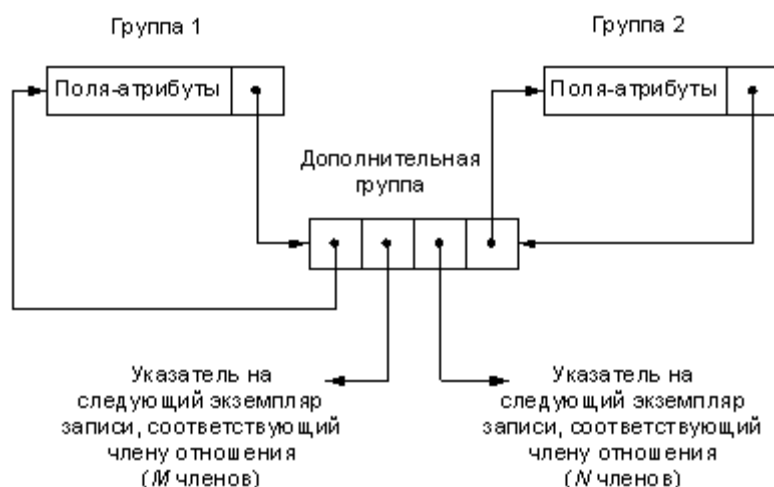


Рис. 6.1. Представление связей типа M:N

Представление связей 1:1, 1:M, N:1 является частным случаем связи типа M:N и осуществляется аналогично рассмотренному выше.

Заметим, что группа может быть членом более чем одного группового отношения. В этом случае вводится несколько *дополнительных групп-указателей*, а в группе – владельце отношений вводится несколько полей – указателей на *дополнительные группы*. Тогда множество записей (групп) и связей между ними образует некую сетевую структуру (ориентированный граф общего вида). Вершинами графа являются группы; дугами графа, направленными от владельца к члену группового отношения, – связи между группами.

Сетевая модель данных поддерживает все необходимые операции над данными, реализованные как действия со *списковыми структурами*. Сетевая модель данных является, вероятно, наиболее общей по возможностям *представления концептуальной модели*. По сути, любая ER-диаграмма без каких-либо изменений представляется средствами *сетевой модели*. К недостаткам *сетевой модели* обычно относят сложность получаемой на её основе концептуальной схемы и большую трудоемкость понимания соответствующей схемы внешним пользователем.

Рассмотрим пример записи части *ER-диаграммы* (СТУДЕНТ и ФАКУЛЬТЕТ) из предыдущей лекции в терминах сетевой СУБД. Для примера рассмотрим несколько *экземпляров сущности* СТУДЕНТ и сущности ФАКУЛЬТЕТ ([рис. 6.2.](#)).

Экземпляры сущности СТУДЕНТ Экземпляры сущности ФАКУЛЬТЕТ

121 Иванов 21.05.89 Н.Новгород	06 Мехмат
125 Петров 15.06.89 Рязань	08 ВМК
231 Сидоров 10.07.89 Чебоксары	
126 Мишин 23.02.89 Саранск	
235 Кашин 07.09.89 Н.Новгород	

Рис. 6.2. Примеры экземпляров сущностей

Пусть студенты Иванов, Петров, Мишин учатся на факультете ВМК, Сидоров и Кашин на механико-математическом факультете. Тогда *сетевая модель* соответствующего фрагмента *ER-диаграммы* будет выглядеть следующим образом ([рис. 6.3.](#)).

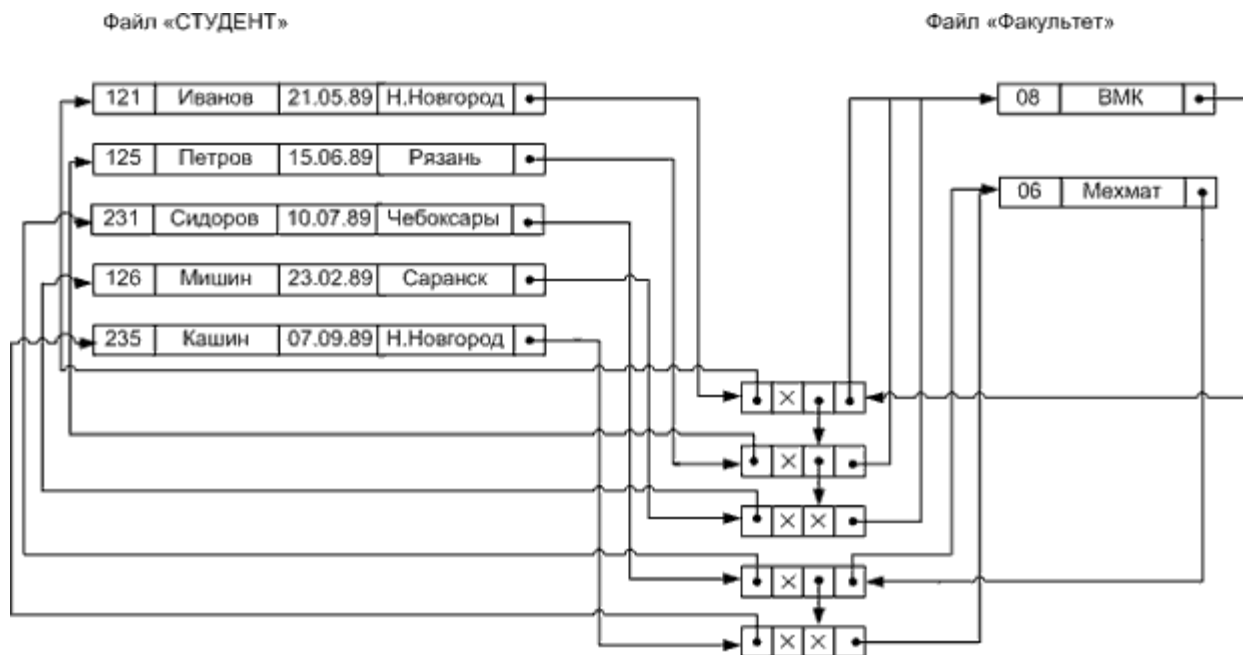


Рис. 6.3. Пример сетевой модели концептуального представления

Заметим, что в дополнительном файле один из указателей не потребовался, так как рассматриваемая связь имеет тип 1:N, а не M:N. Значок x обозначает отсутствие дальнейшей связи.

Наиболее существенным недостатком *сетевой модели* является "жесткость" получаемой концептуальной схемы. Связи закреплены в записях в виде указателей. При появлении новых аспектов использования этих же данных может возникнуть необходимость установления новых связей между ними. Это требует введения в записи новых указателей, т.е. изменения структуры БД, и, соответственно, переформирования всей базы данных.

СУБД, поддерживающие сетевую модель, широко использовались на вычислительных системах серии IBM 360/370 (ЕС ЭВМ). В качестве примеров таких систем можно указать IDMS, UNIBAD (БАНК), и их аналоги СЕДАН, СЕТОР. На персональных компьютерах сетевые СУБД не получили широкого распространения. Примером сетевой СУБД для персонального компьютера является db_VISTA III. Отметим, что система db_VISTA реализована на языке C и поэтому является переносимой. Система может эксплуатироваться на ПЭВМ типа IBM PC, SUN, Macintosh.

6.2.2. Иерархическая модель данных

Это также одна из наиболее ранних моделей данных. Реализация групповых отношений в *иерархической модели*, как и в сетевой, может осуществляться с помощью указателей и представляется в виде графа. Однако, в отличие от *сетевой модели*, здесь существует ряд принципиальных особенностей.

1. Групповые отношения являются отношениями соподчиненности. Группа (запись) – владелец отношения имеет подчиненные группы – члены отношений. Исходная группа называется предком, подчиненная – потомком.

2. Групповые отношения образуют иерархическую структуру, которую можно описать как ориентированный граф следующего вида:

- имеется единственная особая вершина (соответствующая группе), называемая корнем, в которую не заходит ни одно ребро (группа не имеет предков);
- во все остальные вершины входит только одно ребро (все остальные группы имеют одного предка), а исходит произвольное количество ребер (группы имеют произвольное количество потомков);
- отсутствуют циклы.

3. *Иерархическая модель* данных может представлять совокупность нескольких деревьев. В терминологии *иерархической модели* деревья, описывающие структуру

данных, называются деревьями описания данных, а сами структурированные данные (база данных) – деревьями данных.

Особенностью реализации операций поиска в *иерархической модели* является то, что операция всегда начинает поиск с корневой вершины и специфицирует иерархический путь (последовательность связанных вершин) от корня до вершины, экземпляры которой удовлетворяют условиям поиска.

Необходимо отметить, что программы, реализующие операции *иерархической модели*, существенно проще, чем аналогичные программы для *сетевой модели*, т.к. здесь много легче осуществлять навигацию по структуре. Целесообразность появления *иерархической модели* обусловлена, конечно, тем, что большинство организационных систем реального мира имеют иерархическую структуру (административное деление страны, организационная структура предприятия и т.п.). Соответствующее концептуальное представление также будет иметь иерархическую структуру и естественным образом может быть описано в терминах *иерархической модели*. В качестве недостатков *иерархической модели* можно назвать вышеуказанные недостатки сетевой.

СУБД, поддерживающие *иерархическую модель*, достаточно широко использовались на вычислительных системах IBM 360/370 (ЕС ЭВМ). В качестве примеров таких систем можно указать IMS, ОКА и широко тиражируемую в СССР отечественную разработку ИНЕС. Примером иерархической СУБД для *персональных ЭВМ* является отечественная система НИКА (адаптация системы ИНЕС к IBM PC).

6.2.3. Реляционная модель данных

Учитывая отмеченные в предыдущих разделах недостатки сетевых и иерархических моделей, можно сформулировать желательные требования к модели данных:

- модель должна быть понятна пользователю, не имеющему особых навыков в программировании;
- появление новых аспектов использования данных и необходимость введения новых связей не должны приводить к реструктуризации всей модели данных и базы данных в целом.

Моделью данных, удовлетворяющей вышеуказанным требованиям, является *реляционная модель*, часто называемая также **табличной**.

Основным используемым понятием здесь является понятие отношения, представляемого в виде таблицы, столбцы которой соответствуют *атрибутам сущности* (структура строки таблицы аналогична структуре записи). Каждый атрибут может принимать определенное множество значений, называемое доменом. Строка таблицы с конкретными значениями полей здесь называется кортежем (соответствует понятию "экземпляр записи"). Поля таблицы предполагаются элементарными (неделимыми). Таким образом, понятие "таблица" здесь соответствует понятию "файл" модели данных. Первичный ключ здесь – минимальный набор атрибутов, однозначно идентифицирующий кортеж в отношении.

Групповое отношение может представляться двумя способами. При первом способе в таблицы, соответствующие группам – членам отношения, добавляются столбцы ключевых полей (атрибутов) другого члена отношения (связь описывается через ключевые атрибуты). При втором способе групповое отношение определяется как *дополнительная группа* (дополнительная таблица). Столбцами этой дополнительной таблицы являются ключи групп – членов отношения. Таким образом, при любом способе соответствующая модель данных представляет собой совокупность структур таблиц.

Рассмотрим пример записи *ER-диаграммы* (см. [рис. 5.2.](#)) в терминах реляционных баз данных.

Сначала представим таблицы, соответствующие сущностям.

Таблица СТУДЕНТ

Код	Фамилия	Дата рождения	Место рождения

Таблица ФАКУЛЬТЕТ

Номер	Название

Таблица СПЕЦИАЛЬНОСТЬ

Номер	Название

Представим таблицы, описывающие связи.

Таблица "Студент учится на факультете"

Код студента	Номер факультета

Таблица "Студент учится по специальности"

Код студента	Номер специальности

Таблица "На факультете имеются специальности"

Номер факультета	Номер специальности

Заметим, что здесь реализован вышеописанный второй способ представления групповых отношений. Очевидно, что можно построить много вариантов таблиц, представляющих соответствующую ER-диаграмму.

Для приведенных таблиц не указаны домены атрибутов. Отсутствие указания доменов приводит к неоднозначной интерпретации содержания таблицы. Например, две таблицы с одинаковыми атрибутами

Код студента	Фамилия	Дата рождения	Место рождения

могут иметь разное смысловое значение и, соответственно, разное содержание (в одной таблице содержатся сведения о всех студентах факультета, в другой таблице сведения только о старостах факультета). Для устранения неоднозначной интерпретации в случае отсутствия указания доменов используют имя таблицы (СТУДЕНТ; СТАРОСТА)

Для формального описания таблицы используется теоретико-множественное понятие отношения.

Схемой отношения R называется перечень имен атрибутов отношения (соответствующих столбцам таблицы) с указанием доменов этих атрибутов и обозначается $R(A_1, A_2, \dots, A_n); \{A_i\} \subseteq D_i$, где $\{A_i\}$ – множество значений, принимаемых атрибутом $A_i (i=1, n)$.

Совокупность схем отношений, используемых для представления концептуальной модели, называется схемой реляционной базы данных, а текущие значения соответствующих отношений – реляционной базой данных.

В качестве основного недостатка реляционной модели можно указать дублирование информации при представлении связей.

Необходимо отметить, что большинство СУБД для персональных ЭВМ поддерживают именно реляционную модель данных. В качестве примеров таких наиболее распространенных СУБД можно указать все dBase-подобные системы, DB2, Paradox, Access, FoxPro, Oracle, MS SQL Server.

Более подробно реляционная модель данных будет рассмотрена в ["Формализация реляционной модели"](#).

6.2.4. Многомерная модель данных

Вернемся к понятию "сущность" концептуальной модели.

Сущность – это то, о чем накапливается информация в информационной системе. Часто оказывается, что информация об определенной сущности зависит еще от ряда параметров. Рассмотрим, например, сущность УСПЕВАЕМОСТЬ СТУДЕНТОВ со следующими атрибутами: число двоек, число троек, число четверок, число пятерок.

Значение атрибутов зависит от параметров "курс", "учебный год". Если использовать для описания соответствующей концептуальной схемы реляционную модель, то необходимо вводить множество таблиц УСПЕВАЕМОСТЬ СТУДЕНТОВ по каждому году для каждого

курса. Так, при 5 курсах и необходимости анализировать данные за 10 лет число таблиц будет равно пятидесяти. Дублируются аналогичные структуры всех таблиц, достаточно сложна обработка данных, связанная с анализом однотипных данных при изменении значения одного из параметров и т.д.

Наиболее подходящей моделью данных для этого случая является так называемая *многомерная модель*, используемая в технологии OLAP (OnLine Analytical Processing – оперативная аналитическая обработка). Отметим, что многомерность модели данных означает здесь многомерное логическое представление структуры информации и, вообще говоря, не связана с многомерностью визуализации.

Многомерные структуры представляются как гиперкубы данных. Каждая грань куба является размерностью. Основными понятиями, используемыми в многомерных моделях данных, являются "измерение" (dimension) и "ячейка" (cell).

Измерение – упорядоченный набор значений, принимаемых конкретным параметром, соответствующий одной из граней гиперкуба. Для нашего примера можно указать в качестве измерений: учебный год – 2006-2007, 2007-2008, 2008-2009; курсы – 1,2,3 и т.д.

Ячейка или показатель – это поле, соответствующее атрибуту сущности, значение которого однозначно определяется фиксированным набором значений параметров (значениями "измерений", например, 2008-2009 учебный год, первый курс).

В многомерной модели данных определяется ряд дополнительных операций, среди которых можно выделить операции "формирование среза" и "агрегация".

При формировании среза пользователю по его запросу предоставляется некоторое подмножество гиперкуба, полученное в результате фиксации пользователем одного или нескольких значений параметров. Операция "агрегация" обеспечивает переход к более общему представлению информации из гиперкуба пользователю, например суммируя значения показателей по всем значениям одного из параметров, допустим, по всем курсам.

Такая модель позволяет легко сравнивать данные при разных значениях параметров, строить графики зависимости значений конкретных атрибутов от значений определенных параметров (например, изменение атрибута по годам) и т.п. Поэтому основное назначение технологии OLAP – обработка информации для проведения анализа и принятия решения.

Массовое использование СУБД, поддерживающих многомерную модель данных, только начинается. В качестве наиболее известных СУБД такого типа можно указать Oracle Express Server.

6.3. Средства автоматизированного проектирования концептуальной модели

Средства автоматизированного проектирования концептуальной модели привлекают к себе в настоящее время большой интерес и используются в процессе создания структуры *базы данных* и интерфейса пользователя для доступа к данным.

Причина применения этих средств состоит в использовании в подавляющем большинстве реальных разработок баз данных *спиральной модели жизненного цикла* программного обеспечения, что предусматривает последовательное создание нескольких версий программного обеспечения. Каждая следующая версия включает в себя предыдущую (возможно, не полностью) и является ответом на замечания пользователя, полученные в результате тестирования предыдущей версии. При создании баз данных первая модель программного обеспечения, к сожалению, очень редко является удачной. Чаще всего заказчик отвергает первую версию, так как она недостаточно полно отвечает его требованиям. Причина такой ситуации заключается в том, что заказчик не может сразу, до создания начальной версии программы, четко и полно сформулировать свои требования. Обычно после получения первого варианта программного обеспечения заказчик выдвигает дополнительные требования, которые нельзя реализовать в рамках созданной *базы данных*. Это вынуждает разработчиков вносить изменения в структуру *базы данных*, а также, соответственно, в *интерфейс* пользователя для доступа к базе данных. Таких итераций может быть несколько до момента получения решения, адекватного запросам заказчика. Но даже после получения удовлетворительного решения процесс разработки *базы данных* не завершается. Жизнь не

стоит на месте, и запросы заказчика меняются с течением времени. Часть этих изменений можно реализовать без изменения структуры *базы данных*, изменяя только *интерфейс* пользователя, другие же требуют изменения и интерфейса, и структуры *базы данных*. Надо заметить, что подобные изменения являются очень болезненными – работа *по* их внесению может оказаться трудоемкой и, что самое неприятное, потребовать замены большого количества отлаженного программного кода. Иными словами, замененный код был написан впустую, на самом деле его не нужно было писать.

Таким образом, создание работоспособной *базы данных* можно условно разделить на три этапа – проектирование *базы данных*, в процессе которого создаются рабочие прототипы, *кодирование* – создание структур баз данных и законченного интерфейса пользователя и сопровождение готовой *базы данных*.

Основная идея применения средств *автоматизированного проектирования баз данных* заключается в том, что процесс ручного кодирования начинается только после окончания процесса проектирования. На стадии проектирования *схема базы данных* и *интерфейс* пользователя для доступа к базе данных создаются автоматически, исходя из описания концептуальной модели, с помощью так называемых CASE-средств (*Computer Aided Software/System Engineering*). Конечно, созданный таким образом *интерфейс* не является законченным программным продуктом, однако он позволяет заказчику оценить возможности конечного продукта и внести свои коррективы. Только после одобрения заказчиком рабочего прототипа разработчики приступают к ручному кодированию – созданию законченного приложения.

При сопровождении все повторяется, за тем исключением, что генерируется не все *приложение* целиком, а только часть, которую надо изменить.

На практике чаще всего CASE-средства используются для создания *схемы базы данных* в виде ER-диаграмм и генерации структур баз данных для конкретной *СУБД*. После получения от заказчика изменений разработчики вносят соответствующие исправления в диаграмму "сущность – связь" и заново генерируют *структуры баз данных*. Средства автоматической генерации интерфейсов используются реже.

В настоящее время практически каждый производитель *СУБД* предлагает собственный *программный продукт* автоматизированного проектирования. Это *Oracle Designer* (Oracle), *Power Designer* (Sybase) и другие. Демонстрационные версии данных программных продуктов можно загрузить с соответствующих сайтов (<http://www.oracle.com>, <http://www.sybase.com>).

Кроме того, на рынке представлены решения третьих фирм, не производящих *СУБД*. Одними из самых распространенных являются программные продукты фирмы AllFusion – AllFusion ERwin Data Modeler и AllFusion Process Modeler (ранее – BPwin) и другие. На российском рынке данные программы предлагает *фирма Interface Ltd.* (<http://www.interface.ru>). Создание диаграммы "сущность – связь" осуществляется с помощью AllFusion ERwin Data Modeler, дальнейшее *моделирование*, включая генерацию программного кода создания *базы данных* производится с помощью программы AllFusion Process Modeler.

Создав наглядную модель *базы данных* можно оптимизировать структуру *БД* и добиться её полного соответствия требованиям и задачам организации. Визуальное *моделирование* повышает качество создаваемой *базы данных*, продуктивность и скорость её разработки.

На сайте *Interface Ltd.* доступна для загрузки демонстрационная версия AllFusion ERwin Data Modeler, которая представляет собой полнофункциональную версию, ограниченную *по времени*.

Краткие итоги. В лекции рассмотрены вопросы, относящиеся ко второй стадии концептуального проектирования – представлению концептуальной модели в терминах модели данных определенной *СУБД*. Дано описание общего представления о модели данных (основные используемые понятия - элемент, *запись*, *файл*; основные составляющие описания). Рассмотрены *модели данных СУБД* как инструмент *представления концептуальной*

модели (сетевая модель данных, иерархическая модель данных, реляционная модель данных, многомерная модель данных и OLAP-технология). Приведены примеры записи концептуальной модели в терминах конкретной модели данных СУБД. Приводятся сведения о средствах автоматизированного проектирования концептуальной модели.

Вопросы данной лекции рассматриваются в [[1.1](#)] - [[6.8](#)].

РАЗДЕЛ 3. Введение в реляционные базы данных.

Цели занятия.

Вводятся основные понятия теории реляционных баз данных. Изучаются структура данных, ограничения целостности, внутренние ограничения целостности, семантические ограничения целостности.

4. Структура лекционного занятия.

№ п/п	Содержание (кратко)	Методы и средства обучения
1	Правила применения и принципы организации РБД. Использование формального аппарата для оптимизации схем отношений.	1. Персональные компьютеры;
2	Организация обмена между оперативной и внешней памятью. Структуры хранения данных во внешней памяти.	2. Средства доступа в Интернет; 3. Проектор.

5. Содержание лекционного занятия и взаимодействие с аудиторией.

Введение.

Формулирование темы занятия. Использование вводного примера, описание ситуации, демонстрация какого-либо процесса и т. д. (в зависимости от вида, замысла лекции и её структуры)

Тема 3.1. Формализация реляционной модели

Цель 7 лекции: рассмотреть формализованное описание реляционной модели и операций манипулирования данными как основу для использования математических методов проектирования баз данных и основу создания языков запросов к базе данных.

7.1. Формализованное описание отношений и схемы отношений

Как уже отмечалось в п. 6.2.3, реляционная модель описывает *представление* данных в виде двумерной таблицы, называемой *отношением*. Наименованиями столбцов этой таблицы служат **имена атрибутов**. Рассмотрим **формализованное описание соответствующих понятий**.

Пусть $A_1, A_2, \dots, A_n$ имена атрибутов. Каждому имени атрибута A_i соответствует допустимое множество значений, которые может принимать атрибут A_i . Это множество значений D_i называется *доменом атрибута A_i* , $i=1, n$. По определению, домены являются непустыми конечными или счетными множествами. Уточним, что в теории реляционных баз данных домен рассматривается как множество значений одного (причем простого) типа

данных. Понятию домена D_i соответствует множество значений, стоящих в столбце A_i рассматриваемой таблицы.

Схемой отношения $R \{A_1, A_2, \dots, A_n\}$ называется конечное множество имен атрибутов $\{A_1, A_2, \dots, A_n\}$, причем атрибут A_i принимает значение из множества D_i ($i=1, 2, \dots, n$), где n – арность отношения.

Понятию "схема отношения" соответствует описание структуры двумерной таблицы (имена столбцов и допустимые множества значений).

Пусть $D = D_1 \cup D_2 \cup \dots \cup D_n$.

Отношением r со схемой R называется конечное множество отображений $\{t_1, t_2, \dots, t_p\}$ из множества $R: \{A_1, A_2, \dots, A_n\}$ в множество $D: \{D_1 \cup D_2 \cup \dots \cup D_n\}$, таких, что $t_k(A_i) \in D_i, k = \overline{1, p}; i = \overline{1, n}$.

Отображение t_k называется k -м кортежем, n – размерность кортежа.

Понятию k -го кортежа соответствует множество значений, стоящих в k -й строке рассматриваемой таблицы.

Понятию отношения r соответствует множество значений, стоящих во всех строках рассматриваемой таблицы.

Ключом отношения r со схемой R называется минимальное подмножество $K = \{A_{i1}, A_{i2}, \dots, A_{im}\} \subseteq \{A_1, A_2, \dots, A_n\}$, где $\{i_1, i_2, \dots, i_m\} \subseteq \{1, 2, \dots, n\}$, такое, что любые два различных кортежа $t_1, t_2 \in r (t_1 \neq t_2)$ не совпадают по значениям множества $K = \{A_{i1}, A_{i2}, \dots, A_{im}\}$.

Возможны случаи, когда отношение r имеет несколько ключей. Такие ключи называются потенциальными (возможными). **Выбранный из них ключ для идентификации кортежей называется первичным ключом.** Таким образом, достаточно знать значение кортежа на множестве K , чтобы однозначно его идентифицировать. Ключ используется для представления связей между отношениями. С этой целью **первичный** ключ одного отношения включается в структуру (набор атрибутов) связанного с ним отношения. Для второго отношения соответствующий ключ называется **внешним ключом**.

Совокупность схем отношений, используемых для представления концептуальной модели, называется схемой реляционной базы данных (реляционной моделью данных). Текущие значения соответствующих отношений называются реляционной базой данных.

Выпишем реляционную модель данных примера из "Вторая стадия концептуального проектирования (Модели данных СУБД. Представление концептуальной модели средствами модели данных СУБД)" (см. рис. 6.3.). Введем обозначения атрибутов всех соответствующих сущностей. Пусть A_1 – код студента, A_2 – фамилия, A_3 – дата рождения, A_4 – место рождения, A_5 – номер факультета, A_6 – название факультета, A_7 – номер специальности, A_8 – название специальности. Обозначим схему отношения СТУДЕНТ как R_1 , ФАКУЛЬТЕТ как R_2 , СПЕЦИАЛЬНОСТЬ как R_3 , СТУДЕНТ УЧИТСЯ НА ФАКУЛЬТЕТЕ

как **R4**, СТУДЕНТ УЧИТСЯ *ПО* СПЕЦИАЛЬНОСТИ как **R5**, НА ФАКУЛЬТЕТЕ ИМЕЮТСЯ СПЕЦИАЛЬНОСТИ как **R6**.

Тогда реляционная модель соответствующего примера описывается следующей совокупностью схем отношений:

R1(A₁, A₂, A₃, A₄)

R2(A₅, A₆)

R3(A₇, A₈)

R4(A₁, A₅)

R5(A₁, A₇)

R6(A₅, A₇)

Напомним, что понятие "*схема отношения*" соответствует описанию структуры таблицы. Таблице с заполненными значениями (заполненными строками) соответствует понятие "*отношение*". Для данного примера отношения, соответствующие вышеуказанным схемам отношений будем обозначать

r1, r2, r3, r4, r5, r6,

Отметим следующие свойства отношения:

1. *Отношение* имеет имя, которое отличается от имен всех других отношений.
2. Каждое значение элементов кортежей представляется простым (атомарным) типом данных.
3. Каждый *атрибут* имеет уникальное имя.
4. Значения всех атрибутов являются атомарными (неделимыми). Это следует из определения *домена* как множества значений простого типа данных, т.е. среди значений *домена* не могут содержаться множества.
5. Порядок рассмотрения атрибутов в схеме отношения (отношении) не имеет значения, т.к. для ссылки на значение атрибута в *кортеже отношения* всегда используется имя атрибута.
6. Порядок рассмотрения кортежей в отношении не имеет значения, т.к. *отношение* представляет собой множество кортежей, а элементы множества, по определению теории множеств, неупорядочены.

7.2. Манипулирование данными в реляционной модели

Для манипулирования данными в реляционной модели используются два формальных аппарата:

- *реляционная алгебра*, основанная на теории множеств;
- *реляционное исчисление*, базирующееся на исчислении предикатов первого порядка.

Механизмы реляционной алгебры и реляционного исчисления эквивалентны, т.е. для любого допустимого выражения реляционной алгебры можно построить эквивалентную формулу реляционного исчисления и наоборот.

Отличаются два этих формальных аппарата уровнем процедурности. Выражения *реляционной алгебры* строятся на основе алгебраических операций (высокого

уровня), и подобно тому, как интерпретируются арифметические и логические выражения, выражение *реляционной алгебры* также имеет процедурную интерпретацию. Другими словами, *запрос*, представленный на языке *реляционной алгебры*, может быть реализован как последовательность элементарных алгебраических операций с учетом их старшинства и возможного наличия скобок.

Для формулы *реляционного исчисления* однозначная интерпретация (соответствующая однозначная последовательность действий), вообще говоря, отсутствует. Формула только устанавливает условия, которым должны удовлетворять кортежи результирующего отношения. Поэтому языки *реляционного исчисления* являются более непроецедурными или декларативными.

Операции, реализуемые с помощью указанных аппаратов, обладают важным свойством: они замкнуты на множестве отношений. Это означает, что выражения *реляционной алгебры* и формулы *реляционного исчисления* определяются над отношениями *реляционных БД* и результатом вычисления также являются отношения. В результате любое *выражение* или формула могут интерпретироваться как *отношение*, что позволяет использовать их в других выражениях или формулах.

Как мы увидим, *алгебра* и *исчисление* обладают большой выразительной мощностью, очень сложные запросы к базе данных могут быть выражены с помощью одного выражения *реляционной алгебры* или одной формулы *реляционного исчисления*. Именно по этой причине такие механизмы включены в *реляционную модель данных*. Конкретный язык манипулирования *реляционными БД* называется *реляционно полным*, если любой запрос, выражаемый с помощью одной операции *реляционной алгебры* или одной формулы *реляционного исчисления*, может быть выражен с помощью одного оператора этого языка.

Заметим, что крайне редко *алгебра* или *исчисление* принимаются в качестве полной основы какого-либо языка *БД*. Обычно (как, например, в случае языка *SQL*) язык основывается на некоторой смеси алгебраических и логических конструкций. Тем не менее знание алгебраических и логических основ языков баз данных часто бывает полезно на практике.

7.3. Операции реляционной алгебры

Операции реляционной алгебры определены на множестве отношений и являются замкнутыми относительно этого множества (образуют алгебру). Оказывается, что любой произвольный *запрос* к *БД* можно представить в виде последовательности, составленной из пяти основных *операций реляционной алгебры*. Рассмотрим эти *операции*.

Объединение $r \cup s$

Объединением отношений r и s называется множество кортежей, которые принадлежат или r , или s , или им обоим. Для операции объединения требуется одинаковая *арность* отношений.

Для примера, пусть

r			s		
a	b	a	b	g	a

d	a	f	d	a	f
c	b	d			

тогда

$r \cup s$		
a	b	a
d	a	f
c	b	d
b	g	a

Заметим, что с помощью операции объединения может быть реализовано добавление нового кортежа к имеющемуся отношению. В этом случае r – исходное отношение, s – отношение, содержащее один добавляемый кортеж.

Разность $r - s$

Разностью отношений r и s называется множество кортежей, принадлежащих r , но не принадлежащих s . Для этой операции также требуется одинаковая арность отношений.		
$r - s$		
a	b	a
c	b	d

Заметим, что с помощью операции разности может быть реализовано удаление кортежа из имеющегося отношения. В этом случае r – исходное отношение, s – отношение, содержащее один удаляемый кортеж.

Декартово произведение $r \times s$

Пусть r и s – отношения арности k_1 и k_2 соответственно. Декартовым произведением $r \times s$ называется множество кортежей длины $k_1 + k_2$, первые k_1 компонентов которых образуют кортежи, принадлежащие r , а последние k_2 – кортежи, принадлежащие s .					
$r \times s$					
a	b	a	b	g	a
a	b	a	d	a	f
d	a	f	b	g	a
d	a	f	d	a	f
c	b	d	b	g	a

c	b	d	d	a	f

Проекция $\pi_{A_{i1}, A_{i2}, \dots, A_{im}}(r)$

Проекция $\pi_{A_{i1}, A_{i2}, \dots, A_{im}}(r)$ есть множество кортежей, получаемых из кортежей отношения r выбором столбцов с именами $A_{i1}, A_{i2}, \dots, A_{im}$.

Другими словами, это операция построения "вертикального" подмножества, получаемого путем выбора определенных атрибутов и исключения остальных. Повторяющиеся кортежи исключаются.	
$\pi_{1,3}(r)$	
a	a
d	f
c	d

Выбор (селекция) $\sigma_F(r)$

Пусть F – формула, образованная: операндами, являющимися константами или именами атрибутов, арифметическими операторами сравнения, логическими операторами (и, или, не), тогда выбором (селекцией) σ_F называется множество кортежей, компоненты которого удовлетворяют условию, заданному формулой F .

$\sigma_{(1)=(3)}($

Здесь $F:(1)=(3)$ – содержимое первого столбца равно содержимому третьего столбца.

Приведем ряд примеров представления запросов с помощью формальных операций для реляционной модели (СТУДЕНТ, ФАКУЛЬТЕТ, СПЕЦИАЛЬНОСТЬ), рассмотренной выше.

Пример 1.

Сформировать список студентов (фамилия).

Рассмотрим схему отношения СТУДЕНТ.

Атрибут "Фамилия" обозначен здесь A_2 . Для ответа на запрос необходимо взять проекцию отношения r_1 на столбец A_2 .

$\pi_{A_2}(r_1)$

Пример 2.

Выдать список фамилий и дат рождений студентов, которым на текущую дату (date) больше 35 лет.

Рассмотрим то же *отношение* **r1**. Сначала выбираем студентов, которым больше 35 лет:

$$\sigma_{(A3)+35 < date}(r1)$$

Затем берем проекцию полученного отношения на столбцы

$$\pi_{A1, A3}(\sigma_{(A3)+35 < date}(r1))$$

Заметим, что можно было бы выполнить эти две *операции* в другой последовательности – сначала *проекция*, а затем *селекция*. Предлагается оценить, какой из этих вариантов лучше по оценке числа выполняемых элементарных действий и объему требуемой памяти.

Пример 3.

Выдать *список* фамилий студентов, обучающихся по специальности "Информационные технологии". Название специальности является атрибутом отношения **r3**. Если бы в этом отношении присутствовал *атрибут* "фамилия", то задача решалась бы аналогично примеру 2. В отношении **r5** присутствует *атрибут* "код студента", а "фамилия" присутствует в отношении **r1**. Для ответа на этот *запрос* необходимо связывать по "код студента" *отношение* **r3** и *отношение* **r1**.

Сначала выберем из отношения **r3** кортежи с названием специальности "Информационные технологии". Обозначим полученное *отношение* **rp1**. (Дальнейшие промежуточные отношения будем обозначать последовательно **rp1**, **rp2**, **rp3** и т.д.).

$$rp1 = \sigma_{(A8) = \text{"Информационные технологии"}}(r3)$$

Далее нас будет интересовать только *атрибут* **A1** – "код студента". Поэтому возьмем проекцию на эти столбцы.

$$rp2 = \pi_{A1}(rp1).$$

Далее необходимо связать отношения **r1** и **rp2** (склеить таблицы). Для склейки таблиц используется операция "*декартово произведение*":

$$rp3 = r1 \times rp2$$

В отношении **r3** присутствуют два одинаковых столбца: **A1** из отношения **r1** и **A1** из отношения **rp2**. Выбирая из отношения **rp3** строки, в которых значения в соответствующих столбцах совпадают, получим сведения о студентах, обучающихся по специальности "Информационные технологии"

$$rp4 = \sigma_{(A1 * r1) = (A1 * rp2)}(rp3),$$

где **A1 * r1** и **A1 * rp2** обозначают соответственно столбец **A1** соответствующей первой и второй составной части *декартова произведения*. Теперь осталось только выбрать фамилии соответствующих студентов

$$rp5 = \pi_{A1}(rp4)$$

Получаем требуемый результат. Заметим, что для экономии действий и памяти, перед тем как склеивать таблицы, целесообразно было сделать *операцию проекции отношения r1* на столбцы **A₁**, **A₂**. (чтобы не включать в *декартово произведение* лишние столбцы).

Введенные пять основных *операций реляционной алгебры* позволяют реализовать любой *запрос* к реляционной базе данных. Однако наряду с основными операциями достаточно часто удобно использовать так называемые дополнительные *операции реляционной алгебры* (которые могут быть выражены через основные).

Пересечение $r \cap s$

*Пересечением отношений **r** и **s** называется множество кортежей, принадлежащих как **r**, так и **s**. Пересечение может быть выражено через операции разности*

$$r \cap s = r - (r - s).$$

$$r \bowtie_{i \theta j} s$$

θ -соединение **r** и **s** по столбцам **A_i** и **A_j** представляет собой множество таких кортежей в *декартовом произведении r* и **s**, что **i**-й компонент **r** находится в отношении θ с **j**-м компонентом **s**, где θ – *арифметический оператор* сравнения. Если θ является оператором равенства, то эта операция называется *эквисоединением*

$$r \bowtie_{i \theta j} s = \sigma_{i \theta (l+j)}(r \times s),$$

где **l** – *арность* отношения **r**.

Пример.			s		
r					
1	2	3		3	1
4	5	6		6	2
7	8	9			
$r \bowtie_{(2) < (1)} s$					
1	2	3	3		
1	2	3	6		
4	5	6	6		

Заметим, что в примере 3 две последовательно идущие *операции* (*декартово произведение* и *селекция*) вместе как раз представляют операцию соединения. Причем

использование *декартова произведения* для соединения таблиц обязательно обуславливает использование селекции как следующей *операции* для установления связи между таблицами. Поэтому целесообразно использовать такую объединенную операцию и программно реализовывать в *СУБД* именно операцию соединения.

Естественное соединение $r \bowtie s$

Операция применима тогда и только тогда, когда столбцы имеют имена (являются атрибутами). Операция применима к отношениям, у которых есть одинаковые атрибуты.

Пусть

$r = (A_1, \dots, A_k, B_1, \dots, B_n), s = (A_1, \dots, A_k, C_1, \dots, C_m)$,

имена $A_1, \dots, A_k$ совпадают.

Тогда $r \bowtie s$ определяется следующим образом

$$r \bowtie s = \pi_{B_1, \dots, B_n, A_1, \dots, A_k, C_1, \dots, C_m} (\sigma_{r.A_1=s.A_1 \& r.A_2=s.A_2 \& \dots \& r.A_k=s.A_k} (r \times s)).$$

Для подчеркивания важности приведенных *операций реляционной алгебры*, а также для уточнения понятия реляционной *СУБД* приведем следующее *определение* одного из ведущих специалистов в области реляционных баз данных К.Дж. Дейта: " Будем называть систему реляционной, если она поддерживает, по крайней мере, реляционные базы данных, т.е. базы данных, которые могут восприниматься пользователем как таблицы и только как таблицы, операции селекции, проекции и соединения реляционной алгебры, не требуя при этом, чтобы каким-то образом были предопределены физические пути доступа для поддержки этих операций ".

Краткие итоги: В лекции рассматриваются вопросы, связанные с формализацией наиболее распространенной в настоящее время *модели данных СУБД* – реляционной модели. Формальное описание реляционной модели и полученные на этой основе математические методы и алгоритмы позволяют формализовать ряд шагов проектирования реляционной *базы данных*, получить оптимальную (по определенным критериям) структуру *базы данных* и эффективные алгоритмы обработки. Здесь рассматривается формализованное описание отношений, формальные средства манипулирования данными в реляционной модели (дано понятие *реляционного исчисления* и *реляционной алгебры*, приводятся основные *операции реляционной алгебры*). Приводятся примеры представления запросов как последовательность формальных *операций реляционной алгебры*.

Цель 8 лекции: показать возможность эффективного использования формальных методов построения оптимальной (по определенным показателям) структуры реляционной *базы данных* путем нормализации схем отношений.

8.1. Проблема выбора рациональных схем отношений

При представлении концептуальной схемы в виде реляционной модели возможны различные варианты выбора схем отношений. Одни варианты выбора рассматривались в предыдущих разделах (п. 6.2.3), другие получаются объединением (или разбиением) некоторых схем отношений. От правильного выбора схем отношений, представляющих концептуальную схему, в значительной степени будет зависеть эффективность функционирования *базы данных*.

Рассмотрим для примера конкретную схему отношений и проанализируем её недостатки. Предположим, что данные о студентах, факультетах, специальностях, включены в таблицу со следующей схемой отношения: СТУДЕНТ (Код студента, Фамилия, Название факультета, Название специальности).

Эта схема отношений обуславливает следующие недостатки соответствующей базы данных:

- Дублирование информации (избыточность). У студентов, обучающихся на одном факультете, будет повторяться название факультета. Для разных факультетов будут повторяться специальности.
- Потенциальная противоречивость (аномалии обновления). Если, например, изменится название специальности, то изменяя её в одном кортеже (у одного студента), необходимо изменять и во всех других кортежах, где она присутствует.
- Потенциальная возможность потери сведений (аномалии удаления). При удалении информации о всех студентах, поступающих на определенную специальность, мы теряем все сведения об этой специальности.
- Потенциальная возможность не включения информации в базу данных (аномалии включения). В базе данных будут отсутствовать сведения о специальности, если на ней нет обучающихся студентов.

В теории реляционных баз данных существуют формальные методы построения реляционной модели базы данных, в которой отсутствует избыточность и аномалии обновления, удаления и включения.

Нормализация. Первая нормальная форма.

Построение рационального варианта схем отношений (обладающего лучшими свойствами при операциях включения, модификации и удаления данных, чем все остальные наборы схем) осуществляется путем так называемой нормализации схем отношений. Нормализация производится в несколько этапов. На начальном этапе схема отношений должна находиться в первой нормальной форме (1НФ).

Отношение находится в первой нормальной форме, если все атрибуты отношения принимают простые значения (атомарные или неделимые), не являющиеся множеством или кортежем из более элементарных составляющих.

Рассмотрим следующий пример.

Таблица представляет сущность ЭКЗАМЕНАЦИОННАЯ ВЕДОМОСТЬ

Код студента	Фамилия	Код экзамена	Предмет и дата	Оценка
1	Сергеев	1	Математика 5.06.08	4
2	Иванов	1	Математика 5.06.08	5
1	Сергеев	2	Физика 9.06.08	5
2	Иванов	2	Физика 9.06.08	5

Теперь на пересечении любой строки и любого столбца находится одно значение и, следовательно, данная таблица находится в первой нормальной форме.

Далее отношение, представленное в первой нормальной форме, последовательно преобразуется во вторую и третью нормальные формы. Процесс построения второй и третьей нормальных форм будет описан в следующих подразделах. При некоторых предположениях о данных третья нормальная форма является искомым наилучшим вариантом.

Если эти предположения не выполняются, то процесс нормализации продолжается и отношение преобразуется в четвертую и пятую нормальные формы. Построение соответствующих форм описано в литературе и в данной книге не рассматривается.

Прежде чем перейти к построению второй нормальной формы, необходимо определить ряд формальных понятий.

8.2. Функциональные зависимости (зависимости между атрибутами отношения)

Пусть $R(A_1, A_2, \dots, A_n)$ – схема отношения, а X и Y – подмножества $\{A_1, A_2, \dots, A_n\}$.

Функциональная зависимость на отношении R – это утверждение вида "Если два кортежа R совпадают по атрибутам множества $X \subset \{A_1, A_2, \dots, A_n\}$ (т.е. эти кортежи имеют в соответствующих друг другу компонентах одни и те же значения для каждого атрибута множества X), то они должны совпадать и по атрибутам множества $Y \subset \{A_1, A_2, \dots, A_n\}$ ". Формально эта зависимость записывается выражением $X \rightarrow Y$, причем говорится, что X функционально определяет Y . Часто используется другое утверждение: X функционально определяет Y или Y функционально зависит от X (обозначается $X \rightarrow Y$) тогда и только тогда, когда каждое значение множества X отношения R связано с одним значением множества Y отношения R . Иначе говоря, если два кортежа R совпадают по значению X , они совпадают и по значению Y .

Замечание. Вообще говоря, под термином "отношение" могут подразумеваться два понятия:

- отношение как переменная, которая может принимать разные значения (таблица, в строки и столбцы которой могут быть вписаны разные значения);
- отношение, как набор конкретных значений (таблица с заполненными элементами).

Функциональные зависимости характеризуют все отношения, которые могут быть значениями схемы отношения R в принципе. Поэтому единственный способ определить функциональные зависимости – внимательно проанализировать семантику (смысл) атрибутов.

Функциональные зависимости являются, в частности, ограничениями целостности, поэтому целесообразно проверять их при каждом обновлении базы данных.

Пример функциональных зависимостей для отношения ЭКЗАМЕНАЦИОННАЯ ВЕДОМОСТЬ

Код студента $\rightarrow$ Фамилия

Код студента, Код экзамена $\rightarrow$ Оценка

Пример функциональных зависимостей для отношения СТУДЕНТ, приведенного в начале настоящей лекции

Код студента $\rightarrow$ Фамилия,

Код студента $\rightarrow$ Факультет

Заметим, что последняя зависимость существует при условии, что один студент не может обучаться на нескольких факультетах.

Полное множество функциональных зависимостей

Для каждого отношения существует вполне определенное множество функциональных зависимостей между атрибутами данного отношения. Причем из одной или более функциональных зависимостей, присущих рассматриваемому отношению, можно вывести другие функциональные зависимости, также присущие этому отношению.

Заданное множество функциональных зависимостей для отношения R обозначим F , полное множество функциональных зависимостей, которые логически можно получить из F , называется замыканием F и обозначается F^+ .

Если множество функциональных зависимостей совпадает с замыканием данного множества, то такое множество функциональных зависимостей называется полным.

Введенные понятия позволяют формально определить понятие ключа.

Пусть существует некоторая схема R с атрибутами $A_1 A_2 \dots A_n$, F – некоторое множество функциональных зависимостей и X – некоторое подмножество R . Тогда X называется ключом, если, во-первых, в F^+ существует зависимость $X \rightarrow A_1 A_2 \dots A_n$ и, во-вторых, ни для какого подмножества Y , входящего в X , зависимость $Y \rightarrow A_1 A_2 \dots A_n$ не принадлежит F^+ .

Полной функциональной зависимостью называется зависимость неключевого атрибута от всего составного ключа.

Частичной функциональной зависимостью будем называть зависимость неключевого атрибута от части составного ключа.

Для вычисления замыкания множества функциональных зависимостей используются следующие **правила вывода** (аксиомы Армстронга):

Пусть известна некоторая схема отношения $R\{A_1, A_2, \dots, A_n\}$ с множеством атрибутов $U=\{A_1, A_2, \dots, A_n\}$ и множество функциональных зависимостей F , заданных на множестве U .

Аксиома рефлексивности. Если Y входит в X , а X входит в U ($Y \subseteq X \subseteq U$), то $X \rightarrow Y$ логически следует из F . Это правило дает тривиальные зависимости, так как в них правая часть содержится в левой части.

Аксиома пополнения. Если $X \rightarrow Y$ и Z есть подмножество U , то $XZ \rightarrow YZ$. В данном случае функциональная зависимость $X \rightarrow Y$ либо содержалась в исходном множестве F , либо может быть выведена из F с использованием описываемых аксиом.

Аксиома транзитивности. Если $X \rightarrow Y$ и $Y \rightarrow Z$, то $X \rightarrow Z$.

Справедлива следующая **теорема**. Аксиомы Армстронга являются полными и надежными.

Это значит, что используя их мы выведем все возможные функциональные зависимости, логически следующие из F , и не выведем никаких лишних зависимостей.

Существует несколько других правил вывода, которые следуют из аксиом Армстронга.

Правило самоопределения. $X \rightarrow X$.

Правило объединения. Если $X \rightarrow Y$ и $X \rightarrow Z$, то $X \rightarrow Y \cup Z$.

Правило псевдотранзитивности. Если $X \rightarrow Y$ и $W \cup Y \rightarrow Z$, то $X \cup W \rightarrow Z$.

Правило композиции. Если $X \rightarrow Y$ и $Z \rightarrow W$, то $X \cup Z \rightarrow Y \cup W$.

Правило декомпозиции. Если $X \rightarrow Y$ и Z входит в Y , то $X \rightarrow Z$.

Надо отметить, что вычисление замыкания множества функциональных зависимостей является трудоемкой задачей при достаточно большом количестве атрибутов (за счет выписывания большого количества тривиальных зависимостей).

8.3. Декомпозиция схемы отношения

Последовательный переход от одной нормальной формы к другой при нормализации схем отношений реализуется через декомпозицию. Основной операцией, с помощью которой осуществляется декомпозиция, является проекция.

Декомпозицией схемы отношения $R = \{A_1, A_2, \dots, A_n\}$ называется замена ее совокупностью подмножеств R , таких, что их объединение дает R . При этом допускается, чтобы подмножества были пересекающимися.

Алгоритм декомпозиции основан на следующей теореме.

Теорема о декомпозиции. Пусть $R(A, B, C)$ – отношение, A, B, C – атрибуты.

Если R удовлетворяет зависимости $A \rightarrow B$, то R равно соединению его проекций A, B и A, C

$$R(A, B, C) = R(A, B), R(A, C)$$

При нормализации необходимо выбирать такие декомпозиции, которые обладают свойством соединения без потерь. В этом случае, декомпозиция должна обеспечить то, что запросы (выборка данных по условию) к исходному отношению и отношениям, получаемым в результате декомпозиции, дадут одинаковый результат. Соответствующее условие будет выполняться, если каждый кортеж отношения R может быть представлен как естественное соединение его проекций на каждое из подмножеств. Для проверки, обладает ли декомпозиция данным свойством, используются специальные алгоритмы, описанные в литературе (в данной книге не рассматриваются).

Вторым важнейшим желательным свойством декомпозиции является свойство сохранения функциональных зависимостей. Стремление к тому, чтобы декомпозиция сохраняла зависимости, естественно. Функциональные

зависимости являются некоторыми ограничениями на данные. Если декомпозиция не обладает этим свойством, то для того чтобы проверить, не нарушаются ли при вводе данных условия целостности (*функциональные зависимости*), нам приходится соединять все проекции.

Таким образом, для правильно построенного проекта *базы данных* необходимо, чтобы декомпозиции обладали свойством соединения без потерь, и желательно, чтобы они обладали свойством сохранения функциональных зависимостей.

8.4. Выбор рационального набора схем отношений путем нормализации

Вторая нормальная форма (2НФ)

Отношение находится в 2НФ, если оно находится в 1НФ и каждый неключевой атрибут зависит от всего первичного ключа (не зависит от части ключа).

Для перевода отношения в 2НФ необходимо, используя *операцию проекции*, разложить его на несколько отношений следующим образом:

1. построить проекцию без атрибутов, находящихся в *частичной функциональной зависимости* от первичного ключа;
2. построить проекции на части *составного ключа* и атрибуты, зависящие от этих частей.

Третья нормальная форма (3НФ)

Отношение находится в 3НФ, если оно находится в 2НФ и каждый ключевой атрибут нетранзитивно зависит от первичного ключа.

Отношение находится в 3НФ в том и только том случае, если все неключевые атрибуты отношения взаимно независимы и полностью зависят от первичного ключа.

Оказывается, что любая схема отношений может быть приведена к 3НФ декомпозицией, обладающей свойствами соединения без потерь и сохраняющей зависимости.

Мотивировка третьей нормальной формы

Третья нормальная форма исключает избыточность и аномалии включения и удаления.

К сожалению, 3НФ не предотвращает все возможные аномалии.

Нормальная форма Бойса-Кодда (НФБК)

*Если в **R** для каждой зависимости $X \rightarrow A$, где **A** не принадлежит **X**, **X** включает в себя некоторый ключ, то говорят, что данное отношение находится в нормальной форме Бойса-Кодда.*

Детерминантом функциональной зависимости называется минимальная группа атрибутов, от которой зависит некоторый другой атрибут или группа атрибутов, причем эта зависимость нетривиальная.

Отношение находится в НФБК тогда и только тогда, когда каждый его детерминант является потенциальным ключом.

НФБК является более строгой версией 3НФ. Иными словами, любое отношение, находящееся в НФБК, находится в 3НФ. Обратное неверно.

Мотивировка нормальной формы Бойса-Кодда

В нормальной форме Бойса-Кодда не существует избыточности и аномалий включения, удаления и модификации. Оказывается, что любая схема отношения может быть приведена в нормальную форму Бойса-Кодда таким образом, чтобы декомпозиция обладала свойством соединения без потерь. Однако схема отношения может быть неприводимой в НФБК с сохранением зависимостей. В этом случае приходится довольствоваться третьей нормальной формой.

8.5. Пример нормализации до 3НФ

Для улучшения структуры реляционной *базы данных* (устранения возможных аномалий) необходимо привести все таблицы *базы данных* к третьей нормальной форме или в более высокой форме (если это возможно). Таким образом, задача сводится к проверке *нормализации* всех сущностей, отображающихся в таблицы *базы данных*. Если *таблица*, получающаяся из некоторой сущности, не является таблицей в

третьей нормальной форме, то она должна быть заменена на несколько таблиц, находящихся в третьей нормальной форме.

Продолжим рассмотрение примера с отношением ЭКЗАМЕНАЦИОННАЯ ВЕДОМОСТЬ

В начале этой лекции мы привели отношение к первой нормальной форме.

Код студента	Фамилия	Код экзамена	Предмет	Дата	Оценка
1	Сергеев	1	Математика	5.08.03	4
2	Иванов	1	Математика	5.08.03	5
1	Сергеев	2	Физика	9.08.03	5
2	Иванов	2	Физика	9.08.03	5

Ключом данного отношения будет совокупность атрибутов – Код студента и Код экзамена.

Для более краткой записи процесса нормализации введем следующие обозначения:

КС – код студента, КЭ – код экзамена, Ф – фамилия, П – предмет, Д – дата, О – оценка.

Выпишем функциональные зависимости

КС, КЭ $\rightarrow$ Ф, П, Д, О

КС, КЭ $\rightarrow$ Ф

КС, КЭ $\rightarrow$ П

КС, КЭ $\rightarrow$ Д

КС, КЭ $\rightarrow$ О

КЭ $\rightarrow$ П

КЭ $\rightarrow$ Д

КС $\rightarrow$ Ф

В соответствии с определением, отношение находится во второй нормальной форме (2НФ), если оно находясь в 1НФ и каждый неключевой атрибут зависит от первичного ключа и не зависит от части ключа. Здесь атрибуты П, Д, Ф зависят от части ключа. Чтобы избавиться от этих зависимостей необходимо произвести декомпозицию отношения. Для этого используем теорему о декомпозиции.

Имеем отношение R(КС, Ф, КЭ, П, Д, О). Возьмем зависимость КС $\rightarrow$ Ф в соответствии с формулировкой теоремы исходное отношение равно соединению его проекций R1(КС, Ф) и R2(КС, КЭ, П, Д, О).

В отношении R1(КС, Ф) существует функциональная зависимость КС $\rightarrow$ Ф, ключ КС – составной, не ключевой атрибут Ф не зависит от части ключа. Это отношение находится в 2НФ. Так как в этом отношении нет транзитивных зависимостей, отношение R(КС, Ф) находится в 3НФ, что и требовалось.

Рассмотрим отношение R2(КС, КЭ, П, Д, О) с составным ключом КС, КЭ. Здесь есть зависимость КЭ $\rightarrow$ П, КЭ $\rightarrow$ Д, КЭ $\rightarrow$ П, Д. Атрибуты П, Д зависят от части ключа, следовательно отношение не находится в 2НФ. В соответствии с теоремой о декомпозиции исходное отношение (используем функциональную зависимость КЭ $\rightarrow$ П, Д) равно соединению проекций R3(КЭ, П, Д), R4(КС, КЭ, О). В отношении R3(КЭ, П, Д) существуют функциональные зависимости КЭ $\rightarrow$ П, КЭ $\rightarrow$ Д, КЭ $\rightarrow$ П, Д. Зависимости неключевых атрибутов от части ключа нет, следовательно отношение находится в 2НФ. Транзитивных зависимостей в этом отношении так же нет, следовательно отношение находится в 3НФ.

Таким образом, исходное отношение приведено в к трем отношениям, каждое из которых находится в третьей нормальной форме R1(КС, Ф), R3(КЭ, П, Д), R4(КС, КЭ, О).

Заметим, что в отношении R4 атрибуты КС, КЭ являются внешними ключами, используемыми для установления связей с другими отношениями. Представим полученную модель в виде диаграммы объектов-связей (ER-диаграммы). Для наглядности и возможности последующего программирования перейдем к английским названиям объектов (отношений) и атрибутов.

Отношение **R1** представляет объект **student** с атрибутами **id_st** (первичный ключ), surname.

Отношение **R3** представляет объект **exam_st** с атрибутами **id_ex** (первичный ключ), subject, date.

Отношение **R4** представляет объект **mark_st** с атрибутами **id_st** (внешний ключ), **id_ex** (внешний ключ), mark. Первичный ключ здесь **id_st, id_ex**.

Соответствующая ER-диаграмма изображена на [рис. 8.1](#).

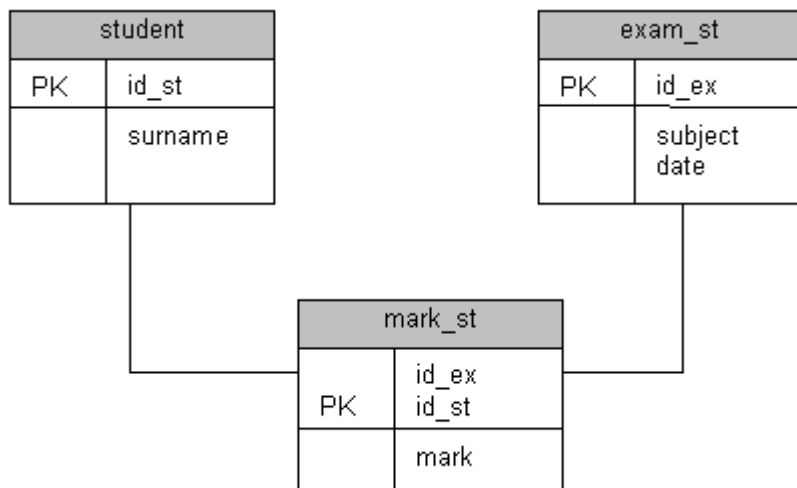


Рис. 8.1. ER-диаграмма, представляющая рассмотренный фрагмент предметной области

8.6. Целостная часть реляционной модели. Реализация условия целостности данных в современных СУБД

Напомним, что под целостностью *базы данных* понимается то, что в ней содержится полная, непротиворечивая и адекватно отражающая предметную часть (правильная) информация. Поддержка целостности в реляционных БД основана на выполнении следующих требований.

1. Первое требование называется **требованием целостности сущностей**. Объекту или сущности реального мира в реляционных БД соответствуют кортежи отношений. Конкретно требование состоит в том, что любой *кортеж* любого отношения отличим от любого другого кортежа этого отношения, т.е., другими словами, любое *отношение* должно обладать определенным первичным ключом. Это требование автоматически удовлетворяется, если в системе не нарушаются базовые свойства отношений.

2. Второе требование называется **требованием целостности по ссылкам**. Очевидно, что при соблюдении нормализованности отношений сложные сущности реального мира представляются в реляционной БД в виде нескольких кортежей нескольких отношений. Связь между отношениями осуществляется с помощью миграции ключа.

Пример внешнего ключа.

СТУДЕНТ (Код студента, Фамилия) сдает ЭКЗАМЕН (Код студента, Предмет, Оценка).

Атрибут Код студента сущности ЭКЗАМЕН называется *внешним ключом*, поскольку его значения однозначно характеризуют сущности, представленные кортежами некоторого другого отношения – отношения Студент (мы предполагаем, что поле Код студента является ключом отношения Студент).

Говорят, что *отношение*, в котором определен *внешний ключ*, ссылается на соответствующее *отношение*, в котором такой же атрибут является первичным ключом.

Требование целостности по ссылкам или требование внешнего ключа состоит в том, что для каждого значения внешнего ключа в ссылающемся отношении в отношении, на которое ведет ссылка, должен найтись кортеж с таким же значением первичного ключа либо значение внешнего ключа должно быть неопределенным (т.е. ни на что не указывать).

Ограничения целостности сущности и по ссылкам должны поддерживаться СУБД. Для соблюдения целостности сущности достаточно гарантировать отсутствие в любом отношении кортежей с одним и тем же значением первичного ключа. (В Access для этого предназначена специальная реализация целочисленного поля – *поле* типа "Счетчик".) С целостностью по ссылкам дела обстоят несколько более сложно.

Понятно, что при обновлении ссылающегося отношения (вставке новых кортежей или модификации значения внешнего ключа в существующих кортежах) достаточно следить за тем, чтобы не появлялись некорректные значения внешнего ключа.

Но как быть при удалении кортежа из отношения, на которое ведет *ссылка*?

Здесь существуют три подхода, каждый из которых поддерживает *целостность* по ссылкам. Первый подход заключается в том, что запрещается производить удаление кортежа, на который существуют ссылки (т.е. сначала нужно либо удалить ссылающиеся кортежи, либо соответствующим образом изменить значения их внешнего ключа). При втором подходе при удалении кортежа, на который имеются ссылки, во всех ссылающихся кортежах *значение* внешнего ключа автоматически становится неопределенным. Наконец, третий подход (*каскадное удаление*) состоит в том, что при удалении кортежа из отношения, на которое ведет *ссылка*, из ссылающегося отношения автоматически удаляются все ссылающиеся кортежи.

В развитых реляционных СУБД обычно можно выбрать способ поддержания целостности по ссылкам для каждой отдельной ситуации определения внешнего ключа. Конечно, для принятия такого решения необходимо анализировать требования конкретной прикладной области.

Заметим, что все современные СУБД поддерживают и *целостность сущностей*, и *целостность* по ссылкам, но позволяют пользователям выключать данные ограничения и, таким образом, строить *базы данных*, не соответствующие реляционной модели. *Опыт* показывает, что отход от основных положений реляционной модели приводит к краткосрочному выигрышу – алгоритмы становятся проще, но впоследствии серьезно усложняют задачу, особенно ее сопровождение.

Краткие итоги: Лекция посвящена вопросам оптимизации схем отношений (структуры реляционной *базы данных*) на основе формальных методов *теории реляционных баз данных*. Здесь рассматривается ряд необходимых для этого понятий (*функциональная зависимость, нормальные формы, декомпозиция схем отношений*). Разбирается пример приведения таблицы к третьей нормальной форме, оптимальной по ряду показателей (исключающей *избыточность, аномалии включения и удаления*). Рассматриваются вопросы реализации целостности данных в реляционных СУБД.

В лекции рассматриваются вопросы использования формального аппарата для оптимизации схем отношений. Сформулирована *проблема выбора рациональных схем отношений* и пути реализации такого выбора путем *нормализации* (последовательного преобразования схемы отношения в ряд *нормальных форм*). Для формального описания соответствующего процесса определены понятие функциональной зависимости (зависимости между атрибутами отношения), ключа, сформулированы правила вывода *множества* функциональных зависимостей, понятие *декомпозиции схемы отношения*. Определены первая, вторая, третья *нормальные формы* и *нормальная форма* Бойса-Кодда. Приведен пример *нормализации* до 3НФ. Рассмотрены вопросы реализации условий целостности данных в реляционных СУБД.

Вопросы настоящей лекции рассматриваются в [[5.4](#)] - [[4.5](#)].

Тема 3.2. Физические модели данных (внутренний уровень). Структура современной СУБД. Программное обеспечение работы с современными базами данных

Цель 9 лекции: дать *представление* об основных типовых способах организации данных в памяти ЭВМ в СУБД с оценкой соответствующих моделей по времени доступа к данным в базе данных и по объему занимаемой памяти.

Как уже отмечалось, концептуальная схема, специфицированная к СУБД, автоматически отображается в *структуру хранения* программами СУБД. Внешний *пользователь* может ничего не знать о том, как его *представление* о данных физически организовано в памяти вычислительной системы. Тем не менее от физического *размещения данных в памяти ЭВМ* существенно зависит время решения прикладных задач. В связи с этим, даже на одном из начальных этапов проектирования *базы данных* – этапе выбора СУБД, желательно знать возможности физических *структур хранения*, представляемых конкретными СУБД, и оценивать временные характеристики проектируемой *базы данных* с учетом этих возможностей.

Способы физической организации данных в различных СУБД, как правило, различны и определяются типом используемой ЭВМ, инструментальными средствами разработки СУБД, а также критериями, которыми руководствуются разработчики СУБД при выборе методов размещения данных и способов доступа к этим данным. Заметим, что наиболее распространенным критерием служит *время доступа* к данным, однако в качестве критерия может выбираться, например, трудоемкость реализации соответствующих методов.

В настоящей лекции будут рассмотрены типовые физические модели организации данных в конкретных СУБД.

Физические модели данных служат для отображения моделей данных. Основными понятиями модели данных являются *поле, логическая запись, логический файл*. Слово "логический" введено, чтобы отличать понятия, относящиеся к *логической модели данных*, от понятий, относящихся к *физической модели данных*. Основными понятиями *физической модели данных*, используемыми для представления *логической модели данных*, являются *поле, физическая запись, физический файл*. В частности, *логическая запись*, состоящая из полей, может быть представлена в виде физической записи (из тех же полей), *логический файл* – в виде физического файла. Прежде чем конкретизировать понятия, относящиеся к *физической модели данных*, рассмотрим структуру памяти ЭВМ.

9.1. Структура памяти ЭВМ

Важнейшей особенностью памяти ЭВМ, в значительной степени определяющей методы организации данных и доступа к ним, является её неоднородность. Существуют два разных типа памяти – оперативная (ОП) и внешняя (ВП), причем *процессор* работает только с данными из оперативной памяти (рис. 9.1.).

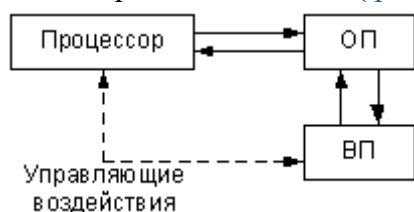


Рис. 9.1. Схема работы ЭВМ

Как уже многократно отмечалось, *базы данных* создаются для работы с большими объемами данных, что обуславливает необходимость использования внешней памяти. Поэтому организация данных и доступа к ним должна учитывать как специфику каждого типа памяти, так и способы их взаимодействия.

Отметим основные свойства оперативной памяти:

- единицей памяти является байт;
- память прямоадресуема (каждый байт имеет адрес);
- процессор выбирает для обработки нужные данные, непосредственно адресуясь к последовательности байтов, содержащих эти данные.

Отметим основные свойства внешней памяти:

- минимальной адресуемой единицей является *физическая запись* ;
- для последующей обработки (например, работы с полями) запись должна быть считана в оперативную память;
- время чтения записи в ОП на несколько порядков выше времени обработки процессором записи из ОП;
- организация обмена осуществляется порциями, т.к. невозможно считать сразу всю базу данных.

9.2. Представление экземпляра логической записи

Логическая запись представляется в оперативной памяти следующим образом:

Логическая запись				Последовательность байтов ОП				
Поле 1	Поле 2	..	Поле N		B1	B2	...	BN
Тип поля					B _i – последовательность байтов ОП, используемая для хранения поля i			
Характеристика поля								
Длина								

Прямая адресация байтов позволяет процессору выбирать для обработки нужное *поле*.

Заметим, что указанное *представление* не делает различий для записей в сетевой, иерархической и реляционных моделях. В случае сетевой и *иерархической моделей* некоторые поля могут являться указателями, тогда последовательность байтов, используемая для хранения этих полей, содержит *адрес* начала последовательности байтов, соответствующей записи – члену отношения.

В большинстве современных *СУБД* используется формат записей фиксированной длины. В этом случае все записи имеют одинаковую длину, определяемую суммарной длиной полей, составляющих *запись*. В *СУБД* другие форматы записей (переменной длины, неопределенной длины) встречаются гораздо реже, поэтому в данной книге эти форматы не рассматриваются. Заметим, что поля записи, принимающие значения существенно разной длины в различных экземплярах записей, в *предметной области* встречаются достаточно часто. Примером может служить *поле резюме* в записи СОТРУДНИК. *Резюме* может составлять полстраницы текста, страницу и т.д. Возникает проблема – как эту информацию переменной длины представить в записи фиксированной длины. Возможным вариантом является установление размера соответствующего поля по максимальному значению. В этом случае у многих экземпляров записи указанное *поле* будет заполнено не полностью и, таким образом, *память ЭВМ* будет использоваться неэффективно. Более эффективный и часто используемый в *СУБД* прием организации таких записей состоит в следующем. Вместо поля (полей), принимающего значение существенно разной длины, в *запись* включается *поле-указатель* на область памяти, где будет размещаться значение исходного поля. Как правило, эта область является областью внешней памяти прямого доступа. В процессе ввода соответствующего значения в выделенной области занимает столько памяти, какова *длина* этого значения.

На *рис. 9.2* представлен пример вышеуказанного представления *экземпляров записей* из *N* полей, причем *поле N* принимает значения соответственно разной длины у разных *экземпляров записей*.



Рис. 9.2. Представление полей переменной длины

Конкретной реализацией такой схемы является *поле* типа MEMO в СУБД (dBase III+, FoxPro, Access и т.д.).

9.3. Организация обмена между оперативной и внешней памятью

Единицей обмена данными между оперативной и внешней памятью является *физическая запись*. *Физическая запись* читается (записывается) за одно обращение к внешней памяти. В частности, *физическая запись* может соответствовать одному экземпляру логической записи. Число обращений к внешней памяти при работе с базой данных определяет время отклика системы. В связи с этим для уменьшения числа обращений к БД при работе с ней увеличивают длину физической записи (объединяют в одну *физическую запись* несколько экземпляров логических записей). В этом случае *физическую запись* называют также блоком, число k экземпляров логических записей, составляющих *физическую запись*, – коэффициентом блокировки.

Ввод исходных данных в БД осуществляется следующим образом:

- в ОП последовательно вводятся k экземпляров логических записей (кортежей);
- введенные k экземпляров объединяются в *физическую запись* (блок);
- *физическая запись* заносится во внешнюю память.

Ввод k экземпляров записей исходной таблицы, составляющих i -ю *физическую запись*, изображен на [рис. 9.3](#).

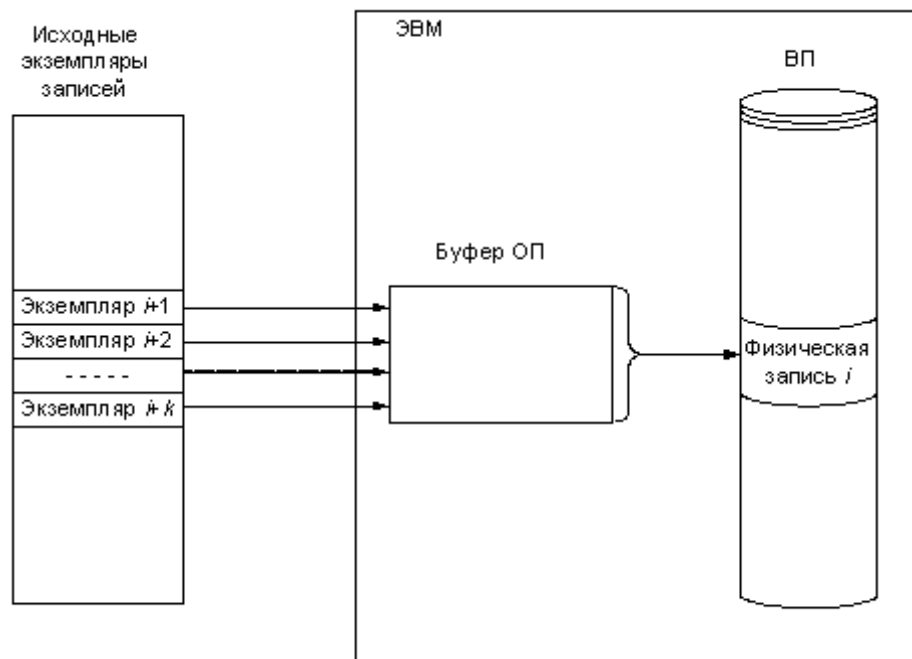


Рис. 9.3. Схема занесения записей во внешнюю память

Обработка данных, хранящихся во внешней памяти, осуществляется следующим образом:

- *физическая запись* (блок) считывается в оперативную память;
- обрабатываются экземпляры логических записей внутри блока (выбираются нужные поля, производится сравнение ключевого поля с заданным значением, осуществляется корректировка полей, выполняются операции удаления и т.п.).

В некоторых СУБД (например, MS SQL Server) единицей обмена между оперативной и внешней памятью является *страница* (вид физической записи, размер которой фиксирован и не зависит от длины логической записи). Организация обмена между оперативной и внешней памятью в этом случае аналогична описанной выше. Отличие здесь будет состоять в том, что экземпляры логических записей формируются в буфере, размером со страницу (если размер страницы не кратен длине логической записи, страница может быть заполнена

неполностью, *физическая запись* на внешнем носителе, соответственно, будет заполнена не полностью).

9.4. Структуры хранения данных во внешней памяти ЭВМ

В современных СУБД наибольшее распространение получили *табличные модели* данных. В связи с этим, а также для большей определенности в настоящем разделе мы будем говорить о *структурах хранения* для табличной модели. Однако отметим, что некоторые из рассматриваемых ниже *структур хранения* могут использоваться и для представления сетевых и *иерархических моделей*.

В качестве внешней памяти мы рассматриваем наиболее распространенную в современных ЭВМ *память* прямого доступа. *Память* прямого доступа дает возможность обращения к любой записи, если известен её *адрес*. Для упрощения изложения мы не будем конкретизировать ряд служебных полей, которые содержит *физическая запись*, и их рассмотрение опускаем.

9.4.1. Последовательное размещение физических записей

В этой структуре хранения записи в памяти размещаются последовательно друг за другом. Как уже отмечалось, считаем, что все записи имеют равную длину. Физический адрес записи может быть легко вычислен по номеру записи (для вычисления необходимо знать формат соответствующей физической записи).

Физическая запись с номером I содержит логические записи с номерами

$$\begin{aligned} &(I-1)k + 1 \\ &(I-1)k + 2 \\ &\dots \\ &(I-1)k + k \\ &I = 1, 2, \dots, \lceil N/k \rceil; \end{aligned}$$

знаком $\lceil N/k \rceil$ обозначим ближайшее целое, большее или равное N/k , – целое сверху.

Рассмотрим, как реализуются основные элементарные операции модели данных в этой структуре хранения, и оценим число этих операций. Напомним, что с точки зрения пользователя в табличной модели данных эти операции являются операциями над строками (столбцами) таблицы.

Поиск записи с заданным значением ключа

При последовательной структуре хранения поиск может осуществляться только перебором. Читается первая *физическая запись*, в ОП она разбивается на k логических записей (разблокируется), заданное значение ключа сравнивается со значением ключа каждой логической записи. При несовпадении читается следующая *физическая запись* и процесс повторяется. В лучшем случае нужная запись будет найдена за одно обращение, в худшем – необходимо считать все физические записи. Среднее число обращений к внешней памяти для поиска нужной записи TP определяется следующей формулой

$$TP = (1 + \lceil N/k \rceil)/2,$$

где N – число логических записей, k – коэффициент блокировки, $\lceil N/k \rceil$ – число физических записей.

Чтение записи с заданным значением ключа

Сначала необходимо найти нужную запись (смотри операцию "поиск"). После окончания операции "поиск" нужная запись уже считана в ОП. Число обращений к ВП равно TP .

Корректировка записи

Сначала необходимо найти нужную запись (смотри операцию "поиск"). После окончания операции "поиск" в ОП найденная *логическая запись* корректируется,

формируется *физическая запись* (блок) и заносится во внешнюю память по тому адресу, откуда она была считана. Число обращений к ВП равно $TP+1$.

Удаление записи

Аналогична операции корректировки. Служебное поле соответствующей логической записи помечается как "удаленная запись". Число обращений к ВП равно $TP+1$.

Добавление записи

Рассмотрим два случая. В первом случае пользователь вводит новую *логическую запись* в конец таблицы. Тогда вводимая *логическая запись* добавляется в конец файла. Она заносится либо в последнюю *физическую запись* (если в ней меньше k логических записей – блок неполон), для чего эта запись должна быть считана в ОП, или формируется новая *физическая запись*, которая заносится в конец файла. Число обращений к ВП равно соответственно либо 2, либо 1.

Во втором случае пользователь вводит новую *логическую запись* в указываемую им i -ю строку таблицы ($i=1, 2, \dots, n$). В этом случае читается *физическая запись* с номером $\lceil (i-1)/k \rceil$, содержащая i -ю *логическую запись*. Если соответствующая *физическая запись* содержит пустые логические записи, то добавляемая запись вставляется в этот блок, блок записывается на свое место в ВП. Число обращений к ВП равно 2. Если указанная *физическая запись* содержит k экземпляров логических записей исходной таблицы, читается *физическая запись* с номером $\lceil i/k \rceil$. Если эта *физическая запись* содержит пустые логические записи, добавляемая запись вставляется в этот блок, блок записывается на свое место в ВП. Суммарное число обращений в этом случае будет на единицу больше и равно 3.

Если физические записи с номерами $\lceil (i-1)/k \rceil$ и $\lceil i/k \rceil$ содержат по k экземпляров исходных логических записей, необходимо формировать дополнительную *физическую запись*. Соответствующий блок будет содержать добавляемую *логическую запись* и $k-1$ пустых логических записей. Блоки с номерами $\lceil i/k \rceil, \lceil (i+1)/k \rceil, \dots, \lceil N/k \rceil$ переписываются на одну позицию ниже (сдвигаются). Сформированная *физическая запись* заносится на освободившееся место (место записи с номером $\lceil i/k \rceil$).

В лучшем случае ($i = N$) ни один блок не сдвигается. В худшем случае ($i = 1$) сдвигаются все блоки. Среднее число обращений к ВП для перезаписи блоков (чтение + запись) составит $2\lceil N/k \rceil/2$. Тогда суммарное число обращений к ВП при добавлении записи в этом случае будет равно $3 + \lceil N/k \rceil$.

Заметим, что если записи упорядочены по значениям ключа поиск может производиться дихотомическим методом и число обращений к внешней памяти будет пропорционально не $(1 + \lceil N/k \rceil)/2$, а $\log_2 \lceil N/k \rceil$, т.е. существенно меньше. Однако добавление записи потребует для сохранения упорядоченности, как правило, сдвига большого числа записей. Поэтому размещение физических записей с упорядочением их по значениям ключа в СУБД не используется.

9.4.2. Размещение физических записей в виде списковой структуры

Основная проблема в использовании изложенного в п. 9.4.1 способа организации записей состоит в отображении добавления логической записи в произвольное место таблицы. При этом приходится переписывать в памяти (сдвигать на одну позицию) физические записи, соответствующие логическим записям таблицы, расположенным ниже места вставки добавляемой строки. Соответствующую проблему можно устранить, используя для представления физических записей связный список (рис. 9.4).

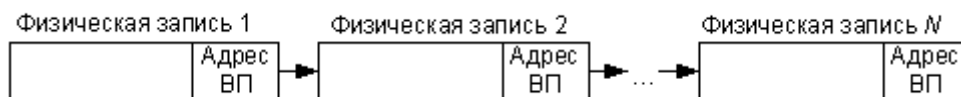


Рис. 9.4. Список физических записей

Кроме этого списка в ВП формируется список свободных элементов ("пустых" физических записей), элементы которого используются при вводе новой записи с данными (рис. 9.5).

Напомним, что каждая *физическая запись* состоит, как и ранее, из k логических записей.

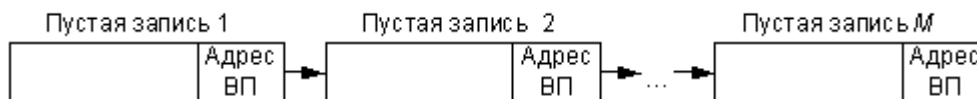


Рис. 9.5. Список свободных элементов

Рассмотрим, как реализуются основные элементарные операции модели данных в этой структуре хранения.

Поиск записи с заданным значением ключа

Заметим, что упорядочение записей по значениям ключа не дает здесь ускорения процедуры поиска. Это связано с тем, что после ряда добавлений новых записей и удаления каких-то имеющихся записей физическая и логическая последовательность записей в списке будут существенно различаться. При этом будет невозможно по номеру записи определить ее адрес и обращаться к записи, соответствующей середине таблицы, для реализации дихотомического метода поиска. Поэтому поиск можно вести только с помощью перебора. В ОП читается первая запись списка, разблокируется, значения ключевых полей логических записей этой физической записи сравниваются с заданным значением. Если значения совпали, нужная запись найдена, если не совпали, из записи выбирается адрес следующей записи списка, читается эта запись. Далее процедура повторяется. Среднее число обращений к ВП будет равно, как и в 9.4.1, $(1 + \lceil N/k \rceil)/2$.

Чтение записи

После завершения предыдущей операции запись считана в ОП. Оценка числа обращений к ВП та же.

Корректировка записи

Считанная запись корректируется и заносится в ВП на свое место (по своему адресу). Число обращений к ВП на единицу больше, чем при чтении.

Удаление записи

Заметим, что мы говорим об операциях над логическими записями. Операция удаления логической записи аналогична операции корректировки. Служебное поле соответствующей логической записи помечается как "удаленная запись". Сформированная *физическая запись* заносится в ВП. Число обращений к ВП равно $TP+1$.

Добавление записи

Для определенности будем считать, что задан ключ логической записи, после которой должна быть добавлена новая запись. Осуществляется операция поиска и чтения физической записи, в которой расположена запись с ключом PK . Если в этом блоке есть *логическая запись*, помеченная как удаленная, добавляемая запись заносится на ее место. Блок записывается в ВП. Число обращений к ВП равно $TP+1$. Если в этом блоке нет логических записей, помеченных как удаленные, необходимо добавлять новую *физическую запись*, выбираемую из списка свободных элементов. С этой целью адрес связи найденной ранее физической записи заменяется на адрес начала списка свободных элементов.

Читается первая *физическая запись* списка свободных элементов. Адрес связи этой записи заменяет адрес начала пустого списка. В ОП формируется новая *физическая запись*, содержащая добавляемую *логическую запись*. В качестве ее адреса связи заносится адрес связи

из физической записи, предшествующей добавляемой. Каждая из этих записей заносится в ВП. Число обращений к ВП при добавлении записи будет примерно равно $TP+3$.

Рассмотренный метод организации *структуры хранения* достаточно эффективно решает проблемы добавления и удаления записей, но не уходит от перебора при поиске нужной записи.

9.4.3. Использование индексов (индексирование)

Как уже отмечалось, упорядочение записей позволяет использовать дихотомический метод поиска нужной записи и тем самым существенно сократить одну из основных составляющих времени поиска – число обращений к ВП. Однако при этом возникают проблемы с добавлением записей, связанные с необходимостью перезаписи части физических записей (сдвига).

Для того чтобы использовать дихотомический поиск и не перемещать физические записи при добавлении новых записей, используется так называемое логическое упорядочение физических записей (индексирование). Основная структура хранения содержит записи исходной таблицы и представлена в виде неупорядоченной последовательности физических записей (см. п. 9.4.1). Для возможной реализации дихотомического поиска по определенному ключу создается дополнительная структура хранения (так называемый индекс). Число записей в индексе равно числу записей исходной таблицы (числу физических записей в основной структуре хранения). Каждая запись индекса имеет два поля: ключевое поле записи основной структуры и указатель – адрес записи основной структуры с соответствующим значением ключа.

Записи индекса (индексного файла) упорядочены по значению ключа. Адреса связи этих записей определяют логическое упорядочение записей основной *структуры хранения*. Пример соответствующей *структуры хранения* приводится в предположении $k=1$ на [рис. 9.6](#).

Рассматриваемую *структуру хранения* называют еще инвертированным списком. Смысл этого термина состоит в следующем. Можно было бы упорядочить записи основной *структуры хранения*, не переставляя их, а объединив в соответствующий упорядоченный список. В нашем случае адреса связи как бы удаляются из списка и включаются в состав файла-индекса (инвертируются). Поэтому полученная структура интерпретируется как инвертированный список.

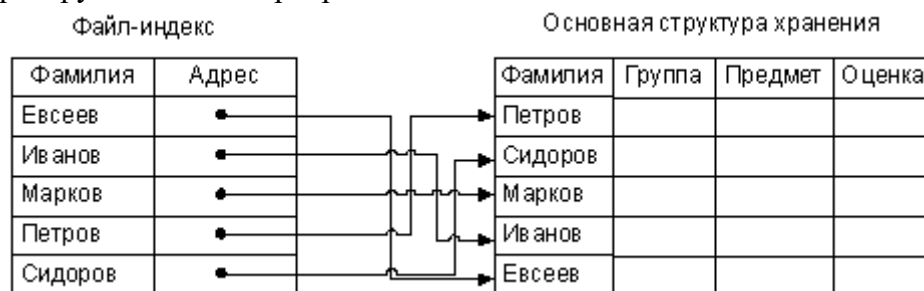


Рис. 9.6. Индексирование

Поиск нужной записи по заданному значению ключа осуществляется в индексном файле *методом половинного деления*. Заметим, что так как записи индекса содержат всего два поля, суммарный объем записей индекса невелик, поэтому индекс, как правило, целиком считывается для обработки в ОП за одно обращение к ВП. После того как в индексном файле обнаружена искомая запись, по адресу связи читается полная соответствующая запись основной *структуры хранения*. Если необходим поиск по другому ключу, строится еще один индекс по соответствующему ключу. Таким образом, по любому ключу поиск можно осуществлять дихотомическим методом.

Оценим число обращений к ВП при реализации элементарных операций. Соответствующие оценки сделаны для случая, когда *физическая запись* состоит из одной логической записи (коэффициент блокировки k равен 1). Расчет оценок для произвольного k производится по аналогии с расчетами пп. 9.4.1–9.4.2.

Поиск записи с заданным значением ключа

Из ВП читается индексный файл (число обращений к ВП для этого зависит от объема индексного файла, как правило, невелико и много меньше числа записей N). После нахождения нужной записи в индексном файле читается соответствующая запись основного файла (одно обращение к ВП).

Чтение записи

В ходе операции поиска искомая запись считана в ОП.

Корректировка записи

Считанная запись корректируется и заносится на свое место (еще одно обращение к ВП).

Удаление записи

Найденная запись помечается как удаленная в основном файле, соответствующая запись в индексном файле удаляется, измененный индекс записывается в ВП. Число обращений к ВП в этом случае по сравнению с числом обращений к ВП при поиске увеличивается на два.

Добавление записи

Добавляемая запись заносится в конец основного файла. Формируется новая запись индекса, соответствующая добавляемой записи. Записи индекса переупорядочиваются по значениям ключа, и индекс заносится в ВП. Число обращений к ВП в этом случае, в основном, определяется чтением-записью индекса.

Таким образом, использование индексов позволяет ценой некоторого увеличения объема используемой памяти (за счет индекса) существенно сократить время реализации основных операций. В связи с этим *индексирование* используется во многих современных СУБД.

9.4.4. В-дерево

Структура В-дерева (сбалансированное дерево) является следствием дальнейшего расширения концепции использования индексов (строится индекс над индексом) и представляет собой многоуровневые индексы.

В-дерево строится следующим образом. Последовательность записей, соответствующая записям исходной таблицы, упорядочивается по значениям первичного ключа. Логические записи объединяются в блоки (по k записей в блоках).

Значением ключа блока является минимальное значение ключа у записей, входящих в блок. Последовательность блоков представляет собой последний уровень *В-дерева*. Строится индекс предыдущего уровня. Записи этого уровня содержат значение ключа блока следующего уровня и указатель-адрес связи соответствующего блока; записи этого уровня также объединяются в блоки (по k записей). Затем аналогично строится индекс более высокого уровня и т.д., пока количество записей индекса на определенном уровне будет не более k .

Рассмотрим процедуру работы с В-деревом на примере. Пусть имеется файл экземпляров логических записей, ключи которых принимают значения 2, 7, 8, 12, 15, 27, 28, 40, 43, 50. Для определенности возьмем $k=2$ (в блок объединяем по 2 экземпляра записей). Построенное для этого примера *В-дерево* изображено на [рис. 9.7](#) (для упрощения рисунка на уровне 4 представлены только ключи логических записей и не представлены значения других полей этих записей).

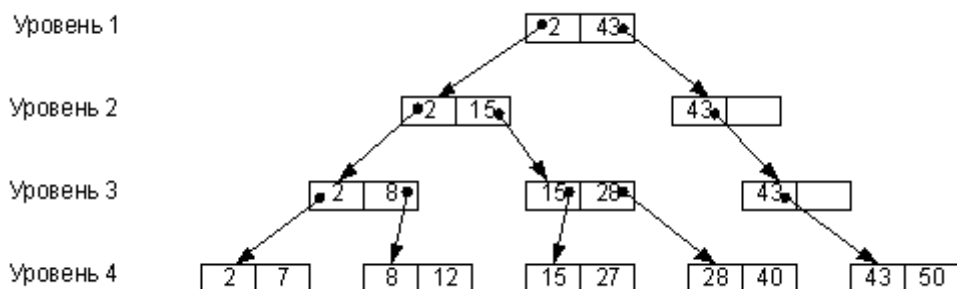


Рис. 9.7. В-дерево

В блоках указано значение ключа соответствующего блока. Значение k принято равным 2.

По построению *В-дерева* все исходные записи находятся на одном расстоянии от верхнего индекса (дерево является сбалансированным).

Рассмотрим реализацию основных операций.

Поиск и чтение записи с заданным значением ключа

Читается верхний индекс. Сравниваем заданное значение ключа со значением ключа последней записи индекса. Если заданное значение ключа больше, чем значение ключа очередной записи индекса (если такая запись имеется), или равно ему, то по адресу связи, указанному в текущей записи, читается блок записей индекса следующего уровня. Далее процесс повторяется.

Считаем, что все блоки расположены в ВП. Тогда число обращений к ВП при поиске информации будет равно числу уровней дерева. Число уровней дерева равно минимальному значению l , при котором выполняется условие $k^l \geq N$ (N – число логических записей).

Модификация (корректировка) записи

После поиска и чтения записи изменяются корректируемые поля. Если корректируется не ключ записи, то измененная запись заносится на свое место. Если изменено значение ключа, то старая запись удаляется (в соответствующем блоке появляется "пустая" запись), а измененная запись заносится так же, как вновь добавляемая.

Удаление записи

После поиска найденная запись удаляется (в соответствующий блок на место этой записи заносится "пустая" запись).

Добавление записи

Прежде всего определяется, где должна быть расположена добавляемая запись с заданным значением ключа. Процедура поиска блока, где должна быть расположена эта запись, аналогична вышеописанной процедуре поиска записей с заданным значением ключа. Если в найденном блоке низшего уровня есть "пустая" запись, добавляемая запись заносится в этот блок (с необходимым переупорядочением записей внутри блока).

Если в соответствующем блоке низшего уровня нет пустого места, блок делится на два блока. В первый из них заносится $\lceil k/2 \rceil$ записей, во второй заносятся остальные. Значением ключа каждого из указанных блоков будет являться, как и описано ранее, минимальное значение ключей у записей, входящих в блок. Добавляемая запись заносится в тот блок, значение ключа которого меньше значения ключа добавляемой записи. Появление нового блока с новым значением ключа обуславливает необходимость формирования соответствующей новой записи в индексе на предыдущем уровне. Эта запись содержит новое значение ключа нового блока и указатель на его месторасположение. Процедура добавления такой записи аналогична описанной выше. Находится блок предыдущего уровня, куда должна быть помещена эта запись. Если в блоке есть пустое место, запись добавляется в блок, если блок полон, он делится на два блока, запись заносится в один из блоков, формируется запись индекса предыдущего уровня и т.д.

Возможен вариант, когда придется делить блок самого верхнего уровня и формировать еще один уровень дерева.

Рассмотрим для примера, изображенного на [рис. 9.7](#), добавление записи с ключом 10.

1. Сравнение на первом уровне.

$2 < 10 < 43$

Движение по левой ветви.

2. Сравнение на втором уровне.

$2 < 10 < 15$

Движение по левой ветви.

3. Сравнение на третьем уровне.

$2 < 8 < 10$

Движение по правой ветви.

Искомый блок

8	12

4. Блок заполнен.

Он делится на 2 блока

8	

12	

Сравнение $8 < 10 < 12$.

Запись с ключом 10 заносится в блок 1

8	10

12	

На низшем уровне появилась новая запись с значением ключа 12. Необходимо добавление новой записи с ключом 12 и указателем на запись низшего уровня к индексу предыдущего уровня.

5. Запись с ключом 12 уровня 3 должна добавляться в блок

2	8

. Блок полон, он делится на два блока

2	

8	

Сравнение $8 < 12$.

Запись добавляется во второй блок

8	12

6. На уровне 3 появился блок с новым ключом 8. Необходимо добавление новой записи с ключом 8 и указателем на соответствующий блок уровня 3 на уровне 2.

7. Запись с ключом 8 уровня 2 должна добавиться в блок

2	15

. Блок полон, он делится на два блока.

2	

15	

$2 < 8 < 15$

Запись добавляется в блок 1

2	8

.

8. На уровне 2 появился блок с новым ключом 15, необходимо добавление новой записи с ключом 15 и указателем на соответствующий блок уровня 2 на уровне 1.

9. Запись с ключом 15 уровня 1 должна добавляться в блок

2	43

. Блок полон, он делится на два блока.

2	

43	

$2 < 15 < 43$

Запись с ключом 15 добавляется в первый блок

2	15

43	

10. Необходимо сформировать еще один уровень дерева

2	43

.

Полученная структура будет иметь вид, представленный на [рис. 9.8](#).

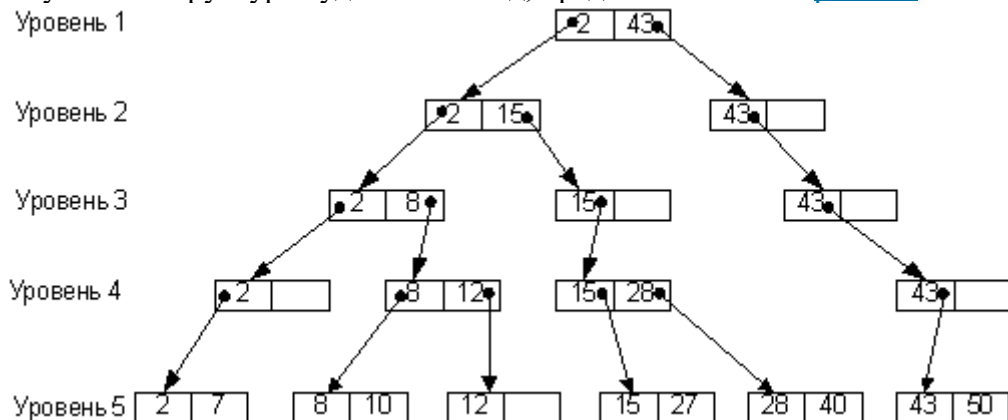


Рис. 9.8. В-дерево после добавления элемента

Необходимо заметить, что используемый прием деления пополам полностью заполненного блока при добавлении в него записи приведет к тому, что блоки будут заполнены, в среднем, наполовину. Тогда процедура добавления записи будет существенно менее трудоемкой (если в нужном блоке есть место, запись добавляется в этот блок и вышестоящие уровни не перестраиваются).

Структура хранения в виде В-дерева позволяет эффективно проводить операции поиска, чтения, удаления, модификации с оценкой числа обращений к внешней памяти числом

уровней дерева l ($l \approx \log kN$), что существенно меньше числа обращений при переборе $\lceil N/k \rceil$.

Процедура добавления записи тоже достаточно эффективна. Соответствующая *структура хранения*, в частности, используется в отечественной СУБД НИКА (ранее использовалась в системе ИНЕС) и на реальных задачах показала высокую эффективность.

9.4.5. Размещение записей с использованием хэширования

Как в любом другом способе организации *структур хранения*, логические записи группируются в физические записи (блоки) по k штук. Однако в отличие от всех других способов организации *структур хранения* здесь выбран особенный способ группировки. *Определенным образом выбирается так называемая хэш-функция f* . Аргументом этой функции является значение x первичного ключа логической записи. Тогда $f(x)$ указывает адрес расположения блока, в котором должна находиться логическая запись со значением ключа x .

Функция f должна, по возможности, равномерно распределять значения x по физическим блокам. Обсуждению возможных *хэш-функций* посвящено достаточно много литературы, поэтому здесь мы не будем касаться этого вопроса. Можно лишь добавить, что иногда, исходя из специфики множества значений x первичного ключа, можно построить функцию f , удовлетворяющую всем необходимым условиям. Таким образом, *логическая запись* таблицы со значением x первичного ключа размещается в блоке внешней памяти по адресу $f(x)$. В этом блоке может находиться не более k записей. Может оказаться, что выбранная функция отображает в один адрес памяти (один блок) более k записей. Возникает так называемая коллизия. Возможным способом разрешения коллизий является использование дополнительной области переполнения следующим образом. Если очередная запись распределяется с помощью функции хэширования в блок, а он полностью заполнен, то в области переполнения формируется список записей, соответствующих этому блоку, с включением в него указанной записи, а в сам блок заносится указатель – адрес связи на первую запись этого списка. Возможны и другие способы разрешения коллизий.

Рассмотрим реализацию основных операций и дадим оценку числа обращений к ВП при их выполнении.

Поиск записи с заданным значением ключа и чтение

По заданному значению ключа x подсчитывается значение функции $f(x)$. Далее из ВП считывается блок, находящийся по адресу $f(x)$. В ОП внутри этого блока перебором ищется нужная запись. Если записей в блоке нет, то по указателю в блоке (адресу связи) читается первая запись списка переполнения, относящаяся к этому блоку. Далее необходимая запись ищется по этому списку. Число обращений к ВП при этом равно:

- единице, если запись находится в блоке;
- единице плюс число записей в соответствующем этому блоку списке области переполнения (как правило, небольшое число).

Модификации записи

Осуществляется поиск и чтение записи, затем в ОП модифицируются поля записи (не являющиеся первичным ключом), запись заносится на свое место. Число обращений к ВП в этом случае на единицу больше, чем при чтении записи. Если модифицируется значение ключа, то занесение записи осуществляется как ввод новой записи (добавление).

Удаление записи

Осуществляется поиск и чтение записи. Если удаляемая запись находилась в блоке основной памяти, на ее место заносится "пустая" запись (или признак "пустой" записи). Если удаляемая запись находилась в списке области переполнения, удаление ее производится по правилам удаления элемента списка. Число обращений к ВП при удалении находится примерно в тех же пределах, что и для предыдущих операций.

Добавление записи

При добавлении записи со значением ключа x подсчитывается адрес соответствующего блока $f(x)$. Блок считывается в ОП. Если в нем есть место, запись заносится в блок, блок записывается в ВП по своему адресу. Если блок заполнен, из него выбирается адрес начала списка записей, переполняющих блок. Далее добавление записи в список производится по правилам добавления элемента в список. Число обращений к ВП при добавлении записей находится примерно в тех же пределах, что и для предыдущих операций.

Таким образом, описанная *структура хранения* с использованием хэширования является наиболее эффективной (из рассмотренных выше) по критерию минимизации числа обращений к ВП при реализации основных операций.

9.4.6. Комбинированные структуры хранения

Необходимо заметить, что в СУБД могут использоваться как каждая из выше рассмотренных структур в отдельности, так и их комбинация. Так, например, в ряде промышленных систем UNIBAD, БАНК для ЭВМ типа IBM 360/370 (ЕС ЭВМ), PARADOX для персональных ЭВМ используются следующие комбинации методов:

- *размещение записей по первичному ключу организовано с использованием хэширования;*
- *последовательность записей по вторичному ключу задается с помощью списковой структуры.*

Краткие итоги: Лекция посвящена вопросам физической организации данных в памяти компьютера (организации *структур хранения*). Физические модели представления данных жестко заложены в структуру конкретной СУБД и различны в различных системах управления базами данных. Заметим, что в данной лекции рассматриваются не *структуры хранения* конкретной СУБД, а некоторые типовые *структуры хранения*, на основе которых и реализуются физические модели организации данных в конкретных СУБД. Здесь описывается двухуровневая структура памяти компьютера как среда размещения данных; организация обмена между внешней и оперативной памятью, определяющая специфику обработки данных. Представлены типовые физические модели (*структуры хранения* данных) во внешней памяти ЭВМ (последовательное размещение физических записей, размещение физических записей в виде *списковой структуры*, использование индексов, организация данных в виде *В-дерева*, размещение записей с использованием хэширования, а также комбинированные *структуры хранения*). Для основных *структур хранения* сделана оценка числа действий при выполнении операций поиска данных, чтения, занесения данных, модификации (корректировки), удаления.

Цель 10 лекции: показать основные элементы структуры современной СУБД (архитектуры базы данных и структуры программного обеспечения) на примере *Microsoft SQL Server 2008*.

10.1 Общая структура СУБД

Для лучшего понимания принципов работы современных СУБД рассмотрим структуру одной из наиболее распространенных клиент-серверных СУБД - *Microsoft SQL Server 2008*. Несмотря на то, что каждая коммерческая СУБД имеет свои отличительные особенности, информации о том, как устроена какая-то из СУБД, обычно бывает достаточно для быстрого первоначального освоения другой СУБД. Краткий обзор возможностей *Microsoft SQL Server - 2008* был приведен в разделе, посвященном краткому обзору современных СУБД. В данном разделе рассмотрим основные моменты, связанные со структурой соответствующей СУБД (архитектурой базы данных и структурой программного обеспечения).

Под архитектурой (структурой) базы данных конкретной СУБД будем понимать основные модели представления данных, используемые в соответствующей СУБД а также взаимосвязи между этими моделями.

В соответствии с рассмотренными в "Различные архитектурные решения, используемые при реализации многопользовательских СУБД. Краткий обзор СУБД" различными уровнями описания данных различают разные уровни абстракции архитектуры базы данных.

Логический уровень (уровень модели данных СУБД) - средство представления концептуальной модели. Здесь каждая СУБД имеет некоторые отличия, но они являются не очень значительными. Отметим, что у разных СУБД существенно отличаются механизмы перехода от логического к физическому уровню представления.

Физический уровень (внутреннее представление данных в памяти ЭВМ - физическая структура базы данных). Данный уровень рассмотрения подразумевает изучение базы данных на уровне файлов, хранящихся на жестком диске. Структура этих файлов – особенность каждой конкретной СУБД, в т.ч. и Microsoft SQL Server.



Рис. 10.1. Архитектура базы данных в Microsoft SQL Server 2008

10.2. Архитектура базы данных. Логический уровень

Рассмотрим *логический уровень представления базы данных* (<http://msdn.microsoft.com>). Microsoft SQL Server 2008 представляет собой реляционную СУБД (данные представляются в виде таблиц). Таким образом, основной структурой модели данных этой СУБД являются таблицы.

Таблицы и типы данных

Таблицы содержат данные о всех сущностях концептуальной модели базы данных. При описании каждого столбца (поля) пользователь должен определить тип соответствующих данных. Microsoft SQL Server 2008 поддерживает как уже ставшие традиционными типы данных (символьная строка с разным представлением, число с плавающей точкой длиной 8 или 4 байта, целое число длины 2 или 4 байта, дата и время, поле примечаний, булево значение и т. д.), так и новые типы данных. Кроме этого Microsoft SQL Server 2008 предоставляет специальный аппарат для создания *пользовательских типов данных*.

Рассмотрим краткую характеристику некоторых новых типов данных, значительно расширяющих возможности пользователя (<http://www.oszone.net>).

Тип данных hierarchyid

Тип данных hierarchyid позволяет создавать отношения между элементами данных в таблице, для того, чтобы задать позицию в иерархии связей между строками таблицы. В результате использования этого типа данных в таблице строки таблицы могут отображать определенную иерархическую структуру, соответствующую связям между данными этой таблицы.

Пространственные типы данных

Пространственные данные – это данные, определяющие географические расположения и формы, преимущественно на Земле. Это могут быть ориентиры, дороги и даже расположение фирмы. В SQL Server 2008 есть географические (geography) и геометрические

(*geometry*) типы данных для работы с этой информацией. Тип данных *geography* работает с информацией для шарообразной земли. Модель шарообразной земли использует при расчетах кривизну земной поверхности. Информация о положении задается широтой и долготой. Эта модель хорошо подходит для приложений, связанных с морскими перевозками, военным планированием и краткосрочными приложениями, имеющими привязку к земной поверхности. Эту модель нужно использовать, если данные хранятся в виде широт и долгот.

Тип данных *geometry* работает с планарной моделью или моделью плоской земли. В этой модели земля считается плоской проекцией из определенной точки. Модель плоской земли не принимает в расчет кривизну поверхности земли, поэтому используется, в первую очередь, для описания коротких расстояний, например, в базе данных приложения, описывающего внутреннюю часть строения.

Типы *geography* и *geometry* создаются из векторных объектов, заданных в форматах Well-Known Text (WKT) или Well-Known Binary (WKB). Это форматы для перенесения пространственных данных, описанные в простых функциях открытого геопространственного консорциума (Open Geospatial Consortium (OGC) Simple Features) для спецификаций SQL (SQL Specification).

Ключи

Для каждой таблицы должен быть определен *первичный ключ* – *минимальный набор атрибутов, уникально идентифицирующий каждую запись в таблице*. Для реализации связи между таблицами в одну из связанных таблиц включается дополнительное поле (несколько полей) – первичный ключ другой таблицы. Дополнительно включенные поле или поля в этом случае называются внешним ключом соответствующей таблицы.

Кроме таблиц, в модель данных *Microsoft SQL Server 2008* входит еще целый ряд компонентов. Дадим краткую характеристику основным из них.

Индексы

В "[Использование формального аппарата для оптимизации схем отношений](#)" рассматривалось понятие индекса. Здесь понятие индекса вынесено на *логический уровень* для удобства пользователя. *Индексы создаются для ускорения поиска нужной информации и содержат информацию об упорядоченности данных по различным критериям*. Индексирование может быть выполнено по одному или нескольким столбцам. Индексирование может быть произведено в любой момент. Индекс содержит ключи, построенные из одного или нескольких столбцов в таблице или представлении. Эти ключи хранятся в виде структуры *сбалансированного дерева*, которая поддерживает быстрый поиск строк по их ключевым значениям в SQL Server.

Представления

Представление — это виртуальная таблица, содержимое которой определяется запросом. Представление формируется на основе SQL-запроса SELECT, формируемого по обычным правилам. Таким образом, представление есть поименованный запрос SELECT.

Как и настоящая таблица, представление состоит из совокупности именованных столбцов и строк данных. Пока представление не будет проиндексировано, оно не существует в базе данных как хранимая совокупность значений. Строки и столбцы данных извлекаются из таблиц, указанных в определяющем представление запросе и динамически создаваемых при обращениях к представлению. Представление выполняет функцию фильтра *базовых таблиц*, на которые оно ссылается. Определяющий представление запрос может быть инициирован в одной или нескольких таблицах или в других представлениях текущей или других баз данных. Кроме того, для определения представлений с данными из нескольких разнородных источников можно использовать распределенные запросы. Это полезно, например, если нужно объединить структурированные подобным образом данные, относящиеся к разным серверам, каждый из которых хранит данные конкретного отдела организации.

Сборки

Сборки являются файлами динамической библиотеки, которые используются в экземпляре SQL Server для развертывания функций, хранимых процедур, триггеров,

определяемых пользователем статистических вычислений и определяемых пользователем типов.

Ограничения

Ограничения позволяют задать метод, с помощью которого компонент СУБД Database Engine автоматически обеспечивает целостность базы данных. Ограничения задают правила допустимости определенных значений в столбцах и представляют собой стандартный механизм обеспечения целостности. Рекомендуется использовать ограничения, а не триггеры, правила и значения по умолчанию. Оптимизатор запросов также использует определения ограничений для построения высокопроизводительных планов выполнения запросов.

Правила

Правила – еще один специальный механизм, предназначенный для обеспечения целостности базы данных, по функциональности напоминающие некоторые типы ограничений. Microsoft отмечает, что при соответствующей возможности использование ограничений по ряду причин предпочтительнее и, возможно, в будущей версии эта возможность будет удалена.

Значения по умолчанию

Значения по умолчанию определяют, какими значениями заполнять столбец, если при вставке строки для этого столбца значение не указано. Значение по умолчанию могут быть любым выражением, результат которого — константа, например собственно константой, встроенной функцией или математическим выражением.

10.3. Архитектура базы данных. Физический уровень

Физический уровень это представление данных в памяти ЭВМ. Как уже отмечалось в "Физические модели данных (внутренний уровень)" основными понятиями, используемыми для представления структуры хранения (физического уровня) являются понятия файла (физического) и единицы обмена между внешней и оперативной памятью (физической записи или страницы). Рассмотрим, как представлены соответствующие понятия в СУБД Microsoft SQL Server 2008 (<http://msdn.microsoft.com>).

Файлы и файловые группы

На физическом уровне база данных в Microsoft SQL Server 2008 представляется набором файлов операционной системы. Данные и сведения журналов транзакций всегда размещаются в разных файлах. Отдельные файлы используются только одной базой данных. Файловые группы представляют собой именованные коллекции файлов и используются для упрощения размещения данных и выполнения задач администрирования, например резервного копирования и восстановления.

Базы данных SQL Server содержат файлы трех типов:

- **Первичные файлы данных.**

Первичный файл данных является отправной точкой базы данных. Он указывает на остальные файлы базы данных. В каждой базе данных имеется один *первичный файл* данных. Для имени первичного файла данных рекомендуется использовать расширение MDF.

- **Вторичные файлы данных.**

Ко вторичным файлам данных относятся все файлы данных, за исключением первичного файла данных. Базы данных могут вообще не содержать *вторичных файлов* данных, или содержать один или несколько *вторичных файлов* данных. Для имени *вторичного файла* данных рекомендуется использовать расширение NDF.

- **Файлы журналов.**

Файлы журналов содержат все сведения журналов, используемые для восстановления базы данных. В каждой базе данных должен быть по меньшей мере один файл журнала, но их может быть и больше. Для имен файлов журналов рекомендуется использовать расширение MDF, NDF и LDF. Однако эти расширения помогают пользователю идентифицировать различные виды файлов и правильно их использовать.

В SQL Server расположение всех файлов базы данных записывается в *первичный файл* базы данных и в специальную служебную структуру СУБД SQL Server, называемую базой данных master. В большинстве случаев при работе с базой данных компонент СУБД (SQL Server Database Engine) использует сведения о размещении файлов, хранимые в базе данных master. Однако в некоторых случаях (например, при восстановлении базы данных master из копии, при определенным образом проводимом присоединении базы данных) компонент Database Engine использует сведения о расположении файлов из первичного файла, чтобы инициализировать записи о расположении файлов в базе данных master.

Файлы SQL Server имеют два имени:

- `logical_file_name` — имя, используемое для ссылки на физический файл во всех инструкциях Transact-SQL. *Логическое имя файла* должно соответствовать правилам для идентификаторов SQL Server и быть уникальным среди логических имен файлов в соответствующей базе данных.
- `os_file_name` — это имя физического файла, включая путь к каталогу. Оно должно соответствовать правилам для имен файлов операционной системы.

Изначально можно указать максимальный размер каждого файла. Если максимальный размер файла не указан, файлы SQL Server могут автоматически увеличиваться в размерах, превосходя первоначально заданные показатели, пока не займут все доступное место на диске. При определении файла пользователь может указывать требуемый шаг роста. Каждый раз при заполнении файла его размер увеличивается на указанный шаг роста. Если в файловой группе имеется несколько файлов, их автоматический рост начинается лишь по заполнении всех файлов. Затем файлы увеличиваются в размерах по кольцевому списку. Эта функция особенно полезна в случаях, когда SQL Server используется в качестве базы данных, внедренной в приложение, где пользователь не имеет удобного доступа к системному администратору. По мере необходимости пользователь может предоставить файлам возможность увеличиваться в размерах автоматически, тем самым снимая с администратора часть забот по наблюдению за свободным пространством базы данных и по распределению дополнительного пространства вручную.

Из объектов баз данных и файлов можно формировать файловые группы, используемые для решения задач распределения и административного управления. Файлы журналов не могут входить в состав файловых групп. Управление пространством журнала отделено от управления пространством данных. Файл не может входить в состав нескольких файловых групп. Таблицы, индексы и данные больших объектов могут быть ассоциированы с указанной файловой группой. В этом случае все их страницы будут размещены внутри файловой группы; либо таблицы и индексы могут быть секционированы. Данные *секционированных таблиц* и индексов разделяются на блоки, каждый из которых может быть помещен в отдельную файловую группу базы данных. В каждой базе данных одна файловая группа назначается файловой группой по умолчанию. Если при создании таблицы или индекса файловая группа не указывается, предполагается, что все страницы будут распределяться из файловой группы по умолчанию. В каждый момент времени лишь одна файловая группа может быть файловой группой по умолчанию.

Страницы и экстенды

Основной единицей хранилища данных и обмена информацией между внешней и оперативной памятью в SQL Server является страница. Место на диске, предоставляемое для размещения файла данных (MDF- или NDF-файл) в базе данных, логически разделяется на страницы с непрерывным перечислением от 0 до n. Дисковые операции ввода-вывода выполняются на уровне страницы. А именно, SQL Server считывает или записывает целые страницы данных. В SQL Server размер страницы составляет 8 КБ. Это значит, что в одном *мегабайте* базы данных SQL Server содержится 128 страниц. Каждая страница начинается с 96-байтового заголовка, который используется для хранения системных данных о странице. Эти данные включают номер страницы, тип страницы, объем свободного места на странице и идентификатор единицы распределения объекта, которому принадлежит страница.

В файлах данных базы данных SQL Server используется 8 типов страниц (данные с типами данных небольших размеров, данные с типами данных больших размеров, записи индекса, сведения о размещении экстендов, сведения о размещении страниц и доступном на них свободном месте и т. д.).

Для эффективного управления памятью страницы объединяются в экстенды, которые являются основными единицами организации пространства.

Экстенд — это коллекция, состоящая из восьми физически непрерывных страниц или 64 Кб; они используются для эффективного управления страницами. Все страницы хранятся в экстендах. Таким образом, в одном мегабайте базы данных SQL Server содержится 16 экстендов.

Чтобы сделать распределение места эффективным, SQL Server не выделяет целые экстенды для таблиц с небольшим объемом данных. SQL Server имеет два типа экстендов:

- Однородные экстенды принадлежат одному объекту (определенной таблице, индексу и т. д.); все восемь страниц могут быть использованы только этим владеющим объектом.
- Смешанные экстенды могут находиться в общем пользовании у не более восьми объектов. Каждая из восьми страниц в экстенде может находиться во владении разных объектов.

Новая таблица или индекс — это обычно страницы, выделенные из смешанных экстендов. При увеличении размера таблицы или индекса до восьми страниц эти таблица или индекс переходят на использование однородных экстендов для последовательных единиц распределения. При создании индекса для существующей таблицы, в которой содержится достаточно строк, чтобы сформировать восемь страниц в индексе, все единицы распределения для индекса находятся в однородных экстендах. Пример размещения объектов в смешанном и однородном экстендах приводится на [рис. 10.2](#).



Рис. 10.2. Размещение объектов в смешанном и однородном экстендах

Страницы файлов данных

Страницы файлов данных SQL Server нумеруются последовательно; первая страница файла получает нулевой номер (0). Каждый файл базы данных имеет уникальный цифровой идентификатор. Чтобы уникальным образом определить страницу базы данных, необходимо использовать как идентификатор файла, так и номер этой страницы. На [рис. 10.3](#) показаны номера страниц базы данных, содержащей *первичный файл* данных объемом в 4 МБ и *вторичный файл* данных объемом в 1 МБ.

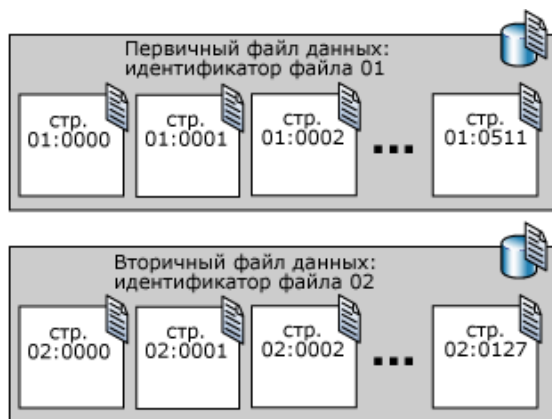


Рис. 10.3. Пример нумерации страниц файлов базы данных

Первая страница каждого файла (страница с номером 0) — это страница заголовка файла; она содержит сведения об атрибутах данного файла. Страницы с номерами 1,2,3 будут описаны ниже.

Организация таблиц и индексов

Таблицы и индексы хранятся в виде коллекции страниц размером 8 КБ.

Страницы таблиц и индексов содержатся в одной или нескольких секциях. *Секция* — это пользовательская единица организации данных. По умолчанию таблица или индекс имеет единственную секцию, которая содержит все страницы таблицы или индекса. Секция располагается в одной файловой группе. Таблица или индекс, имеющие одну секцию, эквивалентны организационной структуре таблиц и индексов предыдущих версий SQL Server.

Если таблица или индекс используют несколько секций, данные секционируются горизонтально, так что группы строк сопоставляются отдельным секциям, основываясь на указанном столбце. Секции могут храниться в одной или нескольких файловых группах в базе данных. Таблица или индекс рассматриваются как единая логическая сущность при выполнении над данными запросов или обновлений. Секция состоит из фрагментов одного или нескольких файлов. Данные внутри фрагмента файла представляются в виде кучи (строки данных хранятся без определенного порядка — последовательное размещение) или *сбалансированного дерева*. Фрагмент файла может иметь один из трех видов: данные с типами небольших размеров (данные IN_ROW_DATA), данные с типами больших размеров (LOB_DATA), данные переменной длины (переполнение строки ROW_OVERFLOW_DATA).

В каждой секции кучи или индекса содержится по крайней мере одна единица распределения IN_ROW_DATA. Кроме того, в зависимости от схемы кучи или индекса, там могут содержаться единицы распределения LOB_DATA или ROW_OVERFLOW_DATA.

Следующая иллюстрация показывает организацию таблицы ([рис. 10.4](#)).

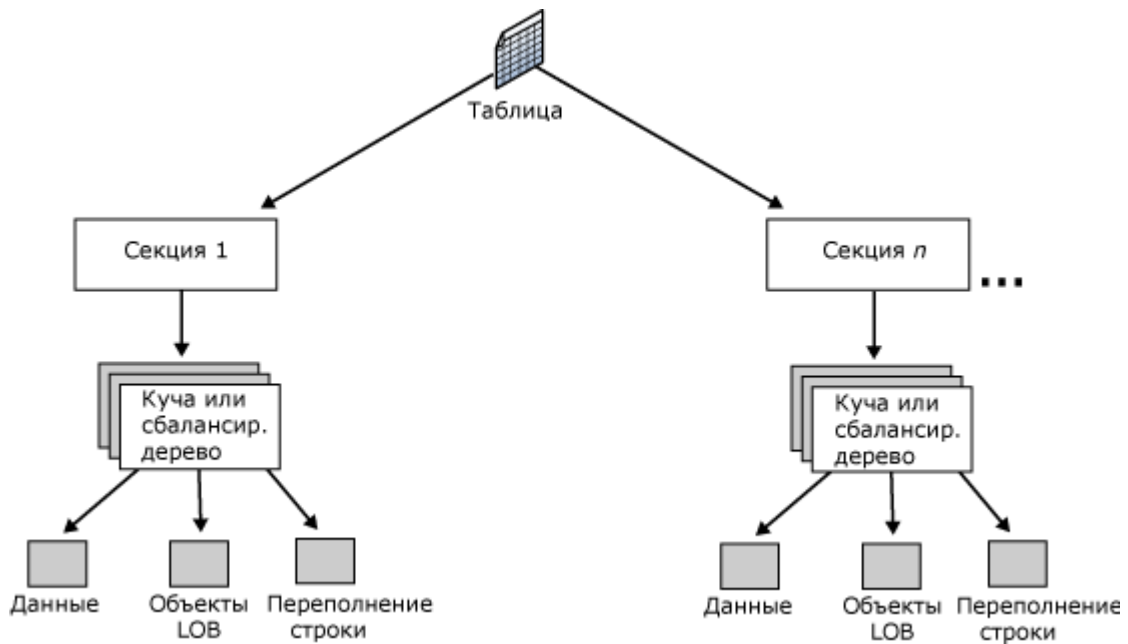


Рис. 10.4. Физическая структура таблицы в базе данных SQL Server

Каждая секция содержит строки данных либо в куче, либо в структуре *кластеризованного индекса*. *Кластеризованный индекс* реализуется в виде структуры индекса *сбалансированного дерева*, которая поддерживает быстрый поиск строк по их ключевым значениям. Страницы в каждом уровне индекса, включая страницы данных на конечном уровне, связаны в двунаправленный список. Однако перемещение из одного уровня на другой выполняется при помощи ключевых значений.

Куча — это *последовательность строк таблицы, которые не имеют кластеризованного индекса*. Строки данных хранятся без определенного порядка, и какой-либо порядок в последовательности страниц данных отсутствует. Страницы данных не связаны в связный список.

Управление работой с экстендами и свободным местом

Структуры данных SQL Server, управляющие использованием экстенда и отслеживанием свободного места, обладают относительно простой структурой. Сведения о свободном месте плотно упакованы, поэтому эти данные содержат относительно небольшое количество страниц. Это приводит к увеличению скорости из-за уменьшения необходимых операций чтения диска для получения сведений о размещении. Также увеличивается вероятность того, что страницы размещения будут оставаться в памяти и повторных операций чтения не потребуется. Большая часть сведений о размещении не связана по цепочке друг с другом. Это упрощает управление сведениями о размещении. Каждое действие по размещению или освобождению страницы может выполняться быстро. Это сокращает конфликты между одновременными задачами использования и освобождения страниц.

SQL Server использует два типа карт для записи сведений об использовании экстендов:

- **Глобальная карта распределения (GAM)**

На GAM-страницах записано, какие экстенды были задействованы. В каждой карте GAM содержатся сведения об использовании 64 000 экстендов или о размещении почти 4 ГБ данных. В карте GAM приходится по одному биту на каждый экстенд в покрываемом им интервале. Если бит равен 1, то экстенд свободен; если бит равен 0, то экстенд задействован.

- **Общая глобальная карта распределения (SGAM)**

На SGAM-страницах записано, какие экстенды в текущий момент используются в качестве смешанных экстендов и имеют как минимум одну неиспользуемую страницу. В каждой карте SGAM содержится сведения об использовании 64 000 экстендов или о размещении почти 4 ГБ данных. В карте SGAM приходится по одному биту на каждый экстенд в покрываемом им интервале. Если бит равен 1, то экстенд используется как

смешанный экстенст и имеет свободную страницу. Если бит равен 0, то экстенст не используется как смешанный экстенст, или он является смешанным экстенстом, но все его страницы используются.

Это дает простые алгоритмы управления экстенстами страниц. Для использования для хранения объекта однородного экстенста компонент СУБД *Database Engine* производит на карте GAM поиск бита 1 и заменяет его на бит 0. Для поиска смешанного экстенста со свободными страницами компонент *Database Engine* производит поиск на карте SGAM бита 1. Для размещения смешанного экстенста компонент *Database Engine* производит на карте GAM поиск бита 1 и заменяет его на бит 0, а затем устанавливает значение соответствующего бита на карте SGAM равным 1. Для освобождения экстенста компонент *Database Engine* устанавливает бит GAM равным 1, а соответствующий бит SGAM равным 0. Внутренние алгоритмы, которые на самом деле используются компонентом *Database Engine*, более сложны, чем это описано в данном подразделе, так как компонент *Database Engine* распространяет данные в базе данных равномерно. Однако даже настоящие алгоритмы упрощаются из-за того, что отпадает необходимость управления цепочками сведений о размещении экстенстов.

Отслеживание свободного места

На страницы *PFS* (Page Free Space) записывается состояние размещения каждой страницы, информация о том, была ли отдельная страница использована или нет, а также количество свободного места на каждой странице. В *PFS* на каждую страницу приходится по одному байту, хранящему информацию о том, была ли страница использована или нет, а если была — то пустая она, или ее заполнение находится в промежутке от 1 до 50 процентов, от 51 до 80 процентов, от 81 до 95 процентов или от 96 до 100 процентов.

После размещения объекта в экстенсте компонент *Database Engine* использует *PFS*-страницы для записи информации о том, какие страницы в экстенсте использованы, а какие свободны. Эти сведения используются компонентом *Database Engine* при выборе новой страницы для размещения объектов. Количеством свободного места на странице можно управлять только в случае кучи и страниц с типами данных "Текст" и "Примечание". Это используется при поиске страницы, обладающей свободным местом, достаточным для сохранения в ней новой добавляемой строки. Для индексов не требуется, чтобы отслеживалось свободное место на странице, так как место, в которое будет вставляться новая строка, назначается значениями *ключа индекса*.

PFS-страница является первой страницей после страницы заголовка файла в файле данных (страница номер 1). Потом следует *GAM*-страница (страница номер 2), а затем *SGAM*-страница (страница номер 3). После первой *PFS*-страницы находится *PFS*-страница размером примерно 8 000 страниц. После первой *GAM*-страницы на странице 2 находится другая *GAM*-страница с 64 000 экстенстов и другая *SGAM*-страница с 64 000 экстенстов находится после первой *SGAM*-страницы на странице номер 3. На [рис. 10.5](#) показана последовательность страниц, используемая компонентом *Database Engine*, для размещения и управления экстенстами.

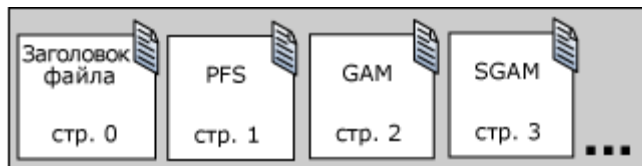


Рис. 10.5. Страницы файла, используемые для размещения и управления экстенстами

Краткие итоги: В лекции рассмотрена архитектура одной из наиболее распространенных клиент-серверных СУБД - Microsoft SQL Server. Описаны основные составляющие архитектуры соответствующей СУБД на разных уровнях абстракции. Рассмотрен логический уровень (уровень модели данных СУБД как средство представления концептуальной модели), включающий следующие понятия: таблицы и типы данных, первичные и внешние ключи, индексы, представления, сборки, ограничения, правила,

значения по умолчанию). Рассмотрен *физический уровень* (внутреннее *представление данных в памяти ЭВМ - физическая структура базы данных*), включающий следующие понятия: файлы и файловые группы, файлы журналов, страницы и экстенды, физическую организацию таблиц и индексов, управление работой с экстендами и памятью).

Цель 11 лекции: дать общее *представление* об основных задачах *программного обеспечения баз данных*, существующих подходов к решению этих задач, в том числе и о *структурированном языке запросов SQL*.

11.1. Основные задачи программного обеспечения баз данных

При работе с реляционными базами данных можно условно выделить две основные задачи:

- собственно работа с базой данных, включающая создание и ведение базы данных (создание структур таблиц, добавление записи в таблицу, удаление записи, обновление, выборка нужной записи);
- создание пользовательских приложений, включающих разработку пользовательского интерфейса по работе с базой данных.

Для решения указанных задач современные *СУБД* в своем составе могут содержать следующие *программные средства*: языки процедурного пошагового программирования, средства визуального программирования (графический *интерфейс*, *диспетчер* проекта, мастера и построители), средства создания объектно-ориентированных приложений. Кроме этого, при разработке пользовательских программ во многих *СУБД* допускается использование других языков программирования, а также использование библиотек разного рода. Так, например, при работе с *СУБД ACCESS* можно использовать *язык программирования ACCESS*, мастер *ACCESS* и *язык программирования VISUAL BASIC*.

При работе с клиент-серверными системами ситуация немного сложнее. Здесь в работе участвуют два типа компьютеров (*сервер* и клиент) и, соответственно, различают клиентское и серверное *программное обеспечение*. Серверное *программное обеспечение* включает *язык программирования*, поддерживающий создание и ведение *базы данных*, также реализацию поступающих от клиентов запросов пользователей к базе данных. Пользовательские приложения создаются и работают на компьютерах-клиентах. Именно эти компьютеры должны иметь, наряду со средствами формирования запросов к базе данных, *средства разработки интерфейса*. В связи с этим, для клиент-серверных *СУБД* *программное обеспечение* разделяется на две части: *программное обеспечение – клиент* и *программное обеспечение – сервер*. Заметим, что наряду с программным обеспечением – клиент, при разработке пользовательских программ в конкретной *СУБД* могут использоваться другие языки программирования, специальные библиотеки, другие системы программирования (определенные для этой *СУБД*). В качестве примера в таблице приводятся возможные варианты использования программного обеспечения для организации клиент-серверного взаимодействия в *СУБД Microsoft SQL Server*.

Таблица 11.1. Возможные варианты использования программного обеспечения в СУБД MS SQL Server	
Средства ведения баз данных на сервере MS SQL	Средства разработки клиентских приложений
Службы SQL-сервер (MS SQL server и др.)	• Программное обеспечение клиента SQL-сервер (Transact SQL, SQL Server Query Analyzer и др.) • MS Access (ODBC) • MS Visual Basic • MS Visual Studio • MS Visual FoxPro • Java (JDBC) • Borland Delphi

	• Borland C++Builder и др. • Библиотеки функций (API, ODBC и др.)

Полное рассмотрение всего спектра программного обеспечения работы *СУБД* очень обширно и выходит за рамки данного пособия. Поэтому в данной работе будет рассмотрены только средства создания и ведения *базы данных*.

11.2. Проблемы создания и ведения реляционных баз данных

При создании *базы данных* и организации работы с ней возникают три основные проблемы:

- собственно создание базы данных (создание таблиц, индексов, ограничений целостности);
- обеспечение безопасности и разграничения доступа;
- организация доступа к элементам таблицы (выборка, редактирование, удаление, добавление).

Первая проблема может быть решена посредством создания в каждой конкретной *СУБД* некоторой утилиты, позволяющей пользователю в определенный момент осуществлять все необходимые действия *по* созданию *базы данных*. Однако это не полностью решает проблему. Такая *утилита* не позволяет создать таблицу динамически во время работы прикладной программы, не позволяет, например, добавить в таблицу сформированный во время работы пользовательской программы столбец. Необходимы средства, дающие возможность формирования во время работы прикладной программы запроса на изменение структуры и содержания *базы данных*. То же самое можно сказать и о решении второй проблемы.

Более подробно рассмотрим возможный *путь* решения третьей проблемы. Организация доступа к базе данных является важнейшей функцией информационной системы. Пользователи постоянно работают с данными. Рассмотрим простой пример. Пусть у нас есть *таблица* *СТУДЕНТ*, хранящая информацию следующего рода:

СТУДЕНТ (Код студента, Фамилия, Имя, Отчество, Дата поступления).

Теперь мы хотим выполнить некоторый *запрос* к базе данных, результатом которого должны стать те строки таблицы *СТУДЕНТ*, для которых дата поступления окажется больше 01.06.2006. Рассмотрим последовательность действий для реализации данного запроса.

1. Получаем доступ к таблице *СТУДЕНТ* и устанавливаем указатель текущей строки на первую строку таблицы.

2. Анализируем поле "Дата поступления" в текущей строке.

3. Если значение "Дата поступления" > "01.06.2006", распечатываем на экране данные об абитуриенте.

4. Если таблица не кончилась, перемещаем указатель текущей строки на следующую строку и переходим к шагу 2, иначе заканчиваем работу.

Любой человек, знакомый с программированием, легко представит себе реализацию подобного алгоритма на любом языке программирования высокого уровня. Вот, в частности, пример реализации на *Object Pascal*:

```
Table.First;
while (not Table.Eof) do
begin
  if FieldByName("Дата поступления").Value >
    "01.06.2006"
  then List.Add(FieldByName("Фамилия").AsString);
  Table.Next;
end;
```

В этом случае разработчик приложения сам организует работу *по* выборке данных, программируя каждое движение *по* таблице (осуществляет навигацию *по* таблице). Такой подход к обработке данных, ориентированный на последовательную работу с отдельными записями, называется *навигационным*. Здесь на конкретном языке программирования мы

описываем процедуру - последовательность действий, необходимых для получения результата. Языки, в которых используется такой подход, называются процедурными. Очевидно, что с возрастанием сложности запроса существенно возрастает сложность процедуры и, соответственно, объем текста программы.

При работе с информационной системой *пользователь* реализует свои запросы к базе данных с помощью разработанных программистами прикладных программ. При навигационном подходе должны быть запрограммированы все возможные запросы. Очевидно, что заранее предугадать все запросы, потребность в которых может возникнуть и запрограммировать их, невозможно. С учетом того, что подавляющее большинство пользователей не владеет навыками программирования, это означает, что объем их действий будет ограничен рамками написанных программ, а именно, теми запросами, реализация которых предусмотрена заранее.

Кроме того, должен учитываться механизм взаимодействия в рамках архитектуры "клиент – сервер". Пользовательская *программа* выполняется на компьютере–клиенте. *Запрос* к базе данных реализуется компьютером – сервером. Необходим механизм формирования в пользовательской программе клиента запроса к базе данных сервера. В этом случае навигационный подход неприемлем. *В связи с этим, для работы с базами данных разработан и используется другой подход, основанный на использовании так называемых языков запросов, которые задают не последовательность необходимых действий, а условия, которым должен удовлетворять результат* (при добавлении столбца, выборке записи, добавлении записи и т. п.). Такой подход решает все три вышеперечисленные проблемы.

С этой целью *разработан и активно используется во всех базах данных – специальный язык запросов SQL*. Особо отметим, что основой языка являются *операции реляционной алгебры*.

Язык SQL (Structured Query Language – структурированный язык запросов) применяется для общения пользователя с реляционной базой данных и состоит из трех частей [[11.2]]:

- DDL (Data Definition Language) – язык определения данных. Предназначен для создания базы данных (таблиц, индексов и т.д.) и редактирования ее схемы.
- DCL (Data Control Language) – язык управления данными. Содержит операторы для разграничения доступа пользователей к объектам базы данных.
- DML (Data Manipulation Language) – язык обработки данных. Содержит операторы для внесения изменений в содержимое таблиц базы данных.

Как видно из написанного выше, SQL решает все рассмотренные ранее вопросы, предоставляя пользователю достаточно простой и понятный *механизм доступа к данным*, не связанный с конструированием алгоритма и его описанием на языке программирования высокого уровня. Так, вместо указания того, как необходимо действовать, *пользователь* при помощи операторов SQL объясняет СУБД, что ему нужно сделать. Далее СУБД сама анализирует текст запроса и определяет, как именно его выполнять.

В архитектуре "клиент – сервер" язык SQL занимает очень важное место. Именно он используется как язык общения клиентского программного обеспечения с серверной СУБД, расположенной на удаленном компьютере. Так, клиент посылает серверу *запрос* на языке SQL, а сервер разбирает его, интерпретирует, выбирает *план выполнения*, выполняет *запрос* и отправляет клиенту результат.

Посмотрим, как выглядит *запрос* на языке SQL, решающий задачу о выборке студентов *по* дате поступления.

```
SELECT Фамилия
FROM Студент
WHERE Дата поступления > "01.06.2006"
```

Может сложиться ложное впечатление, что появление языка SQL является альтернативой языкам программирования высокого уровня. Это не соответствует действительности. Выполнение запроса средствами SQL все равно сводится к работе с

отдельными записями, и от этого никуда не уйти. Важно понимать, что появление языка *SQL* дало, по крайней мере, две новые возможности.

1. Появился новый уровень абстракции между пользователем и СУБД. Этот уровень находится ближе к пользователю, чем уровень программирования на языке высокого уровня, что снижает требования к квалификации пользователей.

2. Многие типовые задачи, возникающие при работе с базами данных, и ранее решаемые каждым программистом по-своему (зачастую дублируя действия другого программиста) решены реализацией языка *SQL*. Таким образом, отпала необходимость самостоятельного решения многих проблем, решенных в СУБД соответствующим образом. Язык *SQL* предоставляет средства для доступа к этим типовым возможностям СУБД.

11.3. Понятие языка *SQL* и его основные части

11.3.1. История возникновения и стандарты языка *SQL*

История возникновения языка *SQL* восходит к 1970 году [[3.1](#)], когда доктор Е.Ф. Кодд предложил реляционную модель в качестве новой модели базы данных. Для доказательства жизнеспособности новой модели данных внутри компании IBM был создан мощный исследовательский проект, получивший название System/R. Проект включал разработку собственно реляционной СУБД и специального языка запросов к базе данных. Так в начале 70-х годов появился первый исследовательский прототип реляционной СУБД. Для этого прототипа разрабатывались и опробовались разные языки запросов, один из которых получил название *SEQUEL* (Structured English Query Language). С момента создания и до наших дней этот язык претерпел массу изменений, но идеология осталась неизменной.

Период с 1979 года (окончание проекта System/R) до настоящего времени характеризуется развитием и совершенствованием языка *SQL* и его постоянно увеличивающейся ролью в индустрии, связанной с созданием и эксплуатацией баз данных. Совершенно очевидно, что язык никогда не получил бы мирового признания, если бы на него не было никаких стандартов. Стандартизация – важная часть технологических процессов конца XX века. Именно наличие разработанных и официально признанных стандартов позволило утвердиться многим современным технологиям (не только в индустрии разработки программного обеспечения, но и в многих других сферах человеческой деятельности). Как обстоит дело со стандартами языка *SQL* и их поддержкой в распространенных СУБД?

Когда ведут речь о стандартах в области, связанной с разработкой программного обеспечения, обычно подразумевают две организации:

- ANSI (American National Standards Institute) – Американский национальный институт стандартов;
- ISO (*International Standards Organization*) – Международную организацию по стандартизации.

Работа над официальным стандартом языка *SQL* началась в 1982 году [8] в рамках комитета ANSI. В 1986 году (обратите внимание, сколько времени ушло на разработку стандарта и согласование деталей!) был утвержден первый вариант стандарта ANSI, а в 1987 году этот стандарт был утвержден и ISO. В 1989 году стандарт претерпел незначительные изменения, но именно этот вариант получил название *SQL-1* или *SQL-89*. В чем особенность *SQL-89*? За время разработки стандарта (1982–1989 гг.) были созданы, представлены на рынке и активно использовались несколько различных СУБД, в которых в том или ином виде был реализован некоторый диалект языка *SQL*. С учетом того, что разработкой стандартов занимались те же люди, кто внедрял *SQL* в СУБД, стандарт *SQL-89* представлял собой плод множества компромиссов, приведших к наличию в нем большого количества "белых пятен", т.е. мест, которые не были описаны, а отданы на усмотрение разработчиков диалекта. В результате чуть ли не все имеющиеся диалекты стали совместимыми со стандартом, но особой пользы это не принесло.

Следующая реализация стандарта была призвана решить эту проблему. В результате длительных обсуждений и согласований в 1992 году был принят новый стандарт ANSI *SQL-2*

или SQL-92. SQL-92, который заполнил многие "белые пятна", впервые добавив в стандарт возможности, еще не реализованные в существующих коммерческих СУБД.

Работа над стандартизацией продолжается и далее. Появились *стандарты SQL-1999, SQL-2003*. Тем не менее, все эти стандарты не решили всех проблем, связанных с наличием нескольких диалектов языка. Как правило, разработчики как игнорировали, так и игнорируют некоторые положения стандарта, с одной стороны, отказываясь реализовывать некоторые его части и, с другой стороны, реализуя то, что отсутствует в стандарте. Несмотря на имеющиеся отличия, все коммерческие СУБД поддерживают некоторое ядро языка, описанное в стандарте, одинаково. Отличий не очень много, они не носят слишком принципиального характера. Хотя каждая СУБД по-прежнему поддерживает свой диалект языка.

В систему управления базами данных Microsoft SQL Server входит язык Transact-SQL, разработанный на основе одного из стандартов SQL.

11.3.2. Достоинства языка SQL

Для ознакомления с достоинствами языка обратимся к соответствующей литературе [[3.1](#)]. Вот некоторые из них:

- межплатформенная переносимость;
- наличие стандартов;
- одобрение и поддержка компанией IBM (СУБД DB2);
- поддержка со стороны компании Microsoft (СУБД SQL Server, протокол ODBC и технология ADO);
- реляционная основа;
- высокоуровневая структура;
- возможность выполнения специальных *интерактивных запросов*;
- обеспечение программного доступа к базам данных;
- возможность различного представления данных;
- полноценность как языка, предназначенного для работы с базами данных;
- возможность динамического определения данных;
- поддержка архитектуры клиент/сервер;
- поддержка корпоративных приложений;
- расширяемость и поддержка объектно-ориентированных технологий;
- возможность доступа к данным в Интернете;
- интеграция с языком Java (протокол JDBC);
- промышленная инфраструктура.

11.3.2. Общая характеристика SQL

Язык запросов SQL основан на операциях реляционной алгебры и, таким образом ориентирован на работу с множествами (отношениями), а не с отдельными записями. Как и в *реляционной алгебре*, операндами языка являются отношения (таблицы), результатами выполнения операции также являются отношения (таблицы). Таким образом, *язык SQL* предназначен для выполнения операций над таблицами, причем как над таблицами в целом (создание, удаление, изменение структуры), так и над данными таблиц (выборка, изменение, добавление и удаление). Отметим, что в явном виде *язык SQL* не является универсальным языком программирования в обычном понимании. В нем отсутствуют операторы условного перехода, организации циклов, позволяющие управлять ходом выполнения программы. Поэтому *язык SQL* относится к классу непроцедурных языков программирования. Это именно *язык запросов к базе данных*, который служит исключительно для организации базы данных и работы с ней. Как уже отмечалось выше, для разработки прикладных программ необходимо использовать другие базовые средства программирования, в который операторы языка SQL будут встраиваться. Языку SQL посвящено большое количество литературы, в том числе и учебников. Подробное изучение языка SQL не входит в задачи настоящего курса, это может занимать отдельный курс. Заметим, что этому языку посвящено большое количество литературы, в том числе и учебников. В связи с этим, здесь

будут изложены только общие сведения о языке, как фундаментальном инструменте работы с базами данных.

Терминология

Под запросом, реализуемым с помощью языка SQL-запросов к базе данных, понимается команда, предназначенная для выполнения (и выполняемая) системой управления базами данных определяемого этой командой действия с базой данных.

Запрос реализуется с помощью операторов языка SQL. Операторы состоят из отдельных логических частей, называемых предложениями. Стандарты языка SQL регламентируют синтаксис операторов. Несмотря на то, что язык SQL работает с реляционной базой данных, вместо термина "отношение" здесь используется термин "таблица", вместо терминов "кортеж" и "атрибут" используются соответственно термины "строка" и "столбец".

Разновидности SQL

Как отмечалось выше, в отличие от "обычных" языков программирования в SQL отсутствует возможность объявления переменных, нет инструкции IF, нет цикла FOR и т.д. Собственно программирование (разработка прикладных программ) на подобном языке практически невозможно. Поэтому к настоящему моменту используются следующие технологии (режимы) работы с базой данных на языке SQL (в некоторых источниках эти технологии называют разновидностями языка SQL):

- формирование непосредственно пользователем запроса на языке SQL в интерактивном режиме (**интерактивный SQL**);
- формирование запроса на языке SQL в прикладной программе (программный или встроенный SQL):
 - статическое формирование запроса (**статический SQL**);
 - динамическое формирование запроса (**динамический SQL**);
 - формирование запроса с помощью библиотек (**API – интерфейсы вызова подпрограмм**).

В интерактивном режиме работы с базой данных: пользователь работает с базой данных в прямом диалоге: вводит запрос на языке SQL – получает результат, вводит другой запрос – получает другой результат и т.д.

Встроенный SQL представляется операторами языка SQL, встроенные в прикладные программы, написанные на других языках программирования (в других программных средах). Это дает возможность работы с базой данных с помощью прикладных программ, написанных на других алгоритмических языках, но требует включения дополнительных средств, обеспечивающих интерфейс между операторами языка SQL и соответствующим языком программирования.

При статическом использовании языка (**статический SQL**) в текст прикладной программы включаются конкретные операторы SQL, и после компиляции исходной программы в выполняемый модуль жестко включаются соответствующие этим операторам функции SQL. Изменения в вызываемых функциях могут здесь определяться только изменениями параметров операторов SQL, инициируемых с помощью переменных языка программирования.

При динамическом использовании языка (**динамический SQL**) формирование SQL-запросов, соответствующие вызовы SQL-функций для обращения к базе данных осуществляется динамически в ходе выполнения программы.

Еще одним способом динамического формирования SQL-запросов в прикладной программе является обращение к соответствующим SQL-функциям с помощью специальных **интерфейсов программирования приложений** (библиотек функций, разработанных для связи прикладной программы и СУБД посредством SQL-запросов).

В настоящем пособии для всех указанных технологий (разновидностей SQL) будут приведены основные идеи и рассмотрены ключевые концепции. Интерактивный SQL будет рассмотрен более подробно, чем программный. Детальное рассмотрение статического,

динамического SQL и различных API-интерфейсов (ODBC, JDBC, DB Library и др.) выходит за рамки нашего курса.

Краткие итоги: В лекции рассматриваются общие принципы организации программного обеспечения работы с реляционными базами данных, включающего

- создание и ведение базы данных;
- создание пользовательских приложений, включающих разработку пользовательского интерфейса по работе с базой данных.

Рассматриваются подходы к организации доступа к данным (навигационный подход и подход, основанный на использовании интерпретируемых языков запросов). Дается общее представление о языке SQL (история возникновения и стандарты языка SQL, достоинства языка SQL, основная терминология, технологии работы).

По языку SQL написано достаточно много литературы. Для более подробного знакомства можно указать, в частности [[3.1](#)] - [[5.4](#)].

РАЗДЕЛ 4. SQL — язык структурированных запросов

Цели занятия.

Определение данных в SQL. Описание учебного проекта. Язык определения данных. Создание таблиц. Модификация структуры таблицы. Удаление таблицы. Язык манипулирования данными. Ввод (добавление) данных в таблицу. Обновление данных. Удаление данных. Язык запросов. Простые запросы. Запросы к связанным таблицам. Декартово произведение. Естественное соединение. Запросы с подзапросами. Теоретико-множественные операции. Представления. Создание представлений. Удаление представлений. Операции над представлениями. Обновление представлений. Индексы

4. Структура лекционного занятия.

№ п/п	Содержание (кратко)	Методы и средства обучения
1	Основные операторы языка SQL.	1. Персональные компьютеры;
2	Программный (встроенный) SQL. Статический SQL. Динамический SQL.	2. Средства доступа в Интернет; 3. Проектор.

5. Содержание лекционного занятия и взаимодействие с аудиторией.

Введение.

Формулирование темы занятия. Использование вводного примера, описание ситуации, демонстрация какого-либо процесса и т. д. (в зависимости от вида, замысла лекции и её структуры)

Тема 4.1. Основные операторы языка SQL. Интерактивный SQL

Цель 12 лекции: дать общую характеристику операторов языка SQL и показать, как записываются основные запросы к базе данных на языке SQL (в интерактивном режиме).

12.1. Общее представление об основных операторах языка SQL

Как уже отмечалось в "Программное обеспечение работы с современными базами данных", все операторы языка SQL разделяются на три составные части: DDL – язык определения данных, DCL – язык управления данными, DML – язык обработки данных.

Приведем примеры основных операторов из вышеуказанных частей (без описания синтаксиса). Описание синтаксиса операторов SQL можно посмотреть в многочисленных книгах по языку SQL, в меню "Справка" конкретных СУБД.

Операторы разграничения доступа пользователей к объектам базы данных (DCL).

GRANT – создание в системе безопасности записи, разрешающей пользователю работать с данными или выполнять определенные операции SQL.

DENY – создание в системе безопасности записи, запрещающей доступ для определенной учетной записи.

Операторы определения данных (язык DDL).

Соответствующие операторы предназначены для создания, удаления, изменения основных объектов модели данных реляционных СУБД: таблиц, представлений, индексов.

CREATE TABLE <имя> – создание новой таблицы в базе данных.

DROP TABLE <имя> – удаление таблицы из базы данных.

ALTER TABLE <имя> – изменение структуры существующей таблицы или ограничений целостности, задаваемых для данной таблицы.

При выполнении аналогичных операций с представлениями или индексами в указанных операторах вместо служебного слова **TABLE** записывается слово **VIEW** (представление) или слово **INDEX** (индекс)

Операторы манипулирования данными (язык DML).

Операторы DML работают с базой данных и используются для изменения данных и получения необходимых сведений.

SELECT – выборка строк, удовлетворяющих заданным условиям. Оператор реализует, в частности, такие операции реляционной алгебры как "селекция" и "проекция".

UPDATE – изменение значений определенных полей в строках таблицы, удовлетворяющих заданным условиям.

INSERT – вставка новых строк в таблицу.

DELETE – удаление строк таблицы, удовлетворяющих заданным условиям. Применение этого оператора учитывает принципы поддержки целостности, поэтому он не всегда может быть выполнен корректно.

12.2 Интерактивный режим работы с SQL (интерактивный SQL)

Соответствующий режим предусматривает непосредственную работу пользователя с базой данных по следующему алгоритму: используя прикладную программу (клиентское приложение) или стандартную утилиту, входящую в СУБД, пользователь:

- устанавливает соединение с БД (подтверждая наличие прав доступа);
- вводит соответствующий оператор SQL, при необходимости в режиме диалога вводит дополнительную информацию;
- инициирует выполнение команды.

Текст запроса поступает в СУБД, которая:

- осуществляет синтаксический анализ запроса (проверяет, является ли запрос корректным);
- проверяет, имеет ли пользователь право выполнять подобный запрос (например, пользователь, у которого определены права только на чтение, пытается что-то удалить);
- выбирает, каким образом осуществлять выполнение запроса – план выполнения запроса;
- выполняет запрос;
- результат выполнения отправляет пользователю.

Схема взаимодействия пользователя и СУБД с использованием интерактивного SQL приводится на рис. 12.1.

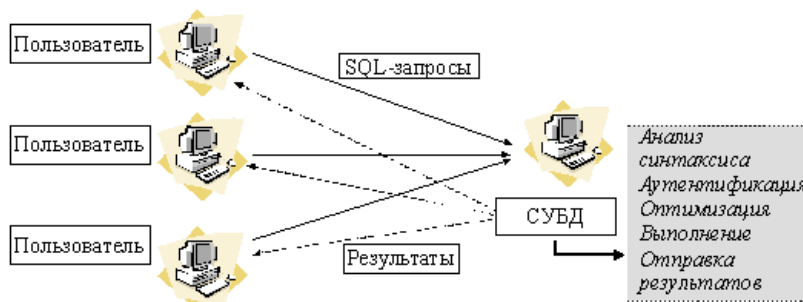


Рис. 12.1. Схема работы интерактивного SQL

12.3. Использование языка SQL для выбора информации из таблицы

Выборка данных осуществляется с помощью оператора **SELECT**, который является самым часто используемым оператором языка SQL. Синтаксис оператора **SELECT** имеет следующий вид:

```

SELECT [ALL/DISTINCT] <список атрибутов>/*
FROM <список таблиц>
[WHERE <условие выборки>]
[ORDER BY <список атрибутов>]
[GROUP BY <список атрибутов>]
[HAVING <условие>]
[UNION<выражение с оператором SELECT>]
  
```

В квадратных скобках указываются элементы, которые могут в запросе отсутствовать.

Ключевое слово **ALL** означает, что результатом будут все строки, удовлетворяющие условию запроса, в том числе и одинаковые строки. **DISTINCT** означает, что в результирующий набор не включаются одинаковые строки. Далее идет *список* атрибутов исходной таблицы, которые будут включены в таблицу-результат. Символ ***** означает, что в таблицу-результат включаются все атрибуты исходной таблицы.

Обязательным ключевым словом является слово **FROM**, за ним следуют имена таблиц, к которым осуществляется *запрос*.

В предложении с ключевым словом **WHERE** задаются условия выборки строк таблицы. В таблицу-результат включаются только те строки, для которых условие, указанное в предложении **WHERE**, принимает значение *истина*.

Ключевое слово **ORDER BY** задает операцию упорядочения строк таблицы-результата *по* указанному списку атрибутов.

В предложении с ключевым словом **GROUP BY** задается *список* атрибутов группировки (разъяснение этого и последующего ключевого слова будет представлено немного позднее).

В предложении **HAVING** задаются условия, накладываемые на каждую группу.

Отдельно отметим, что ключевые слова **FROM**, **WHERE**, **ORDER BY** используются аналогичным образом и в других операторах манипулирования данными языка SQL.

Рассмотрим реализацию запросов для конкретного примера, представленного в "Использование формального аппарата для оптимизации схем отношений" (см. рис. 8.1)

Выдать список всех студентов.

```

SELECT *
FROM student
  
```

или

```

SELECT id_st, surname
FROM student
  
```

Заметим, что если добавить к данному запросу предложение **ORDER BY surname**, то *список* будет упорядочен *по* фамилии. *По* умолчанию подразумевается, что *сортировка* производится *по* возрастанию. Если необходимо упорядочение *по* убыванию, после имени атрибута добавляется слово **DESC**.

Выдать список оценок, которые получил студент с кодом "1".

```
SELECT id_st, mark
FROM mark_st
WHERE id_st = 1
```

Выдать список кодов студентов, которые получили на экзаменах хотя бы одну двойку или тройку.

В предложении *WHERE* можно записывать *выражение* с использованием *арифметических операторов* сравнения (<, >, и т.д.) и логических операторов (*AND*, *OR*, *NOT*) как и в обычных языках программирования.

```
SELECT id_st, mark
FROM mark_st
WHERE ( MARK >= 2 ) AND ( MARK <= 3 )
```

Наряду с операторами сравнения и логическими операторами для составления условий в языке SQL (из-за специфики области применения) существуют ряд специальных операторов, которые, как правило, не имеют аналогов в других языках. Вот эти *операторы*:

- **IN** – вхождение в некоторое множество значений;
- **BETWEEN** – вхождение в некоторый диапазон значений;
- **LIKE** – проверка на совпадение с образцом;
- **IS NULL** – проверка на неопределенное значение.

Оператор **IN** используется для проверки вхождения в некоторое множество значений.

Так, *запрос*

```
SELECT id_st, mark
FROM mark_st
WHERE mark IN (2,3)
```

дает тот же результат, что и вышеуказанный *запрос* (выведет идентификаторы всех абитуриентов, получивших хотя бы одну двойку или тройку на экзаменах).

Того же результата можно добиться, используя оператор **BETWEEN**:

```
SELECT id_st, mark
FROM mark_st
WHERE mark BETWEEN 2 AND 3
```

Выдать список всех студентов, фамилии которых начинаются с буквы А.

В этом случае удобно использовать оператор **LIKE**.

Оператор **LIKE** применим исключительно к символьным полям и позволяет устанавливать, соответствует ли *значение* поля образцу. Образец может содержать специальные символы:

- _** (символ подчеркивания) – замещает любой одиночный символ;
- %** (знак процента) – замещает последовательность любого числа символов.

```
SELECT id_st, surname
FROM student
WHERE surname LIKE 'A%'
```

Очень часто возникает необходимость произвести *вычисление* минимальных, максимальных или средних значений в столбцах. Так, например, может понадобиться вычислить средний балл. Для осуществления подобных вычислений SQL предоставляет специальные *агрегатные функции*:

- **MIN** – минимальное значение в столбце;
- **MAX** – максимальное значение в столбце;
- **SUM** – сумма значений в столбце;
- **AVG** – среднее значение в столбце;
- **COUNT** – количество значений в столбце, отличных от NULL.

Следующий *запрос* считает среднее среди всех баллов, полученных студентами на экзаменах.

```
SELECT AVG(mark)
FROM mark_st
```

Естественно, можно использовать *агрегатные функции* совместно с предложением **WHERE**:

```
SELECT AVG(mark)
FROM mark_st
WHERE id_st = 100
```

Данный *запрос* вычислит средний балл студента с кодом 100 по результатам всех сданных им экзаменов.

```
SELECT AVG(mark)
FROM mark_st
WHERE id_ex = 10
```

Данный *запрос* вычислит средний балл студентов по результатам сдачи экзамена с кодом 10. В дополнение к рассмотренным механизмам язык *SQL* предоставляет мощный аппарат для вычисления агрегатных функций не для всей таблицы результатов запроса, а для разных значений по группам. Для этого в *SQL* существует специальная конструкция **GROUP BY**, предназначенная для указания того столбца, по значениям которого будет производиться группировка. Так, например, мы можем вычислить средний балл по всем экзаменам для каждого студента. Для этого достаточно выполнить следующий *запрос*:

```
SELECT id_st, AVG(mark)
FROM mark_st
GROUP BY id_st
```

Все это, как обычно, может быть совмещено с предложением **WHERE**. При этом, не вдаваясь в тонкости выполнения запроса внутри *СУБД*, можно считать, что сначала выполняется *выборка* тех строк таблицы, которые удовлетворяют условиям из предложения **WHERE**, а потом производится группировка и *агрегирование*.

Приведем *запрос*, который вычисляет средний балл по оценкам, полученным на экзамене с кодом 100, для каждого студента.

```
SELECT id_st, AVG(mark)
FROM mark_st
WHERE id_ex = 100
GROUP BY id_st
```

Заметим, что группировка может производиться более чем по одному полю.

Для запросов, содержащих секцию *GROUP BY* существует важное ограничение: такие запросы могут включать в качестве результата столбцы, по которым производится группировка, и столбцы, которые содержат собственно результаты агрегирования.

Для того чтобы форматировать *вывод*, существуют различные возможности *SQL*. Так, например, допустимым является включение текста в *запрос*. Рассмотрим пример того, как это делается:

```
SELECT 'Средний балл=', AVG(mark)
FROM mark_st
WHERE id_ex = 10
```

В результате данного запроса *пользователь* увидит не просто некоторое число, а число, сопровождаемое поясняющим текстом.

12.4. Использование *SQL* для выбора информации из нескольких таблиц

До сих пор мы рассматривали выбор информации из единственной таблицы. Можно запрашивать информацию из нескольких таблиц, реализуя описанные в соответствующем разделе учебника *реляционные операции*. Стоит упомянуть, что полное рассмотрение темы выходит за рамки данного учебника. Подробно этот вопрос можно изучить при помощи, например, [3.1], [11.2]. Рассмотрим некоторые примеры того, как это делается.

Как правило, в тех случаях когда возникает необходимость выбирать информацию из разных таблиц, они тем или иным образом связаны друг с другом, например отношениями один к многим или один к одному по некоторому полю.

Еще раз вернемся к примеру из "[Физические модели данных \(внутренний уровень\)](#)". Рассмотрим соответствующую *ER-диаграмму* ([рис. 12.2.](#)).

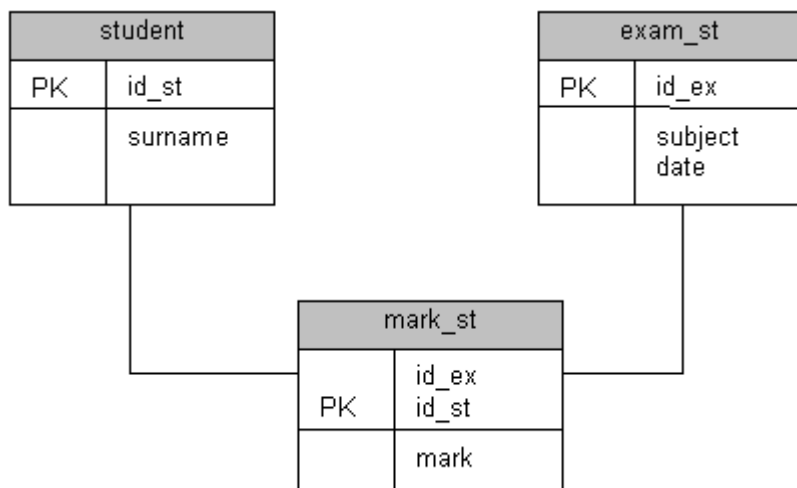


Рис. 12.2. Пример связанных таблиц

В этом примере тоже присутствуют связанные таблицы. Рассмотрим таблицы student, mark_st и exam_st.

Таблица mark_st связана с таблицей exam_st по полю id_ex.

Таблица mark_st связана с таблицей student по полю id_st.

Допустим, требуется распечатать список студентов с оценками, которые они получили на экзаменах. Для этого необходимо выполнить следующий запрос:

```

SELECT student.surname, mark_st.id_ex, mark_st.mark
FROM student, mark_st
WHERE student.id_st = mark_st.id_st
  
```

Отметим следующие изменения по сравнению с запросами к одной таблице.

1. В секции **FROM** указаны две таблицы.

2. Так как таблиц стало больше одной, появилась некоторая неоднозначность при упоминании полей. Так, во многих случаях неизвестно, из какой таблицы из списка FROM брать поле. Для устранения неоднозначности имена полей указываются с префиксом – именем таблицы. Имя таблицы от имени поля отделяется точкой.

3. В предложении **WHERE** указано условие соединения таблиц.

Нетрудно заметить, что использование префиксов-имен таблиц сильно загромождает запрос. Для того чтобы избежать подобного загромождения, используются псевдонимы. Так, можно переписать предыдущий запрос следующим образом:

```

SELECT E.surname, M.id_ex, M.mark
FROM student E, mark_st M
WHERE E.id_st = M.id_st
  
```

12.5. Использование SQL для вставки, редактирования и удаления данных в таблицах

Для добавления данных в таблицу в стандарте SQL предусмотрена команда **INSERT**. Рассмотрим ряд примеров запросов.

```

INSERT INTO mark_st
VALUES (1, 2, 5)
  
```

Данный запрос вставляет в таблицу mark_st строку, содержащую значения, перечисленные в списке **VALUES**. Если не нужно указывать значение какого-то поля, можно присвоить ему **NULL**:

```

INSERT INTO mark
VALUES (1, 2, NULL)
  
```


В случае если необходимо использование для некоторых полей значений *по умолчанию*, SQL позволяет явно указать, какие поля необходимо заполнить конкретными данными, а какие – значениями *по умолчанию*:

```
INSERT INTO mark_st (id_st, id_ex)
VALUES (1, 2)
```

Для удаления данных из таблицы существует команда **DELETE**:

```
DELETE
FROM student
```

Этот *запрос* удаляет все данные из таблицы student.

Можно ограничить *диапазон* удаляемой информации следующим образом:

```
DELETE
FROM student
WHERE surname > 'И'
```

Для обновления данных используется команда **UPDATE**.

```
UPDATE mark_st
SET mark = '5'
WHERE id_st = 100 AND id_ex = 10
```

При помощи этого запроса изменится на "5" оценка у студента с кодом 100 *по экзамену* с кодом 10.

12.5. Язык SQL и операции реляционной алгебры

Язык SQL является средством выражения мощного математического аппарата теории множеств и *реляционной алгебры*. В данном разделе рассматривается *связь* операторов языка SQL с операциями *реляционной алгебры* и теории множеств.

Операция объединения

Средствами языка SQL операция объединения представляется следующим образом:

```
SELECT *
FROM A
UNION
SELECT *
FROM B
```

Операция разности

Средствами языка SQL операция разности представляется следующим образом:

```
SELECT *
FROM A
EXCEPT
SELECT *
FROM B
```

Операция проекции

```
SELECT Fieldi1, ..., Fieldin
FROM A
```

Операция выборки (селекции)

```
SELECT *
FROM A
WHERE (<condition>)
```

Операция пересечения

```
SELECT *
FROM A
INTERSECT
SELECT *
FROM B
```

Операция соединения, эквисоединения

```
SELECT A.Field1, ..., A.Fieldn, B.Field1, ..., B.Fieldm  
FROM A, B  
WHERE (A.Fieldi  $\Theta$  B.Field1)
```

Если Θ – операция "=", то это эквисоединение.

Операция естественного соединения

Пусть есть отношения $A(X_1, \dots, X_n, A_1, \dots, A_m)$ и $B(X_1, \dots, X_n, B_1, \dots, B_r)$.

```
SELECT A.X1, ..., A.Xn, A.A1, ..., A.Am, B.B1, ..., B.Br  
FROM A, B  
WHERE (A.X1 = B.X1) AND ... AND (A.Xn = B.Xn)
```

Краткие итоги: В лекции дается общая характеристика операторов языка SQL, используемых, в частности, для работы с базой данных в интерактивном режиме (создание таблиц, выбор информации из таблиц, добавление, удаление и модификация элементов). Дается понятие интерактивного режима работы с SQL. Рассматриваются основные *операторы SQL*, используемые для манипулирования данными (выбор информации из таблиц, добавление, удаление и модификация элементов). Приводятся примеры записи запросов к базе данных на языке SQL с использованием операторов *select, insert, update, delete*. Рассматривается *связь* между операциями *реляционной алгебры* и операторами языка SQL.

Более подробно материалы лекции рассматриваются в [[3.1](#)] - [[5.4](#)].

Тема 4.2. Использование языка SQL в прикладных программах. Направления развития баз данных

Цель 13 лекции: показать основные возможности формирования запросов к базе данных из прикладных программ.

13.1. Программный (встроенный) SQL

Основная работа с базой данных проводится с использованием прикладных программ, из которых и идут запросы к базам данных. В этом случае *интерактивный режим* работы не может быть использован, текст SQL-запроса должен быть либо включен в прикладную программу (если *запрос* полностью определен заранее), либо формироваться в процессе работы прикладной программы.

Программный SQL предназначен для того, чтобы *встраивать SQL-запросы в прикладную программу, написанную на одном из языков программирования*. При этом возникают следующие вопросы:

- компилятор с алгоритмического языка должен иметь возможность выделения в тексте прикладной программы последовательность операторов SQL.
- компилятор должен объединять возможности языка программирования высокого уровня (переменные, ветвления, циклы) и возможности SQL (запросы на языке, близком к естественному).

Решение этих проблем частично описано в стандарте SQL.

Рассмотрим *алгоритм* выполнения SQL-запросов в интерактивном режиме работы. Легко видеть, что *пользователь* вынужден ожидать результатов выполнения запроса в течение всего времени работы реализации SQL-запроса. Если через некоторое время пользователю снова нужно будет выполнить тот же самый *запрос*, СУБД вновь проделает те же самые действия, что и при предыдущем обращении. Налицо некоторое несовершенство механизма:

- одни и те же этапы выполняются каждый раз заново для одинаковых запросов;
- СУБД не может обрабатывать *интерактивные запросы* с опережением.

Решение подобных проблем очевидно – часть действий по обработке запроса необходимо выполнять один раз, сохранять результат в некотором виде, а потом использовать

столько раз, сколько необходимо. Эта идея является одной из основных идей программного SQL. Таким образом, программный SQL позволяет:

- использовать операторы *интерактивного SQL* в тексте программы на языке программирования высокого уровня;
- наряду с операторами *интерактивного SQL* использовать новые специальные конструкции, дополняющие SQL и увеличивающие его возможности;
- для передачи параметров в запрос использовать в тексте запроса переменные, объявленные в программе;
- для возврата в программу результатов запроса использовать специальные конструкции, отсутствующие в интерактивном SQL;
- осуществлять компиляцию запросов совместно с программой, обеспечивая впоследствии согласованную работу программы и СУБД. Заранее (на этапе компиляции) выполнять действия по анализу и *оптимизации запросов*, экономя время, затрачиваемое на этапе выполнения программы.

На настоящий момент используются три варианта *встраивания* запросов на языке SQL в прикладную программу (программного SQL): *статический SQL*, *динамический SQL* и метод, основанный на различных *интерфейсах программирования приложений* (API). Рассмотрим соответствующие варианты.

13.2. Статический SQL

Статический SQL – разновидность программного SQL, предназначенная для *встраивания SQL-операторов в текст программы на языке программирования высокого уровня*.

Основная особенность статического SQL определяется его названием: встраиваемые запросы должны быть четко определены на стадии написания прикладной программы, так как именно конкретный текст запросов вставляется в прикладную программу.

Рассмотрим два основных этапа, связанных с работой *статического SQL*, – *компиляция программы и работа (выполнение) программы*.

Схема компиляции и сборки программы выглядит следующим образом ([рис. 13.1](#)):

- Программа, включающая операторы языка программирования высокого уровня (ЯПВУ) вместе с операторами SQL, подается на вход специального препроцессора, который выделяет из нее части, связанные с SQL.
- Вместо инструкций *встроенного SQL* препроцессор подставляет вызовы специальных *функций СУБД*. Библиотеки таких функций для связи с языками программирования существуют для всех распространенных серверных СУБД. Стоит особо отметить, что эти библиотеки имеют "закрытый" интерфейс, т.е. разработчики библиотеки могут менять его по своему усмотрению, соответственно обновив препроцессор. Все это говорит о том, что программист не должен вмешиваться в этот процесс.
- Сами инструкции SQL препроцессор выделяет в отдельный файл.
- Программа поступает на вход обычного компилятора языка программирования, после чего получают объектные модули. Далее эти объектные модули вместе с *библиотеками СУБД* собираются в один исполняемый модуль – приложение.
- Наряду с этими операциями происходит работа с файлом, содержащим SQL-инструкции. В литературе этот модуль часто носит название "модуль запросов к базе данных" (Database Request Module, DBRM) [1]. Обработку этого модуля осуществляет специальная утилита, которая обычно носит название BIND. Для каждой инструкции SQL утилита выполняет следующие действия:
 - осуществляет синтаксический анализ запроса (проверяет, является ли запрос корректным);
 - проверяет, существуют ли в базе данных те объекты, на которые ссылается запрос;
 - выбирает, каким образом осуществлять выполнение запроса – *план выполнения* запроса;

- Все планы выполнения запросов сохраняются в СУБД для последующего использования.



Рис. 13.1. Схема компиляции программы с встроенными инструкциями статического SQL

Схема выполнения программы выглядит следующим образом (рис. 13.2.):

Программа запускается на выполнение обычным образом. При необходимости выполнить запрос программой осуществляется вызов специальной функции СУБД, которая отыскивает уже сформированный ранее план выполнения запроса. СУБД выполняет запрос в соответствии с выбранным планом. Результат выполнения запроса поступает в приложение.

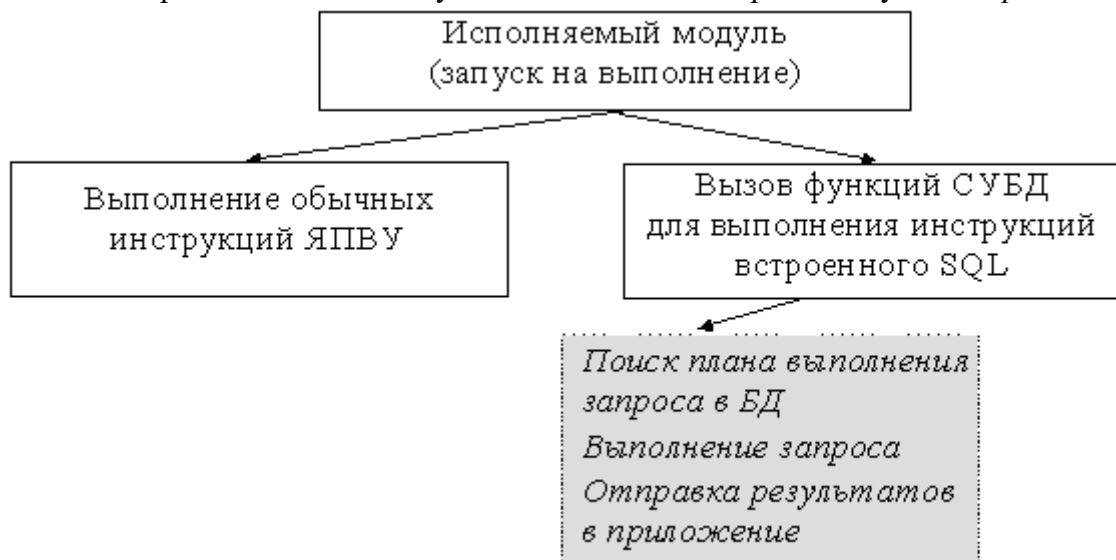


Рис. 13.2. Схема выполнения программы с встроенными инструкциями статического SQL

Для реализации вышеуказанных схем статический SQL должен содержать дополнительные операторы (по сравнению с интерактивным SQL), позволяющие компилятору выделить в тексте программы SQL-запросы, объявлять используемые в этих

запросах таблицы, объявлять переменные для обработки ошибок, как результатов реализации запросов и т. п. Основные команды *статического SQL* приводятся в следующей таблице.

Таблица 13.1. Основные команды <i>статического SQL</i>	
EXEC SQL	Спецификатор, указывающий, что следующая за ним инструкция является инструкцией <i>встроенного SQL</i>
;	В языке С – признак окончания инструкции <i>встроенного SQL</i>
DECLARE TABLE	Объявляет таблицу, которая потом будет использоваться в инструкциях <i>встроенного SQL</i>
SQLCODE	Переменная для обработки ошибок
SQLSTATE	Переменная для обработки ошибок
GET DIAGNOSTICS	Инструкция для обработки ошибок
WHenever SQLERROR SQLWARNING NOT FOUND GOTO CONTINUE	Набор совместно используемых инструкций для упрощения обработки ошибок
BEGIN DECLARE SECTION END DECLARE SECTION	Инструкции для определения области, в которой будут объявлены переменные, впоследствии используемые в запросах SQL
INTO	Используется в операторе SELECT для указания переменной, в которую необходимо поместить результат выполнения запроса
DECLARE CURSOR	Курсор – специальный инструмент, предназначенный для обработки результатов запроса, содержащих более одной строки. Работа с курсором похожа на работу с файлами. Данная инструкция служит для создания курсора и связывания его с конкретным запросом
OPEN	Команда, открывающая курсор и побуждающая СУБД начать выполнение запроса. Устанавливает курсор перед первой строкой результата запроса
FETCH	Команда, перемещающая указатель текущей строки (курсor) на следующую строку. В некоторых СУБД <i>истандарте SQL-92</i> реализованы разные формы команды FETCH, перемещающие курсор на произвольную строку результатов запроса
CLOSE	Закрывает курсор и прекращает доступ к результатам запроса

Использование описанной выше схемы компиляции/сборки/выполнения программы позволяет:

- использовать SQL совместно с программой на языке программирования высокого уровня;
- заранее осуществлять проверку синтаксиса запросов и оптимизацию их выполнения (выбор плана). Понятно, что проверка синтаксиса выполняется быстро, но выбор *плана выполнения* – весьма трудоемкая процедура. Тот факт, что она выполняется один раз на этапе компиляции, позволяет говорить о существенном уменьшении накладных расходов.

Однако статическая разновидность программного SQL имеет некоторые существенные ограничения. Так, переменные в запросах могут использоваться только в тех местах, где в запросах обычно стоят *константы*. Например, нельзя задавать имя таблицы, из которой производится *выборка*, а также названия столбцов, как *параметр*. В связи с этим при использовании статического варианта вложенного (программного) SQL необходимо на этапе написания программы точно знать состав запросов, которые необходимо будет выполнять в прикладной программе. Во многих случаях это ограничение является существенным. Для его устранения была введена новая разновидность программного SQL – *динамический SQL*. Рассмотрим кратко основные идеи *динамического SQL*.

13.3. Динамический SQL

Динамический SQL – разновидность программного SQL, предназначенная для встраивания SQL-операторов в текст программы на языке программирования высокого уровня, допускающая динамическое формирование и выполнение запросов во время работы программы.

История возникновения *динамического SQL* во многом связана с компанией IBM, внедрившей этот мощный инструмент в свою СУБД DB2. Стандарты SQL, в частности SQL-1, не поддерживали *динамического SQL*. Лишь в 1992 году в стандарт SQL-2 были включены спецификации *динамического SQL*. Основной концепцией *динамического SQL* является следующее утверждение: встроенная инструкция SQL не записывается в исходный текст программы, вместо этого программа формирует текст инструкции во время выполнения в одной из своих областей данных, а затем передает сформированную инструкцию в СУБД для динамического выполнения [[3.1]].

Напомним, что при использовании *статического SQL* схема реализации подразумевала два этапа – компиляцию программы и выполнение программы. При этом на этап компиляции ложилась основная нагрузка. Именно здесь решались вопросы проверки, разбора и *оптимизации запросов*, поскольку *запрос* был заранее известен. Совершенно очевидно, что подобную двухэтапную схему нельзя реализовать для *динамического SQL*, так как на этапе компиляции программы *запрос* неизвестен. Поэтому проверку, разборку и *оптимизацию запросов* здесь приходится выполнять непосредственно во время работы программы. Таким образом, если эти операции в статическом SQL выполнялись *вовремя компиляции* один раз, то в *динамическом SQL* они будут выполняться столько раз для одного запроса, сколько раз он будет сформирован в процессе работы прикладной программы. Это определяет существенный недостаток *динамического SQL* – низкую *производительность* по сравнению со статическим. Достоинство *динамического SQL* в том, что он позволяет формировать *запрос* к базе данных во время работы программы, реагируя на те или иные произошедшие события. Такая возможность является жизненно важной для клиент-серверной и *трехзвенной архитектур*, в которых структура *базы данных* и деловые правила имеют тенденцию к изменению, что требует определенной гибкости при организации процесса обработки данных.

Учитывая относительно низкую *производительность динамического SQL*, представляется правильным, там, где только возможно, рекомендовать использование статической разновидности SQL, применяя аппарат *динамического SQL* где это действительно необходимо.

Динамический SQL также должен содержать дополнительные операторы (по сравнению с *интерактивным SQL*). Основные операторы *динамического SQL* приводятся в следующей таблице.

Таблица 13.2. Основные команды <i>динамического SQL</i>	
EXECUTE IMMEDIATE	Немедленное выполнение инструкции
PREPARE	Подготовка инструкции к выполнению
EXECUTE	Выполнение подготовленной ранее инструкции

DESCRIBE	Специальная команда, участвующая при возврате результата выполнения инструкций <i>динамического SQL</i>
DECLARE CURSOR	Разновидность инструкции <i>DECLARE CURSOR</i> , применявшейся ранее в рамках <i>статического SQL</i> , содержащая вместо запроса его имя (связанное с запросом при помощи инструкции PREPARE)
OPEN FETCH CLOSE	Разновидности инструкций для работы с курсором в <i>динамическом SQL</i>

Рассмотрим схему функционирования *динамического SQL* (рис. 13.3). Схема предусматривает одноэтапное и двухэтапное выполнение инструкций.

Одноэтапное выполнение инструкций осуществляется командой **EXECUTE IMMEDIATE**.

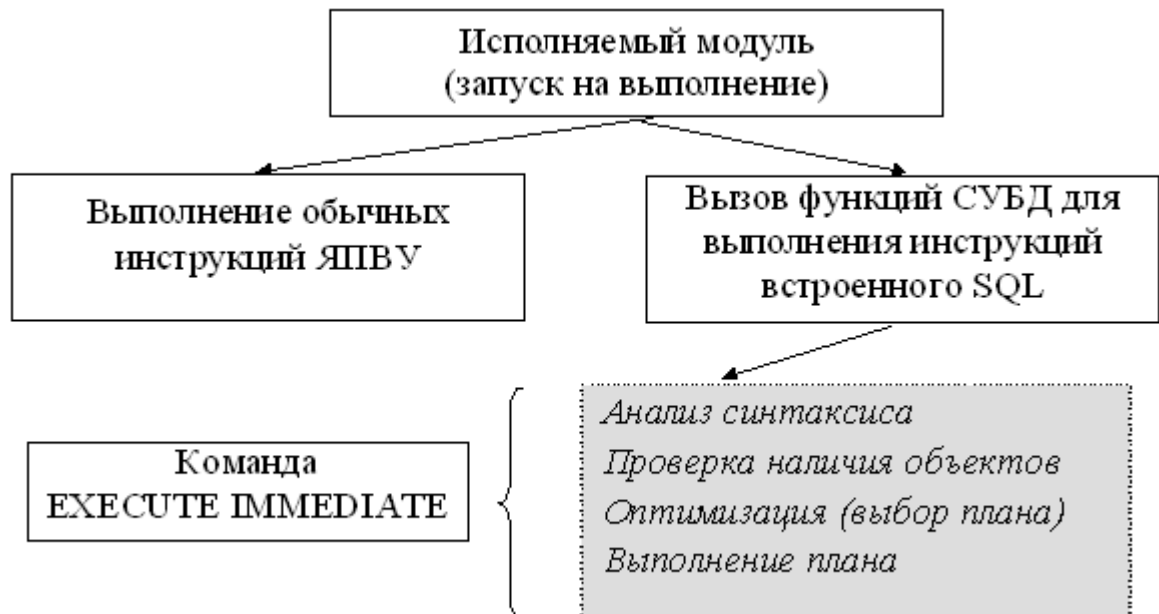


Рис. 13.3. Схема выполнения программы со встроенными инструкциями динамического SQL с применением одноэтапной схемы

Схема выполнения инструкции подразумевает:

- динамическое формирование команды SQL в строковом виде во время работы программы;
- передачу строкового вида инструкции в СУБД при помощи команды **EXECUTE IMMEDIATE**;
- выполнение инструкции системой управления БД, включающее синтаксический анализ, проверку параметров, оптимизацию (выбор плана) и выполнение этого плана.

Основные проблемы одноэтапной схемы заключаются в том, что она не позволяет выполнять инструкции **SELECT** (ибо нет средств для возврата в *приложение* результатов запроса) и приводит к нерациональному расходованию вычислительных ресурсов (т.к. при повторном выполнении той же инструкции вновь будет затрачено время на все те же действия по ее интерпретации и выполнению).

Двухэтапное выполнение инструкций основано на следующем соображении: скорее всего, команда *динамического SQL* в таком виде, как она поступает на выполнение, будет выполняться неоднократно. При этом могут меняться какие-то конкретные детали. А это значит, что инструкцию можно параметризовать. Использование параметризованных инструкций позволяет сделать схему выполнения двухэтапной, разделив процесс на "подготовку инструкции" и "выполнение инструкции" (рис. 13.4).



Рис. 13.4. Схема выполнения программы со встроенными инструкциями динамического SQL с применением двухэтапной схемы

На этапе подготовки можно осуществить синтаксический анализ инструкции, интерпретировать ее и подготовиться к выполнению, выбрав план выполнения.

На этапе выполнения СУБД подставляет значения параметров (полученные из программы) и использует сформированный ранее план выполнения для достижения результата.

При этом реализуется идея однократного выполнения тех действий, которые можно выполнить один раз. Так, подготовленная одна инструкция может быть выполнена десятки раз с разными параметрами.

13.4. Интерфейсы программирования приложений (API). DB-Library, ODBC, OCI, JDBC

Как замечено выше, программный SQL отличается от обычной, интерактивной формы наличием некоторых специальных инструкций, а также механизмом трансляции и выполнения запросов. Таким образом, для применения программного SQL в тексте своих программ программистам необходимо ознакомиться с некоторым специфическим набором инструкций. Стоит заметить, что в разных СУБД эти наборы инструкций, вообще говоря, могут несколько отличаться друг от друга. В результате возникает некоторая проблема, связанная с переносимостью программы.

Наряду с описанным выше механизмом существует и активно применяется еще один подход, связанный с наличием специальных интерфейсов – API (application programming interface – интерфейс программирования приложений). Эти API представляют собой библиотеки функций, разработанные для обеспечения связи прикладной программы с СУБД посредством выполнения SQL-запросов. Прикладная программа вызывает специальные функции библиотеки для передачи в СУБД SQL-запроса в текстовом виде и для получения результатов выполнения запросов, а также различной служебной информации.

Применение подобного подхода приводит к тому, что программистам более не требуется изучать специальные наборы инструкций SQL, а необходимо лишь изучить специальную библиотеку функций. С учетом того, что механизм использования API является

широко используемым и стандартным подходом (чего только стоит использование мощного аппарата *Windows API*), для специалистов нет ничего нового в изучении еще одной библиотеки, в данном случае – для общения с *СУБД*.

Кроме этого, *программа*, содержащая вызовы некоторых функций специализированной библиотеки, ничем не отличается по схеме компиляции и выполнения от обычной программы. Так, подобная *программа* не требует применения специализированного препроцессора с механизмом раздельной компиляции. Может показаться, что подход, связанный с использованием библиотек *API*, является наиболее прогрессивным, на самом же деле такой *вывод* вряд ли верен. Так, на настоящий момент очень активно используются и *динамический SQL* и библиотеки *API*. В каждом из этих подходов существуют свои достоинства, недостатки и границы разумной применимости. Как обычно, выбор того, каким из подходов воспользоваться, лежит на административной группе разработчиков *базы данных*, которая принимает решения в зависимости от особенностей конкретной задачи и имеющихся специалистов.

В данном разделе рассматривается подход, основанный на *интерфейсе программирования приложений*.

Посмотрим, как работают прикладные программы, использующие различные *API*. Принципы работы разных библиотек аналогичны. Схема работы приложения совместно с *SQL API* выглядит следующим образом [[3.1](#)]:

- программа получает доступ к базе данных путем вызова одной или нескольких *API-функций*, подключающих программу к *СУБД* и к конкретной базе данных;
- для пересылки инструкций *SQL* в *СУБД* программа формирует инструкцию в виде текстовой строки и затем передает эту строку в качестве параметра при вызове *API-функции*;
- программа вызывает выполнение *API-функции* для проверки состояния переданной в *СУБД* инструкции и обработки ошибок;
- если инструкция *SQL* представляет собой запрос на выборку, то, вызывая *API-функции*, программа записывает результаты запроса в свои переменные; обычно за один вызов возвращается одна строка или столбец данных;
- свое обращение к базе данных программа заканчивает вызовом *API-функции*, отключающей ее от *СУБД*.

Из имеющихся для реализации *SQL-запросов* интерфейсов *API* на настоящий момент выделилось несколько библиотек, "стандартных" в том смысле, что они активно применяются множеством разработчиков по всему миру. Данное пособие не является подробным руководством по всем этим библиотекам. Более того, для их профессионального освоения необходимо серьезное изучение соответствующей литературы. В рамках данного курса мы лишь приведем обзор этих библиотек, рассмотрим основные заложенные в них идеи. Подробно эти *SQL API* описаны, например в [[3.1](#)].

Протокол ODBC

ODBC (Open Database Connectivity – открытый доступ к базам данных) – разработанный компанией Microsoft универсальный интерфейс программирования приложений для доступа к базам данных [[3.1](#)].

Основной целью разработки протокола ODBC считается стандартизация механизмов взаимодействия с различными *СУБД*. Основная проблема, связанная с разработкой приложений, взаимодействующих с базами данных на основе специальных *SQL API*, состояла в том, что каждая *СУБД* имела собственный программный интерфейс доступа, каждый из них имел свои особенности и функционировал не совсем так, как другие. В связи с этим разработка приложения существенно зависела от используемой *СУБД*. Компания Microsoft сделала важный шаг для решения этой проблемы. Основная идея заключалась в разработке универсального интерфейса на уровне семейства операционных систем Windows, который мог бы быть поддержан в разных *СУБД*.

Рассмотрим кратко структуру программного обеспечения ODBC [[3.1](#)]:

- **интерфейс вызовов функций ODBC**: это так называемый верхний уровень ODBC, содержащий API, который и используется непосредственно приложениями. Данный API реализован в виде библиотеки динамической компоновки DLL и входит в состав операционной системы Windows;

- **драйверы ODBC**: это так называемый нижний уровень ODBC, содержащий набор драйверов для СУБД, поддерживающих протокол ODBC. В рамках технологии для каждой СУБД может быть разработан соответствующий ODBC-драйвер, который будет являться промежуточным звеном между прикладной программой и СУБД, транслируя вызовы *функций СУБД* в вызовы внутренних специализированных *функций СУБД*. Таким образом решается проблема стандартизации. Для многих современных СУБД существуют специализированные *драйверы ODBC*, отдельно устанавливаемые в операционную систему;

- **диспетчер драйверов ODBC**: данный программный механизм представляет средний уровень ODBC, управляя процессом загрузки необходимых драйверов.

Схема выполнения программы с использованием протокола ODBC для доступа к данным приводится на [рис. 13.5](#).

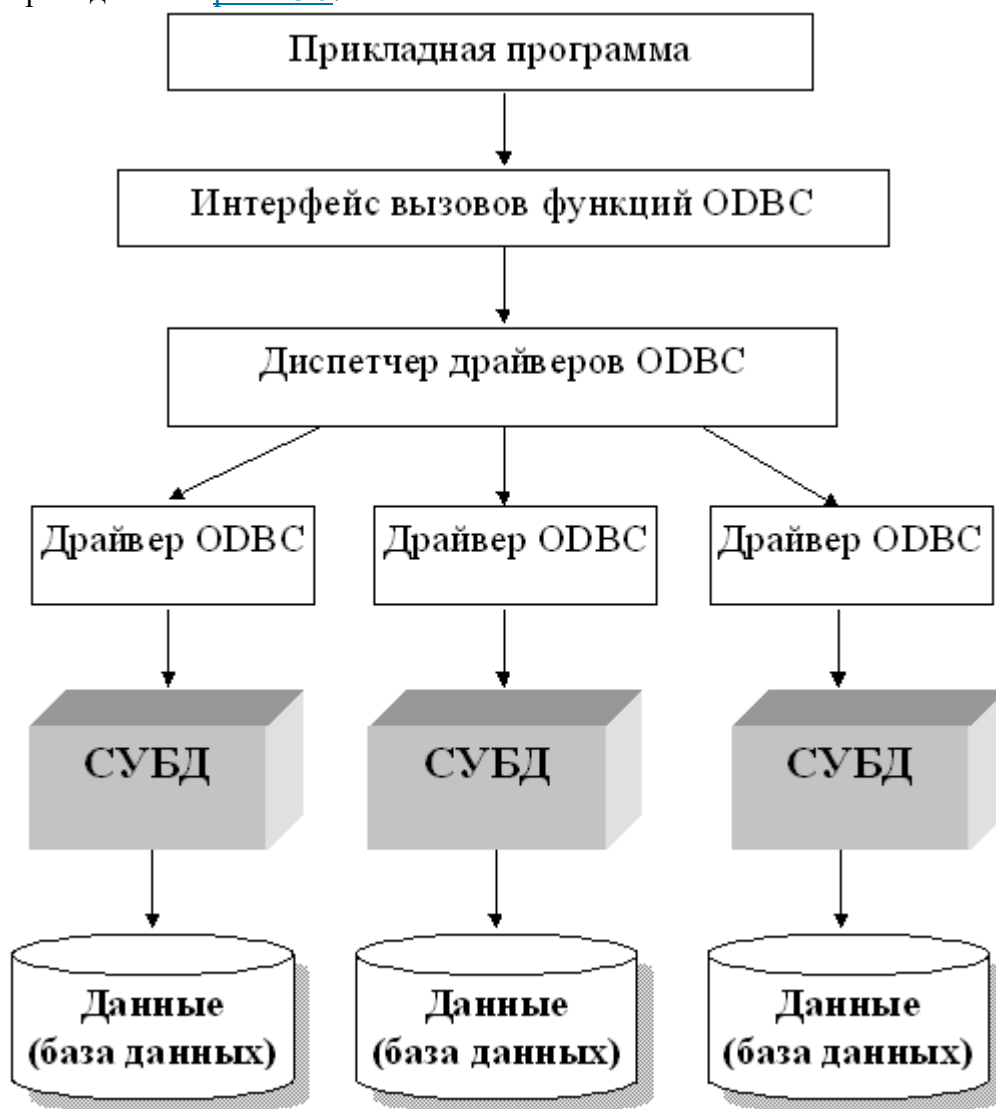


Рис. 13.5. Схема выполнения программы с использованием протокола ODBC для доступа к данным

Перечень некоторых базисных функций ODBC API приводится в следующей таблице.

Таблица 13.3. Базисные функции ODBC API		
Назначение	Функция	Описание

Соединение с источником данных	SQLAllocEnv	Получает указатель окружения. Одно окружение может служить для создания нескольких соединений.
	SQLAllocConnect	Получает указатель соединения.
	SQLConnect	Соединяется с указанным драйвером, используя имя источника данных, идентификатор пользователя и пароль.
Подготовка SQL запросов	SQLAllocStmt	Размещает указатель выражения.
	SQLPrepare	Подготавливает SQL выражение для дальнейшего использования.
	SQLGetCursorName	Возвращает имя, связанное с указателем выражения.
	SQLSetCursorName	Устанавливает имя курсора.
Выполнение запросов	SQLExecute	Выполняет заранее подготовленный запрос.
	SQLExecDirect	Выполняет запрос.
Выборка результатов и информации о результатах	SQLRowCount	Возвращает количество записей, задействованных в операциях вставки, удаления, модификации.
	SQLNumResultCols	Возвращает количество колонок в выбранном наборе данных.
	SQLDescribeCol	Описывает колонку в выбранном наборе данных.
	SQLColumns	Описывает атрибуты колонки в выбранном наборе данных.
	SQLBindCol	Присваивает место в памяти для колонки в выбранном наборе данных и указывает ее тип данных.
	SQLFetch	Возвращает несколько наборов данных.

Протокол JDBC

JDBC (Java Database Connectivity) представляет собой *API* для выполнения SQL-запросов к базам данных из программ, написанных на языке Java [[3.1](#)].

Рассмотрим основные принципы JDBC.

С развитием глобальных сетей, в частности Интернета, и всех сопутствующих технологий стали появляться новые языки, специально предназначенные для работы в новых условиях. Одним из таких языков является язык *программирования Java*. В настоящее время Интернет-приложения занимают существенное место на рынке, работая в рамках 2-, 3- и *многозвенной архитектуры*. При этом значение языка Java как средства создания приложений, работающих с базами данных, существенно возрастает. Именно это и явилось одной из основных причин разработки нового программного интерфейса – JDBC. Первоначально интерфейс JDBC был разработан компанией Sun Microsystems, в настоящий момент этот *API* поддерживается всеми ведущими коммерческими СУБД.

Известно несколько различных версий JDBC. Так, версия 1.0 содержала некоторые средства доступа к данным:

- диспетчер драйверов (для подключения к разным СУБД);

- механизм управления сеансами (для одновременной работы с несколькими СУБД);
- механизм передачи инструкций SQL на выполнение в СУБД;
- механизм работы с курсорами (для передачи результатов выполнения запросов из СУБД в приложение).

Этот перечень определенным образом напоминает аналогичный функциональный аппарат протокола ODBC.

Версия JDBC 2.0 содержит существенные отличия. Так, вследствие увеличения возможностей интерфейса было проведено его идеологическое разделение на две основные части: Core API (основные возможности) и Extensions API (так называемые расширения).

В [[3.1](#)] указаны следующие возможности JDBC:

- **Пакетные операции.** Программа на Java может осуществить обновление базы данных в пакетном режиме, т.е. одна функция JDBC может добавить в базу данных несколько записей, что положительно сказывается на производительности программ.
- **Курсоры с произвольным доступом.** В JDBC 2.0 существует средство, позволяющее перемещаться по результатам запроса произвольным образом.
- **Обновляемые курсоры.** В JDBC 2.0 курсоры, наряду с функцией возврата результата запроса, используются и при обновлении базы данных. Обновления производятся при добавлении или изменении одной из строк в результатах запроса.
- **Организация связанного пула.** Несколько программ на языке Java могут пользоваться совместным доступом к базе данных, уменьшая затраты на подключения к базе данных и отключения от нее. Данный перечень можно продолжить (распределенные транзакции, поддержка JNDI и т.д.).

Версия JDBC 3.0 появилась совсем недавно и содержит такие новации, как объектно-реляционные расширения SQL и улучшенные механизмы обработки транзакций. Архитектура JDBC берет свое начало от ODBC и в существенной части повторяет ее, поэтому схема выполнения программы на Java с использованием протокола JDBC для доступа к данным полностью аналогична схеме на [рис. 13.5](#) (слова ODBC заменяются на слова JDBC). В отличие от ODBC, драйверы JDBC подразделяются на четыре типа. Основные отличия между этими типами связаны с местонахождением API СУБД (на клиентской или серверной СУБД) и способом доступа к базе данных (через собственный API СУБД или через ODBC).

Библиотека DB-Library

Библиотека DB-Library реализует *интерфейс программирования приложений* для совместной работы с широко распространенной СУБД Microsoft SQL Server. Данная библиотека является весьма обширной и содержит более 100 функций. Основными из них являются:

- **dblogin(); dbopen()** – подключение к БД;
- **dbopen(); dbexit()** – установка/разрыв соединения с БД;
- **dbcmd()** – передача инструкции (пакета инструкций) SQL в СУБД в текстовом виде;
- **dbSQLexec()** – требование к СУБД выполнить текущий пакет инструкций;
- **dbcancel()** – прекращение выполнения пакета инструкций SQL;
- **dbresults()** – получение результатов выполнения очередной инструкции SQL в текущем пакете;
- **dbbind(), dbdata(), dbnextrow(), dbnumcols(), dbdatlen()** и др. – обработка результатов запросов на выборку данных.

Логика работы прикладной программы, обрабатывающей данные, хранящиеся в базе данных под управлением Microsoft SQL Server, выглядит следующим образом:

- при помощи указанных выше функций (**dblogin(), dbopen()**) прикладная программа формирует сведение об авторизации и пытается установить соединение с СУБД;
- при помощи СУБД программа открывает конкретную базу данных, с которой будет происходить работа (**dbopen()**);

- при помощи специальной функции (**dbcmd()**) программа передает в СУБД текст SQL-инструкции, которую далее необходимо будет выполнить; в библиотеке DB-Library поддерживается так называемый пакетный режим работы. Данный режим подразумевает возможность создания пакетов инструкций. Так, вызывая функцию **dbcmd()** несколько раз, вы можете передать в СУБД текст нескольких команд SQL, которые впоследствии будут выполнены как одна команда;
- используя функцию **dbSQLexec()**, программа вызывает выполнение инструкций, переданных ранее при помощи вызовов функций **dbcmd()** ;
- вызывая функцию **dbresults()**, программа может определить, удалось ли СУБД выполнить очередную инструкцию (как правило, число вызовов **dbresults()** соответствует числу инструкций в очередном пакете);
- в случае если мы имеем дело с запросом, возвращающим набор строк в качестве результата (запросом на выборку), программа при помощи вызовов функции **dbbind()** осуществляет связывание каждого поля результатов запроса с некоторой областью оперативной памяти. Далее при помощи функции **dbnextrow()** программа выполняет переход к следующей строке результатов запроса, что приводит к помещению в буфер новых данных;
- при помощи функции **dbexit()** программа разрывает соединение с базой данных. Библиотека DB-Library представляет собой большой и сложный механизм. Так, в библиотеке предусмотрены специальные механизмы обработки ошибок, разные способы передачи результатов выполнения запросов в прикладную программу и т.д.

Краткие итоги: В лекции рассматриваются разные технологии формирования запросов на языке *SQL* в прикладных программах (программный *SQL*). Дается понятие *статического SQL*, *динамического SQL* и приводятся соответствующие основные *операторы*. Рассматриваются *интерфейсы программирования приложений (API)* (протокол *ODBC*, протокол *JDBC*, библиотека DB-Library).

Цель 14 лекции: выделить основные черты в новых направлениях развития теории и практики создания баз данных (новые свойства, присущие объектно-ориентированным и распределенным базам данных) и хранилищ данных.

14.1. Объектно-ориентированный подход к организации баз данных

В начале 90-х годов XX века начались активные попытки по внедрению объектно-ориентированных технологий в отрасль проектирования и разработки баз данных. Бытовала точка зрения о том, что соответствующие технологии быстро вытеснят все остальные, так же как и во многих других программистских отраслях, но ничего подобного не произошло.

Объектно-ориентированное программирование

Рассмотрим термин "объектно-ориентированное программирование". Заметим, что это термин, принятый преимущественно в российской литературе. В западной литературе [[14.2](#)] под этим понимается сразу три аспекта:

- Объектно-ориентированный анализ – *OOA, object-oriented analysis*. **Объектно-ориентированный анализ** – это методология, при которой требования к системе воспринимаются с точки зрения *классов и объектов*, выявленных в предметной области.
- *Объектно-ориентированное проектирование* – *OOD, object-oriented design*. **Объектно-ориентированное проектирование** – это методология проектирования, соединяющая в себе процесс объектной декомпозиции и приемы представления логической и физической, а также статической и динамической моделей проектируемой системы.
- *Объектно-ориентированное программирование* – *OOP, object-oriented programming*. **Объектно-ориентированное программирование** – это методология программирования, основанная на представлении программы в виде

совокупности объектов , каждый из которых является экземпляром определенного класса , а классы образуют иерархию наследования.

Здесь и далее по тексту условимся не отступать от традиций и понимать под объектно-ориентированным программированием (ООП) сразу три указанных выше аспекта.

Основой объектно-ориентированной технологии является так называемая объектная модель, которая возникает как результат объектно-ориентированной декомпозиции. Она выделяет основные абстракции предметной области, определяет классы абстракций и выясняет, какими данными (атрибутами) описывается каждая абстракция, какую функциональность эти абстракции должны обеспечивать. В отличие от традиционных технологий программирования объектно-ориентированная технология представляет программу как совокупность классов объектов, взаимодействующих друг с другом.

Объект – конкретная материализация абстракции; сущность с хорошо определенными границами, в которой инкапсулированы состояние и поведение.

Объект ООП – инкапсулированная структура, имеющая атрибуты и методы.

Термин "инкапсулированная структура" означает, что объект является самодостаточным, программы, внешние по отношению к объекту, ничего "не знают" о его структуре и такое "знание" им не требуется. "Внешний" вид объекта называется его интерфейсом.

В таком понимании *объект* – это черный ящик, нам неизвестно, что у него внутри, мы лишь можем вызвать его методы и только через них взаимодействовать с ним. Кроме этого, объекты могут принадлежать иерархии "от общего к частному", которая реализуется путем наследования. Инкапсулированные состояния объекта могут быть как простыми типами данных, так и другими объектами, или даже массивами объектов. Каждый *объект* содержит определенную совокупность методов, *классы* взаимодействуют друг с другом посредством механизма сообщений. *Объекты* идентифицируются с помощью специальных указателей – дескрипторов. Методы объектов ООП представляют собой последовательности инструкций, выполняемых объектом. Например, у объекта может быть метод, отображающий данный объект, создающий данный объект и изменяющий его.

*Предметная область моделируется как множество классов взаимодействующих объектов. Объект характеризуется набором свойств, которые являются как бы его пассивными характеристиками, и набором методов работы с этим объектом. Работать с объектом можно только с использованием его методов. Атрибуты объекта могут принимать множество допустимых значений, набор конкретных значений атрибутов определяет состояние объекта. Используя методы работы с объектом можно изменять значение его атрибутов и тем самым как бы изменить состояние самого объекта. Множество объектов с одним и тем же набором атрибутов и методов образует *класс* объектов. *Класс*, объекты которого могут служить значениями атрибута объектов другого класса, называется доменом этого атрибута.*

К числу основных идей объектно-ориентированной технологии, как правило, относят [[14.1](#)]: **абстрагирование, инкапсуляцию, модульность, иерархичность, типизацию, полиморфизм, наследование.**

Инкапсуляция ограничивает область видимости имени атрибута пределами того объекта, в котором оно определено. Смысл этого атрибута будет определяться тем объектом, в котором оно инкапсулировано.

Полиморфизм – способность одного и того же программного кода работать с разнообразными данными. Другими словами, он допускает возможность в объектах разных типов иметь методы (процедуры или функции) с одинаковыми именами. Во время выполнения объектной программы одни и те же методы оперируют с разными объектами в зависимости от типа аргумента.

Наследование. Допускается порождение нового класса на основе уже существующего класса, и этот процесс называется наследованием. В этом случае новый класс, называемый подклассом существующего класса, наследует все атрибуты и методы класса. В подклассе,

кроме того, могут быть определены дополнительные атрибуты и методы. Различают случаи простого и множественного наследования. В первом случае подкласс может определяться только на основе одного класса, во втором случае – на основе нескольких классов. Набор классов образует иерархическую структуру.

Объектно-ориентированные базы данных

К настоящему моменту терминология еще не устоялась, существует много разных определений и трактовок. *Представляется, что объектно-ориентированная база данных (ООБД) – база данных, основанная на принципах объектно-ориентированной технологии.* К основным описательным моментам, связанным с ООБД, в литературе [[14.2](#)] относят:

- *объекты* (в ООБД любая сущность – объект и обрабатывается как объект); отметим, что здесь используется понятие "объект" объектно-ориентированного программирования, которое отличается от понятия "объект", рассматриваемого ранее в данном учебном пособии.
- *классы* (понятие "тип данных" реляционной модели заменяется понятиями "класс" и "подкласс");
- *наследование* (классы образуют иерархию наследования, заимствуя свойства друг друга);
- атрибуты (характеристики объекта моделируются его атрибутами);
- сообщения и методы (каждый класс имеет определенную совокупность методов, классы взаимодействуют друг с другом посредством механизма сообщений);
- инкапсуляция (внутренняя структура объектов скрыта);
- *идентификаторы объектов* – дескрипторы.

Схема представления объекта приводится на [рис. 14.1](#)

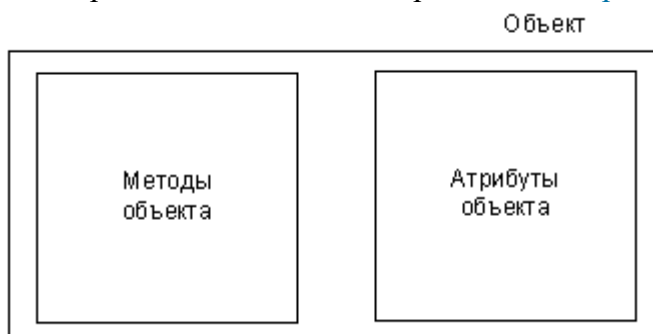


Рис. 14.1. Схема представления объекта

Система управления объектно-ориентированной базой данных называется объектно-ориентированной СУБД (ООСУБД). Цель ООСУБД – обеспечение постоянного хранения объектов, причем в отличие от традиционной СУБД ООСУБД должна хранить в составе объекта данные и программы.

Поскольку каждый объект данного класса имеет один и тот же набор методов, методы сохраняются только один раз – как *методы класса* (данные каждого экземпляра объекта хранятся отдельно).

Схема представления класса объектов приводится на [рис. 14.2](#)



Рис. 14.2. Схема представления класса объектов

Используя *наследование*, всем объектам ПОДРАЗДЕЛЕНИЕ можно приписать свойство объекта-родителя (ФАКУЛЬТЕТ) – название факультета, номер факультета. Схема представления объектов ФАКУЛЬТЕТ и ПОДРАЗДЕЛЕНИЕ приводится на [рис. 14.3](#).

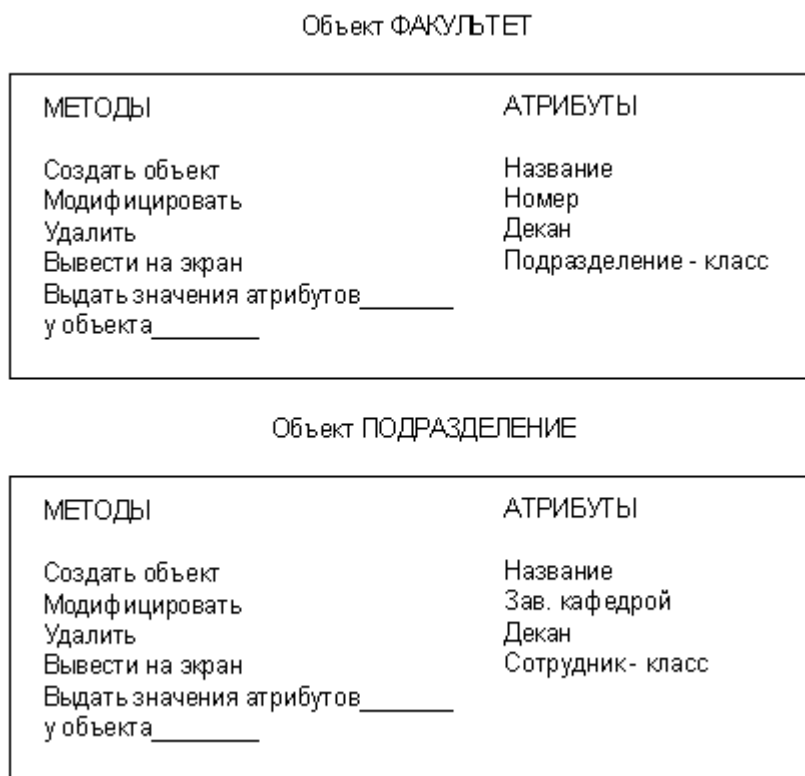


Рис. 14.3. Фрагменты представления конкретных объектов

Сравнивая объектно-ориентированный и реляционный подходы к БД, можно отметить следующие особенности. В реляционных БД (РБД) реальные объекты представляются как структуры, состоящие из набора *элементарных типов* данных. Такое представление имеет понятную интерпретацию – строка в плоской таблице. В том случае, когда специфика предметной области позволяет работать с такого рода приближением реальных объектов, РБД отлично справляются со своей задачей. Довольно часто реляционная модель и ее способ описания предметной области в виде набора плоских таблиц не отражают внутренней структуры для многих предметных областей, являются искусственными и становятся совершенно непонятными при увеличении количества таблиц. Основная причина несостоятельности реляционного подхода заключается в слишком сильной абстракции реального объекта, что ведет к потере семантики.

В отличие от реляционных баз данных *объектно-ориентированные базы данных* обладают простой и естественной связью с предметной областью, представляя ее структуру и состав, что облегчает проектирование и положительно сказывается на понимании принципов функционирования программ. Так, в сложных неоднородных предметных областях использование ООБД (в частности, там, где разные объекты имеют разные методы) должно действительно упростить процесс проектирования и разработки.

К сожалению, в ООБД существуют свои проблемы. В ООБД отсутствует универсальная модель данных, и соответственно, отсутствует мощная математическая база, как, например, в реляционной модели. В связи с этим у ООБД нет языка запросов высокого уровня, аналогичного SQL, и при доступе к данным используется мало эффективный навигационный подход. ООСУБД отличаются от реляционных СУБД тем, что программный интерфейс создания приложения либо очень слаб, либо вообще отсутствует. Это означает, для

написании приложения, работающего с ООБД, не существует мастеров и конструкторов (не считая, например, конструктора создания списка полей в объекте, который поставляется вместе с ООСУБД ObjectStore). Поэтому разработчик создает приложения на одном из алгоритмических языков.

По нашему мнению, существенным ограничением развития объектно-ориентированного подхода к созданию баз данных является то, что методы объекта содержатся внутри объекта и неразрывно связаны с ним. Это делает, по сути, невозможным создание для объектно-ориентированной базы данных соответствующей системы управления базой данных в традиционном понимании СУБД, функциями которой, в частности, является реализация операций обработки данных. Поэтому ООСУБД часто является не системой управления базами данных, а библиотекой программ, с помощью которой можно построить объектно-ориентированную базу данных. Примером такой библиотеки является ООСУБД ObjectStore. В связи с этим, возникает проблема реализации непредвиденных запросов.

Для перехода к *объектно-ориентированным БД* стандарт объектного программирования был дополнен стандартизованными средствами доступа к базам данных (стандарт ODMG 93; Object Database Management Group – группа управления *объектно-ориентированными базами данных*). К настоящему времени этот стандарт не реализован. Состояние проблемы подробно описано также в работах [[14.2](#)], [[2.1](#)], [[1.2](#)], [[6.5](#)], [[1.3](#)] и др.]. Отметим только, что ООБД используются, но пока не стали реальной альтернативой реляционным базам данных.

Объектно-ориентированные возможности появляются в ведущих современных СУБД, таких, как, например, Oracle. Предпринимаются попытки внесения изменений в стандарты языка SQL с целью его частичной адаптации к ООБД. Так, новый стандарт SQL-3 включает большой раздел, посвященный этому вопросу.

Объектно-реляционные СУБД

В настоящее время реляционные СУБД доминируют среди систем управления данными. Преимущества объектно-ориентированного подхода для создания сложных специализированных приложений с одной стороны, и стремление разработчиков систем управления базами данных с другой стороны расширить границы применения соответствующих СУБД обусловили *включение объектно-ориентированных компонент (расширяемая пользователем система типов, инкапсуляция, наследование, полиморфизм и т. п.) в модель данных реляционной СУБД. Соответствующие СУБД, называемые объектно-реляционными*, соединяют в себе лучшие качества реляционных и *объектно-ориентированных баз данных*. Отметим, что в разных СУБД реализован разный набор из перечисленных объектно-ориентированных компонент. Таким образом, не существует общепринятой объектно-реляционной модели, а скорее имеется несколько таких моделей, поддерживающих определенный набор объектно-ориентированных компонент. Однако, основой всех таких моделей являются реляционные таблицы, используется язык запросов, включено понятие объекта, а в некоторых дополнительно реализована возможность сохранения методов в базе данных.

Соответствующие изменения реляционной модели обусловили необходимость расширения стандарта языка запросов SQL. Первый вариант такого стандарта получил название SQL3. Работа над стандартом продолжается и в настоящее время.

В качестве примера в максимальной степени объектно-ориентированной СУБД можно указать исследовательскую СУБД Postgres [[2.1](#)].

Отметим считающиеся объектными расширениями элементы СУБД Microsoft Server 2008.

- **Пользовательские расширения.** Пользователи имеют возможность вмешиваться в изначально предоставляемый СУБД инструментарий, создавая, в частности, новые *пользовательские типы данных*.

- **Хранение больших объемов данных.** Наряду с теми данными, которые хранились в БД традиционно, *Microsoft SQL Server 2008* позволяет хранить в столбцах таблицы данные больших размеров (поддерживаются соответствующие типы данных).

- **Новые, ориентированные на определенные классы объектов, типы данных.** В системе определены новые типы данных (*geometry*, *geography*), характерные для тех направлений, в которых объектно-ориентированный подход весьма эффективен и часто используется (картография и соответствующие приложения, геометрическое представление объектов самой разной природы).

- **Хранимые процедуры.** В определенном смысле хранимые процедуры также являются объектным расширением, осуществляя необходимые пользователю воздействия на данные (стандартный для ООП процедурный подход).

14.2. Распределенные базы данных

База данных – интегрированная совокупность данных, с которой работают много пользователей. Изложение всех предыдущих разделов предполагало единую базу данных, размещаемую на одном компьютере. Напомним основные принципы, положенные в основу теории баз данных:

- централизованное хранение данных;
- централизованное обслуживание данных (ввод, корректировка, чтение, контроль целостности).

Заметим, что *базы данных* появились в период господства больших ЭВМ. *База данных* велась на одной ЭВМ, все пользователи работали именно на ЭВМ (возможные режимы работы описаны в "Различные архитектурные решения, используемые при реализации многопользовательских СУБД. Краткий обзор СУБД"). Других вариантов использования вычислительной техники в то время просто не существовало. Если проанализировать работу пользователей с данными в компаниях, организациях, предприятиях в "докомпьютерное" время, то нетрудно заметить, что на отдельных участках пользователи работали со "своими" данными (осуществляли сбор определенных данных, их хранение, обработку, передачу обработанных данных на другие участки или уровни управления).

У такой технологии были существенные недостатки, которые уже отмечались в предыдущих разделах: дублирование некоторых данных, отсутствие возможности сравнительного анализа данных всех участков. Однако у этой технологии были и существенные достоинства: данные вводились и хранились в местах их порождения; с этими данными работал *пользователь*, являющийся специалистом именно по этим данным, что позволяло ему вести эффективный *контроль* правильности данных на всех стадиях обработки; данные находились непосредственно у пользователя, что давало возможность их оперативной обработки. Централизация данных на одной ЭВМ, несомненно, дающая эффективные возможности хранения и обработки данных, не позволяла реализовывать вышеназванные достоинства.

Развитие вычислительных компьютерных сетей обусловило новые возможности в организации и ведении баз данных, позволяющие каждому пользователю иметь на своем компьютере свои данные и работать с ними и в то же время позволяющие работать всем пользователям со всей совокупностью данных как с единой централизованной базой данных. Соответствующая совокупность данных называется распределенной базой данных.

Термин "**распределенная база данных**" достаточно часто встречается в литературе [[14.2], [2.1], [1.2], [6.5], [1.3]]. Однако в разных источниках под этим термином понимаются совершенно разные вещи. Часть авторов понимают под распределенной базой данных то, что имеется удаленный *сервер*, на котором расположены данные, а также клиентские компьютеры, расположенные территориально в другом месте. Такая трактовка нам представляется неправильной. Настоящая **распределенная база данных** располагается на нескольких компьютерах. При этом часть файлов расположена на одном компьютере, часть на другом и т.д. Более того, возможна и даже часто встречается ситуация, когда *информация* на этих компьютерах пересекается, дублируется.

Распределенная база данных – совокупность логически взаимосвязанных *разделяемых данных* (и описаний их структур), физически распределенных в компьютерной сети.

Система управления распределенной базой данных – программная система, обеспечивающая работу с распределенной базой данных и позволяющая пользователю работать как с его локальными данными, так и со всей базой данных в целом.

Система управления распределенной базой данных (РаСУБД) является распределенной системой. Каждый фрагмент *базы данных* работает под управлением отдельной *СУБД*, которая осуществляет *доступ* к данным этого фрагмента. Пользователи взаимодействуют с распределенной базой данных через локальные и глобальные приложения. Локальные приложения дают пользователю возможность работать со своими локальными данными и не требуют доступа к другим фрагментам. Глобальные приложения дают пользователю возможность работать с другими фрагментами *базы данных*, расположенными на других компьютерах сети. Общая схема распределенной *базы данных* представлена на [рис. 14.4](#).

Объединение данных организуется виртуально. Соответствующий подход, по сути, отражает организационную структуру предприятия (и даже общества в целом), состоящего из отдельных подразделений. Причем, хотя каждое подразделение обрабатывает свой набор данных (эти наборы, как правило, пересекаются), существует необходимость доступа к этим данным как к единому целому (в частности, для управления всем предприятием).

Одним из примеров реализации такой модели может служить *сеть Интернет*: данные вводятся и хранятся на разных компьютерах по всему миру, любой *пользователь* может получить *доступ* к этим данным, не задумываясь о том, где они физически расположены.

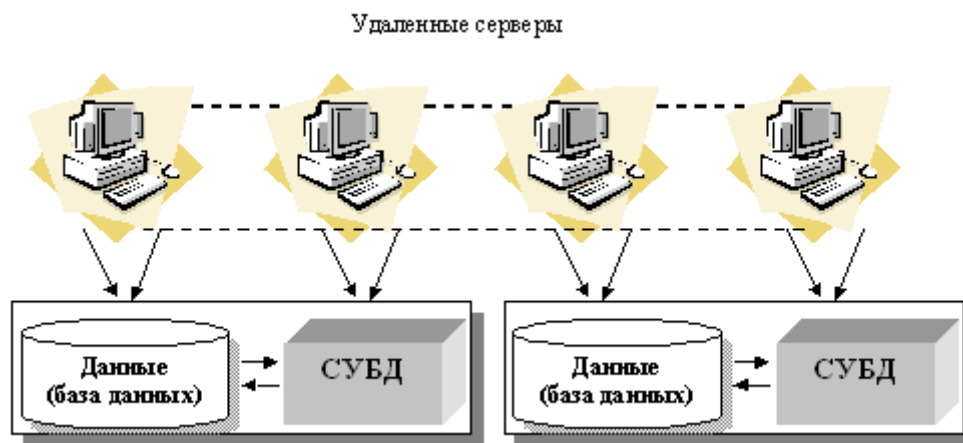


Рис. 14.4. Распределенная база данных

К.Дж. Дейт провозглашает следующий фундаментальный принцип распределенной *базы данных* [[2.1]]. Для пользователя распределенная система должна выглядеть точно так же, как нераспределенная. Из этого принципа следует ряд правил:

1. Локальная автономия.
2. Независимость от центрального узла.
3. Непрерывное функционирование.
4. Независимость от расположения.
5. Независимость от фрагментации.
6. Независимость от репликации.
7. Обработка распределенных запросов.
8. Управление *распределенными транзакциями*.
9. Независимость от аппаратного обеспечения.
10. Независимость от операционной системы.
11. Независимость от сети.
12. Независимость от СУБД.

Заметим, что понятие *распределенной базы данных* можно интерпретировать как следующий шаг в развитии понятий о данных (см. ["Введение в базы данных. Общая](#)

характеристика основных понятий"), обусловленный распределенностью данных в реальных предметных областях, а также новым этапом развития средств вычислительной техники – широким использованием вычислительных сетей.

В этой интерпретации распределенную базу данных можно понимать как совокупность логически взаимосвязанных распределенных по разным компьютерам баз данных.

Перечислим основные проблемы создания распределенной базы данных.

1. Фрагментация данных и распределение по компьютерам.
2. Составление глобального каталога, содержащего информацию о каждом фрагменте БД и его местоположении в сети. (Каталог может храниться на одном узле или быть распределенным)
3. Организация обработки запросов (синхронизация нескольких запросов к одним и тем же данным, исключение аномалий удаления и обновления одних и тех же данных, расположенных на различных узлах, оптимизация последовательности шагов при обработке запроса и т.д.).

Значительным достоинством этой модели является приближение данных к месту их порождения, что позволяет существенно повысить их достоверность, недостатком – достаточно высокая сложность управления данными как единым целым.

К сожалению, процесс создания и обслуживания распределенных баз данных связан и с техническими трудностями, среди которых можно выделить жесткие требования к пропускной способности каналов связи, а также низкую производительность, обусловленную значительными затратами коммуникационных и вычислительных ресурсов при их синхронизации во время выполнения транзакций (особенно при интенсивных обращениях из разных узлов к одному фрагменту).

В задачу данного учебника не входит подробное изучение принципов построения распределенных баз данных. Интересующимся рекомендуем обратиться к соответствующей литературе, например [[2.1], [1.2], [6.5], [1.3]] Здесь мы хотим лишь обрисовать проблему и сделать некоторые выводы по перспективам ее решения. Технология, связанная с использованием распределенных баз данных, в наибольшей степени соответствует организационной человеческой деятельности (информация распределена по месту деятельности людей, и они обмениваются ей в процессе работы) и позволяет наиболее успешно решать важнейшие проблемы ведения баз данных:

- повысить достоверность информации (информация вводится в месте ее порождения лицом, которое лучше всех понимает ее смысловое значение);
- повысить оперативность локальной обработки информации (соответствующие вопросы решаются на локальном компьютере с фрагментом базы данных).

Поэтому очевидно, что задача проектирования, создания и функционирования распределенных баз данных является весьма существенной, активно изучается в настоящее время и будет решаться и далее.

14.3. Хранилища данных

Как уже неоднократно отмечалось, технологии баз данных предназначены, как правило, для решения текущих задач обработки данных организации. В базу данных постоянно вносятся изменения, то есть база данных отражает моментальный снимок определенной области деятельности предприятия. Для эффективного принятия решений руководством при управлении организацией важно не только знать текущее положение дел, но и иметь возможность анализировать динамику (изменение во времени) основных показателей, причем, зачастую из разных баз данных. Такую возможность дает технология так называемых хранилищ данных.

Приведем определение хранилища данных (Bill Inmon).

Хранилище данных – предметно-ориентированный, интегрированный, привязанный ко времени и неизменяемый набор данных, предназначенный для поддержки принятия решений.

Под **предметной ориентированностью** здесь понимается ориентированность на предметы (определенные группы данных), а не на конкретные приложения. Например, ориентация на данные о сотрудниках, а не только о расчете их заработной платы.

Под **интегрированностью** здесь понимается возможное *объединение* данных из разных источников (баз данных), имеющих разных формат и несогласованных.

Привязка ко времени предполагает, что для всех данных указан момент или промежуток времени, в который они корректны.

Данные в хранилище не изменяются, они лишь регулярно пополняются из оперативных баз данных.

Общая схема взаимодействия информационного хранилища и баз данных приводится на [рис. 14.5](#).

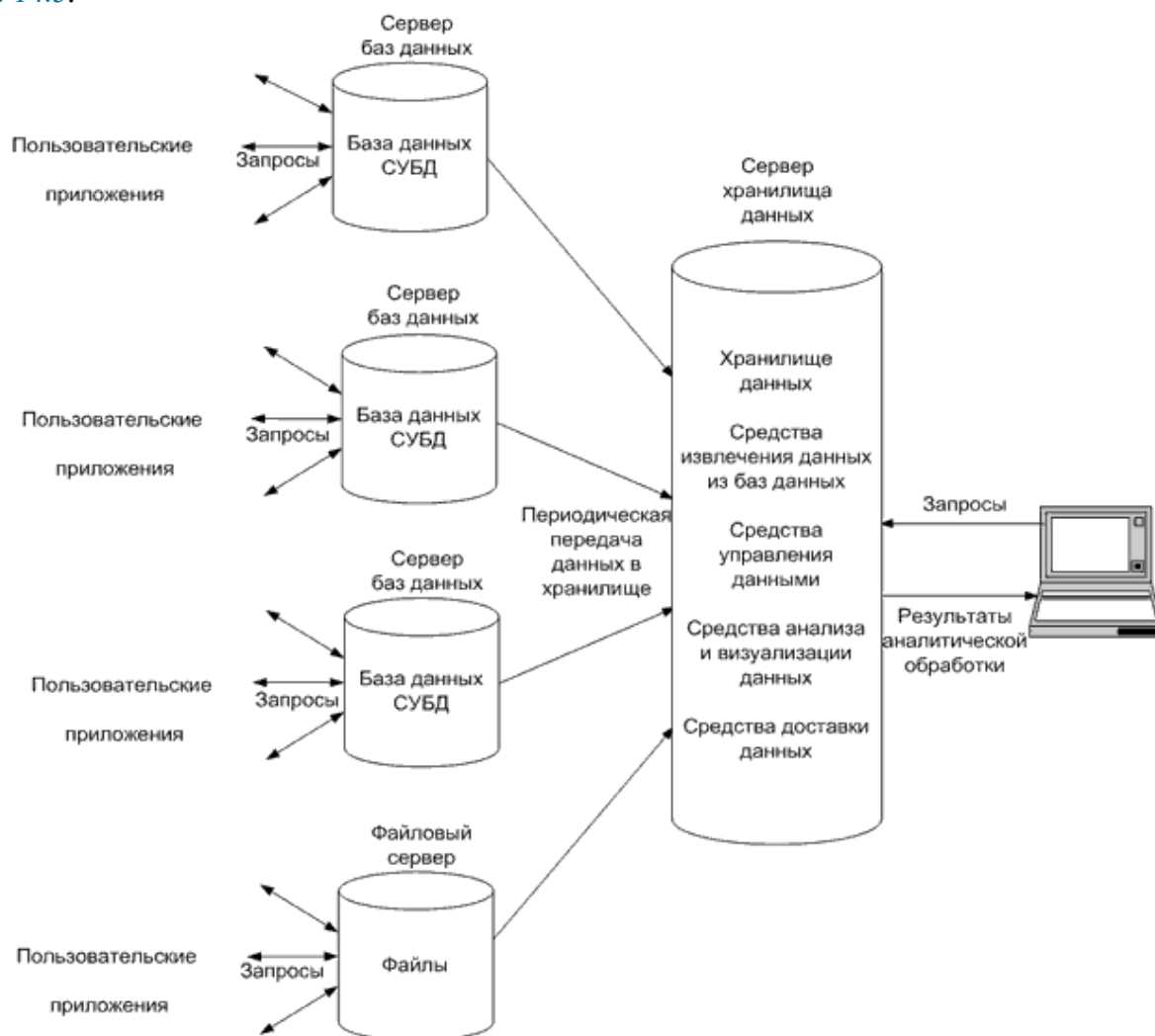


Рис. 14.5. Схема организации работы хранилища данных

Еще раз подчеркнем, что основной целью хранилищ данных является бизнес-анализ или информационная *поддержка* принятия управленческих решений.

Для реализации всей необходимой обработки информации в соответствии с этой схемой необходимы следующие *программные средства*:

- средства извлечения данных из баз данных;
- средства управления данными хранилища (система управления базой данных хранилища);
- средства анализа данных хранилища (используется OLAP-технология);
- средства доставки данных;
- средства визуализации результатов обработки для конечных пользователей.

Для работы соответствующих программных средств необходимо описание структуры содержимого информационного хранилища (метаописание).

Для самого общего случая, если данные берутся из баз данных, управляемых разными СУБД, из файлов разных типов, а данные разнородны, средства управления данными хранилища пока не созданы. Однако, если данные в информационное хранилище выбираются только из реляционных баз данных, то в качестве средств управления данными хранилища может быть взята мощная реляционная СУБД. Поэтому разработчики современных СУБД включают в состав программного обеспечения СУБД средства организации работы с хранилищами данных.

Рассмотрим в качестве **примера** *возможности СУБД Microsoft SQL Server 2008 для организации хранилищ данных.*

Microsoft SQL Server 2008 содержит в своем составе средства извлечения, преобразования и загрузки данных (SQL Server 2008 Integration Services), способные интегрировать данные из различных источников, проверять данные на допустимость и преобразовывать перед загрузкой в хранилище. Эти средства также способствуют перемещению данных, поддерживают текстовый анализ и нечеткий поиск. Нужно отметить также среду визуальной разработки (Business Intelligence Development Studio) для создания многомерных кубов, отчетов, пакетов извлечения, преобразования и загрузки данных.

Существенной особенностью хранилищ данных является их очень большой объем. Microsoft SQL Server 2008 как средство управления данными хранилища позволяет работать с большими объемами данных, причем для сокращения времени обработки предусмотрена поддержка параллельных вычислений (путем разделения таблиц и индексов на секции и обеспечение параллельной обработки секций). В системе предусмотрена возможность сжатия данных (таблиц), что позволяет уменьшить физический размер таблиц и существенно сокращает время обмена между оперативной и внешней памятью.

В качестве средств анализа данных хранилища используется SQL Server 2008 Analysis Services, применяемый для построения многомерных кубов (многомерных моделей данных). Это средство содержит семь эффективных алгоритмов анализа данных с целью поддержки принятия управленческих решений, в том числе анализ тенденций и статистический анализ данных.

В качестве средств представления аналитических данных пользователям предлагается использовать средство генерации отчетов SQL Server 2008 Reporting Services.

Таким образом, Microsoft SQL Server 2008 является эффективным средством реализации хранилищ данных на основе реляционных баз данных.

Краткие итоги: В лекции рассмотрены перспективные направления в теории и практике создания баз данных – объектно-ориентированные и распределенные базы данных. Здесь описываются основные идеи объектно-ориентированного программирования (объект, класс, методы класса, наследование) и их приложение к теории баз данных. Отмечены основные достоинства и недостатки объектно-ориентированных баз данных.

Рассматривается понятие распределенных баз данных как следующий шаг в развитии понятий о данных. Отмечены основные достоинства распределенных баз данных и проблемы, возникающие при их разработке.

Рассматривается понятие хранилища данных, в качестве примера системы управления данными хранилища приводится СУБД Microsoft SQL Server 2008.

**Приложение № 2 к методическим материалам
по дисциплине (модулю). Конспекты
практических (семинарских) занятий по
дисциплине (модулю)**

**КОНСПЕКТЫ ПРАКТИЧЕСКИХ (СЕМИНАРСКИХ) ЗАНЯТИЙ ПО
ДИСЦИПЛИНЕ (МОДУЛЮ)**

1. Учебная дисциплина.

2. Тема практического (семинарского) занятия.

3. Цели занятия.

(Цели занятия можно формулировать следующим образом: сформировать представление о ... , сформировать понимание ..., раскрыть основные положения ... , раскрыть сущность ...)

4. Структура практического (семинарского) занятия.

№
п/п

Содержание (кратко)

Методы и средства
обучения

5. Содержание практического (семинарского) занятия и взаимодействие с аудиторией.

Введение.

Формулирование темы занятия. Использование вводного примера, описание ситуации, демонстрация какого-либо процесса и т. д. (в зависимости от вида, замысла лекции и её структуры)

1. Тема практического (семинарского) занятия.

Вопросы к обсуждению:

Практические задания:

Требования к выполнению практического задания:

2. Тема практического (семинарского) занятия.

Вопросы к обсуждению:

Практические задания:

Требования к выполнению практического задания:

КОНСПЕКТЫ ЛАБОРАТОРНЫХ ЗАНЯТИЙ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)

РАЗДЕЛ 1. ТЕОРИЯ БАЗ ДАННЫХ

ЗАДАНИЯ К ЛАБОРАТОРНЫМ ЗАНЯТИЯМ РАЗДЕЛА 1

Тема лабораторных занятий: Установка PostgreSQL.

Перечень изучаемых элементов содержания

Вводятся основные понятия объект, сущность, параметр, атрибут, таблица, запись, поле, домен, кортеж, первичный ключ и т.д. Три вида архитектур: локальная, «файл-сервер» и «клиент-сервер» (два варианта). Способы физического хранения данных и обмена данными. Сравнительный анализ и рекомендации по использованию каждого вида архитектур. Подробно рассматривается базовая технология COM. В краткой форме с указанием преимуществ и особенностей работы рассматриваются технологии ADO, CORBA и проч. Необходимость создания журнала аудита. Доступность и приемы заполнения журнала аудита. Назначение и использование просмотров. Просмотры редактируемые и не редактируемые. Область применения и преимущества каждого вида просмотра. Подробно рассматривается пять этапов проектирования базы данных. Состав работ, выполняемых на каждом этапе.

Форма практического задания: лабораторный практикум.

Задания лабораторного практикума

1. Установка PostgreSQL
 - 1.1 Установка PostgreSQL для Windows
 - 1.2 Установка PostgreSQL для Linux
 - 1.3 Утилита pgAdmin III
 - 1.4 Утилита psql

РУБЕЖНЫЙ КОНТРОЛЬ К РАЗДЕЛУ 1

форма рубежного контроля – отчет по лабораторной работе с ответами на контрольные вопросы.

1. Перечислите основные параметры команды утилиты psql для подключения к базе данных.
2. Перечислите основные команды утилиты psql для работы с буфером запросов.
3. Перечислите основные команды утилиты psql для работы со списком объектов.

РАЗДЕЛ 2. ПРОЕКТИРОВАНИЕ ПРИЛОЖЕНИЯ БАЗЫ ДАННЫХ

Перечень изучаемых элементов содержания

Два способа создания файла базы данных: на уровне команд и диалоговый режим работы с утилитой IBConsol. Приемы модификации структуры базы данных на уровне команд,

с рассмотрением последствий не корректных действий при выполнении данной операции. Команда удаления файла базы данных. Команды по созданию, модификации и удалению таблиц, с указанием особенностей работы каждой из команд. Допустимые типы полей и способы их задания и приемы указания дополнительных характеристик полей. Понятие целостности и достоверности данных. Опция СHECK в командах создания и модификации таблицы. Приемы создания вычисляемых (виртуальных) полей, установка и удаление условий ссылочной целостности данных. Понятия индекса и ключа, с указанием общих свойств и различий. Типы и виды ключей и два способа создания ключей. Приемы создания именованных и не именованных условий ссылочной целостности и особенностей их работы. Понятие набора данных, его размера. Методы перемещения курсора по записям набора данных. Четыре метода доступа к полям и их значениям. Методы поиска данных на стороне клиента, как по произвольному полю (полям), так и по индексному полю (полям), с указанием особенностей каждого вида поиска. Поиск на частичное совпадение и поиск без учета регистра. Поиск по индексному полю: на точное совпадение и пошаговое приближение (инкрементальный локатор). Варианты формирования запросов на добавление, редактирование и удаление записи: Особенности и преимущества каждого из вариантов. Методы сортировки данных на стороне сервера и клиента. Сортировка по-простому и сложному индексу. Запросы на извлечение записей с группировкой. Запросов с фильтрацией записей. Комбинированные запросы с сортировкой, группировкой и фильтрацией. Понятие отчета. Процедура формирования отчета. Отчеты с вычислениями, группировкой и сортировкой. Вывод отчета на экран и печать. Понятие и назначение генератора. Команды создания, запуска, модификации, управления и удаления генератора. Понятие хранимой процедуры, её назначение. Создание различных типов хранимых процедур и их модификации. Назначение хранимых процедур, преимущества использования хранимых процедур. Вызов хранимых процедур. Понятие и назначение триггера. Создание, изменение и удаление триггера. Понятие отчета и его назначение. Компоненты, методы и свойства, с помощью которых формируется отчет. Редактирование отчета. Предварительный просмотр и вывод отчета на печать. Понятие транзакции. Излагаются методы создания и управления транзакциями (два способа). Сравнительная характеристика двух способов управления транзакциями. Состояния, в которых могут находиться транзакции. Конкурирующие транзакции. Понятие изоляции транзакций. Конфликты и приемы их разрешения.

ЗАДАНИЯ К ЛАБОРАТОРНЫМ ЗАНЯТИЯМ РАЗДЕЛА 2

Темы лабораторных занятий:

1. Описание практического примера.

Форма практического задания: лабораторный практикум.

Задания лабораторного практикума

1. Наполнение тестовой БД
 - 1.1 Создание базы данных
 - 1.2 Создание таблиц
 - 1.3 Создание последовательностей
 - 1.4 Ввод данных
 - 1.5 Создание индексов

РУБЕЖНЫЙ КОНТРОЛЬ К РАЗДЕЛУ 2

форма рубежного контроля – отчет по лабораторной работе с ответами на контрольные вопросы.

1. Для чего нужны последовательности?
2. Для чего нужны индексы?

РАЗДЕЛ 3. ВВЕДЕНИЕ В РЕЛЯЦИОННЫЕ БАЗЫ ДАННЫХ.

Перечень изучаемых элементов содержания

Вводятся основные понятия теории реляционных баз данных. Изучаются структура данных, ограничения целостности, внутренние ограничения целостности, семантические ограничения целостности.

ЗАДАНИЯ К ЛАБОРАТОРНЫМ ЗАНЯТИЯМ РАЗДЕЛА 3

Темы лабораторных занятий:

1. Введение в язык SQL

Форма практического задания: лабораторный практикум.

Задания лабораторного практикума

3.1 Средства определения данных языка SQL

3.1.1 Оператор CREATE TABLE

3.1.2 Оператор ALTER TABLE

3.1.3 Операторы DROP TABLE и DROP

3.2 Средства запроса данных языка SQL

3.2.1 Чтение заданных столбцов из одиночной таблицы

3.2.2 Чтение заданных строк из одиночной таблицы

3.2.3 Чтение заданных строк и столбцов из одиночной таблицы

3.2.4 Диапазоны, специальные символы и пустые значения в предложениях WHERE

3.2.5 Сортировка результатов

3.2.6 Агрегатные функции SQL

3.2.7 Агрегатные функции и группировка

3.2.8 Оконные функции

3.2.9 Чтение данных из нескольких таблиц с применением вложенных запросов

3.2.10 Чтение данных из нескольких таблиц с помощью операции соединения

3.2.11 Внешние соединения

3.2.12 Объединения

3.3 Средства модификации данных языка SQL

3.3.1 Вставка данных, оператор INSERT

3.3.2 Изменение данных, оператор UPDATE

3.3.3 Удаление данных, оператор DELETE

3.4 Транзакции

РУБЕЖНЫЙ КОНТРОЛЬ К РАЗДЕЛУ 3

форма рубежного контроля – отчет по лабораторной работе с ответами на контрольные вопросы.

1. Оператор CREATE TABLE.
2. Определение ограничений CONSTRAINT.
3. Оператор вставки данных INSERT.
4. Оператор обновления UPDATE.
5. Оператор удаления строк DELETE.
6. Чтение строк и столбцов.
7. Шаблоны поиска в SQL.
8. Сортировка результатов запроса.
9. Встроенные функции SQL.
10. Группировка в запросах.
11. Вложенные запросы.
12. Внешние и внутренние соединения.

РАЗДЕЛ 4. SQL — ЯЗЫК СТРУКТУРИРОВАННЫХ ЗАПРОСОВ

Перечень изучаемых элементов содержания

Определение данных в SQL. Описание учебного проекта. Язык определения данных. Создание таблиц. Модификация структуры таблицы. Удаление таблицы. Язык манипулирования данными. Ввод (добавление) данных в таблицу. Обновление данных. Удаление данных. Язык запросов. Простые запросы. Запросы к связанным таблицам. Декартово произведение. Естественное соединение. Запросы с подзапросами. Теоретико-множественные операции. Представления. Создание представлений. Удаление представлений. Операции над представлениями. Обновление представлений. Индексы.

ЗАДАНИЯ К ЛАБОРАТОРНЫМ ЗАНЯТИЯМ РАЗДЕЛА 4

Темы лабораторных занятий:

1. Применение SQL
2. Дополнительные возможности

Форма практического задания: лабораторный практикум.

Задания лабораторного практикума

- 4.1 SQL-представления
 - 4.1.1 Использование представлений для скрывания столбцов и строк
 - 4.1.2 Использование представлений для отображения вычисляемых столбцов
 - 4.1.3 Использование представлений для скрывания сложного синтаксиса
- 4.2 SQL-запросы в прикладных программах
- 4.3 Хранимые процедуры
- 4.4 Триггеры
 - 4.4.1 Использование триггеров для проверки допустимости вводимых данных
 - 4.4.2 Использование триггеров для присвоения значений по умолчанию
 - 4.4.3 Триггер, обновляющий представление
 - 4.4.4 Триггер, обеспечивающий ссылочную целостность
- 4.5 Правила
- 4.6 Модуль TABLEFUNC
- 4.7 Словарь метаданных
- 4.8 Функция ROW_NUMBER
- 4.9 Функция COALESCE
- 4.10 Числовые функции
- 4.11 Тригонометрические функции
- 4.12 Работа со строками
- 4.13 Работа с датой и временем

РУБЕЖНЫЙ КОНТРОЛЬ К РАЗДЕЛУ 4

форма рубежного контроля – отчет по лабораторной работе с ответами на контрольные вопросы.

10. Среда выполнения PL/SQL.
11. Неименованный блок PL/SQL.
12. Курсоры в PL/SQL.
13. Атрибуты курсора.
14. Неявные курсоры.
15. Курсоры-циклы.
16. Курсоры с параметрами.
17. Курсоры с обновлением.
18. Процедуры и функции PL/SQL.
19. Строковые и операторные триггеры.
20. Псевдозаписи триггера.
21. Триггерные предикаты.
22. Перечислите основные функции PostgreSQL для работы с числами.

23. Перечислите основные тригонометрические функции PostgreSQL.
24. Перечислите основные функции PostgreSQL для работы со строками.
25. Перечислите основные функции PostgreSQL для работы с датой и временем.

УЧЕБНО-НАГЛЯДНЫЕ ПОСОБИЯ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)

РАЗДЕЛ 1. Теория баз данных

Тема 1.1. Введение в базы данных. Общая характеристика основных понятий

1.Маркин, А. В. Программирование на SQL в 2 ч. Часть 1 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 429 с. — (Высшее образование). — ISBN 978-5-534-15817-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509818> (дата обращения: 08.03.2023).

2.Маркин, А. В. Программирование на SQL в 2 ч. Часть 2 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., испр. и доп. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-534-15818-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509819> (дата обращения: 08.03.2023).

Тема 1.2. Системы управления базами данных

1.Маркин, А. В. Программирование на SQL в 2 ч. Часть 1 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 429 с. — (Высшее образование). — ISBN 978-5-534-15817-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509818> (дата обращения: 08.03.2023).

2.Маркин, А. В. Программирование на SQL в 2 ч. Часть 2 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., испр. и доп. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-534-15818-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509819> (дата обращения: 08.03.2023).

РАЗДЕЛ 2. Проектирование приложения базы данных

Тема 2.1. Различные архитектурные решения, используемые при реализации многопользовательских СУБД. Краткий обзор СУБД

1.Маркин, А. В. Программирование на SQL в 2 ч. Часть 1 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 429 с. — (Высшее образование). — ISBN 978-5-534-15817-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509818> (дата обращения: 08.03.2023).

2.Маркин, А. В. Программирование на SQL в 2 ч. Часть 2 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., испр. и доп. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-534-15818-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509819> (дата обращения: 08.03.2023).

Тема 2.2. С Различные представления о данных в базах данных. Основные этапы проектирования баз данных

1.Маркин, А. В. Программирование на SQL в 2 ч. Часть 1 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 429 с. — (Высшее образование). — ISBN 978-5-534-15817-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509818> (дата обращения: 08.03.2023).

2.Маркин, А. В. Программирование на SQL в 2 ч. Часть 2 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., испр. и доп. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-534-15818-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509819> (дата обращения: 08.03.2023).

РАЗДЕЛ 3. Введение в реляционные базы данных

Тема 3.1. Формализация реляционной модели

1.Маркин, А. В. Программирование на SQL в 2 ч. Часть 1 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 429 с. — (Высшее образование). — ISBN 978-5-534-15817-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509818> (дата обращения: 08.03.2023).

2.Маркин, А. В. Программирование на SQL в 2 ч. Часть 2 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., испр. и доп. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-534-15818-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509819> (дата обращения: 08.03.2023).

Тема 3.2. Физические модели данных (внутренний уровень). Структура современной СУБД. Программное обеспечение работы с современными базами данных

1.Маркин, А. В. Программирование на SQL в 2 ч. Часть 1 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 429 с. — (Высшее образование). — ISBN 978-5-534-15817-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509818> (дата обращения: 08.03.2023).

2.Маркин, А. В. Программирование на SQL в 2 ч. Часть 2 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., испр. и доп. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-534-15818-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509819> (дата обращения: 08.03.2023).

РАЗДЕЛ 4. SQL — язык структурированных запросов

Тема 4.1. Основные операторы языка SQL. Интерактивный SQL

1.Маркин, А. В. Программирование на SQL в 2 ч. Часть 1 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 429 с. — (Высшее образование). — ISBN 978-5-534-15817-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509818> (дата обращения: 08.03.2023).

2.Маркин, А. В. Программирование на SQL в 2 ч. Часть 2 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., испр. и доп. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-534-15818-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509819> (дата обращения: 08.03.2023).

Тема 4.2. Использование языка SQL в прикладных программах. Направления развития баз данных

1.Маркин, А. В. Программирование на SQL в 2 ч. Часть 1 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 429 с. — (Высшее образование). — ISBN 978-5-534-15817-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509818> (дата обращения: 08.03.2023).

2.Маркин, А. В. Программирование на SQL в 2 ч. Часть 2 : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., испр. и доп. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-534-15818-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509819> (дата обращения: 08.03.2023).

ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

№ п/п	Содержание изменения	Реквизиты документа об утверждении изменения	Дата введения изменения
1.			
2.			
3.			
4.			



**Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Российский государственный социальный университет»**

УТВЕРЖДАЮ

Декан факультета политических и социальных
технологий

/Пивнева С.В./

28 февраля 2024 г.

**МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)
ИНТЕГРАЦИЯ ИНФОРМАЦИОННЫХ СИСТЕМ**

Направление подготовки
«Информационные системы и технологии»

Направленность
«Интеллектуальные информационные системы и технологии»

**ОБРАЗОВАТЕЛЬНАЯ ПРОГРАММА ВЫСШЕГО ОБРАЗОВАНИЯ - ПРОГРАММА
БАКАЛАВРИАТА**

Форма обучения
Очная, заочная

Москва, 2024 г.

Методические материалы по дисциплине (модулю) «Интеграция информационных систем» разработана на основании федерального государственного образовательного стандарта высшего образования – бакалавриата по направлению подготовки 09.03.02 Информационные системы и технологии, утвержденного приказом Министерства образования и науки Российской Федерации от 19.09.2017 № 926, учебного плана по основной профессиональной образовательной программе высшего образования – программе бакалавриата по направлению подготовки 09.03.02 Информационные системы и технологии.

Методические материалы по дисциплине (модулю) «Интеграция информационных систем» разработана рабочей группой в составе: канд. пед. наук, доцент Мнацакян О.Л.

Методические материалы по дисциплине (модулю) обсуждены и утверждены на заседании кафедры информационных технологий, искусственного интеллекта и общественно-социальных технологий цифрового общества факультета социальных и политических технологий.

Протокол № 11 от «28» февраля 2024 г.

Заведующий кафедрой
канд. пед. наук, доцент

С.В. Пивнева

(подпись)

СОДЕРЖАНИЕ

1. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ К ЛЕКЦИОННЫМ, ЛАБОРАТОРНЫМ ЗАНЯТИЯМ.....	4
1.1. Методические материалы к проведению лекционных занятий по дисциплине (модулю)	4
1.2. Методические материалы по подготовке к лабораторным занятиям по дисциплине (модулю)	6
2. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ) И САМОСТОЯТЕЛЬНОЙ РАБОТЕ.....	6
3. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНИВАНИЯ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ И (ИЛИ) ОПЫТА ДЕЯТЕЛЬНОСТИ, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ.....	16
3.1. Организационные основы применения балльно-рейтинговой системы оценки успеваемости обучающихся по дисциплине (модулю)	16
3.2. Проведение текущего контроля успеваемости обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося.....	16
3.3. Проведение промежуточной аттестации обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося	17
ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ	19

1. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ К ЛЕКЦИОННЫМ, ЛАБОРАТОРНЫМ ЗАНЯТИЯМ

1.1. Методические материалы к проведению лекционных занятий по дисциплине (модулю)

Лекция – один из методов обучения, одна из основных системообразующих форм организации учебного процесса в вузе. Лекционное занятие представляет собой систематическое, последовательное, монологическое изложение педагогическим работником учебного материала, как правило, теоретического характера. Такое занятие представляет собой элемент технологии представления учебного материала путем логически стройного, систематически последовательного и ясного изложения.

Цель лекции – организация целенаправленной познавательной деятельности обучающихся по овладению программным материалом дисциплины (модуля). Чтение курса лекций позволяет дать связанное, последовательное изложение материала в соответствии с новейшими данными науки, сообщить слушателям основное содержание предмета в целостном, систематизированном виде. В ряде случаев лекция выполняет функцию основного источника информации: при отсутствии учебников и учебных пособий, чаще по новым курсам; в случае, когда новые научные данные по той или иной теме не нашли отражения в учебниках; отдельные разделы и темы очень сложны для самостоятельного изучения. В таких случаях только лектор может методически помочь обучающимся в освоении сложного материала.

Возможные формы проведения лекций:

- Вводная лекция – один из наиболее важных и трудных видов лекции при чтении систематических курсов. От успеха этой лекции во многом зависит успех усвоения всего курса. Она может содержать: определение дисциплины (модуля); краткую историческую справку о дисциплине (модуле); цели и задачи дисциплины (модуля), ее роль в общей системе обучения и связь со смежными дисциплинами (модулями); основные проблемы (понятия и определения) данной науки; основную и дополнительную учебную литературу; особенности самостоятельной работы обучающихся над дисциплиной (модулем) и формы участия в научно-исследовательской работе; отчетность по курсу.

- Информационная лекция ориентирована на изложение и объяснение обучающимся научной информации, подлежащей осмыслению и запоминанию. Это самый традиционный тип лекций в практике высшей школы.

- Заключительная лекция предназначена для обобщения полученных знаний и раскрытия перспектив дальнейшего развития данной науки.

- Обзорная лекция – это систематизация научных знаний на высоком уровне, допускающая большое число ассоциативных связей в процессе осмысления информации, излагаемой при раскрытии внутрисубъектной и междисциплинарных связей, исключая детализацию и конкретизацию. Как правило, стержень излагаемых теоретических положений составляет научно-понятийная и концептуальная основа всего курса или крупных его разделов.

- Лекция-беседа – непосредственный контакт педагогического работника с аудиторией – диалог. По ходу лекции педагогический работник задает вопросы для выяснения мнений и уровня осведомленности обучающихся по рассматриваемой проблеме.

- Лекция-дискуссия – свободный обмен мнениями в ходе изложения лекционного материала. Педагогический работник активизирует участие в обсуждении отдельными вопросами, сопоставляет между собой различные мнения и тем самым развивает дискуссию, стремясь направить ее в нужное русло.

- Лекция с применением обратной связи включает в себе то, что в начале и конце каждого раздела лекции задаются вопросы. Первый – для того, чтобы узнать, насколько обучающиеся ориентируются в излагаемом материале, вопрос в конце раздела предназначен для выяснения степени усвоения только что изложенного материала. При неудовлетворительных результатах контрольного опроса педагогический работник возвращается к уже прочитанному разделу, изменив при этом методику подачи материала.

- Проблемная лекция опирается на логику последовательно моделируемых проблемных ситуаций путем постановки проблемных вопросов или предъявления проблемных задач. Проблемный вопрос – это диалектическое противоречие, требующее для своего решения размышления, сравнения, поиска, приобретения и применения новых знаний. Проблемная задача содержит дополнительную вводную информацию и при необходимости некоторые ориентиры поиска ее решения.

- Программированная лекция-консультация – педагогический работник сам составляет и предлагает обучающимся вопросы. На подготовленные вопросы педагогический работник сначала просит ответить обучающихся, а затем проводит анализ и обсуждение неправильных ответов. В лекциях можно использовать наглядные материалы, а также подготовить презентацию. Что касается презентации, то в качестве визуальной поддержки ее можно органично интегрировать во все вышеупомянутые лекции. В то же время лекцию-презентацию возможно выделить и в качестве самостоятельной формы. Лекция-презентация должна отражать суть основных и (или) проблемных вопросов лекции, на которые особо следует обратить внимание обучающихся. В условиях применения активного метода проведения занятий презентация представляется весьма удачным способом донесения информации до слушателей. Единственное, на что следует обратить внимание при подготовке слайдов, - это их оформление и текст. Слайд не должен быть перегружен картинками и лишней информацией, которая будет отвлекать от основного аспекта того или иного вопроса лекции. Во время лекции можно задавать вопросы аудитории в отношении того или иного слайда, тем самым еще больше вовлекая обучающихся в проблематику.

Краткое содержание лекционных занятий

Наименование разделов и тем	Содержание учебного материала
РАЗДЕЛ 1. Информационная система и ее роль в осуществлении деятельности производственной организации	
Тема 1.1. Основы применения информационных систем в управлении	Производственная организация как система. Информационные системы управления организациями: основные понятия. Автоматизированные системы управления технологическими процессами. Автоматизированные системы управления предприятием. Информационно-аналитические системы. Системы поддержки принятия решений.
Тема 1.2. Интегрированные информационные системы в управлении организацией	Интегрированная информационная система управления предприятием. Классификация информационных систем производственного предприятия. Структура управления производственным предприятием. Основные классы систем автоматизации производственного предприятия. Классификация информационных систем по уровням управления. Системы поддержки принятия управленческих решений. Концепция Lean Production. Теория ограничений. Концепция быстро реагирующего производства. Концепция гибкого производства.
РАЗДЕЛ 2. Базовые концепции автоматизации управления на производственных предприятиях	
Тема 2.1. Интегрированные информационные системы управления ресурсами организации	Функциональные и технологические возможности современных информационных систем класса ERP. Контур управления основными данными. Контур управления цепями поставок. Контур управления финансами. Контур

	управления производством. Управление проектами. Управление персоналом. Аналитическая отчетность.
Тема 2.2. Информационные системы управления производством	Системы управления производственными процессами. Системы компьютеризированного интегрированного производства. Системы непрерывного управления жизненным циклом изделия. Концепция и системы управления непрерывным жизненным циклом изделия. Цифровое производство. Системы управления эффективностью деятельности организации. Концепция управления эффективностью деятельности организации. Характеристика основных процессов управления BPM-систем. Архитектура BPM-систем.

1.2. Методические материалы по подготовке к лабораторным занятиям по дисциплине (модулю)

Лабораторные занятия – одна из форм учебного занятия, направленная на развитие самостоятельности обучающихся и приобретение умений и навыков. Данные учебные занятия углубляют, расширяют, детализируют полученные ранее знания. Лабораторное занятие предполагает выполнение обучающимися по заданию и под руководством преподавателей одной или нескольких работ.

Цель лабораторных занятий состоит в развитии познавательных способностей, самостоятельности мышления и творческой активности обучающихся; углублении, расширении, детализировании знаний, полученных на лекции в обобщенной форме, и содействии выработке навыков профессиональной деятельности. В отдельных случаях на лабораторных занятиях руководителем занятия сообщаются дополнительные знания.

Для достижения поставленных целей и решения требуемого перечня задач лабораторные занятия проводятся традиционными технологиями или с использованием активных и интерактивных образовательных технологий.

При подготовке и работе во время проведения лабораторных занятий следует обратить внимание на следующие моменты: на процесс предварительной подготовки, на работу во время занятия, обработку полученных результатов, исправление полученных замечаний.

Предварительная подготовка к лабораторному занятию заключается в изучении теоретического материала в отведенное для самостоятельной работы время, ознакомление с инструктивными материалами с целью осознания задач занятия, техники безопасности при работе с приборами, веществами.

Работа во время проведения занятия включает:

- консультирование студентов преподавателями и вспомогательным персоналом с целью предоставления исчерпывающей информации, необходимой для самостоятельного выполнения предложенных преподавателем задач.
- самостоятельное выполнение заданий согласно обозначенной рабочей программой дисциплины (модуля) тематики.

2. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ) И САМОСТОЯТЕЛЬНОЙ РАБОТЕ

Освоение обучающимся дисциплины (модуля) «Интеграция информационных систем» предполагает изучение материалов дисциплины на аудиторных занятиях и в ходе самостоятельной работы. Аудиторные занятия проходят в форме лекций, семинаров, практических и лабораторных занятий. Самостоятельная работа включает разнообразный комплекс видов и форм работы обучающихся.

Для успешного освоения дисциплины (модуля) и достижения поставленных целей необходимо внимательно ознакомиться с рабочей программы дисциплины (модуля), доступной в электронной информационно-образовательной среде РГСУ.

Следует обратить внимание на списки основной и дополнительной литературы, на предлагаемые преподавателем ресурсы информационно-телекоммуникационной сети Интернет. Эта информация необходима для самостоятельной работы обучающегося.

При подготовке к аудиторным занятиям необходимо помнить особенности каждой формы его проведения.

Подготовка к учебному занятию лекционного типа заключается в следующем.

С целью обеспечения успешного обучения обучающийся должен готовиться к лекции, поскольку она является важнейшей формой организации учебного процесса, поскольку:

- знакомит с новым учебным материалом;
- разъясняет учебные элементы, трудные для понимания;
- систематизирует учебный материал;
- ориентирует в учебном процессе.

С этой целью:

- внимательно прочитайте материал предыдущей лекции;
- ознакомьтесь с учебным материалом по учебнику и учебным пособиям с темой прочитанной лекции;
- внесите дополнения к полученным ранее знаниям по теме лекции на полях лекционной тетради;
- запишите возможные вопросы, которые вы зададите лектору на лекции по материалу изученной лекции;
- постарайтесь уяснить место изучаемой темы в своей подготовке;
- узнайте тему предстоящей лекции (по тематическому плану, по информации лектора) и запишите информацию, которой вы владеете по данному вопросу.

Подготовка к лабораторному занятию.

При подготовке и работе во время проведения лабораторных занятий следует обратить внимание на следующие моменты: на процесс предварительной подготовки, на работу во время занятия, обработку полученных результатов, исправление полученных замечаний.

Предварительная подготовка к лабораторному занятию заключается в изучении теоретического материала в отведенное для самостоятельной работы время, ознакомление с инструктивными материалами с целью осознания задач занятия, техники безопасности при работе с приборами, веществами.

Работа во время проведения занятия включает:

- консультирование студентов преподавателями и вспомогательным персоналом с целью предоставления исчерпывающей информации, необходимой для самостоятельного выполнения предложенных преподавателем задач.
- самостоятельное выполнение заданий согласно обозначенной рабочей программой дисциплины (модуля) тематики.

Самостоятельная работа.

Самостоятельная работа - планируемая учебная, учебно-исследовательская, научно-исследовательская работа студентов, выполняемая во внеаудиторное (аудиторное) время по заданию и при методическом руководстве преподавателя, но без его непосредственного участия (при частичном непосредственном участии преподавателя, оставляющем ведущую роль за работой студентов).

Для более углубленного изучения темы задания для самостоятельной работы рекомендуется выполнять параллельно с изучением данной темы. При выполнении заданий по возможности используйте наглядное представление материала.

Самостоятельная работа студентов в ВУЗе является важным видом учебной и научной деятельности студента. Самостоятельная работа студентов играет значительную роль в рейтинговой технологии обучения.

К современному специалисту в области медицины общество предъявляет достаточно широкий перечень требований, среди которых немаловажное значение имеет наличие у выпускников определенных навыков (компетенций) и умения самостоятельно добывать знания из различных источников, систематизировать полученную информацию, давать оценку конкретной профессиональной ситуации. Формирование такого умения происходит в течение всего периода обучения.

В процессе самостоятельной работы студент приобретает необходимые для будущей специальности компетенции, навыки самоорганизации, самоконтроля, самоуправления, саморефлексии и становится активным самостоятельным субъектом учебной деятельности.

Самостоятельная работа студентов должна оказывать важное влияние на формирование личности будущего специалиста, его компетентность. Каждый студент самостоятельно определяет режим своей работы и меру труда, затрачиваемого на овладение учебным содержанием по каждой дисциплине. Он выполняет внеаудиторную работу по личному индивидуальному плану, в зависимости от его подготовки, времени и других условий.

Виды самостоятельной работы.

Работа с литературой.

При работе с книгой необходимо подобрать литературу, научиться правильно ее читать, вести записи. Для подбора литературы в библиотеке используются алфавитный и систематический каталоги. Важно помнить, что рациональные навыки работы с книгой - это всегда большая экономия времени и сил. Правильный подбор учебников рекомендуется преподавателем, читающим лекционный курс. Необходимая литература может быть также указана в методических разработках по данному курсу. Изучая материал по учебнику, следует переходить к следующему вопросу только после правильного уяснения предыдущего, описывая на бумаге все выкладки и вычисления (в том числе те, которые в учебнике опущены или на лекции даны для самостоятельного вывода). При изучении любой дисциплины большую и важную роль играет самостоятельная индивидуальная работа. Особое внимание следует обратить на определение основных понятий курса. Студент должен подробно разбирать примеры, которые поясняют такие определения, и уметь строить аналогичные примеры самостоятельно. Нужно добиваться точного представления о том, что изучаешь. Полезно составлять опорные конспекты. При изучении материала по учебнику полезно в тетради (на специально отведенных полях) дополнять конспект лекций. Там же следует отмечать вопросы, выделенные студентом для консультации с преподавателем. Выводы, полученные в результате изучения, рекомендуется в конспекте выделять, чтобы они при перечитывании записей лучше запоминались. Опыт показывает, что многим студентам помогает составление листа опорных сигналов, содержащего важнейшие и наиболее часто употребляемые формулы и понятия. Такой лист помогает запомнить формулы, основные положения лекции, а также может служить постоянным справочником для студента. Различают два вида чтения: первичное и вторичное. Первичное - это внимательное, неторопливое чтение, при котором можно остановиться на трудных местах.

Задача вторичного чтения - полное усвоение смысла целого (по счету это чтение может быть и не вторым, а третьим или четвертым). Самостоятельная работа с учебниками и книгами (а также самостоятельное теоретическое исследование проблем, обозначенных преподавателем на лекциях) – это важнейшее условие формирования у себя научного способа познания.

Чтение научного текста является частью познавательной деятельности. Ее цель – извлечение из текста необходимой информации. От того на сколько осознанно читающим собственная внутренняя установка при обращении к печатному слову (найти нужные сведения, усвоить информацию полностью или частично, критически проанализировать материал и т.п.) во многом зависит эффективность осуществляемого действия.

Методические рекомендации по составлению конспекта:

1. Внимательно прочитайте текст. Уточните в справочной литературе непонятные слова. При записи не забудьте вынести справочные данные на поля конспекта;
2. Выделите главное, составьте план;

3. Кратко сформулируйте основные положения текста, отметьте аргументацию автора;
4. Законспектируйте материал, четко следуя пунктам плана. При конспектировании старайтесь выразить мысль своими словами. Записи следует вести четко, ясно.
5. Грамотно записывайте цитаты. Цитируя, учитывайте лаконичность, значимость мысли. В тексте конспекта желательно приводить не только тезисные положения, но и их доказательства. При оформлении конспекта необходимо стремиться к емкости каждого предложения. Мысли автора книги следует излагать кратко, заботясь о стиле и выразительности написанного. Число дополнительных элементов конспекта должно быть логически обоснованным, записи должны распределяться в определенной последовательности, отвечающей логической структуре произведения. Для уточнения и дополнения необходимо оставлять поля. Овладение навыками конспектирования требует от студента целеустремленности, повседневной самостоятельной работы.

Методические материалы по самостоятельному решению задач

При самостоятельном решении задач нужно обосновывать каждый этап решения, исходя из теоретических положений курса. Если студент видит несколько путей решения проблемы (задачи), то нужно сравнить их и выбрать самый рациональный. Полезно до начала вычислений составить краткий план решения проблемы (задачи). Решение проблемных задач или примеров следует излагать подробно, вычисления располагать в строгом порядке, отделяя вспомогательные вычисления от основных. Решения при необходимости нужно сопровождать комментариями, схемами, чертежами и рисунками. Следует помнить, что решение каждой учебной задачи должно доводиться до окончательного логического ответа, которого требует условие, и по возможности с выводом. Полученный ответ следует проверить способами, вытекающими из существа данной задачи. Полезно также (если возможно) решать несколькими способами и сравнить полученные результаты. Решение задач данного типа нужно продолжать до приобретения твердых навыков в их решении.

Методические материалы к выполнению реферата

Реферат (от лат. *refere* – сообщать) – краткое изложение в письменном виде или в форме публикации доклада, содержания научного труда (трудов), литературы по теме. Работа над рефератом условно разделяется на выбор темы, подбор литературы, подготовку и защиту плана; написание теоретической части и всего текста с указанием библиографических данных используемых источников, подготовку доклада, выступление с ним. Тематика рефератов полностью связана с основными вопросами изучаемого курса.

Список литературы к темам не дается, и обучающиеся самостоятельно ведут библиографический поиск, причем им не рекомендуется ограничиваться университетской библиотекой.

Важно учитывать, что написание реферата требует от обучающихся определенных усилий. Особое внимание следует уделить подбору литературы, методике ее изучения с целью отбора и обработки собранного материала, обоснованию актуальности темы и теоретического уровня обоснованности используемых в качестве примеров фактов какой-либо деятельности.

Выбрав тему реферата, начав работу над литературой, необходимо составить план. Изучая литературу, продолжается обдумывание темы, осмысливание прочитанного, делаются выписки, сопоставляются точки зрения разных авторов и т.д. Реферативная работа сводится к тому, чтобы в ней выделились две взаимосвязанные стороны: во-первых, ее следует рассматривать как учебное задание, которое должен выполнить обучаемый, а во-вторых, как форму научной работы, творческого воображения при выполнении учебного задания.

Наличие плана реферата позволяет контролировать ход работы, избежать формального переписывания текстов из первоисточников.

Оформление реферата включает титульный лист, оглавление и краткий список использованной литературы. Список использованной литературы размещается на последней странице рукописи или печатной форме реферата. Реферат выполняется в письменной или печатной форме на белых листах формата А4 (210 x 297 мм). Шрифт Times New Roman, кегель 14, через 1,5 интервала при соблюдении следующих размеров текста: верхнее поле – 25 мм,

нижнее – 20 мм, левое – 30 мм, правое – 15 мм. Нумерация страниц производится сверху листа, по центру. Титульный лист нумерации не подлежит.

Рефераты должны быть написаны простым, ясным языком, без претензий на наукообразность. Следует избегать сложных грамматических оборотов, непривычных терминов и символов. Если же такие термины и символы все-таки приводятся, то необходимо разъяснять их значение при первом упоминании в тексте реферата.

Объем реферата предполагает тщательный отбор информации, необходимой для краткого изложения вопроса. Важнейший этап – редактирование готового текста реферата и подготовка к обсуждению. Обсуждение требует хорошей ориентации в материале темы, умения выделить главное, поставить дискуссионный вопрос, привлечь внимание слушателей к интересной литературе, логично и убедительно изложить свои мысли.

Рефераты обязательно подлежат защите. Процедура защиты начинается с определения оппонентов защищающего свою работу. Они стремятся дать основательный анализ работы обучающимся, обращают внимание на положительные моменты и недостатки реферата, дают общую оценку содержанию, форме преподнесения материала, характеру использованной литературы. Иногда они дополняют тот или иной раздел реферата. Последнее особенно ценно, ибо говорит о глубоком знании обучающимся-оппонентом изучаемой проблемы.

Обсуждение не ограничивается заслушиванием оппонентов. Другие обучающиеся имеют право уточнить или опровергнуть какое-либо утверждение. Преподаватель предлагает любому обучающемуся задать вопрос по существу доклада или попытаться подвести итог обсуждению.

Алгоритм работы над рефератом

1. Выбор темы

Тема должна быть сформулирована грамотно (с литературной точки зрения);

В названии реферата следует поставить четкие рамки рассмотрения темы;

Желательно избегать слишком длинных названий;

Следует по возможности воздерживаться от использования в названии спорных с научной точки зрения терминов, излишней наукообразности, а также чрезмерного упрощения формулировок.

2. Реферат следует составлять из пяти основных частей: введения; основной части; заключения; списка литературы; приложений.

3. Основные требования к введению:

Во введении не следует концентрироваться на содержании; введение должно включать краткое обоснование актуальности темы реферата, где требуется показать, почему данный вопрос может представлять научный интерес и есть ли связь представляемого материала с современностью. Таким образом, тема реферата должна быть актуальна либо с научной точки зрения, либо с современных позиций.

Очень важно выделить цель, а также задачи, которые требуется решить для выполнения цели.

Введение должно содержать краткий обзор изученной литературы, в котором указывается взятый из того или иного источника материал, кратко анализируются изученные источники, показываются их сильные и слабые стороны;

Объем введения составляет две страницы текста.

4. Требования к основной части реферата:

Основная часть содержит материал, отобранный для рассмотрения проблемы;

Также основная часть должна включать в себя собственно мнение обучающихся и сформулированные самостоятельные выводы, опирающиеся на приведенные факты;

Материал, представленный в основной части, должен быть логически изложен и распределен по параграфам, имеющим свои названия;

В изложении основной части необходимо использовать сноски (в первую очередь, когда приводятся цифры и чьи-то цитаты);

Основная часть должна содержать иллюстративный материал (графики, таблицы и т. д.);

Объем основной части составляет около 10 страниц.

5. Требования к заключению:

В заключении формулируются выводы по параграфам, обращается внимание на выдвинутые во введении задачи и цели;

Заключение должно быть четким, кратким, вытекающим из содержания основной части.

6. Требования к оформлению списка литературы (по ГОСТу):

Необходимо соблюдать правильность последовательности записи источников: сначала следует писать фамилию, а после инициалы; название работы не ставится в кавычки; после названия сокращенно пишется место издания; затем идет год издания; наконец, называется процитированная страница.

Критерии оценки реферата

Обучающийся, защищающий реферат, должен рассказать о его актуальности, поставленных целях и задачах, изученной литературе, структуре основной части, сделанных в ходе работы выводах.

По окончанию выступления ему может быть задано несколько вопросов по представленной проблеме.

Оценка складывается из соблюдения требований к реферату, грамотного раскрытия темы, умения четко рассказывать о представленном реферате, способности понять суть задаваемых по работе вопросов и найти точные ответы на них.

Методические материалы к выполнению эссе

Эссе – литературное произведение небольшого объема, обычно прозаическое, свободной композиции, передающее индивидуальные впечатления, суждения, соображения автора о той или иной проблеме, теме, о том или ином событии или явлении. Это вид самостоятельной исследовательской работы обучающихся, с целью углубления и закрепления теоретических знаний и освоения практических навыков. Цель эссе состоит в развитии самостоятельного творческого мышления и письменного изложения собственных мыслей. При написании эссе обучающийся должен представить развернутый письменный ответ на теоретический или практический актуальный вопрос, объявленный преподавателем в аудитории непосредственно перед ее написанием. В процессе написания эссе разрешается пользоваться нормативно-правовыми актами, конспектом лекций (в печатном виде). Использование интернет-ресурсов не допускается. Темы эссе преподаватель предлагает из числа тех, которые обучающиеся уже рассматривали на лекциях или семинарских занятиях, исходя из содержания заданий в составе оценочных средств. По решению преподавателя, в качестве темы эссе может быть выбрана одна или несколько тем, которые могут быть распределены между обучающимися по желанию.

Требования к выполнению эссе:

1. Проводится письменно.

2. Эссе выполняется на компьютере (гарнитура Times New Roman, шрифт 14) через 1,5 интервала с полями: верхнее, нижнее – 2; правое – 3; левое – 1,5. Отступ первой строки абзаца – 1,25. Сноски – постраничные. Таблицы и рисунки встраиваются в текст работы. При этом обязательный заголовок таблицы надо размещать над табличным полем, а рисунки сопровождать подрисуночными подписями. При включении в эссе нескольких таблиц и/или рисунков их нумерация обязательна. Обязательна и нумерация страниц. Их целесообразно проставлять внизу страницы – по середине или в правом углу. Номер страницы не ставится на титульном листе, но в общее число страниц он включается. Объем эссе, без учета приложений, не должен превышать 5 страниц. Значительное превышение установленного объема является недостатком работы и указывает на то, что обучающийся не сумел отобрать и переработать необходимый материал.

3. Работа должна содержать собственные умозаключения по сути поставленной проблемы, включать самостоятельно проведенный анализ по сути этой проблемы, выводы, обобщающие авторскую позицию по поставленной проблеме.

Критерии оценки эссе:

«Отлично» – исключительные знания материала, абсолютное понимание сути, безукоризненное знание основных понятий и положений, логически и лексически грамотно изложенный, содержательный, аргументированный, конкретный и исчерпывающий ответ.

«Хорошо» – глубокие знания материала, правильное понимание сути, знание основных понятий и положений, содержательный, полный и конкретный ответ.

«Удовлетворительно» – твердые, но недостаточно полные знания, верное понимание сути, в целом правильный ответ.

«Неудовлетворительно» – непонимание сущности задания, грубые ошибки в ответе.

Методические материалы по выполнению тестирования.

Тестовые задания содержат вопросы и 3-4 варианта ответа по базовым положениям изучаемой темы/раздела, составлены с расчетом на знания, полученные обучающимся в процессе изучения темы/раздела.

Тестовые задания выполняются в письменной форме и сдаются преподавателю, ведущему дисциплину (модуль). На выполнение тестовых заданий обучающимся отводится 45 минут.

При обработке результатов оценочной процедуры используются: критерии оценки по содержанию и качеству полученных ответов, ключи, оценочные листы.

Критерии оценки теста:

«Зачтено» - если обучающийся правильно выполнил не менее 2/3 всей работы или допустил не более одной грубой ошибки и двух недочетов, не более одной грубой и одной негрубой ошибки, не более трех негрубых ошибок, одной негрубой ошибки и трех недочетов, при наличии четырех-пяти недочетов.

«Не зачтено» - если число ошибок и недочетов превысило норму для оценки 3 или правильно выполнено менее 2/3 всей работы.

Методические материалы по выполнению доклада.

Рекомендуется следующая структура доклада:

1. титульный лист, содержание доклада;
2. краткое изложение;
3. цели и задачи;
4. изложение характера исследований и рассмотренных проблем, гипотезы, спорные вопросы;
5. источники информации, методы сбора и анализа данных, степень их полноты и достоверности;
6. анализ и толкование полученных в работе результатов;
7. выводы и оценки;
8. библиография и приложения.

Время выступления докладчика не должно превышать 10 минут.

Основные требования к оформлению доклада:

- титульный лист должен включать название доклада, наименование предметной (цикловой) комиссии, фамилию обучающегося;
- все использованные литературные источники сопровождаются библиографическим описанием;
- приводимая цитата из источника берется в кавычки (оформляются сноски);
- единицы измерения должны применяться в соответствии с действующими стандартами;
- все названия литературных источников следует приводить в соответствии с новейшими изданиями;
- рекомендуется включение таблиц, графиков, схем, если они отражают основное содержание или улучшают ее наглядность;
- названия фирм, учреждений, организаций и предприятий должны именоваться так, как они указываются в источнике;

Критерии оценки доклада

При выполнении доклада обучающийся должен продемонстрировать умение кратко излагать прочитанный материал, а также умение обобщать и анализировать материал по теме доклада.

Презентация

Методические материалы к презентациям

1. Объём презентации 10 -20 слайдов.
2. На титульном слайде должно быть отражено:
 - наименование факультета;
 - тема презентации;
 - фамилия, имя, отчество, направление подготовки/ специальность, направленность (профиль)/ специализация, форма обучения, номер группы автора презентации;
 - фамилия, имя, отчество, степень, звание, должность руководитель работы;
 - год выполнения работы.
3. В презентации должны быть отражены обоснование актуальности представляемого материала, цели и задачи работы.
4. Содержание презентации должно включать наиболее значимый материал доклада, а также, при необходимости, таблицы, диаграммы, рисунки, фотографии, карты, видео – вставки, звуковое сопровождение.
5. Заключительный слайд должен содержать информацию об источниках информации для презентации.

Критерии оценки презентации

1. Объём презентации 10 -20 слайдов.
2. Правильность оформления титульного слайда.
3. Актуальность отобранного материала, обоснованность формулировки цели и задач работы.
4. Наглядность и логичность презентации, обоснованность использования таблиц, диаграмм, рисунков, фотографий, карт, видео – вставок, звукового сопровождения; правильный выбор шрифтов, фона, других элементов дизайна слайда.
5. Объём и качество источников информации (не менее 2-х интернет – источников и не менее 2-х литературных источников).

Методические материалы по подготовке к опросу

Самостоятельная работа обучающихся включает подготовку к опросу на практических занятиях. Для этого обучающийся изучает лекции, основную и дополнительную литературу, публикации, информацию из Интернет-ресурсов.

Тема и вопросы к практическим занятиям, вопросы для самоконтроля содержатся в рабочей программе и доводятся до студентов заранее.

Для подготовки к опросу обучающемуся необходимо ознакомиться с материалом, посвященным теме практического занятия, в учебнике или другой рекомендованной литературе, конспекте лекции, обратить внимание на усвоение основных понятий дисциплины, выявить неясные вопросы и подобрать дополнительную литературу для их освещения.

Критерии оценки опроса

- «Отлично»:
- дан полный, развернутый ответ на поставленный вопрос;
 - в ответе прослеживается четкая структура, логическая последовательность, отражающая сущность раскрываемых понятий, теорий, явлений;
 - знание по предмету демонстрируется на фоне понимания его в системе данной науки и междисциплинарных связей;
 - свободное владение терминологией;
 - ответы на дополнительные вопросы четкие, краткие;
- «Хорошо»:
- дан полный, развернутый ответ на поставленный вопрос, показано умение выделять существенные и несущественные признаки, причинно-следственные связи;

- ответ недостаточно логичен с единичными ошибками в частностях, исправленные студентом с помощью преподавателя;
- единичные ошибки в терминологии;
- ответы на дополнительные вопросы правильные, недостаточно полные и четкие.

«Удовлетворительно»:

– ответ не полный, с ошибками в деталях, умение раскрыть значение обобщённых знаний не показано, речевое оформление требует поправок, коррекции;

– логика и последовательность изложения имеют нарушения, студент не способен самостоятельно выделить существенные и несущественные признаки и причинно-следственные связи;

- ошибки в раскрываемых понятиях, терминах;
- студент не ориентируется в теме, допускает серьезные ошибки;
- студент не может ответить на большую часть дополнительных вопросов.

«Неудовлетворительно»:

– ответ представляет собой разрозненные знания с существенными ошибками по вопросу;

– присутствуют фрагментарность, нелогичность изложения, студент не осознает связь обсуждаемого вопроса с другими объектами дисциплины, речь неграмотная;

- незнание терминологии;
- ответы на дополнительные вопросы неправильные.

Методические материалы по выполнению практического задания

При выполнении практического задания обучающийся придерживается следующего алгоритма:

1. Записать дату, тему и цель задания;
2. Ознакомиться с правилами и условия выполнения практического задания;
3. Повторить теоретические задания, необходимые для рациональной работы и других практических действий, используя конспекты лекций и рекомендованную литературу, представленной в программе;
4. Выполнить работу по предложенному алгоритму действий;
5. Обобщить результаты работы, сформулировать выводы / дать ответы на контрольные вопросы;

Работа должна быть выполнена грамотно, с соблюдением культуры изложения. При использовании данных из учебных, методических пособий и другой литературы, периодических изданий, Интернет-источников должны иметься ссылки на вышеперечисленные.

Критерии оценки практического задания:

«Отлично» – правильный ответ, дается четкое обоснование принятому решению; рассуждения четкие последовательные логические; используются ссылки на полученные при изучении дисциплины знания; правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Хорошо» – правильный ответ, дается обоснование принятому решению; но с не существенными ошибками, в рассуждениях отсутствует логическая последовательность; используются ссылки на полученные при изучении дисциплины знания, правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Удовлетворительно» – правильный ответ, допускаются грубые ошибки в обосновании принятого решения; рассуждения не последовательные сумбурные; используются ссылки на полученные при изучении дисциплины знания; используются формулы, процедуры, понятия, имеющие прямое значение для подтверждения принятого решения, однако, при обращении к ним допускаются серьезные ошибки, студент не может правильно ими воспользоваться.

«Неудовлетворительно, не зачтено» – ответ неверный, отсутствует обоснование принятому решению; студент демонстрирует полное непонимание сути вопроса.

Для оценки решения ситуационной задачи (аналитического задания):

Оценка «отлично» выставляется, если задача решена грамотно, ответы на вопросы сформулированы четко. Эталонный ответ полностью соответствует решению студента, которое хорошо обосновано теоретически.

Оценка «хорошо» выставляется, если задача решена, ответы на вопросы сформулированы не достаточно четко. Решение студента в целом соответствует эталонному ответу, но не достаточно хорошо обосновано теоретически.

Оценка «удовлетворительно» выставляется, если задача решена не полностью, ответы не содержат всех необходимых обоснований решения.

Оценка «неудовлетворительно» выставляется, если задача не решена или имеет грубые теоретические ошибки в ответе на поставленные вопросы.

Методические материалы по выполнению лабораторного задания

При выполнении лабораторного задания обучающийся придерживается следующего алгоритма:

1. Записать дату, тему и цель задания;
2. Ознакомиться с правилами и условия выполнения задания;
3. Повторить теоретические задания, необходимые для рациональной работы и других практических действий, используя конспекты лекций и рекомендованную литературу, представленной в программе;
4. Выполнить работу по предложенному алгоритму действий;
5. Обобщить результаты работы, сформулировать выводы / дать ответы на контрольные вопросы;

Работа должна быть выполнена грамотно, с соблюдением культуры изложения. При использовании данных из учебных, методических пособий и другой литературы, периодических изданий, Интернет-источников должны иметься ссылки на вышеперечисленные.

Критерии оценки лабораторного задания:

«Отлично» – правильный ответ, дается четкое обоснование принятому решению; рассуждения четкие последовательные логические; используются ссылки на полученные при изучении дисциплины знания; правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Хорошо» – правильный ответ, дается обоснование принятому решению; но с не существенными ошибками, в рассуждениях отсутствует логическая последовательность; используются ссылки на полученные при изучении дисциплины знания, правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Удовлетворительно» – правильный ответ, допускаются грубые ошибки в обосновании принятого решения; рассуждения не последовательные сумбурные; используются ссылки на полученные при изучении дисциплины знания; используются формулы, процедуры, понятия, имеющие прямое значение для подтверждения принятого решения, однако, при обращении к ним допускаются серьезные ошибки, студент не может правильно ими воспользоваться.

«Неудовлетворительно, не зачтено» – ответ неверный, отсутствует обоснование принятому решению; студент демонстрирует полное непонимание сути вопроса.

Методические указания для подготовки к промежуточной аттестации.

Изучение учебных дисциплин (модулей) завершается зачетом/зачетом с оценкой или экзаменом. Подготовка к промежуточной аттестации способствует закреплению, углублению и обобщению знаний, получаемых, в процессе обучения, а также применению их к решению практических задач. Готовясь к экзамену, студент ликвидирует имеющиеся пробелы в знаниях, углубляет, систематизирует и упорядочивает свои знания. На зачете или экзамене студент демонстрирует то, что он освоил в процессе обучения по дисциплине (модулю).

Вначале следует просмотреть весь материал по дисциплине (модулю), отметить для себя трудные вопросы. Обязательно в них разобраться. В заключение еще раз целесообразно повторить основные положения, используя при этом листы опорных сигналов. Систематическая подготовка к занятиям в течение семестра позволит использовать время промежуточной аттестации для систематизации знаний.

3. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНИВАНИЯ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ И (ИЛИ) ОПЫТА ДЕЯТЕЛЬНОСТИ, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ

3.1. Организационные основы применения балльно-рейтинговой системы оценки успеваемости обучающихся по дисциплине (модулю)

Оценка качества освоения обучающимися дисциплины (модуля) реализуется в формате балльно-рейтинговой системы оценки успеваемости обучающихся (БРСО).

БРСО в ходе текущего контроля успеваемости и промежуточной аттестации осуществляется по 100-балльной шкале.

Академический рейтинг обучающегося по дисциплине (модулю) складывается из результатов:

- текущего контроля успеваемости (максимальный текущий рейтинг обучающегося 80 рейтинговых баллов);

- промежуточной аттестации (максимальный рубежный рейтинг обучающегося 20 рейтинговых баллов).

Условия оценки освоения обучающимся дисциплины (модуля) в формате БРСО доводятся преподавателем до сведения обучающихся на первом учебном занятии, а также размещены в свободном доступе в электронной информационно-образовательной среде Университета.

3.2. Проведение текущего контроля успеваемости обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося

В течение учебного семестра до промежуточной аттестации на основании утвержденной рабочей программы дисциплины (модуля) формируется текущий рейтинг обучающегося. Текущий рейтинг обучающегося складывается как сумма рейтинговых баллов, полученных им в течение учебного семестра по всем видам учебных занятий по учебной дисциплине.

В процессе текущего контроля оцениваются следующие действия обучающегося, направленные на освоение компетенций в рамках изучения учебной дисциплины:

- академическая активность (посещаемость учебных занятий, самостоятельное изучение содержания учебной дисциплины в электронной информационно-образовательной среде, соблюдение сроков сдачи практических заданий и текущих контрольных мероприятий и др.);

- выполнение и сдача текущих и итогового практических заданий (эссе, рефераты, творческие задания, кейс-задания, лабораторные работы, расчетные задания и др., активное участие в групповых интерактивных занятиях (дискуссии, WiKi-проекты и др.), защита проектов и др.);

- прохождение рубежей текущего контроля, включая соблюдение графика их прохождения в электронной информационно-образовательной среде.

Для планирования расчета текущего рейтинга обучающегося используются следующие пропорции:

Вид учебного действия	Максимальная рейтинговая оценка, баллов
академическая активность	10
практические задания	40
из них: текущие практические задания	20
итоговое практическое задание	20
рубежи текущего контроля	30
ИТОГО:	80

В течение учебного семестра по дисциплине (модулю) обучающимся должен быть накоплен текущий рейтинг не менее 52 рейтинговых баллов (65% от максимального значения текущего рейтинга).

Необходимыми условиями допуска обучающегося к промежуточной аттестации по дисциплине являются положительное прохождение обучающимся не менее 65% рубежей текущего контроля с накоплением не менее 65% максимального рейтингового балла за каждый рубеж текущего контроля и положительное выполнение итогового практического задания с накоплением не менее 65% максимального рейтингового балла, установленного за итоговое практическое задание.

Невыполнение вышеуказанных условий является текущей академической задолженностью, которая должна быть ликвидирована обучающимся до контрольного мероприятия промежуточной аттестации.

Сведения о наличии у обучающихся текущей академической задолженности, сроках и порядке добора рейтинговых баллов для её ликвидации доводятся до обучающихся педагогическим работником.

В случае неликвидации текущей академической задолженности, педагогический работник обязан во время контрольного мероприятия промежуточной аттестации поставить обучающемуся 0 рейтинговых баллов. В этом случае ликвидация текущей академической задолженности возможна в периоды проведения повторной промежуточной аттестации.

3.3. Проведение промежуточной аттестации обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося

Промежуточная аттестация по дисциплине (модулю) проводится в соответствии с Положением о промежуточной аттестации обучающихся по основным профессиональным образовательным программам высшего образования – программ специалитета в Российском государственном социальном университете и Положением о балльно-рейтинговой системе оценки успеваемости обучающихся по основным профессиональным образовательным программам высшего образования - программам бакалавриата, программам специалитета, программам магистратуры в Российском государственном социальном университете.

На промежуточную аттестацию отводится 20 рейтинговых баллов.

Ответы обучающегося на контрольном мероприятии промежуточной аттестации оцениваются педагогическим работником по 20 - балльной шкале, а итоговая оценка по дисциплине (модулю) выставляется по пятибалльной системе для дифференцированного зачета.

Критерии выставления оценки определяются Положением о балльно-рейтинговой системе оценки успеваемости обучающихся по основным профессиональным образовательным программам высшего образования - программам бакалавриата, программам специалитета, программам магистратуры в Российском государственном социальном университете.

В процессе определения рубежного рейтинга обучающегося используется следующая шкала:

Рубежный рейтинг	Критерии оценки освоения обучающимся учебной дисциплины в ходе контрольных мероприятий промежуточной аттестации
19-20 рейтинговых баллов	обучающийся глубоко и прочно усвоил программный материал, исчерпывающе, последовательно, грамотно и логически стройно его излагает, тесно увязывает с задачами и будущей деятельностью, не затрудняется с ответом при видоизменении задания, свободно справляется с задачами и практическими заданиями, правильно обосновывает принятые решения, умеет самостоятельно обобщать и излагать материал, не допуская ошибок

16-18 рейтинговых баллов	обучающийся твердо знает программный материал, грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос, может правильно применять теоретические положения и владеет необходимыми умениями и навыками при выполнении практических заданий
13-15 рейтинговых баллов	обучающийся освоил основной материал, но не знает отдельных деталей, допускает неточности, недостаточно правильные формулировки, нарушает последовательность в изложении программного материала и испытывает затруднения в выполнении практических заданий
1-12 рейтинговых баллов	обучающийся не знает значительной части программного материала, допускает существенные ошибки, с большими затруднениями выполняет практические задания
0 рейтинговых баллов	не аттестован

Если результат контроля успеваемости в рамках проведения контрольных мероприятий промежуточной аттестации (рубежный рейтинг обучающегося) неудовлетворительный (получено менее 13 рейтинговых баллов), то промежуточная аттестация по учебной дисциплине (модулю) невозможна даже при наличии высокого текущего рейтинга, полученного по итогам текущего контроля по учебной дисциплине (модулю).

ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

№ п/п	Содержание изменения	Реквизиты документа об утверждении изменения	Дата введения изменения
1.	Утверждены и введены в действие решением Ученого совета факультета на основании Федерального государственного образовательного стандарта (указываем реквизиты ФГОС)	Протокол заседания Ученого совета факультета № _____ от «_____» _____ 20____ года	____.____.____
2.	*	Протокол заседания Ученого совета факультета № _____ от «_____» _____ 20____ года	____.____.____
3.	*	Протокол заседания Ученого совета факультета № _____ от «_____» _____ 20____ года	____.____.____
4.	*	Протокол заседания Ученого совета факультета № _____ от «_____» _____ 20____ года	____.____.____



**Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Российский государственный социальный университет»**

УТВЕРЖДАЮ

декан факультета политических и социальных

технологий  /Пивнева С.В./

«28» февраля 2024 года

МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ

ПРОГРАММИРОВАНИЕ

Направление подготовки

«Информационные системы и технологии»

Направленность

«Интеллектуальные информационные системы и технологии»

**ОБРАЗОВАТЕЛЬНАЯ ПРОГРАММА ВЫСШЕГО ОБРАЗОВАНИЯ - ПРОГРАММА
БАКАЛАВРИАТА**

Форма обучения

Очная, заочная

Методические материалы по дисциплине (модулю) «Программирование» разработаны на основании федерального государственного образовательного стандарта высшего образования – *бакалавриата* по направлению подготовки *09.03.02 Информационные системы и технологии*, утвержденного приказом Министерства образования и науки Российской Федерации от 19.09.2017 № 926, учебного плана по основной профессиональной образовательной программе высшего образования – программе бакалавриата по направлению подготовки *09.03.02 Информационные системы и технологии* (далее – «ОПОП»).

Методические материалы по дисциплине (модулю) разработаны рабочей группой в составе: М.Е. Головкин.

Методические материалы по дисциплине (модулю) обсуждены и утверждены на заседании кафедры информационных технологий, искусственного интеллекта и общественно-социальных технологий цифрового общества факультета социальных и политических технологий. Протокол № 11 от «28» февраля 2024 года.

Заведующий кафедрой
канд. пед. наук, доцент



С.В. Пивнева

СОДЕРЖАНИЕ

1. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ К ЛЕКЦИОННЫМ, ЛАБОРАТОРНЫМ ЗАНЯТИЯМ	4
1.1. Методические материалы к проведению лекционных занятий по дисциплине (модулю)	4
1.2. Методические материалы по подготовке к практическим занятиям по дисциплине (модулю)	7
1.3. Учебно-наглядные пособия по разделам (темам) дисциплины (модуля)	14
1.4. Методические материалы по подготовке к лабораторным занятиям по дисциплине (модулю)	18
2. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ) И САМОСТОЯТЕЛЬНОЙ РАБОТЕ	18
3. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНИВАНИЯ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ И (ИЛИ) ОПЫТА ДЕЯТЕЛЬНОСТИ, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ.....	28
3.1. Организационные основы применения балльно-рейтинговой системы оценки успеваемости обучающихся по дисциплине (модулю)	28
3.2. Проведение текущего контроля успеваемости обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося.....	28
3.3. Проведение промежуточной аттестации обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося.....	29
ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ	31

1. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ К ЛЕКЦИОННЫМ, ЛАБОРАТОРНЫМ ЗАНЯТИЯМ

1.1. Методические материалы к проведению лекционных занятий по дисциплине (модулю)

Лекция - один из методов обучения, одна из основных системообразующих форм организации учебного процесса в вузе. Лекционное занятие представляет собой систематическое, последовательное, монологическое изложение педагогическим работником учебного материала, как правило, теоретического характера. Такое занятие представляет собой элемент технологии представления учебного материала путем логически стройного, систематически последовательного и ясного изложения.

Цель лекции – организация целенаправленной познавательной деятельности обучающихся по овладению программным материалом дисциплины (модуля). Чтение курса лекций позволяет дать связанное, последовательное изложение материала в соответствии с новейшими данными науки, сообщить слушателям основное содержание предмета в целостном, систематизированном виде. В ряде случаев лекция выполняет функцию основного источника информации: при отсутствии учебников и учебных пособий, чаще по новым курсам; в случае, когда новые научные данные по той или иной теме не нашли отражения в учебниках; отдельные разделы и темы очень сложны для самостоятельного изучения. В таких случаях только лектор может методически помочь обучающимся в освоении сложного материала.

Возможные формы проведения лекций:

- Вводная лекция – один из наиболее важных и трудных видов лекции при чтении систематических курсов. От успеха этой лекции во многом зависит успех усвоения всего курса. Она может содержать: определение дисциплины (модуля); краткую историческую справку о дисциплине (модуле); цели и задачи дисциплины (модуля), ее роль в общей системе обучения и связь со смежными дисциплинами (модулями); основные проблемы (понятия и определения) данной науки; основную и дополнительную учебную литературу; особенности самостоятельной работы обучающихся над дисциплиной (модулем) и формы участия в научно-исследовательской работе; отчетность по курсу.

- Информационная лекция ориентирована на изложение и объяснение обучающимся научной информации, подлежащей осмыслению и запоминанию. Это самый традиционный тип лекций в практике высшей школы.

- Заключительная лекция предназначена для обобщения полученных знаний и раскрытия перспектив дальнейшего развития данной науки.

- Обзорная лекция – это систематизация научных знаний на высоком уровне, допускающая большое число ассоциативных связей в процессе осмысления информации, излагаемой при раскрытии внутрипредметной и междпредметной связей, исключая детализацию и конкретизацию. Как правило, стержень излагаемых теоретических положений составляет научно-понятийная и концептуальная основа всего курса или крупных его разделов.

- Лекция-беседа - непосредственный контакт педагогического работника с аудиторией - диалог. По ходу лекции педагогический работник задает вопросы для выяснения мнений и уровня осведомленности обучающихся по рассматриваемой проблеме.

- Лекция-дискуссия - свободный обмен мнениями в ходе изложения лекционного материала. Педагогический работник активизирует участие в обсуждении отдельными вопросами, сопоставляет между собой различные мнения и тем самым развивает дискуссию, стремясь направить ее в нужное русло.

- Лекция с применением обратной связи включает в себе то, что в начале и конце каждого раздела лекции задаются вопросы. Первый - для того, чтобы узнать, насколько обучающиеся ориентируются в излагаемом материале, вопрос в конце раздела предназначен для выяснения степени усвоения только что изложенного материала. При

неудовлетворительных результатах контрольного опроса педагогический работник возвращается к уже прочитанному разделу, изменив при этом методику подачи материала.

- Проблемная лекция опирается на логику последовательно моделируемых проблемных ситуаций путем постановки проблемных вопросов или предъявления проблемных задач. Проблемный вопрос - это диалектическое противоречие, требующее для своего решения размышления, сравнения, поиска, приобретения и применения новых знаний. Проблемная задача содержит дополнительную вводную информацию и при необходимости некоторые ориентиры поиска ее решения.

- Программированная лекция - консультация – педагогический работник сам составляет и предлагает обучающимся вопросы. На подготовленные вопросы педагогический работник сначала просит ответить обучающихся, а затем проводит анализ и обсуждение неправильных ответов. В лекциях можно использовать наглядные материалы, а также подготовить презентацию. Что касается презентации, то в качестве визуальной поддержки ее можно органично интегрировать во все вышеупомянутые лекции. В то же время лекцию-презентацию возможно выделить и в качестве самостоятельной формы. Лекция-презентация должна отражать суть основных и (или) проблемных вопросов лекции, на которые особо следует обратить внимание обучающихся. В условиях применения активного метода проведения занятий презентация представляется весьма удачным способом донесения информации до слушателей. Единственное, на что следует обратить внимание при подготовке слайдов, - это их оформление и текст. Слайд не должен быть перегружен картинками и лишней информацией, которая будет отвлекать от основного аспекта того или иного вопроса лекции. Во время лекции можно задавать вопросы аудитории в отношении того или иного слайда, тем самым еще больше вовлекая обучающихся в проблематику.

Краткое содержание лекционных занятий

Наименование разделов и тем	Содержание учебного материала
РАЗДЕЛ 1. Алгоритмы. Языки программирования.	
Тема 1.1. Основы алгоритмизации.	Разработка алгоритма как один из начальных этапов программирования.
Тема 1.2. Языки и системы программирования.	Общее описание языков и систем программирования.
РАЗДЕЛ 2. Теоретические основы разработки алгоритмов и программ.	
Тема 2.1. Программирование вычислительных алгоритмов на языке высокого уровня (по выбору: Паскаль, JAVA, C).	Примеры разработки алгоритмов и программ на различных языках высокого уровня.
Тема 2.2. Методы программирования.	Изучение основных методик программирования
РАЗДЕЛ 3. Сортировка данных.	
Тема 3.1. Пузырьковая сортировка. Метод декомпозиции.	Реализация пузырьковой сортировки данных и метода декомпозиции в языках программирования.
Тема 3.2. Оценки эффективности алгоритмов сортировки	Критерии эффективности алгоритмов сортировки.
РАЗДЕЛ 4. Структуры данных.	

Тема 4.1. Стеки, очереди, списки и операции над ними.	Описание алгоритмов работы стеков, очередей, списков, а также связей их элементов друг с другом.
Тема 4.2. Бинарные деревья. Операции с бинарным деревом поиска.	Описание бинарных деревьев и алгоритмов работы с ними.
РАЗДЕЛ 5. Основные понятия	
Тема 5.1. Основные понятия языка С.	Основные понятия языка программирования С.
Тема 5.2. Современные системы программирования	Системы программирования для работы на языке С.
РАЗДЕЛ 6. Переменные, выражения и работа с операторами	
Тема 6.1. Переменные и типы данных. Выражения.	Имена и типы переменных, правила их объявления. Примеры выражений.
Тема 6.2. Операторы if, else, switch, while, do-while, for.	Правила и особенности работы с операторами ветвления и цикла
РАЗДЕЛ 7. Структуры данных. Работа с указателями.	
Тема 7.1. Массивы. Структуры.	Объявление массивов и структур. Принципы работы с ними.
Тема 7.2. Указатели. Строки.	Объявление указателей и правила работы с ними. Альтернативные представления строк как массивов символов и как указателей на элементы типа char.
РАЗДЕЛ 8. Основы объектно-ориентированного программирования на языке С++	
Тема 8.1. Понятие класса. Поля класса.	Класс как множество объектов с одинаковым поведением. Переменные в составе класса.
Тема 8.2. Методы в классе. Конструкторы класса.	Описание функций класса – методов. Принципы работы с конструкторами класса.
РАЗДЕЛ 9. Работа с объектами	
Тема 9.1. Создание объектов класса. Обращение к полям и методам класса.	Способы создания объектов класса. Способы обращения к полям и методам объектов классов.
Тема 9.2. Деструктор класса. Освобождение памяти, выделенной для объекта класса.	Описание деструктора класса. Способы очистки памяти в современном объектно-ориентированном программировании.
РАЗДЕЛ 10. Наследование классов	
Тема 10.1. Понятие о наследовании.	Наследование как способ создания новых классов на основе существующих.
Тема 10.2. Правила наследования полей и методов для различных модификаторов доступа.	Воздействие модификаторов на степень доступа к полям и классам базового класса для объекта производного класса.
РАЗДЕЛ 11. Инкапсуляция и полиморфизм.	
Тема 11.1. Инкапсуляция и способы её достижения в языке С++.	Инкапсуляция как основной способ сокрытия данных внутри класса.
Тема 11.2. Полиморфизм и его использование в языке С++.	Проявления полиморфизма и особенности перегрузки функций, а также методов классов в С++.
РАЗДЕЛ 12. Основы программирования на языке С#.	

Тема 12.1. Структура программы на C#.	Структура кода программы на языке C#.
Тема 12.2. Особенности работы с объектами и классами на C#.	Отличия синтаксиса C# от C++.
РАЗДЕЛ 13. Обработка исключительных ситуаций.	
Тема 13.1. Исключительные ситуации и их классы.	Понятие об исключительной ситуации как о критической ошибке. Классы и типы исключительных ситуаций.
Тема 13.2. Блоки try, catch, finally, throw.	Использование блоков try, catch, finally, throw в обработке исключительных ситуаций.
РАЗДЕЛ 14. Интерфейсы. Делегаты и лямбда-выражения.	
Тема 14.1. Объявление интерфейсов. Реализация множественного наследования.	Интерфейс как особый вид абстрактного класса. Реализация интерфейсов и множественного наследования с их применением.
Тема 14.2. Объявление и применение делегатов. Работа с лямбда-выражениями.	Делегаты и лямбда-выражения как альтернатива использованию функций в C#.
РАЗДЕЛ 15. Разработка приложений с графическим интерфейсом.	
Тема 15.1. Создание приложений с графическим интерфейсом в Visual C#. Форма и проект программы.	Принципы конструирования приложений с оконным интерфейсом. Свойства компонентов интерфейса и события.
Тема 15.2. Работа с кнопками, меню, списками ListBox и ComboBox.	Основные свойства и события для компонентов Button (кнопка), ListBox, ComboBox. Создание меню приложения.

1.2. Методические материалы по подготовке к практическим занятиям по дисциплине (модулю)

Практические (семинарские) занятия - одна из форм учебного занятия, направленная на развитие самостоятельности обучающихся и приобретение умений и навыков. Данные учебные занятия углубляют, расширяют, детализируют полученные ранее знания. Практическое занятие предполагает выполнение обучающимися по заданию и под руководством преподавателей одной или нескольких практических работ.

Цель практических занятий и семинаров состоит в развитии познавательных способностей, самостоятельности мышления и творческой активности обучающихся; углублении, расширении, детализировании знаний, полученных на лекции в обобщенной форме, и содействии выработке навыков профессиональной деятельности. В отдельных случаях на практических занятиях и семинарах руководителем занятия сообщаются дополнительные знания.

Для достижения поставленных целей и решения требуемого перечня задач практические занятия и семинары проводятся традиционными технологиями или с использованием активных и интерактивных образовательных технологий.

Возможные формы проведения практических (семинарских) занятий:

- Деловая игра - это метод группового обучения совместной деятельности в процессе решения общих задач в условиях максимально возможного приближения к реальным проблемным ситуациям. Имитационные игры - на занятиях имитируется деятельность какой-либо организации, предприятия или его подразделения. Имитироваться могут события, конкретная деятельность людей (деловое совещание, обсуждение плана) и обстановка, условия, в которых происходит событие или осуществляется деятельность (кабинет начальника цеха, зал заседаний). Исполнение ролей (ролевые игры) - в этих играх

отрабатывается тактика поведения, действий, выполнение функций и обязанностей конкретного лица. Для проведения игр с исполнением роли разрабатывается модель-пьеса ситуации, между студентами распределяются роли с «обязательным содержанием», характеризующиеся различными интересами; в процессе их взаимодействия должно быть найдено компромиссное решение. «Деловой театр» (метод инсценировки) - в нем разыгрывается какая-либо ситуация, поведение человека в этой обстановке, обучающийся должен вжиться в образ определенного лица, понять его действия, оценить обстановку и найти правильную линию поведения. Основная задача метода инсценировки - научить ориентироваться в различных обстоятельствах, давать объективную оценку своему поведению, учитывать возможности других людей, влиять на их интересы, потребности и деятельность, не прибегая к формальным атрибутам власти, к приказу.

- Игровое проектирование - является практическим занятием или циклом занятий, суть которых состоит в разработке инженерного, конструкторского, технологического и других видов проектов в игровых условиях, максимально воссоздающих реальность. Этот метод отличается высокой степенью сочетания индивидуальной и совместной работы обучающихся.

- Познавательно-дидактические игры не относятся к деловым играм. Они предполагают лишь включение изучаемого материала в необычный игровой контекст и иногда содержат лишь элементы ролевых игр. Такие игры могут проводиться в виде копирования научных, культурных, социальных явлений (конкурс знатоков, «Поле чудес», КВН и т.д.) и в виде предметно-содержательных моделей, (например, игры-путешествия, когда надо разработать рациональный маршрут, пользуясь различными картами).

- Анализ конкретных ситуаций. Конкретная ситуация – это любое событие, которое содержит в себе противоречие или вступает в противоречие с окружающей средой. Ситуации могут нести в себе как позитивный, так и отрицательный опыт. Все ситуации делятся на простые, критические и экстремальные.

- Кейс-метод (от английского case – случай, ситуация) – усовершенствованный метод анализа конкретных ситуаций, метод активного проблемно-ситуационного анализа, основанный на обучении путем решения конкретных задач – ситуаций (решение кейсов). Непосредственная цель метода case-study - обучающиеся должны проанализировать ситуацию, разобраться в сути проблем, предложить возможные решения и выбрать лучшее из них. Кейсы делятся на практические (отражающие реальные жизненные ситуации), обучающие (искусственно созданные, содержащие значительные элементы условности при отражении в нем жизни) и исследовательские (ориентированные на проведение исследовательской деятельности посредством применения метода моделирования). Метод конкретных ситуаций (метод case-study) относится к неигровым имитационным активным методам обучения.

- Тренинг (англ. training от train — обучать, воспитывать) – метод активного обучения, направленный на развитие знаний, умений и навыков и социальных установок. Тренинг – форма интерактивного обучения, целью которого является развитие компетентности межличностного и профессионального поведения в общении. Достоинство тренинга заключается в том, что он обеспечивает активное вовлечение всех участников в процесс обучения. Можно выделить основные типы тренингов по критерию направленности воздействия и изменений – навыковый, психотерапевтический, социально-психологический, бизнес-тренинг.

- Метод Сократа (Майевтика) – метод вопросов, предполагающих критическое отношение к догматическим утверждениям, называется еще как метод «сократовской иронии». Это умение извлекать скрытое в человеке знание с помощью искусных наводящих вопросов, подразумевающего короткий, простой и заранее предсказуемый ответ.

- Интерактивная лекция – выступление ведущего обучающего перед большой аудиторией с применением следующих активных форм обучения: дискуссия, беседа, демонстрация слайдов или учебных фильмов, мозговой штурм.

- Групповая, научная дискуссия, диспут. Дискуссия — это целенаправленное обсуждение конкретного вопроса, сопровождающееся обменом мнениями, идеями между двумя и более лицами. Задача дискуссии - обнаружить различия в понимании вопроса и в споре установить истину. Дискуссии могут быть свободными и управляемыми. К технике управляемой дискуссии относятся: четкое определение цели, прогнозирование реакции оппонентов, планирование своего поведения, ограничение времени на выступления и их заданная очередность. Разновидностью свободной дискуссии является форум, где каждому желающему дается неограниченное время на выступление, при условии, что его выступление вызывает интерес аудитории. Каждый конкретный форум имеет свою тематику — достаточно широкую, чтобы в её пределах можно было вести многоплановое обсуждение.

- Дебаты — это чётко структурированный и специально организованный публичный обмен мыслями между двумя сторонами по актуальным темам. Это разновидность публичной дискуссии участников дебатов, направляющая на переубеждение в своей правоте третьей стороны, а не друг друга. Поэтому вербальные и невербальные средства, которые используются участниками дебатов, имеют целью получения определённого результата — сформировать у слушателей положительное впечатление от собственной позиции.

- Метод работы в малых группах. Групповое обсуждение кого-либо вопроса направлено на достижение лучшего взаимопонимания и нахождения истины. Групповое обсуждение способствует лучшему усвоению изучаемого материала. Оптимальное количество участников - 5-7 человек. Перед обучающимися ставится проблема, выделяется определенное время, в течение которого они должны подготовить аргументированный обдуманный ответ. Педагогический работник может устанавливать правила проведения группового обсуждения — задавать определенные рамки обсуждения, ввести алгоритм выработки общего мнения, назначить лидера и др.

- Круглый стол - общество, собрание в рамках более крупного мероприятия (съезда, симпозиума, конференции). Мероприятие, как правило, на которое приглашаются эксперты и специалисты из разных сфер деятельности для обсуждения актуальных вопросов. Данная модель обсуждения, основываясь на соглашениях, в качестве итогов даёт результаты, которые, в свою очередь, являются новыми соглашениями.

- Коллоквиум - (лат. colloquium — разговор, беседа) - одна из форм учебных занятий в системе образования, имеющая целью выяснение и повышение знаний обучающихся. На коллоквиумах обсуждаются: отдельные части, разделы, темы, вопросы изучаемого курса (обычно не включаемые в тематику семинарских и других практических учебных занятий), рефераты, проекты и др. работы обучающихся. Это научные собрания, на которых заслушиваются и обсуждаются доклады. Коллоквиум — это и форма контроля, массового опроса, позволяющая преподавателю в сравнительно небольшой срок выяснить уровень знаний студентов по данной теме дисциплины. Коллоквиум проходит обычно в форме дискуссии, в ходе которой обучающимся предоставляется возможность высказать свою точку зрения на рассматриваемую проблему, учиться обосновывать и защищать ее. Аргументируя и отстаивая свое мнение, обучающийся в то же время демонстрирует, насколько глубоко и осознанно он усвоил изученный материал.

- Метод «мозговой штурм» (мозговой штурм, мозговая атака, англ. brainstorming) — оперативный метод решения проблемы на основе стимулирования творческой активности, при котором участникам обсуждения предлагают высказывать как можно большее количество вариантов решения, в том числе самых фантастических. Затем из общего числа высказанных идей отбирают наиболее удачные, которые могут быть использованы на практике. Является методом экспертного оценивания.

- Метод проектов - это способ достижения дидактической цели через детальную разработку проблемы (технологию), которая должна завершиться вполне реальным, осязаемым практическим результатом, оформленным тем или иным образом; это совокупность приёмов, действий обучающихся в их определённой последовательности для достижения поставленной задачи — решения проблемы, лично значимой для учащихся и

оформленной в виде некоего конечного продукта. Основное предназначение метода проектов состоит в предоставлении учащимся возможности самостоятельного приобретения знаний в процессе решения практических задач или проблем, требующего интеграции знаний из различных предметных областей.

- Брифинг - (англ. briefing от англ. brief – короткий, недолгий) – краткая пресс-конференция, посвященная одному вопросу. Основное отличие: отсутствует презентационная часть. То есть практически сразу идут ответы на вопросы журналистов.

- Метод портфолио (итал. portfolio — 'портфель, англ. - папка для документов) - современная образовательная технология, в основе которой используется метод аутентичного оценивания результатов образовательной и профессиональной деятельности. Портфолио как подборка сертифицированных достижений, наиболее значимых работ и отзывов на них.

Вопросы для самоподготовки к практическим (семинарским) занятиям по разделам (темам) дисциплины (модуля)

РАЗДЕЛ 1. Алгоритмы. Языки программирования.

Тема 1.1. Основы алгоритмизации.

Вопросы для самоподготовки:

1. Основные парадигмы программирования.
2. Определение алгоритма

Тема 1.2. Языки и системы программирования.

Вопросы для самоподготовки:

1. Цели использования современных языков программирования высокого уровня.
2. Особенности работы с Microsoft Visual Studio.

РАЗДЕЛ 2. Теоретические основы разработки алгоритмов и программ.

Тема 2.1. Программирование вычислительных алгоритмов на языке высокого уровня (по выбору: Паскаль, JAVA, C).

Вопросы для самоподготовки:

1. Классификация алгоритмов.

Тема 2.2. Методы программирования.

Вопросы для самоподготовки:

1. Основные понятия из программирования. Переменные, области видимости,

РАЗДЕЛ 3. Сортировка данных.

Тема 3.1. Пузырьковая сортировка. Метод декомпозиции.

Вопросы для самоподготовки:

1. Реализация пузырьковой сортировки данных и метода декомпозиции.

Тема 3.2. Оценки эффективности алгоритмов сортировки.

Вопросы для самоподготовки:

1. Реализация алгоритма пузырьковой сортировки и оценка его эффективности.

РАЗДЕЛ 4. Структуры данных.

Тема 4.1. Стеки, очереди, списки и операции над ними.

Вопросы для самоподготовки:

1. Стеки, очереди, списки. Принципы организации и отличия.

Тема 4.2. Бинарные деревья. Операции с бинарным деревом поиска.

Вопросы для самоподготовки:

1. Бинарные деревья. Алгоритмы поиска с использованием бинарных деревьев.

РАЗДЕЛ 5. Основные понятия

Тема 5.1. Основные понятия языка С.

Вопросы для самоподготовки:

3. Основные парадигмы программирования.
4. История создания и развития языка С.

Тема 5.2. Современные системы программирования.

Вопросы для самоподготовки:

3. Особенности работы с Borland Builder.
4. Особенности работы с Microsoft Visual Studio.

РАЗДЕЛ 6. Переменные, выражения и работа с операторами

Тема 6.1. Переменные и типы данных. Выражения.

Вопросы для самоподготовки:

1. Чем различаются типы данных?
2. Какими могут быть выражения?

Тема 6.2. Операторы if, else, switch, while, do-while, for.

Вопросы для самоподготовки:

1. Принципы работы с условными операторами и операторами ветвления.
2. В чём отличия друг от друга циклов в языке С?

РАЗДЕЛ 7. Структуры данных. Работа с указателями.

Тема 7.1. Массивы. Структуры.

Вопросы для самоподготовки:

1. Реализация операций на списках, стеках и очередях.

Тема 7.2. Указатели. Строки.

Вопросы для самоподготовки:

1. Реализация и принципы использования указателей

РАЗДЕЛ 8. Основы объектно-ориентированного программирования на языке C++.

Тема 8.1. Понятие класса. Поля класса.

Вопросы для самоподготовки:

1. Принципы объектно-ориентированной парадигмы программирования
2. Основные понятия языка программирования C++.

Тема 8.2. Методы в классе. Конструкторы класса.

Вопросы для самоподготовки:

5. Характеристики объектов.
6. Описание полей класса.
7. Описание методов класса.

РАЗДЕЛ 9. Работа с объектами.

Тема 9.1. Создание объектов класса. Обращение к полям и методам класса.

Вопросы для самоподготовки:

1. Принципы действия модификаторов доступа.
2. Обращение к полям класса.

Тема 9.2. Деструктор класса. Освобождение памяти, выделенной для объекта класса.

Вопросы для самоподготовки:

1. Вызов методов класса
2. Способы описания методов класса

РАЗДЕЛ 10. Наследование классов

Тема 10.1. Понятие о наследовании.

Вопросы для самоподготовки:

1. Наследование как способ образования новых классов на основе объявленных.

Тема 10.2. Правила наследования полей и методов для различных модификаторов доступа.

Вопросы для самоподготовки:

1. Влияние модификаторов доступа на работу с полями и методами.

РАЗДЕЛ 11. Инкапсуляция и полиморфизм.

Тема 11.1. Инкапсуляция и способы её достижения в языке C++.

Вопросы для самоподготовки:

1. Значение сокрытия данных для объектно-ориентированного программирования.

Тема 11.2. Полиморфизм и его использование в языке C++.

Вопросы для самоподготовки:

1. Примеры реализации полиморфизма

РАЗДЕЛ 12. Основы программирования на языке C#.

Тема 12.1. Структура программы на C#.

Вопросы для самоподготовки:

1. Основные принципы программирования на C#.

Тема 12.2. Особенности работы с объектами и классами на C#.

Вопросы для самоподготовки:

1. Отличия синтаксиса C# от C++.

РАЗДЕЛ 13. Обработка исключительных ситуаций.

Тема 13.1. Исключительные ситуации и их классы.

Вопросы для самоподготовки:

1. Основные классы исключительных ситуаций.

Тема 13.2. Блоки try, catch, finally, throw.

Вопросы для самоподготовки:

1. Роль блоков try, catch, finally, throw в работе механизма обработки исключительных ситуаций.

РАЗДЕЛ 14. Интерфейсы. Делегаты и лямбда-выражения.

Тема 14.1. Объявление интерфейсов. Реализация множественного наследования.

Вопросы для самоподготовки:

1. Интерфейсы как особый вид абстрактных классов.

Тема 14.2. Объявление и применение делегатов. Работа с лямбда-выражениями.**Вопросы для самоподготовки:**

1. Делегаты и лямбда-функции как альтернативы традиционным функциям в C#.

РАЗДЕЛ 15. Разработка приложений с графическим интерфейсом.**Тема 15.1. Создание приложений с графическим интерфейсом в Visual C#. Форма и проект программы.****Вопросы для самоподготовки:**

1. Принципы конструирования приложений с оконным интерфейсом. Свойства компонентов интерфейса и события.

Тема 15.2. Работа с кнопками, меню, списками ListBox и ComboBox.**Вопросы для самоподготовки:**

1. Основные свойства и события для компонентов Button (кнопка), ListBox, ComboBox. Создание меню приложения.

1.3. Учебно-наглядные пособия по разделам (темам) дисциплины (модуля)¹**РАЗДЕЛ 1. Алгоритмы. Языки программирования.****Тема 1.1. Основы алгоритмизации.**

Зыков, С. В. Программирование : учебник и практикум для вузов / С. В. Зыков. — Москва : Издательство Юрайт, 2023. — 320 с. — (Высшее образование). — ISBN 978-5-534-02444-9. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511712> (дата обращения: 07.03.2023).

Тема 1.2. Языки и системы программирования.

Зыков, С. В. Программирование : учебник и практикум для вузов / С. В. Зыков. — Москва : Издательство Юрайт, 2023. — 320 с. — (Высшее образование). — ISBN 978-5-534-02444-9. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511712> (дата обращения: 07.03.2023).

РАЗДЕЛ 2. Теоретические основы разработки алгоритмов и программ.**Тема 2.1. Программирование вычислительных алгоритмов на языке высокого уровня (по выбору: Паскаль, JAVA, C).**

¹ Раздел может быть оформлен в виде приложения к методическим материалам по дисциплине (модулю).

Зыков, С. В. Программирование : учебник и практикум для вузов / С. В. Зыков. — Москва : Издательство Юрайт, 2023. — 320 с. — (Высшее образование). — ISBN 978-5-534-02444-9. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511712> (дата обращения: 07.03.2023).

Тема 2.2. Методы программирования.

Зыков, С. В. Программирование : учебник и практикум для вузов / С. В. Зыков. — Москва : Издательство Юрайт, 2023. — 320 с. — (Высшее образование). — ISBN 978-5-534-02444-9. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511712> (дата обращения: 07.03.2023).

РАЗДЕЛ 3. Сортировка данных.

Тема 3.1. Пузырьковая сортировка. Метод декомпозиции.

Зыков, С. В. Программирование : учебник и практикум для вузов / С. В. Зыков. — Москва : Издательство Юрайт, 2023. — 320 с. — (Высшее образование). — ISBN 978-5-534-02444-9. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511712> (дата обращения: 07.03.2023).

Тема 3.2. Оценки эффективности алгоритмов сортировки.

Зыков, С. В. Программирование : учебник и практикум для вузов / С. В. Зыков. — Москва : Издательство Юрайт, 2023. — 320 с. — (Высшее образование). — ISBN 978-5-534-02444-9. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511712> (дата обращения: 07.03.2023).

РАЗДЕЛ 4. Структуры данных.

Тема 4.1. Стеки, очереди, списки и операции над ними.

Зыков, С. В. Программирование : учебник и практикум для вузов / С. В. Зыков. — Москва : Издательство Юрайт, 2023. — 320 с. — (Высшее образование). — ISBN 978-5-534-02444-9. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511712> (дата обращения: 07.03.2023).

Тема 4.2. Бинарные деревья. Операции с бинарным деревом поиска.

Зыков, С. В. Программирование : учебник и практикум для вузов / С. В. Зыков. — Москва : Издательство Юрайт, 2023. — 320 с. — (Высшее образование). — ISBN 978-5-534-02444-9. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511712> (дата обращения: 07.03.2023).

РАЗДЕЛ 5. Основные понятия

Тема 5.1. Основные понятия языка С.

Б. Керниган, Д. Ритчи «Язык программирования С» с примечаниями.

Тема 5.2. Современные системы программирования.

Б. Керниган, Д. Ритчи «Язык программирования С» с примечаниями.

РАЗДЕЛ 6. Переменные, выражения и работа с операторами

Тема 6.1. Переменные и типы данных. Выражения.

Б. Керниган, Д. Ритчи «Язык программирования С» с примечаниями.

Тема 6.2. Операторы if, else, switch, while, do-while, for.

Б. Керниган, Д. Ритчи «Язык программирования С» с примечаниями.

РАЗДЕЛ 7. Структуры данных. Работа с указателями.

Тема 7.1. Массивы. Структуры.

Б. Керниган, Д. Ритчи «Язык программирования С» с примечаниями.

Тема 7.2. Указатели. Строки.

Б. Керниган, Д. Ритчи «Язык программирования С» с примечаниями.

РАЗДЕЛ 8. Основы объектно-ориентированного программирования на языке С++.

Тема 8.1. Понятие класса. Поля класса.

Х. Дейтел, П. Дейтел «Как программировать на С++» с примечаниями.

Тема 8.2. Методы в классе. Конструкторы класса.

Х. Дейтел, П. Дейтел «Как программировать на С++» с примечаниями.

РАЗДЕЛ 9. Работа с объектами.

Тема 9.1. Создание объектов класса. Обращение к полям и методам класса.

Х. Дейтел, П. Дейтел «Как программировать на С++» с примечаниями.

Тема 9.2. Деструктор класса. Освобождение памяти, выделенной для объекта класса.

Х. Дейтел, П. Дейтел «Как программировать на С++» с примечаниями.

РАЗДЕЛ 10. Наследование классов

Тема 10.1. Понятие о наследовании.

Х. Дейтел, П. Дейтел «Как программировать на C++» с примечаниями.

Тема 10.2. Правила наследования полей и методов для различных модификаторов доступа.

Х. Дейтел, П. Дейтел «Как программировать на C++» с примечаниями.

РАЗДЕЛ 11. Инкапсуляция и полиморфизм.

Х. Дейтел, П. Дейтел «Как программировать на C++» с примечаниями.

Тема 11.2. Полиморфизм и его использование в языке C++.

Х. Дейтел, П. Дейтел «Как программировать на C++» с примечаниями.

РАЗДЕЛ 12. Основы программирования на языке C#.

Тема 12.1. Структура программы на C#.

Х. Дейтел, П. Дейтел «Как программировать на Visual C# 2012» с примечаниями.

Тема 12.2. Особенности работы с объектами и классами на C#.

Х. Дейтел, П. Дейтел «Как программировать на Visual C# 2012» с примечаниями.

РАЗДЕЛ 13. Обработка исключительных ситуаций.

Тема 13.1. Исключительные ситуации и их классы.

Х. Дейтел, П. Дейтел «Как программировать на Visual C# 2012» с примечаниями.

Тема 13.2. Блоки try, catch, finally, throw.

Х. Дейтел, П. Дейтел «Как программировать на Visual C# 2012» с примечаниями.

РАЗДЕЛ 14. Интерфейсы. Делегаты и лямбда-выражения.

Тема 14.1. Объявление интерфейсов. Реализация множественного наследования.

Х. Дейтел, П. Дейтел «Как программировать на Visual C# 2012» с примечаниями.

Тема 14.2. Объявление и применение делегатов. Работа с лямбда-выражениями.

Х. Дейтел, П. Дейтел «Как программировать на Visual C# 2012» с примечаниями.

РАЗДЕЛ 15. Разработка приложений с графическим интерфейсом.

Тема 15.1. Создание приложений с графическим интерфейсом в Visual C#. Форма и проект программы.

Х. Дейтел, П. Дейтел «Как программировать на Visual C# 2012» с примечаниями.

Тема 15.2. Работа с кнопками, меню, списками ListBox и ComboBox.

Х. Дейтел, П. Дейтел «Как программировать на Visual C# 2012» с примечаниями.

1.4. Методические материалы по подготовке к лабораторным занятиям по дисциплине (модулю)

Лабораторные занятия - одна из форм учебного занятия, направленная на развитие самостоятельности обучающихся и приобретение умений и навыков. Данные учебные занятия углубляют, расширяют, детализируют полученные ранее знания. Лабораторное занятие предполагает выполнение обучающимися по заданию и под руководством преподавателей одной или нескольких работ.

Цель лабораторных занятий состоит в развитии познавательных способностей, самостоятельности мышления и творческой активности обучающихся; углублении, расширении, детализировании знаний, полученных на лекции в обобщенной форме, и содействии выработке навыков профессиональной деятельности. В отдельных случаях на лабораторных занятиях руководителем занятия сообщаются дополнительные знания.

Для достижения поставленных целей и решения требуемого перечня задач лабораторные занятия проводятся традиционными технологиями или с использованием активных и интерактивных образовательных технологий.

При подготовке и работе во время проведения лабораторных занятий следует обратить внимание на следующие моменты: на процесс предварительной подготовки, на работу во время занятия, обработку полученных результатов, исправление полученных замечаний.

Предварительная подготовка к лабораторному занятию заключается в изучении теоретического материала в отведенное для самостоятельной работы время, ознакомление с инструктивными материалами с целью осознания задач занятия, техники безопасности при работе с приборами, веществами.

Работа во время проведения занятия включает:

- консультирование студентов преподавателями и вспомогательным персоналом с целью предоставления исчерпывающей информации, необходимой для самостоятельного выполнения предложенных преподавателем задач.
- самостоятельное выполнение заданий согласно обозначенной рабочей программой дисциплины (модуля) тематики.

2. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ) И САМОСТОЯТЕЛЬНОЙ РАБОТЕ

Освоение обучающимся дисциплины (модуля) «*Наименование дисциплины (модуля)*» предполагает изучение материалов дисциплины на аудиторных занятиях и в ходе самостоятельной работы. Аудиторные занятия проходят в форме лекций, семинаров, практических и лабораторных занятий. Самостоятельная работа включает разнообразный комплекс видов и форм работы обучающихся.

Для успешного освоения дисциплины (модуля) и достижения поставленных целей необходимо внимательно ознакомиться с рабочей программы дисциплины (модуля), доступной в электронной информационно-образовательной среде РГСУ.

Следует обратить внимание на списки основной и дополнительной литературы, на предлагаемые преподавателем ресурсы информационно-телекоммуникационной сети Интернет. Эта информация необходима для самостоятельной работы обучающегося.

При подготовке к аудиторным занятиям необходимо помнить особенности каждой формы его проведения.

Подготовка к учебному занятию лекционного типа заключается в следующем.

С целью обеспечения успешного обучения обучающийся должен готовиться к лекции, поскольку она является важнейшей формой организации учебного процесса, поскольку:

- знакомит с новым учебным материалом;
- разъясняет учебные элементы, трудные для понимания;
- систематизирует учебный материал;
- ориентирует в учебном процессе.

С этой целью:

- внимательно прочитайте материал предыдущей лекции;
- ознакомьтесь с учебным материалом по учебнику и учебным пособиям с темой прочитанной лекции;
- внесите дополнения к полученным ранее знаниям по теме лекции на полях лекционной тетради;
- запишите возможные вопросы, которые вы зададите лектору на лекции по материалу изученной лекции;
- постарайтесь уяснить место изучаемой темы в своей подготовке;
- узнайте тему предстоящей лекции (по тематическому плану, по информации лектора) и запишите информацию, которой вы владеете по данному вопросу.

Подготовка к занятию семинарского типа.

При подготовке и работе во время проведения занятий семинарского типа следует обратить внимание на следующие моменты: на процесс предварительной подготовки, на работу во время занятия, обработку полученных результатов, исправление полученных замечаний.

Предварительная подготовка к учебному занятию семинарского типа заключается в изучении теоретического материала в отведенное для самостоятельной работы время, ознакомление с инструктивными материалами с целью осознания задач практического занятия, техники безопасности при работе с приборами, веществами.

Работа во время проведения учебного занятия семинарского типа включает:

- консультирование студентов преподавателями и вспомогательным персоналом с целью предоставления исчерпывающей информации, необходимой для самостоятельного выполнения предложенных преподавателем задач.
- самостоятельное выполнение заданий согласно обозначенной рабочей программой дисциплины (модуля) тематики.

Самостоятельная работа.

Самостоятельная работа - планируемая учебная, учебно-исследовательская, научно-исследовательская работа студентов, выполняемая во внеаудиторное (аудиторное) время по заданию и при методическом руководстве преподавателя, но без его непосредственного участия (при частичном непосредственном участии преподавателя, оставляющем ведущую роль за работой студентов).

Для более углубленного изучения темы задания для самостоятельной работы рекомендуется выполнять параллельно с изучением данной темы. При выполнении заданий по возможности используйте наглядное представление материала.

Самостоятельная работа студентов в ВУЗе является важным видом учебной и научной деятельности студента. Самостоятельная работа студентов играет значительную роль в рейтинговой технологии обучения.

К современному специалисту в области медицины общество предъявляет достаточно широкий перечень требований, среди которых немаловажное значение имеет наличие у

выпускников определенных навыков (компетенций) и умения самостоятельно добывать знания из различных источников, систематизировать полученную информацию, давать оценку конкретной профессиональной ситуации. Формирование такого умения происходит в течение всего периода обучения.

В процессе самостоятельной работы студент приобретает необходимые для будущей специальности компетенции, навыки самоорганизации, самоконтроля, самоуправления, саморефлексии и становится активным самостоятельным субъектом учебной деятельности.

Самостоятельная работа студентов должна оказывать важное влияние на формирование личности будущего специалиста, его компетентность. Каждый студент самостоятельно определяет режим своей работы и меру труда, затрачиваемого на овладение учебным содержанием по каждой дисциплине. Он выполняет внеаудиторную работу по личному индивидуальному плану, в зависимости от его подготовки, времени и других условий.

Виды самостоятельной работы.

Работа с литературой.

При работе с книгой необходимо подобрать литературу, научиться правильно ее читать, вести записи. Для подбора литературы в библиотеке используются алфавитный и систематический каталоги. Важно помнить, что рациональные навыки работы с книгой - это всегда большая экономия времени и сил. Правильный подбор учебников рекомендуется преподавателем, читающим лекционный курс. Необходимая литература может быть также указана в методических разработках по данному курсу. Изучая материал по учебнику, следует переходить к следующему вопросу только после правильного уяснения предыдущего, описывая на бумаге все выкладки и вычисления (в том числе те, которые в учебнике опущены или на лекции даны для самостоятельного вывода). При изучении любой дисциплины большую и важную роль играет самостоятельная индивидуальная работа. Особое внимание следует обратить на определение основных понятий курса. Студент должен подробно разбирать примеры, которые поясняют такие определения, и уметь строить аналогичные примеры самостоятельно. Нужно добиваться точного представления о том, что изучаешь. Полезно составлять опорные конспекты. При изучении материала по учебнику полезно в тетради (на специально отведенных полях) дополнять конспект лекций. Там же следует отмечать вопросы, выделенные студентом для консультации с преподавателем. Выводы, полученные в результате изучения, рекомендуется в конспекте выделять, чтобы они при перечитывании записей лучше запоминались. Опыт показывает, что многим студентам помогает составление листа опорных сигналов, содержащего важнейшие и наиболее часто употребляемые формулы и понятия. Такой лист помогает запомнить формулы, основные положения лекции, а также может служить постоянным справочником для студента. Различают два вида чтения: первичное и вторичное. Первичное - это внимательное, неторопливое чтение, при котором можно остановиться на трудных местах.

Задача вторичного чтения - полное усвоение смысла целого (по счету это чтение может быть и не вторым, а третьим или четвертым). Самостоятельная работа с учебниками и книгами (а также самостоятельное теоретическое исследование проблем, обозначенных преподавателем на лекциях) – это важнейшее условие формирования у себя научного способа познания.

Чтение научного текста является частью познавательной деятельности. Ее цель – извлечение из текста необходимой информации. От того, насколько осознанно читающим собственная внутренняя установка при обращении к печатному слову (найти нужные сведения, усвоить информацию полностью или частично, критически проанализировать материал и т.п.) во многом зависит эффективность осуществляемого действия.

Методические рекомендации по составлению конспекта:

1. Внимательно прочитайте текст. Уточните в справочной литературе непонятные слова. При записи не забудьте вынести справочные данные на поля конспекта;
2. Выделите главное, составьте план;
3. Кратко сформулируйте основные положения текста, отметьте аргументацию автора;

4. Законспектируйте материал, четко следуя пунктам плана. При конспектировании старайтесь выразить мысль своими словами. Записи следует вести четко, ясно.

5. Грамотно записывайте цитаты. Цитируя, учитывайте лаконичность, значимость мысли. В тексте конспекта желательно приводить не только тезисные положения, но и их доказательства. При оформлении конспекта необходимо стремиться к емкости каждого предложения. Мысли автора книги следует излагать кратко, заботясь о стиле и выразительности написанного. Число дополнительных элементов конспекта должно быть логически обоснованным, записи должны распределяться в определенной последовательности, отвечающей логической структуре произведения. Для уточнения и дополнения необходимо оставлять поля. Овладение навыками конспектирования требует от студента целеустремленности, повседневной самостоятельной работы.

Методические материалы по самостоятельному решению задач

При самостоятельном решении задач нужно обосновывать каждый этап решения, исходя из теоретических положений курса. Если студент видит несколько путей решения проблемы (задачи), то нужно сравнить их и выбрать самый рациональный. Полезно до начала вычислений составить краткий план решения проблемы (задачи). Решение проблемных задач или примеров следует излагать подробно, вычисления располагать в строгом порядке, отделяя вспомогательные вычисления от основных. Решения при необходимости нужно сопровождать комментариями, схемами, чертежами и рисунками. Следует помнить, что решение каждой учебной задачи должно доводиться до окончательного логического ответа, которого требует условие, и по возможности с выводом. Полученный ответ следует проверить способами, вытекающими из существа данной задачи. Полезно также (если возможно) решать несколькими способами и сравнить полученные результаты. Решение задач данного типа нужно продолжать до приобретения твердых навыков в их решении.

Методические материалы к выполнению реферата

Реферат (от лат. *refere* – сообщать) – краткое изложение в письменном виде или в форме публикации доклада, содержания научного труда (трудов), литературы по теме. Работа над рефератом условно разделяется на выбор темы, подбор литературы, подготовку и защиту плана; написание теоретической части и всего текста с указанием библиографических данных используемых источников, подготовку доклада, выступление с ним. Тематика рефератов полностью связана с основными вопросами изучаемого курса.

Список литературы к темам не дается, и обучающиеся самостоятельно ведут библиографический поиск, причем им не рекомендуется ограничиваться университетской библиотекой.

Важно учитывать, что написание реферата требует от обучающихся определенных усилий. Особое внимание следует уделить подбору литературы, методике ее изучения с целью отбора и обработки собранного материала, обоснованию актуальности темы и теоретического уровня обоснованности используемых в качестве примеров фактов какой-либо деятельности.

Выбрав тему реферата, начав работу над литературой, необходимо составить план. Изучая литературу, продолжается обдумывание темы, осмысливание прочитанного, делаются выписки, сопоставляются точки зрения разных авторов и т.д. Реферативная работа сводится к тому, чтобы в ней выделились две взаимосвязанные стороны: во-первых, ее следует рассматривать как учебное задание, которое должен выполнить обучаемый, а во-вторых, как форму научной работы, творческого воображения при выполнении учебного задания.

Наличие плана реферата позволяет контролировать ход работы, избежать формального переписывания текстов из первоисточников.

Оформление реферата включает титульный лист, оглавление и краткий список использованной литературы. Список использованной литературы размещается на последней странице рукописи или печатной форме реферата. Реферат выполняется в письменной или печатной форме на белых листах формата А4 (210 x 297 мм). Шрифт Times New Roman, кегель 14, через 1,5 интервала при соблюдении следующих размеров текста: верхнее поле – 25

мм, нижнее – 20 мм, левое – 30 мм, правое – 15 мм. Нумерация страниц производится сверху листа, по центру. Титульный лист нумерации не подлежит.

Рефераты должны быть написаны простым, ясным языком, без претензий на наукообразность. Следует избегать сложных грамматических оборотов, непривычных терминов и символов. Если же такие термины и символы все-таки приводятся, то необходимо разъяснять их значение при первом упоминании в тексте реферата.

Объем реферата предполагает тщательный отбор информации, необходимой для краткого изложения вопроса. Важнейший этап – редактирование готового текста реферата и подготовка к обсуждению. Обсуждение требует хорошей ориентации в материале темы, умения выделить главное, поставить дискуссионный вопрос, привлечь внимание слушателей к интересной литературе, логично и убедительно изложить свои мысли.

Рефераты обязательно подлежат защите. Процедура защиты начинается с определения оппонентов защищающего свою работу. Они стремятся дать основательный анализ работы обучающимся, обращают внимание на положительные моменты и недостатки реферата, дают общую оценку содержанию, форме преподнесения материала, характеру использованной литературы. Иногда они дополняют тот или иной раздел реферата. Последнее особенно ценно, ибо говорит о глубоком знании обучающимся-оппонентом изучаемой проблемы.

Обсуждение не ограничивается заслушиванием оппонентов. Другие обучающиеся имеют право уточнить или опровергнуть какое-либо утверждение. Преподаватель предлагает любому обучающемуся задать вопрос по существу доклада или попытаться подвести итог обсуждению.

Алгоритм работы над рефератом

1. Выбор темы

Тема должна быть сформулирована грамотно (с литературной точки зрения);

В названии реферата следует поставить четкие рамки рассмотрения темы;

Желательно избегать слишком длинных названий;

Следует по возможности воздерживаться от использования в названии спорных с научной точки зрения терминов, излишней наукообразности, а также чрезмерного упрощения формулировок.

2. Реферат следует составлять из пяти основных частей: введения; основной части; заключения; списка литературы; приложений.

3. Основные требования к введению:

Во введении не следует концентрироваться на содержании; введение должно включать краткое обоснование актуальности темы реферата, где требуется показать, почему данный вопрос может представлять научный интерес и есть ли связь представляемого материала с современностью. Таким образом, тема реферата должна быть актуальна либо с научной точки зрения, либо с современных позиций.

Очень важно выделить цель, а также задачи, которые требуется решить для выполнения цели.

Введение должно содержать краткий обзор изученной литературы, в котором указывается взятый из того или иного источника материал, кратко анализируются изученные источники, показываются их сильные и слабые стороны;

Объем введения составляет две страницы текста.

4. Требования к основной части реферата:

Основная часть содержит материал, отобранный для рассмотрения проблемы;

Также основная часть должна включать в себя собственно мнение обучающихся и сформулированные самостоятельные выводы, опирающиеся на приведенные факты;

Материал, представленный в основной части, должен быть логически изложен и распределен по параграфам, имеющим свои названия;

В изложении основной части необходимо использовать сноски (в первую очередь, когда приводятся цифры и чьи-то цитаты);

Основная часть должна содержать иллюстративный материал (графики, таблицы и т. д.);

Объем основной части составляет около 10 страниц.

5. Требования к заключению:

В заключении формулируются выводы по параграфам, обращается внимание на выдвинутые во введении задачи и цели;

Заключение должно быть четким, кратким, вытекающим из содержания основной части.

6. Требования к оформлению списка литературы (по ГОСТу):

Необходимо соблюдать правильность последовательности записи источников: сначала следует писать фамилию, а после инициалы; название работы не ставится в кавычки; после названия сокращенно пишется место издания; затем идет год издания; наконец, называется процитированная страница.

Критерии оценки реферата

Обучающийся, защищающий реферат, должен рассказать о его актуальности, поставленных целях и задачах, изученной литературе, структуре основной части, сделанных в ходе работы выводах.

По окончании выступления ему может быть задано несколько вопросов по представленной проблеме.

Оценка складывается из соблюдения требований к реферату, грамотного раскрытия темы, умения четко рассказывать о представленном реферате, способности понять суть задаваемых по работе вопросов и найти точные ответы на них.

Методические материалы к выполнению эссе

Эссе – литературное произведение небольшого объема, обычно прозаическое, свободной композиции, передающее индивидуальные впечатления, суждения, соображения автора о той или иной проблеме, теме, о том или ином событии или явлении. Это вид самостоятельной исследовательской работы обучающихся, с целью углубления и закрепления теоретических знаний и освоения практических навыков. Цель эссе состоит в развитии самостоятельного творческого мышления и письменного изложения собственных мыслей. При написании эссе обучающийся должен представить развернутый письменный ответ на теоретический или практический актуальный вопрос, объявленный преподавателем в аудитории непосредственно перед ее написанием. В процессе написания эссе разрешается пользоваться нормативно-правовыми актами, конспектом лекций (в печатном виде). Использование интернет-ресурсов не допускается. Темы эссе преподаватель предлагает из числа тех, которые обучающиеся уже рассматривали на лекциях или семинарских занятиях, исходя из содержания заданий в составе оценочных средств. По решению преподавателя, в качестве темы эссе может быть выбрана одна или несколько тем, которые могут быть распределены между обучающимися по желанию.

Требования к выполнению эссе:

1. Проводится письменно.

2. Эссе выполняется на компьютере (гарнитура Times New Roman, шрифт 14) через 1,5 интервала с полями: верхнее, нижнее – 2; правое – 3; левое – 1,5. Отступ первой строки абзаца – 1,25. Сноски – постраничные. Таблицы и рисунки встраиваются в текст работы. При этом обязательный заголовок таблицы надо размещать над табличным полем, а рисунки сопровождать подрисовочными подписями. При включении в эссе нескольких таблиц и/или рисунков их нумерация обязательна. Обязательна и нумерация страниц. Их целесообразно проставлять внизу страницы – по середине или в правом углу. Номер страницы не ставится на титульном листе, но в общее число страниц он включается. Объем эссе, без учета приложений, не должен превышать 5 страниц. Значительное превышение установленного объема является недостатком работы и указывает на то, что обучающийся не сумел отобрать и переработать необходимый материал.

3. Работа должна содержать собственные умозаключения по сути поставленной проблемы, включать самостоятельно проведенный анализ по сути этой проблемы, выводы, обобщающие авторскую позицию по поставленной проблеме.

Критерии оценки эссе:

«Отлично» – исключительные знания материала, абсолютное понимание сути, безукоризненное знание основных понятий и положений, логически и лексически грамотно изложенный, содержательный, аргументированный, конкретный и исчерпывающий ответ.

«Хорошо» – глубокие знания материала, правильное понимание сути, знание основных понятий и положений, содержательный, полный и конкретный ответ.

«Удовлетворительно» – твердые, но недостаточно полные знания, верное понимание сути, в целом правильный ответ.

«Неудовлетворительно» – непонимание сущности задания, грубые ошибки в ответе.

Методические материалы по выполнению тестирования.

Тестовые задания содержат вопросы и 3-4 варианта ответа по базовым положениям изучаемой темы/раздела, составлены с расчетом на знания, полученные обучающимся в процессе изучения темы/раздела.

Тестовые задания выполняются в письменной форме и сдаются преподавателю, ведущему дисциплину (модуль). На выполнение тестовых заданий обучающимся отводится 45 минут.

При обработке результатов оценочной процедуры используются: критерии оценки по содержанию и качеству полученных ответов, ключи, оценочные листы.

Критерии оценки теста:

«Зачтено» - если обучающийся правильно выполнил не менее 2/3 всей работы или допустил не более одной грубой ошибки и двух недочетов, не более одной грубой и одной негрубой ошибки, не более трех негрубых ошибок, одной негрубой ошибки и трех недочетов, при наличии четырех-пяти недочетов.

«Не зачтено» - если число ошибок и недочетов превысило норму для оценки 3 или правильно выполнено менее 2/3 всей работы.

Методические материалы по выполнению доклада.

Рекомендуется следующая структура доклада:

1. титульный лист, содержание доклада;
2. краткое изложение;
3. цели и задачи;
4. изложение характера исследований и рассмотренных проблем, гипотезы, спорные вопросы;
5. источники информации, методы сбора и анализа данных, степень их полноты и достоверности;
6. анализ и толкование полученных в работе результатов;
7. выводы и оценки;
8. библиография и приложения.

Время выступления докладчика не должно превышать 10 минут.

Основные требования к оформлению доклада:

- титульный лист должен включать название доклада, наименование предметной (цикловой) комиссии, фамилию обучающегося;
- все использованные литературные источники сопровождаются библиографическим описанием;
- приводимая цитата из источника берется в кавычки (оформляются сноски);
- единицы измерения должны применяться в соответствии с действующими стандартами;
- все названия литературных источников следует приводить в соответствии с новейшими изданиями;

- рекомендуется включение таблиц, графиков, схем, если они отражают основное содержание или улучшают ее наглядность;

- названия фирм, учреждений, организаций и предприятий должны именоваться так, как они указываются в источнике;

Критерии оценки доклада

При выполнении доклада обучающийся должен продемонстрировать умение кратко излагать прочитанный материал, а также умение обобщать и анализировать материал по теме доклада.

Презентация

Методические материалы к презентациям

1. Объем презентации 10 -20 слайдов.

2. На титульном слайде должно быть отражено:

- наименование факультета;
- тема презентации;
- фамилия, имя, отчество, направление подготовки/ специальность, направленность (профиль)/ специализация, форма обучения, номер группы автора презентации;
- фамилия, имя, отчество, степень, звание, должность руководитель работы;
- год выполнения работы.

3. В презентации должны быть отражены обоснование актуальности представляемого материала, цели и задачи работы.

4. Содержание презентации должно включать наиболее значимый материал доклада, а также, при необходимости, таблицы, диаграммы, рисунки, фотографии, карты, видео – вставки, звуковое сопровождение.

5. Заключительный слайд должен содержать информацию об источниках информации для презентации.

Критерии оценки презентации

1. Объем презентации 10 -20 слайдов.

2. Правильность оформления титульного слайда.

3. Актуальность отобранного материала, обоснованность формулировки цели и задач работы.

4. Наглядность и логичность презентации, обоснованность использования таблиц, диаграмм, рисунков, фотографий, карт, видео – вставок, звукового сопровождения; правильный выбор шрифтов, фона, других элементов дизайна слайда.

5. Объем и качество источников информации (не менее 2-х интернет – источников и не менее 2-х литературных источников).

Методические материалы по подготовке к опросу

Самостоятельная работа обучающихся включает подготовку к опросу на практических занятиях. Для этого обучающийся изучает лекции, основную и дополнительную литературу, публикации, информацию из Интернет-ресурсов.

Тема и вопросы к практическим занятиям, вопросы для самоконтроля содержатся в рабочей программе и доводятся до студентов заранее.

Для подготовки к опросу обучающемуся необходимо ознакомиться с материалом, посвященным теме практического занятия, в учебнике или другой рекомендованной литературе, конспекте лекции, обратить внимание на усвоение основных понятий дисциплины, выявить неясные вопросы и подобрать дополнительную литературу для их освещения.

Критерии оценки опроса

«Отлично»:

- дан полный, развернутый ответ на поставленный вопрос;
- в ответе прослеживается четкая структура, логическая последовательность, отражающая сущность раскрываемых понятий, теорий, явлений;

- знание по предмету демонстрируются на фоне понимания его в системе данной науки и междисциплинарных связей;
 - свободное владение терминологией;
 - ответы на дополнительные вопросы четкие, краткие;
- «Хорошо»:
- дан полный, развернутый ответ на поставленный вопрос, показано умение выделять существенные и несущественные признаки, причинно-следственные связи;
 - ответ недостаточно логичен с единичными ошибками в частностях, исправленные студентом с помощью преподавателя;
 - единичные ошибки в терминологии;
 - ответы на дополнительные вопросы правильные, недостаточно полные и четкие.
- «Удовлетворительно»:
- ответ не полный, с ошибками в деталях, умение раскрыть значение обобщённых знаний не показано, речевое оформление требует поправок, коррекции;
 - логика и последовательность изложения имеют нарушения, студент не способен самостоятельно выделить существенные и несущественные признаки и причинно-следственные связи;
 - ошибки в раскрываемых понятиях, терминах;
 - студент не ориентируется в теме, допускает серьезные ошибки;
 - студент не может ответить на большую часть дополнительных вопросов.
- «Неудовлетворительно»:
- ответ представляет собой разрозненные знания с существенными ошибками по вопросу;
 - присутствуют фрагментарность, нелогичность изложения, студент не осознает связь обсуждаемого вопроса с другими объектами дисциплины, речь неграмотная;
 - незнание терминологии;
 - ответы на дополнительные вопросы неправильные.
- Методические материалы по выполнению практического задания**
- При выполнении практического задания обучающийся придерживается следующего алгоритма:
1. Записать дату, тему и цель задания;
 2. Ознакомиться с правилами и условия выполнения практического задания;
 3. Повторить теоретические задания, необходимые для рациональной работы и других практических действий, используя конспекты лекций и рекомендованную литературу, представленной в программе;
 4. Выполнить работу по предложенному алгоритму действий;
 5. Обобщить результаты работы, сформулировать выводы / дать ответы на контрольные вопросы;
- Работа должна быть выполнена грамотно, с соблюдением культуры изложения. При использовании данных из учебных, методических пособий и другой литературы, периодических изданий, Интернет-источников должны иметься ссылки на вышеперечисленные.
- Критерии оценки практического задания:**
- «Отлично» – правильный ответ, дается четкое обоснование принятому решению; рассуждения четкие последовательные логические; используются ссылки на полученные при изучении дисциплины знания; правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.
- «Хорошо» – правильный ответ, дается обоснование принятому решению; но с не существенными ошибками, в рассуждениях отсутствует логическая последовательность; используются ссылки на полученные при изучении дисциплины знания, правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Удовлетворительно» – правильный ответ, допускаются грубые ошибки в обосновании принятого решения; рассуждения не последовательные сумбурные; используются ссылки на полученные при изучении дисциплины знания; используются формулы, процедуры, понятия, имеющие прямое значение для подтверждения принятого решения, однако, при обращении к ним допускаются серьезные ошибки, студент не может правильно ими воспользоваться.

«Неудовлетворительно, не зачтено» – ответ неверный, отсутствует обоснование принятому решению; студент демонстрирует полное непонимание сути вопроса.

Для оценки решения ситуационной задачи (аналитического задания):

Оценка «отлично» выставляется, если задача решена грамотно, ответы на вопросы сформулированы четко. Эталонный ответ полностью соответствует решению студента, которое хорошо обосновано теоретически.

Оценка «хорошо» выставляется, если задача решена, ответы на вопросы сформулированы не достаточно четко. Решение студента в целом соответствует эталонному ответу, но не достаточно хорошо обосновано теоретически.

Оценка «удовлетворительно» выставляется, если задача решена не полностью, ответы не содержат всех необходимых обоснований решения.

Оценка «неудовлетворительно» выставляется, если задача не решена или имеет грубые теоретические ошибки в ответе на поставленные вопросы.

Методические материалы по выполнению лабораторного задания

При выполнении лабораторного задания обучающийся придерживается следующего алгоритма:

1. Записать дату, тему и цель задания;
2. Ознакомиться с правилами и условия выполнения задания;
3. Повторить теоретические задания, необходимые для рациональной работы и других практических действий, используя конспекты лекций и рекомендованную литературу, представленной в программе;
4. Выполнить работу по предложенному алгоритму действий;
5. Обобщить результаты работы, сформулировать выводы / дать ответы на контрольные вопросы;

Работа должна быть выполнена грамотно, с соблюдением культуры изложения. При использовании данных из учебных, методических пособий и другой литературы, периодических изданий, Интернет-источников должны иметься ссылки на вышеперечисленные.

Критерии оценки лабораторного задания:

«Отлично» – правильный ответ, дается четкое обоснование принятому решению; рассуждения четкие последовательные логические; используются ссылки на полученные при изучении дисциплины знания; правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Хорошо» – правильный ответ, дается обоснование принятому решению; но с не существенными ошибками, в рассуждениях отсутствует логическая последовательность; используются ссылки на полученные при изучении дисциплины знания, правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Удовлетворительно» – правильный ответ, допускаются грубые ошибки в обосновании принятого решения; рассуждения не последовательные сумбурные; используются ссылки на полученные при изучении дисциплины знания; используются формулы, процедуры, понятия, имеющие прямое значение для подтверждения принятого решения, однако, при обращении к ним допускаются серьезные ошибки, студент не может правильно ими воспользоваться.

«Неудовлетворительно, не зачтено» – ответ неверный, отсутствует обоснование принятому решению; студент демонстрирует полное непонимание сути вопроса.

Методические указания для подготовки к промежуточной аттестации.

Изучение учебных дисциплин (модулей) завершается зачетом/зачетом с оценкой или экзаменом. Подготовка к промежуточной аттестации способствует закреплению, углублению и обобщению знаний, получаемых, в процессе обучения, а также применению их к решению практических задач. Готовясь к экзамену, студент ликвидирует имеющиеся пробелы в знаниях, углубляет, систематизирует и упорядочивает свои знания. На зачете или экзамене студент демонстрирует то, что он освоил в процессе обучения по дисциплине (модулю).

Вначале следует просмотреть весь материал по дисциплине (модулю), отметить для себя трудные вопросы. Обязательно в них разобраться. В заключение еще раз целесообразно повторить основные положения, используя при этом листы опорных сигналов. Систематическая подготовка к занятиям в течение семестра позволит использовать время промежуточной аттестации для систематизации знаний.

3. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНИВАНИЯ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ И (ИЛИ) ОПЫТА ДЕЯТЕЛЬНОСТИ, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ

3.1. Организационные основы применения балльно-рейтинговой системы оценки успеваемости обучающихся по дисциплине (модулю)

Оценка качества освоения обучающимися дисциплины (модуля) реализуется в формате балльно-рейтинговой системы оценки успеваемости обучающихся (БРСО).

БРСО в ходе текущего контроля успеваемости и промежуточной аттестации осуществляется по 100-балльной шкале.

Академический рейтинг обучающегося по дисциплине (модулю) складывается из результатов:

- текущего контроля успеваемости (максимальный текущий рейтинг обучающегося 80 рейтинговых баллов);
- промежуточной аттестации (максимальный рубежный рейтинг обучающегося 20 рейтинговых баллов).

Условия оценки освоения обучающимся дисциплины (модуля) в формате БРСО доводятся преподавателем до сведения обучающихся на первом учебном занятии, а также размещены в свободном доступе в электронной информационно-образовательной среде Университета.

3.2. Проведение текущего контроля успеваемости обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося

В течение учебного семестра до промежуточной аттестации на основании утвержденной рабочей программы дисциплины (модуля) формируется текущий рейтинг обучающегося. Текущий рейтинг обучающегося складывается как сумма рейтинговых баллов, полученных им в течение учебного семестра по всем видам учебных занятий по учебной дисциплине.

В процессе текущего контроля оцениваются следующие действия обучающегося, направленные на освоение компетенций в рамках изучения учебной дисциплины:

- академическая активность (посещаемость учебных занятий, самостоятельное изучение содержания учебной дисциплины в электронной информационно-образовательной среде, соблюдение сроков сдачи практических заданий и текущих контрольных мероприятий и др.);
- выполнение и сдача текущих и итогового практических заданий (эссе, рефераты, творческие задания, кейс-задания, лабораторные работы, расчетные задания и др., активное участие в групповых интерактивных занятиях (дискуссии, WiKi-проекты и др.), защита проектов и др.);
- прохождение рубежей текущего контроля, включая соблюдение графика их прохождения в электронной информационно-образовательной среде.

Для планирования расчета текущего рейтинга обучающегося используются следующие пропорции:

Вид учебного действия	Максимальная рейтинговая оценка, баллов
академическая активность	10

практические задания	40
<i>из них: текущие практические задания</i>	20
<i>итоговое практическое задание</i>	20
рубежи текущего контроля	30
<i>ИТОГО:</i>	80

В течение учебного семестра по дисциплине (модулю) обучающимся должен быть накоплен текущий рейтинг не менее 52 рейтинговых баллов (65% от максимального значения текущего рейтинга).

Необходимыми условиями допуска обучающегося к промежуточной аттестации по дисциплине являются положительное прохождение обучающимся не менее 65% рубежей текущего контроля с накоплением не менее 65% максимального рейтингового балла за каждый рубеж текущего контроля и положительное выполнение итогового практического задания с накоплением не менее 65% максимального рейтингового балла, установленного за итоговое практическое задание.

Невыполнение вышеуказанных условий является текущей академической задолженностью, которая должна быть ликвидирована обучающимся до контрольного мероприятия промежуточной аттестации.

Сведения о наличии у обучающихся текущей академической задолженности, сроках и порядке добора рейтинговых баллов для её ликвидации доводятся до обучающихся педагогическим работником.

В случае неликвидации текущей академической задолженности, педагогический работник обязан во время контрольного мероприятия промежуточной аттестации поставить обучающемуся 0 рейтинговых баллов. В этом случае ликвидация текущей академической задолженности возможна в периоды проведения повторной промежуточной аттестации.

3.3. Проведение промежуточной аттестации обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося

Промежуточная аттестация по дисциплине (модулю) проводится в соответствии с Положением о промежуточной аттестации обучающихся по основным профессиональным образовательным программам высшего образования – программ специалитета в Российском государственном социальном университете и Положением о балльно-рейтинговой системе оценки успеваемости обучающихся по основным профессиональным образовательным программам высшего образования - программам бакалавриата, программам специалитета, программам магистратуры в Российском государственном социальном университете.

На промежуточную аттестацию отводится 20 рейтинговых баллов.

Ответы обучающегося на контрольном мероприятии промежуточной аттестации оцениваются педагогическим работником по 20 - балльной шкале, а итоговая оценка по дисциплине (модулю) выставляется по пятибалльной системе для экзамена и системе зачтено/не зачтено для зачета.

Критерии выставления оценки определяются Положением о балльно-рейтинговой системе оценки успеваемости обучающихся по основным профессиональным образовательным программам высшего образования - программам бакалавриата, программам специалитета, программам магистратуры в Российском государственном социальном университете.

В процессе определения рубежного рейтинга обучающегося используется следующая шкала:

Рубежный рейтинг	Критерии оценки освоения обучающимся учебной дисциплины в ходе контрольных мероприятий промежуточной аттестации
19-20 рейтинговых баллов	обучающийся глубоко и прочно усвоил программный материал, исчерпывающе, последовательно, грамотно и логически стройно его излагает, тесно увязывает с задачами и будущей деятельностью, не затрудняется с ответом при видоизменении задания, свободно справляется с задачами и практическими заданиями, правильно обосновывает принятые решения, умеет самостоятельно обобщать и излагать материал, не допуская
16-18 рейтинговых баллов	обучающийся твердо знает программный материал, грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос, может правильно применять теоретические положения и владеет необходимыми умениями и навыками при выполнении практических
13-15 рейтинговых баллов	обучающийся освоил основной материал, но не знает отдельных деталей, допускает неточности, недостаточно правильные формулировки, нарушает последовательность в изложении программного материала и испытывает затруднения в выполнении практических заданий
1-12 рейтинговых баллов	обучающийся не знает значительной части программного материала, допускает существенные ошибки, с большими затруднениями выполняет практические задания
0 рейтинговых баллов	не аттестован

Если результат контроля успеваемости в рамках проведения контрольных мероприятий промежуточной аттестации (рубежный рейтинг обучающегося) неудовлетворительный (получено менее 13 рейтинговых баллов), то промежуточная аттестация по учебной дисциплине (модулю) невозможна даже при наличии высокого текущего рейтинга, полученного по итогам текущего контроля по учебной дисциплине (модулю).

ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

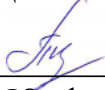
№ п/п	Содержание изменения	Реквизиты документа об утверждении изменения	Дата введения изменения
1.			
2.			
3.			
4.			



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Российский государственный социальный университет»

УТВЕРЖДАЮ

Декан факультета политических и
социальных технологий

 / С.В. Пивнева
«28» февраля 2024 г.

МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)

«ФИЗИКА»

Направление подготовки

«Информационные системы и технологии»

Направленность

«Интеллектуальные информационные системы и технологии»

**ОБРАЗОВАТЕЛЬНАЯ ПРОГРАММА ВЫСШЕГО ОБРАЗОВАНИЯ - ПРОГРАММА
БАКАЛАВРИАТА**

Форма обучения

Очная, заочная

Москва, 2024 г.

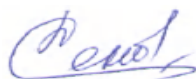
Методические материалы по дисциплине (модулю) «Физика» разработаны на основании федерального государственного образовательного стандарта высшего образования – *бакалавриата* по направлению подготовки 09.03.02 *Информационные системы и технологии*, утвержденного приказом Министерства образования и науки Российской Федерации от 19.09.2017 № 926, учебного плана по основной профессиональной образовательной программе высшего образования – программе бакалавриата по направлению подготовки 09.03.02 *Информационные системы и технологии* (далее – «ОПОП»).

Методические материалы по дисциплине (модулю) разработаны рабочей группой в составе: канд. пед. наук, доцент Пивнева С.В., старший преподаватель Бекбулатов Д.Р.

Методические материалы по дисциплине (модулю) разработаны на заседании кафедры комплекса естественно-научных дисциплин.

Протокол № 9 от «27» февраля 2024 года

Заведующий кафедрой
канд. техн. наук, доцент



Д.А. Денисова

(подпись)

СОДЕРЖАНИЕ

1. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ К ЛЕКЦИОННЫМ И ЛАБОРАТОРНЫМ ЗАНЯТИЯМ	5
1.1. Методические материалы к проведению лекционных занятий по дисциплине (модулю)	5
1.2. Методические материалы по подготовке к лабораторным занятиям по дисциплине (модулю)	7
2. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ) И САМОСТОЯТЕЛЬНОЙ РАБОТЕ.....	12
3. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНИВАНИЯ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ И (ИЛИ) ОПЫТА ДЕЯТЕЛЬНОСТИ, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ.....	21
3.1. Организационные основы применения балльно-рейтинговой системы оценки успеваемости обучающихся по дисциплине (модулю)	21
3.2. Проведение текущего контроля успеваемости обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося.....	21
3.3. Проведение промежуточной аттестации обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося.....	22
ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ	24

1. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ К ЛЕКЦИОННЫМ И ЛАБОРАТОРНЫМ ЗАНЯТИЯМ

1.1. Методические материалы к проведению лекционных занятий по дисциплине (модулю)

Лекция - один из методов обучения, одна из основных системообразующих форм организации учебного процесса в вузе. Лекционное занятие представляет собой систематическое, последовательное, монологическое изложение педагогическим работником учебного материала, как правило, теоретического характера. Такое занятие представляет собой элемент технологии представления учебного материала путем логически стройного, систематически последовательного и ясного изложения.

Цель лекции – организация целенаправленной познавательной деятельности обучающихся по овладению программным материалом дисциплины (модуля). Чтение курса лекций позволяет дать связанное, последовательное изложение материала в соответствии с новейшими данными науки, сообщить слушателям основное содержание предмета в целостном, систематизированном виде. В ряде случаев лекция выполняет функцию основного источника информации: при отсутствии учебников и учебных пособий, чаще по новым курсам; в случае, когда новые научные данные по той или иной теме не нашли отражения в учебниках; отдельные разделы и темы очень сложны для самостоятельного изучения. В таких случаях только лектор может методически помочь обучающимся в освоении сложного материала.

Возможные формы проведения лекций:

- Вводная лекция – один из наиболее важных и трудных видов лекции при чтении систематических курсов. От успеха этой лекции во многом зависит успех усвоения всего курса. Она может содержать: определение дисциплины (модуля); краткую историческую справку о дисциплине (модуле); цели и задачи дисциплины (модуля), ее роль в общей системе обучения и связь со смежными дисциплинами (модулями); основные проблемы (понятия и определения) данной науки; основную и дополнительную учебную литературу; особенности самостоятельной работы обучающихся над дисциплиной (модулем) и формы участия в научно-исследовательской работе; отчетность по курсу.

- Информационная лекция ориентирована на изложение и объяснение обучающимся научной информации, подлежащей осмыслению и запоминанию. Это самый традиционный тип лекций в практике высшей школы.

- Заключительная лекция предназначена для обобщения полученных знаний и раскрытия перспектив дальнейшего развития данной науки.

- Обзорная лекция – это систематизация научных знаний на высоком уровне, допускающая большое число ассоциативных связей в процессе осмысления информации, излагаемой при раскрытии внутрипредметной и междпредметной связей, исключая детализацию и конкретизацию. Как правило, стержень излагаемых теоретических положений составляет научно-понятийная и концептуальная основа всего курса или крупных его разделов.

- Лекция-беседа - непосредственный контакт педагогического работника с аудиторией - диалог. По ходу лекции педагогический работник задает вопросы для выяснения мнений и уровня осведомленности обучающихся по рассматриваемой проблеме.

- Лекция-дискуссия - свободный обмен мнениями в ходе изложения лекционного материала. Педагогический работник активизирует участие в обсуждении отдельными вопросами, сопоставляет между собой различные мнения и тем самым развивает дискуссию, стремясь направить ее в нужное русло.

- Лекция с применением обратной связи включает в себе то, что в начале и конце каждого раздела лекции задаются вопросы. Первый - для того, чтобы узнать, насколько обучающиеся ориентируются в излагаемом материале, вопрос в конце раздела предназначен для выяснения степени усвоения только что изложенного материала. При

неудовлетворительных результатах контрольного опроса педагогический работник возвращается к уже прочитанному разделу, изменив при этом методику подачи материала.

- Проблемная лекция опирается на логику последовательно моделируемых проблемных ситуаций путем постановки проблемных вопросов или предъявления проблемных задач. Проблемный вопрос - это диалектическое противоречие, требующее для своего решения размышления, сравнения, поиска, приобретения и применения новых знаний. Проблемная задача содержит дополнительную вводную информацию и при необходимости некоторые ориентиры поиска ее решения.

- Программированная лекция - консультация – педагогический работник сам составляет и предлагает обучающимся вопросы. На подготовленные вопросы педагогический работник сначала просит ответить обучающихся, а затем проводит анализ и обсуждение неправильных ответов. В лекциях можно использовать наглядные материалы, а также подготовить презентацию. Что касается презентации, то в качестве визуальной поддержки ее можно органично интегрировать во все вышеупомянутые лекции. В то же время лекцию-презентацию возможно выделить и в качестве самостоятельной формы. Лекция-презентация должна отражать суть основных и (или) проблемных вопросов лекции, на которые особо следует обратить внимание обучающихся. В условиях применения активного метода проведения занятий презентация представляется весьма удачным способом донесения информации до слушателей. Единственное, на что следует обратить внимание при подготовке слайдов, - это их оформление и текст. Слайд не должен быть перегружен картинками и лишней информацией, которая будет отвлекать от основного аспекта того или иного вопроса лекции. Во время лекции можно задавать вопросы аудитории в отношении того или иного слайда, тем самым еще больше вовлекая обучающихся в проблематику.

Краткое содержание лекционных занятий

Наименование разделов	Содержание учебного материала
Раздел 1. Постоянный ток	
Тема 1.1. Природа электрического тока	Закон сохранения электрического заряда. Закон Кулона. Напряженность электростатического поля.
Тема 1.2. Законы Ома. Правила Кирхгофа	Принцип суперпозиции электростатических полей. Теорема Гаусса для электростатического поля. Циркуляция вектора напряженности электростатического поля.
Тема 1.3 Соединения проводников	Потенциал электростатического поля. Связь напряженности с потенциалом. Эквипотенциальные поверхности. Энергия взаимодействия системы зарядов.
Раздел 2. Законы постоянного тока	
Тема 2.1. Электрическое сопротивление. Удельное сопротивление вещества	Электрический ток, сила и плотность тока. Сторонние силы. Электродвижущая Постоянный электрический ток: сила и напряжение.
Тема 2.2. Электродвижущая сила. Удельное сопротивление проводника	Закон Ома для однородного и неоднородного участков цепи. Работа и мощность тока. Закон Джоуля-Ленца. Правила Кирхгофа для разветвленных цепей.
Тема 2.3 Работа электрического тока. Закон Джоуля-Ленца	Равновесие зарядов на проводнике. Проводник во внешнем электрическом поле. Емкость уединенного проводника. Конденсаторы.
Раздел 3. Электромагнитные колебания	
Тема 3.1. Свободные электромагнитные колебания.	Намагничивание вещества. Напряженность магнитного поля. Магнитная восприимчивость и

Колебательный контур	магнитная проницаемость.
Тема 3.2. Вынужденные электромагнитные колебания. Резонанс	Законы Кирхгофа в комплексной форме. Характеристики элементов цепи в установившемся синусоидальном режиме. Метод комплексных амплитуд.
Тема 3.3. Гармонические электромагнитные колебания	Расчет установившегося синусоидального режима в простых цепях; векторные диаграммы; простейшие резонансы напряжений и токов. Мощность в установившемся синусоидальном режиме.
Раздел 4. Переменный ток	
Тема 4.1. Переменный ток. Производство, передача и потребление электрической энергии	Резонанс в электрических цепях общего вида. Частотные характеристики цепи, методы определения и построения амплитудно-частотной и фазочастотной характеристик.
Тема 4.2. Емкостное сопротивление. Электромагнитное поле	Общая характеристика трехфазных цепей. Соединение звездой и треугольником. Свойства симметричных трехфазных цепей. Векторные диаграммы.
Тема 4.3. Свойства электромагнитных волн. Различные виды электромагнитных излучений и их применение	Понятие о переходных процессах; коммутация, собственные колебания цепи и вынужденный режим. Переходные процессы в цепях первого порядка при включении источников постоянных сигналов. Переходные процессы в цепи, содержащей индуктивный, емкостной и резистивный элементы (колебательный, апериодический и критический режимы).

1.2. Методические материалы по подготовке к лабораторным занятиям по дисциплине (модулю)

Лабораторные занятия - одна из форм учебного занятия, направленная на развитие самостоятельности обучающихся и приобретение умений и навыков. Данные учебные занятия углубляют, расширяют, детализируют полученные ранее знания. Лабораторное занятие предполагает выполнение обучающимися по заданию и под руководством преподавателей одной или нескольких работ.

Цель лабораторных занятий состоит в развитии познавательных способностей, самостоятельности мышления и творческой активности обучающихся; углублении, расширении, детализировании знаний, полученных на лекции в обобщенной форме, и содействии выработке навыков профессиональной деятельности. В отдельных случаях на лабораторных занятиях руководителем занятия сообщаются дополнительные знания.

Для достижения поставленных целей и решения требуемого перечня задач лабораторные занятия проводятся традиционными технологиями или с использованием активных и интерактивных образовательных технологий.

При подготовке и работе во время проведения лабораторных занятий следует обратить внимание на следующие моменты: на процесс предварительной подготовки, на работу во время занятия, обработку полученных результатов, исправление полученных замечаний.

Предварительная подготовка к лабораторному занятию заключается в изучении теоретического материала в отведенное для самостоятельной работы время, ознакомление с инструктивными материалами с целью осознания задач занятия, техники безопасности при работе с приборами, веществами.

Работа во время проведения занятия включает:

- консультирование студентов преподавателями и вспомогательным персоналом с целью предоставления исчерпывающей информации, необходимой для самостоятельного выполнения предложенных преподавателем задач.
- самостоятельное выполнение заданий согласно обозначенной рабочей программой дисциплины (модуля) тематики.

Задания для самостоятельной работы к Разделу 1

1. ЭДС источника тока 2 В, внутреннее сопротивление 1 Ом. Определить силу тока в цепи, если внешняя цепь потребляет мощность 0,75 Вт.
2. Имеется катушка медной проволоки сечением 0,1 мм². Масса проволоки 0,3 кг. Определить сопротивление проволоки. Удельное сопротивление меди $0,017 \cdot 10^{-6}$ Ом · м, а плотность меди равна 8900 кг/м³.
3. Две электрические лампочки включены в сеть параллельно. Сопротивление первой лампочки 360 Ом, второй – 240 Ом. Какая из лампочек поглощает большую мощность и во сколько раз?
4. В бытовой электроплитке, рассчитанной на напряжение 220 В, имеются две спирали, сопротивление каждой из которых 80,7 Ом. Найти мощность электроплитки, если спирали включены параллельно.

Вопросы для самостоятельной работы к Разделу 1

Постоянный ток. Законы постоянного тока. Сила тока. Условия, необходимые для существования тока. Виды соединения проводников. Понятие постоянного тока. Параллельное и последовательное соединение проводников. Взаимодействие проводников с током. Сила электрического тока. Электрическое напряжение. Энергия электрического поля. Тепловые двигатели. Необратимость тепловых процессов. Напряженность электрического поля. Принцип суперпозиции. Графическое изображение полей

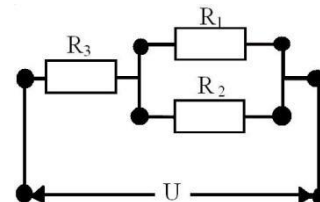
Литература для самостоятельного изучения к Разделу 1.

1. Интеллектуальные информационные системы и технологии. Айзензон, А. Е. Физика: учебник и практикум для вузов / А. Е. Айзензон. — Москва: Издательство Юрайт, 2023. — 335 с. — (Высшее образование). — ISBN 978-5-534-00487-8. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://www.urait.ru/bcode/511373> (дата обращения: 07.03.2023).
2. Бондарев, Б. В. Курс общей физики в 3 кн. Книга 1: механика: учебник для бакалавров / Б. В. Бондарев, Н. П. Калашников, Г. Г. Спирин. — Москва: Издательство Юрайт, 2022. — 353 с. — (Бакалавр. Академический курс). — ISBN 978-5-9916-1753-6. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509098> (дата обращения: 07.03.2023).
3. Бондарев, Б. В. Курс общей физики в 3 кн. Книга 2: электромагнетизм, оптика, квантовая физика: учебник для бакалавров / Б. В. Бондарев, Н. П. Калашников, Г. Г. Спирин. — 2-е изд. — Москва: Издательство Юрайт, 2022. — 441 с. — (Бакалавр. Академический курс). — ISBN 978-5-9916-1754-3. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509100> (дата обращения: 07.03.2023).
4. Бондарев, Б. В. Курс общей физики в 3 кн. Книга 3: термодинамика, статистическая физика, строение вещества: учебник для бакалавров / Б. В. Бондарев, Н. П. Калашников, Г. Г. Спирин. — 2-е изд. — Москва: Издательство Юрайт, 2022. — 369 с. — (Бакалавр.

Задания для самостоятельной работы к Разделу 2

1. Цепь состоит из трех последовательно соединенных проводников. Сопротивление первого проводника 4 Ом, второго — 6 Ом. Напряжение на концах третьего проводника равно 4 В. Общее напряжение на проводниках 24 В. Найти силу тока в цепи и напряжение на концах первого и второго проводников.

2. Сопротивление $R_1 = 2$ Ом и $R_2 = 3$ Ом соединены параллельно. К ним последовательно присоединено сопротивление $R_3 = 1$ Ом. Определить мощность, выделяющуюся на каждом сопротивлении, если общее напряжение в цепи равно $U = 4,4$ В.



Вопросы для самостоятельной работы к Разделу 2

Понятие постоянного тока. Параллельное и последовательное соединение проводников. Взаимодействие проводников с током. Сила электрического тока. Электрическое напряжение. Энергия электрического поля.

Литература для самостоятельного изучения к Разделу 2.

1. Интеллектуальные информационные системы и технологии. Айзензон, А. Е. Физика: учебник и практикум для вузов / А. Е. Айзензон. — Москва: Издательство Юрайт, 2023. — 335 с. — (Высшее образование). — ISBN 978-5-534-00487-8. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://www.urait.ru/bcode/511373> (дата обращения: 07.03.2023).

2. Бондарев, Б. В. Курс общей физики в 3 кн. Книга 1: механика: учебник для бакалавров / Б. В. Бондарев, Н. П. Калашников, Г. Г. Спирин. — Москва: Издательство Юрайт, 2022. — 353 с. — (Бакалавр. Академический курс). — ISBN 978-5-9916-1753-6. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509098> (дата обращения: 07.03.2023).

3. Бондарев, Б. В. Курс общей физики в 3 кн. Книга 2: электромагнетизм, оптика, квантовая физика: учебник для бакалавров / Б. В. Бондарев, Н. П. Калашников, Г. Г. Спирин. — 2-е изд. — Москва: Издательство Юрайт, 2022. — 441 с. — (Бакалавр. Академический курс). — ISBN 978-5-9916-1754-3. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509100> (дата обращения: 07.03.2023).

4. Бондарев, Б. В. Курс общей физики в 3 кн. Книга 3: термодинамика, статистическая физика, строение вещества: учебник для бакалавров / Б. В. Бондарев, Н. П. Калашников, Г. Г. Спирин. — 2-е изд. — Москва: Издательство Юрайт, 2022. — 369 с. — (Бакалавр. Академический курс). — ISBN 978-5-9916-1755-0. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/508976> (дата обращения: 07.03.2023).

Задания для самостоятельной работы к Разделу 3

1. Электрический чайник, содержащий 0,6 л воды при температуре 10°C , включили и забыли выключить. Через сколько времени после включения выкипит вся вода? Сопротивление нагревательного элемента чайника 14,4 Ом, напряжение в цепи 110В, КПД

чайника 60 %. Удельная теплоемкость воды $4,2 \text{ кДж}/(\text{кг} \cdot \text{Л})$, удельная теплота парообразования равна $2260 \text{ кДж}/\text{кг}$.

2. Какой длины надо взять никелиновую проволоку сечением $0,84 \text{ мм}^2$, чтобы изготовить нагреватель, при помощи которого можно было бы нагреть 2 л воды от температуры 20° до кипения за 10 мин при КПД, равном 80 %? Удельное сопротивление никелина $\rho = 0,420 \cdot 10^{-6} \text{ Ом} \cdot \text{м}$, напряжение в сети 220В

3. При замыкании на проводник сопротивлением 5 Ом сила тока равна 1А. Сила тока короткого замыкания батареи равна 6А. Какую наибольшую полезную мощность может дать батарея?

Вопросы для самостоятельной работы к Разделу 3

1. Что такое переменный ток?
2. Как можно получить переменный ток?
3. Активное сопротивление в цепи переменного тока.
4. В чем суть резонанса напряжений и токов?
5. Как определить мгновенную мощность тока?
6. Чему равна мгновенная мощность при наличии переменного тока на участке цепи, имеющей: а) только активное сопротивление; б) только реактивное сопротивление?
7. Чему равна средняя мощность, если на участке включены: а) активное сопротивление; б) реактивное сопротивление; в) активное и реактивное сопротивления?
8. Как определить активное сопротивление участка цепи переменному току, зная мощность, выделяющуюся на этом участке, и показания амперметра?

Литература для самостоятельного изучения к Разделу 3.

1. Интеллектуальные информационные системы и технологии. Айзензон, А. Е. Физика: учебник и практикум для вузов / А. Е. Айзензон. — Москва: Издательство Юрайт, 2023. — 335 с. — (Высшее образование). — ISBN 978-5-534-00487-8. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://www.urait.ru/bcode/511373> (дата обращения: 07.03.2023).

2. Бондарев, Б. В. Курс общей физики в 3 кн. Книга 1: механика: учебник для бакалавров / Б. В. Бондарев, Н. П. Калашников, Г. Г. Спирин. — Москва: Издательство Юрайт, 2022. — 353 с. — (Бакалавр. Академический курс). — ISBN 978-5-9916-1753-6. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509098> (дата обращения: 07.03.2023).

3. Бондарев, Б. В. Курс общей физики в 3 кн. Книга 2: электромагнетизм, оптика, квантовая физика: учебник для бакалавров / Б. В. Бондарев, Н. П. Калашников, Г. Г. Спирин. — 2-е изд. — Москва: Издательство Юрайт, 2022. — 441 с. — (Бакалавр. Академический курс). — ISBN 978-5-9916-1754-3. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509100> (дата обращения: 07.03.2023).

4. Бондарев, Б. В. Курс общей физики в 3 кн. Книга 3: термодинамика, статистическая физика, строение вещества: учебник для бакалавров / Б. В. Бондарев, Н. П. Калашников, Г. Г. Спирин. — 2-е изд. — Москва: Издательство Юрайт, 2022. — 369 с. — (Бакалавр. Академический курс). — ISBN 978-5-9916-1755-0. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/508976> (дата обращения: 07.03.2023).

Задания для самостоятельной работы к Разделу 4

1. На деревянное кольцо, диаметр которого $d=30$ см, а поперечное сечение имеет вид круга, навита обмотка из медной проволоки массой $m=2$ кг. Обмотку присоединяют сначала к сети постоянного тока, затем к сети переменного тока (50 Гц), причем действующее напряжение равно напряжению постоянного тока. В каком случае потребляется большая мощность?

2. Какое сопротивление переменному току в 1000 Гц представляет реостат, если его активное сопротивление 100 Ом, число витков 2000 , длина 50 см и площадь витка 15 см²?

3. В сеть переменного тока (частота 50 Гц) включены последовательно соединенные реостат с сопротивлением $R=103$ Ом и катушка индуктивности, состоящая из железного цилиндрического сердечника с обмоткой из $N=400$ витков медной проволоки сечением $0,5$ мм². Катушка имеет длину 40 см и диаметр $d=4$ см. В цепи наблюдается сдвиг фаз между напряжением и током $\varphi = 30^\circ$. Определите среднюю магнитную проницаемость железа сердечника.

4. В цепь переменного тока с частотой $f=50$ Гц включена катушка с индуктивностью $L=0,01$ Гн и активным сопротивлением $R=2$ Ом. Пользуясь методом комплексных амплитуд, определите: а) напряжение, обеспечивающее в катушке ток с амплитудой $I_t=0,5$ А; б) сдвиг фаз между током и напряжением.

5. В цепь с амплитудой напряжения 440 В и частотой 50 Гц включены последовательно нормально горящая лампочка накаливания и конденсатор. Какова емкость конденсатора, если на лампочке написано: «55 Вт, 110 В»? Какова разность фаз между током и напряжением в цепи?

Литература для самостоятельного изучения к Разделу 4.

1. Интеллектуальные информационные системы и технологии. Айзензон, А. Е. Физика: учебник и практикум для вузов / А. Е. Айзензон. — Москва: Издательство Юрайт, 2023. — 335 с. — (Высшее образование). — ISBN 978-5-534-00487-8. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://www.urait.ru/bcode/511373> (дата обращения: 07.03.2023).

2. Бондарев, Б. В. Курс общей физики в 3 кн. Книга 1: механика: учебник для бакалавров / Б. В. Бондарев, Н. П. Калашников, Г. Г. Спирин. — Москва: Издательство Юрайт, 2022. — 353 с. — (Бакалавр. Академический курс). — ISBN 978-5-9916-1753-6. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509098> (дата обращения: 07.03.2023).

3. Бондарев, Б. В. Курс общей физики в 3 кн. Книга 2: электромагнетизм, оптика, квантовая физика: учебник для бакалавров / Б. В. Бондарев, Н. П. Калашников, Г. Г. Спирин. — 2-е изд. — Москва: Издательство Юрайт, 2022. — 441 с. — (Бакалавр. Академический курс). — ISBN 978-5-9916-1754-3. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509100> (дата обращения: 07.03.2023).

4. Бондарев, Б. В. Курс общей физики в 3 кн. Книга 3: термодинамика, статистическая физика, строение вещества: учебник для бакалавров / Б. В. Бондарев, Н. П. Калашников, Г. Г. Спирин. — 2-е изд. — Москва: Издательство Юрайт, 2022. — 369 с. — (Бакалавр. Академический курс). — ISBN 978-5-9916-1755-0. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/508976> (дата обращения: 07.03.2023).

2. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ) И САМОСТОЯТЕЛЬНОЙ РАБОТЕ

Освоение обучающимся дисциплины (модуля) «Технологии возможностей и безбарьерной среды» предполагает изучение материалов дисциплины на аудиторных занятиях и в ходе самостоятельной работы. Аудиторные занятия проходят в форме лекций, семинаров, практических и лабораторных занятий. Самостоятельная работа включает разнообразный комплекс видов и форм работы обучающихся.

Для успешного освоения дисциплины (модуля) и достижения поставленных целей необходимо внимательно ознакомиться с рабочей программы дисциплины (модуля), доступной в электронной информационно-образовательной среде РГСУ.

Следует обратить внимание на списки основной и дополнительной литературы, на предлагаемые преподавателем ресурсы информационно-телекоммуникационной сети Интернет. Эта информация необходима для самостоятельной работы обучающегося.

При подготовке к аудиторным занятиям необходимо помнить особенности каждой формы его проведения.

Подготовка к учебному занятию лекционного типа заключается в следующем.

С целью обеспечения успешного обучения обучающийся должен готовиться к лекции, поскольку она является важнейшей формой организации учебного процесса, поскольку:

- знакомит с новым учебным материалом;
- разъясняет учебные элементы, трудные для понимания;
- систематизирует учебный материал;
- ориентирует в учебном процессе.

С этой целью:

- внимательно прочитайте материал предыдущей лекции;
- ознакомьтесь с учебным материалом по учебнику и учебным пособиям с темой прочитанной лекции;
- внесите дополнения к полученным ранее знаниям по теме лекции на полях лекционной тетради;
- запишите возможные вопросы, которые вы зададите лектору на лекции по материалу изученной лекции;
- постарайтесь уяснить место изучаемой темы в своей подготовке;
- узнайте тему предстоящей лекции (по тематическому плану, по информации лектора) и запишите информацию, которой вы владеете по данному вопросу.

Подготовка к занятию семинарского типа.

При подготовке и работе во время проведения занятий семинарского типа следует обратить внимание на следующие моменты: на процесс предварительной подготовки, на работу во время занятия, обработку полученных результатов, исправление полученных замечаний.

Предварительная подготовка к учебному занятию семинарского типа заключается в изучении теоретического материала в отведенное для самостоятельной работы время, ознакомление с инструктивными материалами с целью осознания задач практического занятия, техники безопасности при работе с приборами, веществами.

Работа во время проведения учебного занятия семинарского типа включает:

- консультирование студентов преподавателями и вспомогательным персоналом с целью предоставления исчерпывающей информации, необходимой для самостоятельного выполнения предложенных преподавателем задач.
- самостоятельное выполнение заданий согласно обозначенной рабочей программой дисциплины (модуля) тематики.

Самостоятельная работа.

Самостоятельная работа - планируемая учебная, учебно-исследовательская, научно-исследовательская работа студентов, выполняемая во внеаудиторное (аудиторное) время по заданию и при методическом руководстве преподавателя, но без его непосредственного

участия (при частичном непосредственном участии преподавателя, оставляющем ведущую роль за работой студентов).

Для более углубленного изучения темы задания для самостоятельной работы рекомендуется выполнять параллельно с изучением данной темы. При выполнении заданий по возможности используйте наглядное представление материала.

Самостоятельная работа студентов в ВУЗе является важным видом учебной и научной деятельности студента. Самостоятельная работа студентов играет значительную роль в рейтинговой технологии обучения.

К современному специалисту в области медицины общество предъявляет достаточно широкий перечень требований, среди которых немаловажное значение имеет наличие у выпускников определенных навыков (компетенций) и умения самостоятельно добывать знания из различных источников, систематизировать полученную информацию, давать оценку конкретной профессиональной ситуации. Формирование такого умения происходит в течение всего периода обучения.

В процессе самостоятельной работы студент приобретает необходимые для будущей специальности компетенции, навыки самоорганизации, самоконтроля, самоуправления, саморефлексии и становится активным самостоятельным субъектом учебной деятельности.

Самостоятельная работа студентов должна оказывать важное влияние на формирование личности будущего специалиста, его компетентность. Каждый студент самостоятельно определяет режим своей работы и меру труда, затрачиваемого на овладение учебным содержанием по каждой дисциплине. Он выполняет внеаудиторную работу по личному индивидуальному плану, в зависимости от его подготовки, времени и других условий.

Виды самостоятельной работы.

Работа с литературой.

При работе с книгой необходимо подобрать литературу, научиться правильно ее читать, вести записи. Для подбора литературы в библиотеке используются алфавитный и систематический каталоги. Важно помнить, что рациональные навыки работы с книгой - это всегда большая экономия времени и сил. Правильный подбор учебников рекомендуется преподавателем, читающим лекционный курс. Необходимая литература может быть также указана в методических разработках по данному курсу. Изучая материал по учебнику, следует переходить к следующему вопросу только после правильного уяснения предыдущего, описывая на бумаге все выкладки и вычисления (в том числе те, которые в учебнике опущены или на лекции даны для самостоятельного вывода). При изучении любой дисциплины большую и важную роль играет самостоятельная индивидуальная работа. Особое внимание следует обратить на определение основных понятий курса. Студент должен подробно разбирать примеры, которые поясняют такие определения, и уметь строить аналогичные примеры самостоятельно. Нужно добиваться точного представления о том, что изучаешь. Полезно составлять опорные конспекты. При изучении материала по учебнику полезно в тетради (на специально отведенных полях) дополнять конспект лекций. Там же следует отмечать вопросы, выделенные студентом для консультации с преподавателем. Выводы, полученные в результате изучения, рекомендуется в конспекте выделять, чтобы они при перечитывании записей лучше запоминались. Опыт показывает, что многим студентам помогает составление листа опорных сигналов, содержащего важнейшие и наиболее часто употребляемые формулы и понятия. Такой лист помогает запомнить формулы, основные положения лекции, а также может служить постоянным справочником для студента. Различают два вида чтения: первичное и вторичное. Первичное - это внимательное, неторопливое чтение, при котором можно остановиться на трудных местах.

Задача вторичного чтения - полное усвоение смысла целого (по счету это чтение может быть и не вторым, а третьим или четвертым). Самостоятельная работа с учебниками и книгами (а также самостоятельное теоретическое исследование проблем, обозначенных преподавателем на лекциях) – это важнейшее условие формирования у себя научного способа познания.

Чтение научного текста является частью познавательной деятельности. Ее цель – извлечение из текста необходимой информации. От того, насколько осознанна читающим собственная внутренняя установка при обращении к печатному слову (найти нужные сведения, усвоить информацию полностью или частично, критически проанализировать материал и т.п.) во многом зависит эффективность осуществляемого действия.

Методические рекомендации по составлению конспекта:

1. Внимательно прочитайте текст. Уточните в справочной литературе непонятные слова. При записи не забудьте вынести справочные данные на поля конспекта;
2. Выделите главное, составьте план;
3. Кратко сформулируйте основные положения текста, отметьте аргументацию автора;
4. Законспектируйте материал, четко следуя пунктам плана. При конспектировании старайтесь выразить мысль своими словами. Записи следует вести четко, ясно.
5. Грамотно записывайте цитаты. Цитируя, учитывайте лаконичность, значимость мысли. В тексте конспекта желательно приводить не только тезисные положения, но и их доказательства. При оформлении конспекта необходимо стремиться к емкости каждого предложения. Мысли автора книги следует излагать кратко, заботясь о стиле и выразительности написанного. Число дополнительных элементов конспекта должно быть логически обоснованным, записи должны распределяться в определенной последовательности, отвечающей логической структуре произведения. Для уточнения и дополнения необходимо оставлять поля. Овладение навыками конспектирования требует от студента целеустремленности, повседневной самостоятельной работы.

Методические материалы по самостоятельному решению задач

При самостоятельном решении задач нужно обосновывать каждый этап решения, исходя из теоретических положений курса. Если студент видит несколько путей решения проблемы (задачи), то нужно сравнить их и выбрать самый рациональный. Полезно до начала вычислений составить краткий план решения проблемы (задачи). Решение проблемных задач или примеров следует излагать подробно, вычисления располагать в строгом порядке, отделяя вспомогательные вычисления от основных. Решения при необходимости нужно сопровождать комментариями, схемами, чертежами и рисунками. Следует помнить, что решение каждой учебной задачи должно доводиться до окончательного логического ответа, которого требует условие, и по возможности с выводом. Полученный ответ следует проверить способами, вытекающими из существа данной задачи. Полезно также (если возможно) решать несколькими способами и сравнить полученные результаты. Решение задач данного типа нужно продолжать до приобретения твердых навыков в их решении.

Методические материалы к выполнению реферата

Реферат (от лат. *refere* – сообщать) – краткое изложение в письменном виде или в форме публикации доклада, содержания научного труда (трудов), литературы по теме. Работа над рефератом условно разделяется на выбор темы, подбор литературы, подготовку и защиту плана; написание теоретической части и всего текста с указанием библиографических данных используемых источников, подготовку доклада, выступление с ним. Тематика рефератов полностью связана с основными вопросами изучаемого курса.

Список литературы к темам не дается, и обучающиеся самостоятельно ведут библиографический поиск, причем им не рекомендуется ограничиваться университетской библиотекой.

Важно учитывать, что написание реферата требует от обучающихся определенных усилий. Особое внимание следует уделить подбору литературы, методике ее изучения с целью отбора и обработки собранного материала, обоснованию актуальности темы и теоретического уровня обоснованности используемых в качестве примеров фактов какой-либо деятельности.

Выбрав тему реферата, начав работу над литературой, необходимо составить план. Изучая литературу, продолжается обдумывание темы, осмысливание прочитанного, делаются выписки, сопоставляются точки зрения разных авторов и т.д. Реферативная работа сводится к тому, чтобы в ней выделились две взаимосвязанные стороны: во-первых, ее следует

рассматривать как учебное задание, которое должен выполнить обучаемый, а во-вторых, как форму научной работы, творческого воображения при выполнении учебного задания.

Наличие плана реферата позволяет контролировать ход работы, избежать формального переписывания текстов из первоисточников.

Оформление реферата включает титульный лист, оглавление и краткий список использованной литературы. Список использованной литературы размещается на последней странице рукописи или печатной форме реферата. Реферат выполняется в письменной или печатной форме на белых листах формата А4 (210 x 297 мм). Шрифт Times New Roman, кегель 14, через 1,5 интервала при соблюдении следующих размеров текста: верхнее поле – 25 мм, нижнее – 20 мм, левое – 30 мм, правое – 15 мм. Нумерация страниц производится сверху листа, по центру. Титульный лист нумерации не подлежит.

Рефераты должны быть написаны простым, ясным языком, без претензий на наукообразность. Следует избегать сложных грамматических оборотов, непривычных терминов и символов. Если же такие термины и символы все-таки приводятся, то необходимо разъяснять их значение при первом упоминании в тексте реферата.

Объем реферата предполагает тщательный отбор информации, необходимой для краткого изложения вопроса. Важнейший этап – редактирование готового текста реферата и подготовка к обсуждению. Обсуждение требует хорошей ориентации в материале темы, умения выделить главное, поставить дискуссионный вопрос, привлечь внимание слушателей к интересной литературе, логично и убедительно изложить свои мысли.

Рефераты обязательно подлежат защите. Процедура защиты начинается с определения оппонентов защищающего свою работу. Они стремятся дать основательный анализ работы обучающимся, обращают внимание на положительные моменты и недостатки реферата, дают общую оценку содержанию, форме преподнесения материала, характеру использованной литературы. Иногда они дополняют тот или иной раздел реферата. Последнее особенно ценно, ибо говорит о глубоком знании обучающимся-оппонентом изучаемой проблемы.

Обсуждение не ограничивается заслушиванием оппонентов. Другие обучающиеся имеют право уточнить или опровергнуть какое-либо утверждение. Преподаватель предлагает любому обучающемуся задать вопрос по существу доклада или попытаться подвести итог обсуждению.

Алгоритм работы над рефератом

1. Выбор темы

Тема должна быть сформулирована грамотно (с литературной точки зрения);

В названии реферата следует поставить четкие рамки рассмотрения темы;

Желательно избегать слишком длинных названий;

Следует по возможности воздерживаться от использования в названии спорных с научной точки зрения терминов, излишней наукообразности, а также чрезмерного упрощения формулировок.

2. Реферат следует составлять из пяти основных частей: введения; основной части; заключения; списка литературы; приложений.

3. Основные требования к введению:

Во введении не следует концентрироваться на содержании; введение должно включать краткое обоснование актуальности темы реферата, где требуется показать, почему данный вопрос может представлять научный интерес и есть ли связь представляемого материала с современностью. Таким образом, тема реферата должна быть актуальна либо с научной точки зрения, либо с современных позиций.

Очень важно выделить цель, а также задачи, которые требуется решить для выполнения цели.

Введение должно содержать краткий обзор изученной литературы, в котором указывается взятый из того или иного источника материал, кратко анализируются изученные источники, показываются их сильные и слабые стороны;

Объем введения составляет две страницы текста.

4. Требования к основной части реферата:

Основная часть содержит материал, отобранный для рассмотрения проблемы;

Также основная часть должна включать в себя собственно мнение обучающихся и сформулированные самостоятельные выводы, опирающиеся на приведенные факты;

Материал, представленный в основной части, должен быть логически изложен и распределен по параграфам, имеющим свои названия;

В изложении основной части необходимо использовать сноски (в первую очередь, когда приводятся цифры и чьи-то цитаты);

Основная часть должна содержать иллюстративный материал (графики, таблицы и т. д.);

Объем основной части составляет около 10 страниц.

5. Требования к заключению:

В заключении формулируются выводы по параграфам, обращается внимание на выдвинутые во введении задачи и цели;

Заключение должно быть четким, кратким, вытекающим из содержания основной части.

6. Требования к оформлению списка литературы (по ГОСТу):

Необходимо соблюдать правильность последовательности записи источников: сначала следует писать фамилию, а после инициалы; название работы не ставится в кавычки; после названия сокращенно пишется место издания; затем идет год издания; наконец, называется процитированная страница.

Критерии оценки реферата

Обучающийся, защищающий реферат, должен рассказать о его актуальности, поставленных целях и задачах, изученной литературе, структуре основной части, сделанных в ходе работы выводах.

По окончании выступления ему может быть задано несколько вопросов по представленной проблеме.

Оценка складывается из соблюдения требований к реферату, грамотного раскрытия темы, умения четко рассказывать о представленном реферате, способности понять суть задаваемых по работе вопросов и найти точные ответы на них.

Методические материалы к выполнению эссе

Эссе – литературное произведение небольшого объема, обычно прозаическое, свободной композиции, передающее индивидуальные впечатления, суждения, соображения автора о той или иной проблеме, теме, о том или ином событии или явлении. Это вид самостоятельной исследовательской работы обучающихся, с целью углубления и закрепления теоретических знаний и освоения практических навыков. Цель эссе состоит в развитии самостоятельного творческого мышления и письменного изложения собственных мыслей. При написании эссе обучающийся должен представить развернутый письменный ответ на теоретический или практический актуальный вопрос, объявленный преподавателем в аудитории непосредственно перед ее написанием. В процессе написания эссе разрешается пользоваться нормативно-правовыми актами, конспектом лекций (в печатном виде). Использование интернет-ресурсов не допускается. Темы эссе преподаватель предлагает из числа тех, которые обучающиеся уже рассматривали на лекциях или семинарских занятиях, исходя из содержания заданий в составе оценочных средств. По решению преподавателя, в качестве темы эссе может быть выбрана одна или несколько тем, которые могут быть распределены между обучающимися по желанию.

Требования к выполнению эссе:

1. Проводится письменно.

2. Эссе выполняется на компьютере (гарнитура Times New Roman, шрифт 14) через 1,5 интервала с полями: верхнее, нижнее – 2; правое – 3; левое – 1,5. Отступ первой строки абзаца – 1,25. Сноски – постраничные. Таблицы и рисунки встраиваются в текст работы. При этом обязательный заголовок таблицы надо размещать над табличным полем, а рисунки

сопровождать подрисуночными подписями. При включении в эссе нескольких таблиц и/или рисунков их нумерация обязательна. Обязательна и нумерация страниц. Их целесообразно проставлять внизу страницы – по середине или в правом углу. Номер страницы не ставится на титульном листе, но в общее число страниц он включается. Объем эссе, без учета приложений, не должен превышать 5 страниц. Значительное превышение установленного объема является недостатком работы и указывает на то, что обучающийся не сумел отобрать и переработать необходимый материал.

3. Работа должна содержать собственные умозаключения по сути поставленной проблемы, включать самостоятельно проведенный анализ по сути этой проблемы, выводы, обобщающие авторскую позицию по поставленной проблеме.

Критерии оценки эссе:

«Отлично» – исключительные знания материала, абсолютное понимание сути, безукоризненное знание основных понятий и положений, логически и лексически грамотно изложенный, содержательный, аргументированный, конкретный и исчерпывающий ответ.

«Хорошо» – глубокие знания материала, правильное понимание сути, знание основных понятий и положений, содержательный, полный и конкретный ответ.

«Удовлетворительно» – твердые, но недостаточно полные знания, верное понимание сути, в целом правильный ответ.

«Неудовлетворительно» – непонимание сущности задания, грубые ошибки в ответе.

Методические материалы по выполнению тестирования.

Тестовые задания содержат вопросы и 3-4 варианта ответа по базовым положениям изучаемой темы/раздела, составлены с расчетом на знания, полученные обучающимся в процессе изучения темы/раздела.

Тестовые задания выполняются в письменной форме и сдаются преподавателю, ведущему дисциплину (модуль). На выполнение тестовых заданий обучающимся отводится 45 минут.

При обработке результатов оценочной процедуры используются: критерии оценки по содержанию и качеству полученных ответов, ключи, оценочные листы.

Критерии оценки теста:

«Зачтено» - если обучающийся правильно выполнил не менее 2/3 всей работы или допустил не более одной грубой ошибки и двух недочетов, не более одной грубой и одной негрубой ошибки, не более трех негрубых ошибок, одной негрубой ошибки и трех недочетов, при наличии четырех-пяти недочетов.

«Не зачтено» - если число ошибок и недочетов превысило норму для оценки 3 или правильно выполнено менее 2/3 всей работы.

Методические материалы по выполнению доклада.

Рекомендуется следующая структура доклада:

1. титульный лист, содержание доклада;
2. краткое изложение;
3. цели и задачи;
4. изложение характера исследований и рассмотренных проблем, гипотезы, спорные вопросы;
5. источники информации, методы сбора и анализа данных, степень их полноты и достоверности;
6. анализ и толкование полученных в работе результатов;
7. выводы и оценки;
8. библиография и приложения.

Время выступления докладчика не должно превышать 10 минут.

Основные требования к оформлению доклада:

- титульный лист должен включать название доклада, наименование предметной (цикловой) комиссии, фамилию обучающегося;

- все использованные литературные источники сопровождаются библиографическим описанием;
- приводимая цитата из источника берется в кавычки (оформляются сноски);
- единицы измерения должны применяться в соответствии с действующими стандартами;
- все названия литературных источников следует приводить в соответствии с новейшими изданиями;
- рекомендуется включение таблиц, графиков, схем, если они отражают основное содержание или улучшают ее наглядность;
- названия фирм, учреждений, организаций и предприятий должны именоваться так, как они указываются в источнике;

Критерии оценки доклада

При выполнении доклада обучающийся должен продемонстрировать умение кратко излагать прочитанный материал, а также умение обобщать и анализировать материал по теме доклада.

Презентация

Методические материалы к презентациям

1. Объем презентации 10 -20 слайдов.
2. На титульном слайде должно быть отражено:
 - наименование факультета;
 - тема презентации;
 - фамилия, имя, отчество, направление подготовки/ специальность, направленность (профиль)/ специализация, форма обучения, номер группы автора презентации;
 - фамилия, имя, отчество, степень, звание, должность руководитель работы;
 - год выполнения работы.
3. В презентации должны быть отражены обоснование актуальности представляемого материала, цели и задачи работы.
4. Содержание презентации должно включать наиболее значимый материал доклада, а также, при необходимости, таблицы, диаграммы, рисунки, фотографии, карты, видео – вставки, звуковое сопровождение.
5. Заключительный слайд должен содержать информацию об источниках информации для презентации.

Критерии оценки презентации

1. Объем презентации 10 -20 слайдов.
2. Правильность оформления титульного слайда.
3. Актуальность отобранного материала, обоснованность формулировки цели и задач работы.
4. Наглядность и логичность презентации, обоснованность использования таблиц, диаграмм, рисунков, фотографий, карт, видео – вставок, звукового сопровождения; правильный выбор шрифтов, фона, других элементов дизайна слайда.
5. Объем и качество источников информации (не менее 2-х интернет – источников и не менее 2-х литературных источников).

Методические материалы по подготовке к опросу

Самостоятельная работа обучающихся включает подготовку к опросу на практических занятиях. Для этого обучающийся изучает лекции, основную и дополнительную литературу, публикации, информацию из Интернет-ресурсов.

Тема и вопросы к практическим занятиям, вопросы для самоконтроля содержатся в рабочей программе и доводятся до студентов заранее.

Для подготовки к опросу обучающемуся необходимо ознакомиться с материалом, посвященным теме практического занятия, в учебнике или другой рекомендованной литературе, конспекте лекции, обратить внимание на усвоение основных понятий

дисциплины, выявить неясные вопросы и подобрать дополнительную литературу для их освещения.

Критерии оценки опроса

«Отлично»:

- дан полный, развернутый ответ на поставленный вопрос;
- в ответе прослеживается четкая структура, логическая последовательность, отражающая сущность раскрываемых понятий, теорий, явлений;
- знание по предмету демонстрируется на фоне понимания его в системе данной науки и междисциплинарных связей;
- свободное владение терминологией;
- ответы на дополнительные вопросы четкие, краткие;

«Хорошо»:

- дан полный, развернутый ответ на поставленный вопрос, показано умение выделять существенные и несущественные признаки, причинно-следственные связи;
- ответ недостаточно логичен с единичными ошибками в частностях, исправленные студентом с помощью преподавателя;
- единичные ошибки в терминологии;
- ответы на дополнительные вопросы правильные, недостаточно полные и четкие.

«Удовлетворительно»:

- ответ не полный, с ошибками в деталях, умение раскрыть значение обобщённых знаний не показано, речевое оформление требует поправок, коррекции;
- логика и последовательность изложения имеют нарушения, студент не способен самостоятельно выделить существенные и несущественные признаки и причинно-следственные связи;
- ошибки в раскрываемых понятиях, терминах;
- студент не ориентируется в теме, допускает серьезные ошибки;
- студент не может ответить на большую часть дополнительных вопросов.

«Неудовлетворительно»:

- ответ представляет собой разрозненные знания с существенными ошибками по вопросу;
- присутствуют фрагментарность, нелогичность изложения, студент не осознает связь обсуждаемого вопроса с другими объектами дисциплины, речь неграмотная;
- незнание терминологии;
- ответы на дополнительные вопросы неправильные.

Методические материалы по выполнению практического задания

При выполнении практического задания обучающийся придерживается следующего алгоритма:

1. Записать дату, тему и цель задания;
2. Ознакомиться с правилами и условия выполнения практического задания;
3. Повторить теоретические задания, необходимые для рациональной работы и других практических действий, используя конспекты лекций и рекомендованную литературу, представленной в программе;
4. Выполнить работу по предложенному алгоритму действий;
5. Обобщить результаты работы, сформулировать выводы / дать ответы на контрольные вопросы;

Работа должна быть выполнена грамотно, с соблюдением культуры изложения. При использовании данных из учебных, методических пособий и другой литературы, периодических изданий, Интернет-источников должны иметься ссылки на вышеперечисленные.

Критерии оценки практического задания:

«Отлично» – правильный ответ, дается четкое обоснование принятому решению; рассуждения четкие последовательные логические; используются ссылки на полученные при

изучении дисциплины знания; правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Хорошо» – правильный ответ, дается обоснование принятому решению; но с не существенными ошибками, в рассуждениях отсутствует логическая последовательность; используются ссылки на полученные при изучении дисциплины знания, правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Удовлетворительно» – правильный ответ, допускаются грубые ошибки в обосновании принятого решения; рассуждения не последовательные сумбурные; используются ссылки на полученные при изучении дисциплины знания; используются формулы, процедуры, понятия, имеющие прямое значение для подтверждения принятого решения, однако, при обращении к ним допускаются серьезные ошибки, студент не может правильно ими воспользоваться.

«Неудовлетворительно, не зачтено» – ответ неверный, отсутствует обоснование принятому решению; студент демонстрирует полное непонимание сути вопроса.

Для оценки решения ситуационной задачи (аналитического задания):

Оценка «отлично» выставляется, если задача решена грамотно, ответы на вопросы сформулированы четко. Эталонный ответ полностью соответствует решению студента, которое хорошо обосновано теоретически.

Оценка «хорошо» выставляется, если задача решена, ответы на вопросы сформулированы не достаточно четко. Решение студента в целом соответствует эталонному ответу, но не достаточно хорошо обосновано теоретически.

Оценка «удовлетворительно» выставляется, если задача решена не полностью, ответы не содержат всех необходимых обоснований решения.

Оценка «неудовлетворительно» выставляется, если задача не решена или имеет грубые теоретические ошибки в ответе на поставленные вопросы.

Методические материалы по выполнению лабораторного задания

При выполнении лабораторного задания обучающийся придерживается следующего алгоритма:

1. Записать дату, тему и цель задания;
2. Ознакомиться с правилами и условия выполнения задания;
3. Повторить теоретические задания, необходимые для рациональной работы и других практических действий, используя конспекты лекций и рекомендованную литературу, представленной в программе;
4. Выполнить работу по предложенному алгоритму действий;
5. Обобщить результаты работы, сформулировать выводы / дать ответы на контрольные вопросы;

Работа должна быть выполнена грамотно, с соблюдением культуры изложения. При использовании данных из учебных, методических пособий и другой литературы, периодических изданий, Интернет-источников должны иметься ссылки на вышеперечисленные.

Критерии оценки лабораторного задания:

«Отлично» – правильный ответ, дается четкое обоснование принятому решению; рассуждения четкие последовательные логические; используются ссылки на полученные при изучении дисциплины знания; правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Хорошо» – правильный ответ, дается обоснование принятому решению; но с не существенными ошибками, в рассуждениях отсутствует логическая последовательность; используются ссылки на полученные при изучении дисциплины знания, правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Удовлетворительно» – правильный ответ, допускаются грубые ошибки в обосновании принятого решения; рассуждения не последовательные сумбурные; используются ссылки на полученные при изучении дисциплины знания; используются формулы, процедуры, понятия, имеющие прямое значение для подтверждения принятого решения, однако, при обращении к ним допускаются серьезные ошибки, студент не может правильно ими воспользоваться.

«Неудовлетворительно, не зачтено» – ответ неверный, отсутствует обоснование принятому решению; студент демонстрирует полное непонимание сути вопроса.

Методические указания для подготовки к промежуточной аттестации.

Изучение учебных дисциплин (модулей) завершается зачетом/зачетом с оценкой или экзаменом. Подготовка к промежуточной аттестации способствует закреплению, углублению и обобщению знаний, получаемых, в процессе обучения, а также применению их к решению практических задач. Готовясь к экзамену, студент ликвидирует имеющиеся пробелы в знаниях, углубляет, систематизирует и упорядочивает свои знания. На зачете или экзамене студент демонстрирует то, что он освоил в процессе обучения по дисциплине (модулю).

Вначале следует просмотреть весь материал по дисциплине (модулю), отметить для себя трудные вопросы. Обязательно в них разобраться. В заключение еще раз целесообразно повторить основные положения, используя при этом листы опорных сигналов. Систематическая подготовка к занятиям в течение семестра позволит использовать время промежуточной аттестации для систематизации знаний.

3. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНИВАНИЯ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ И (ИЛИ) ОПЫТА ДЕЯТЕЛЬНОСТИ, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ

3.1. Организационные основы применения балльно-рейтинговой системы оценки успеваемости обучающихся по дисциплине (модулю)

Оценка качества освоения обучающимися дисциплины (модуля) реализуется в формате балльно-рейтинговой системы оценки успеваемости обучающихся (БРСО).

БРСО в ходе текущего контроля успеваемости и промежуточной аттестации осуществляется по 100-балльной шкале.

Академический рейтинг обучающегося по дисциплине (модулю) складывается из результатов:

- текущего контроля успеваемости (максимальный текущий рейтинг обучающегося 80 рейтинговых баллов);
- промежуточной аттестации (максимальный рубежный рейтинг обучающегося 20 рейтинговых баллов).

Условия оценки освоения обучающимся дисциплины (модуля) в формате БРСО доводятся преподавателем до сведения обучающихся на первом учебном занятии, а также размещены в свободном доступе в электронной информационно-образовательной среде Университета.

3.2. Проведение текущего контроля успеваемости обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося

В течение учебного семестра до промежуточной аттестации на основании утвержденной рабочей программы дисциплины (модуля) формируется текущий рейтинг обучающегося. Текущий рейтинг обучающегося складывается как сумма рейтинговых баллов, полученных им в течение учебного семестра по всем видам учебных занятий по учебной дисциплине.

В процессе текущего контроля оцениваются следующие действия обучающегося, направленные на освоение компетенций в рамках изучения учебной дисциплины:

- академическая активность (посещаемость учебных занятий, самостоятельное изучение содержания учебной дисциплины в электронной информационно-образовательной среде, соблюдение сроков сдачи практических заданий и текущих контрольных мероприятий и др.);

- выполнение и сдача текущих и итогового практических заданий (эссе, рефераты, творческие задания, кейс-задания, лабораторные работы, расчетные задания и др., активное участие в групповых интерактивных занятиях (дискуссии, WiKi-проекты и др.), защита проектов и др.);

- прохождение рубежей текущего контроля, включая соблюдение графика их прохождения в электронной информационно-образовательной среде.

Для планирования расчета текущего рейтинга обучающегося используются следующие пропорции:

Вид учебного действия	Максимальная рейтинговая оценка, баллов
академическая активность	10
практические задания	40
<i>из них: текущие практические задания</i>	20
<i>итоговое практическое задание</i>	20
рубежи текущего контроля	30
<i>ИТОГО:</i>	80

В течение учебного семестра по дисциплине (модулю) обучающимся должен быть накоплен текущий рейтинг не менее 52 рейтинговых баллов (65% от максимального значения текущего рейтинга).

Необходимыми условиями допуска обучающегося к промежуточной аттестации по дисциплине являются положительное прохождение обучающимся не менее 65% рубежей текущего контроля с накоплением не менее 65% максимального рейтингового балла за каждый рубеж текущего контроля и положительное выполнение итогового практического задания с накоплением не менее 65% максимального рейтингового балла, установленного за итоговое практическое задание.

Невыполнение вышеуказанных условий является текущей академической задолженностью, которая должна быть ликвидирована обучающимся до контрольного мероприятия промежуточной аттестации.

Сведения о наличии у обучающихся текущей академической задолженности, сроках и порядке добора рейтинговых баллов для её ликвидации доводятся до обучающихся педагогическим работником.

В случае неликвидации текущей академической задолженности, педагогический работник обязан во время контрольного мероприятия промежуточной аттестации поставить обучающемуся 0 рейтинговых баллов. В этом случае ликвидация текущей академической задолженности возможна в периоды проведения повторной промежуточной аттестации.

3.3. Проведение промежуточной аттестации обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося

Промежуточная аттестация по дисциплине (модулю) проводится в соответствии с Положением о промежуточной аттестации обучающихся по основным профессиональным образовательным программам высшего образования – программ специалитета в Российском государственном социальном университете и Положением о балльно-рейтинговой системе оценки успеваемости обучающихся по основным профессиональным образовательным программам высшего образования - программам бакалавриата, программам специалитета, программам магистратуры в Российском государственном социальном университете.

На промежуточную аттестацию отводится 20 рейтинговых баллов.

Ответы обучающегося на контрольном мероприятии промежуточной аттестации оцениваются педагогическим работником по 20 - балльной шкале, а итоговая оценка по дисциплине (модулю) выставляется по системе зачтено/не зачтено.

Критерии выставления оценки определяются Положением о балльно-рейтинговой системе оценки успеваемости обучающихся по основным профессиональным образовательным программам высшего образования - программам бакалавриата, программам специалитета, программам магистратуры в Российском государственном социальном университете.

В процессе определения рубежного рейтинга обучающегося используется следующая шкала:

Рубежный рейтинг	Критерии оценки освоения обучающимся учебной дисциплины в ходе контрольных мероприятий промежуточной аттестации
19-20 рейтинговых баллов	обучающийся глубоко и прочно усвоил программный материал, исчерпывающе, последовательно, грамотно и логически стройно его излагает, тесно увязывает с задачами и будущей деятельностью, не затрудняется с ответом при видоизменении задания, свободно справляется с задачами и практическими заданиями, правильно обосновывает принятые решения, умеет самостоятельно обобщать и излагать материал, не допуская
16-18 рейтинговых баллов	обучающийся твердо знает программный материал, грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос, может правильно применять теоретические положения и владеет необходимыми умениями и навыками при выполнении практических
13-15 рейтинговых баллов	обучающийся освоил основной материал, но не знает отдельных деталей, допускает неточности, недостаточно правильные формулировки, нарушает последовательность в изложении программного материала и испытывает затруднения в выполнении практических заданий
1-12 рейтинговых баллов	обучающийся не знает значительной части программного материала, допускает существенные ошибки, с большими затруднениями выполняет практические задания
0 рейтинговых баллов	не аттестован

Если результат контроля успеваемости в рамках проведения контрольных мероприятий промежуточной аттестации (рубежный рейтинг обучающегося) неудовлетворительный (получено менее 13 рейтинговых баллов), то промежуточная аттестация по учебной дисциплине (модулю) невозможна даже при наличии высокого текущего рейтинга, полученного по итогам текущего контроля по учебной дисциплине (модулю).

ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

№ п/п	Содержание изменения	Реквизиты документа об утверждении изменения	Дата введения изменения
1.			
2.			
3.			
4.			



**Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Российский государственный социальный университет»**

УТВЕРЖДАЮ

Декан факультета политических и социальных

технологий  /Пивнева С.В./

28 февраля 2024 г.

**МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ
ПРОЕКТИРОВАНИЕ БАЗ ДАННЫХ**

Направление подготовки
«Информационные системы и технологии»

Направленность
«Интеллектуальные информационные системы и технологии»

**ОБРАЗОВАТЕЛЬНАЯ ПРОГРАММА ВЫСШЕГО ОБРАЗОВАНИЯ - ПРОГРАММА
БАКАЛАВРИАТА**

Форма обучения
Очная, заочная

Москва 2024

Методические материалы по дисциплине (модулю) «Проектирование баз данных» разработаны на основании федерального государственного образовательного стандарта высшего образования – *бакалавриата* по направлению подготовки *09.03.02 Информационные системы и технологии*, утвержденного приказом Министерства образования и науки Российской Федерации от 19.09.2017 № 926, учебного плана по основной профессиональной образовательной программе высшего образования – программе бакалавриата по направлению подготовки *09.03.02 Информационные системы и технологии* (далее – «ОПОП»).

Методические материалы по дисциплине (модулю) разработаны рабочей группой в составе:

Методические материалы по дисциплине (модулю) обсуждены и утверждены на заседании кафедры информационных технологий, искусственного интеллекта и общественно-социальных технологий цифрового общества факультета социальных и политических технологий.

Протокол № 11 от «28» февраля 2024 года

Заведующий кафедрой
канд. пед. наук, доцент

_____ 

_____ С.В. Пивнева

СОДЕРЖАНИЕ

1. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ К ЛЕКЦИОННЫМ, ЛАБОРАТОРНЫМ ЗАНЯТИЯМ.....	4
1.1. Методические материалы к проведению лекционных занятий по дисциплине (модулю)	4
1.2. Методические материалы по подготовке к практическим занятиям по дисциплине (модулю)	6
1.3. Учебно-наглядные пособия по разделам (темам) дисциплины (модуля)	12
1.4. Методические материалы по подготовке к лабораторным занятиям по дисциплине (модулю)	16
2. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ) И САМОСТОЯТЕЛЬНОЙ РАБОТЕ.....	16
3. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНИВАНИЯ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ И (ИЛИ) ОПЫТА ДЕЯТЕЛЬНОСТИ, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ.....	25
3.1. Организационные основы применения балльно-рейтинговой системы оценки успеваемости обучающихся по дисциплине (модулю)	26
3.2. Проведение текущего контроля успеваемости обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося.....	26
3.3. Проведение промежуточной аттестации обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося.....	27
Приложение № 1 к методическим материалам по дисциплине (модулю). Конспекты лекционных занятий по дисциплине (модулю).....	29
КОНСПЕКТЫ ЛЕКЦИОННЫХ ЗАНЯТИЙ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)	29
Приложение № 3 к методическим материалам по дисциплине (модулю). Конспекты лабораторных занятий по дисциплине (модулю).....	131
КОНСПЕКТЫ ЛАБОРАТОРНЫХ ЗАНЯТИЙ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)	131
Приложение № 4 к методическим материалам по дисциплине (модулю). Учебно-наглядные пособия по дисциплине (модулю).....	141
УЧЕБНО-НАГЛЯДНЫЕ ПОСОБИЯ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)	141
ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ	145

1. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ К ЛЕКЦИОННЫМ, ЛАБОРАТОРНЫМ ЗАНЯТИЯМ

1.1. Методические материалы к проведению лекционных занятий по дисциплине (модулю)

Лекция - один из методов обучения, одна из основных системообразующих форм организации учебного процесса в вузе. Лекционное занятие представляет собой систематическое, последовательное, монологическое изложение педагогическим работником учебного материала, как правило, теоретического характера. Такое занятие представляет собой элемент технологии представления учебного материала путем логически стройного, систематически последовательного и ясного изложения.

Цель лекции – организация целенаправленной познавательной деятельности обучающихся по овладению программным материалом дисциплины (модуля). Чтение курса лекций позволяет дать связанное, последовательное изложение материала в соответствии с новейшими данными науки, сообщить слушателям основное содержание предмета в целостном, систематизированном виде. В ряде случаев лекция выполняет функцию основного источника информации: при отсутствии учебников и учебных пособий, чаще по новым курсам; в случае, когда новые научные данные по той или иной теме не нашли отражения в учебниках; отдельные разделы и темы очень сложны для самостоятельного изучения. В таких случаях только лектор может методически помочь обучающимся в освоении сложного материала.

Возможные формы проведения лекций:

- Вводная лекция – один из наиболее важных и трудных видов лекции при чтении систематических курсов. От успеха этой лекции во многом зависит успех усвоения всего курса. Она может содержать: определение дисциплины (модуля); краткую историческую справку о дисциплине (модуле); цели и задачи дисциплины (модуля), ее роль в общей системе обучения и связь со смежными дисциплинами (модулями); основные проблемы (понятия и определения) данной науки; основную и дополнительную учебную литературу; особенности самостоятельной работы обучающихся над дисциплиной (модулем) и формы участия в научно-исследовательской работе; отчетность по курсу.

- Информационная лекция ориентирована на изложение и объяснение обучающимся научной информации, подлежащей осмыслению и запоминанию. Это самый традиционный тип лекций в практике высшей школы.

- Заключительная лекция предназначена для обобщения полученных знаний и раскрытия перспектив дальнейшего развития данной науки.

- Обзорная лекция – это систематизация научных знаний на высоком уровне, допускающая большое число ассоциативных связей в процессе осмысления информации, излагаемой при раскрытии внутрипредметной и межпредметной связей, исключая детализацию и конкретизацию. Как правило, стержень излагаемых теоретических положений составляет научно-понятийная и концептуальная основа всего курса или крупных его разделов.

- Лекция-беседа - непосредственный контакт педагогического работника с аудиторией - диалог. По ходу лекции педагогический работник задает вопросы для выяснения мнений и уровня осведомленности обучающихся по рассматриваемой проблеме.

- Лекция-дискуссия - свободный обмен мнениями в ходе изложения лекционного материала. Педагогический работник активизирует участие в обсуждении отдельными вопросами, сопоставляет между собой различные мнения и тем самым развивает дискуссию, стремясь направить ее в нужное русло.

- Лекция с применением обратной связи включает в себе то, что в начале и конце каждого раздела лекции задаются вопросы. Первый - для того, чтобы узнать, насколько обучающиеся ориентируются в излагаемом материале, вопрос в конце раздела предназначен для выяснения степени усвоения только что изложенного материала. При

неудовлетворительных результатах контрольного опроса педагогический работник возвращается к уже прочитанному разделу, изменив при этом методику подачи материала.

- Проблемная лекция опирается на логику последовательно моделируемых проблемных ситуаций путем постановки проблемных вопросов или предъявления проблемных задач. Проблемный вопрос - это диалектическое противоречие, требующее для своего решения размышления, сравнения, поиска, приобретения и применения новых знаний. Проблемная задача содержит дополнительную вводную информацию и при необходимости некоторые ориентиры поиска ее решения.

- Программированная лекция - консультация – педагогический работник сам составляет и предлагает обучающимся вопросы. На подготовленные вопросы педагогический работник сначала просит ответить обучающихся, а затем проводит анализ и обсуждение неправильных ответов. В лекциях можно использовать наглядные материалы, а также подготовить презентацию. Что касается презентации, то в качестве визуальной поддержки ее можно органично интегрировать во все вышеупомянутые лекции. В то же время лекцию-презентацию возможно выделить и в качестве самостоятельной формы. Лекция-презентация должна отражать суть основных и (или) проблемных вопросов лекции, на которые особо следует обратить внимание обучающихся. В условиях применения активного метода проведения занятий презентация представляется весьма удачным способом донесения информации до слушателей. Единственное, на что следует обратить внимание при подготовке слайдов, - это их оформление и текст. Слайд не должен быть перегружен картинками и лишней информацией, которая будет отвлекать от основного аспекта того или иного вопроса лекции. Во время лекции можно задавать вопросы аудитории в отношении того или иного слайда, тем самым еще больше вовлекая обучающихся в проблематику.

Краткое содержание лекционных занятий

Наименование разделов и тем	Содержание учебного материала
РАЗДЕЛ 1. Теория баз данных	
Тема 1.1. Введение в базы данных. Общая характеристика основных понятий	Основные понятия теории баз данных
Тема 1.2. Системы управления базами данных	Описание целей и задач СУБД.
РАЗДЕЛ 2. Общие принципы построения (архитектура) баз данных	
Тема 2.1. Различные архитектурные решения, используемые при реализации многопользовательских СУБД.	Типы архитектур СУБД
Тема 2.2. Краткий обзор СУБД	Обзор СУБД
РАЗДЕЛ 3. Модели данных	
Тема 3.1. Различные представления о данных в базах данных. Основные этапы проектирования баз данных	Проектирование баз данных.
Тема 3.2. Первая стадия концептуального проектирования базы данных (концептуальное моделирование. Вторая стадия концептуального проектирования (Модели данных СУБД. Представление концептуальной модели средствами модели данных СУБД)	Первая стадия концептуального проектирования базы данных (концептуальное моделирование). Вторая стадия концептуального проектирования (Модели данных СУБД. Представление концептуальной модели средствами модели данных СУБД).
РАЗДЕЛ 4. Базисные операции с реляционными данными	
Тема 4.1. Формализация реляционной модели	Правила применения и принципы организации РБД. Использование формального аппарата для оптимизации схем отношений.

РАЗДЕЛ 5. Нормальные формы в реляционных базах данных	
Тема 5.1. Использование формального аппарата для оптимизации схем отношений	Использования формальных методов построения оптимальной (по определенным показателям) структуры реляционной <i>базы данных</i> путем нормализации схем отношений.
РАЗДЕЛ 6. Проектирование баз данных с использованием семантического подхода	
Тема 6.1. Физические модели данных (внутренний уровень)	Основные типовые способы организации данных в памяти ЭВМ в <i>СУБД</i> с оценкой соответствующих моделей по времени доступа к данным в базе данных и по объему занимаемой памяти
РАЗДЕЛ 7. Проектирование баз данных	
Тема 7.1. Программное обеспечение работы с современными базами данных	Основные задачи программного обеспечения баз данных, существующих подходов к решению этих задач, в том числе и о структурированном языке запросов SQL.
Тема 7.2. Основные операторы языка SQL. Интерактивный SQL. Использование языка SQL в прикладных программах	Общая характеристика операторов языка SQL и показать, как записываются основные запросы к базе данных на языке SQL (в интерактивном режиме).
РАЗДЕЛ 8. Физическое проектирование БД	
Тема 8.1. Направления развития баз данных	Основные черты в новых направлениях развития теории и практики создания баз данных (новые свойства, присущие объектно-ориентированным и распределенным базам данных) и хранилищ данных

1.2. Методические материалы по подготовке к практическим занятиям по дисциплине (модулю)

Практические (семинарские) занятия - одна из форм учебного занятия, направленная на развитие самостоятельности обучающихся и приобретение умений и навыков. Данные учебные занятия углубляют, расширяют, детализируют полученные ранее знания. Практическое занятие предполагает выполнение обучающимися по заданию и под руководством преподавателей одной или нескольких практических работ.

Цель практических занятий и семинаров состоит в развитии познавательных способностей, самостоятельности мышления и творческой активности обучающихся; углублении, расширении, детализировании знаний, полученных на лекции в обобщенной форме, и содействии выработке навыков профессиональной деятельности. В отдельных случаях на практических занятиях и семинарах руководителем занятия сообщаются дополнительные знания.

Для достижения поставленных целей и решения требуемого перечня задач практические занятия и семинары проводятся традиционными технологиями или с использованием активных и интерактивных образовательных технологий.

Возможные формы проведения практических (семинарских) занятий:

- Деловая игра - это метод группового обучения совместной деятельности в процессе решения общих задач в условиях максимально возможного приближения к реальным проблемным ситуациям. Имитационные игры - на занятиях имитируется деятельность какой-либо организации, предприятия или его подразделения. Имитироваться могут события, конкретная деятельность людей (деловое совещание, обсуждение плана) и обстановка, условия, в которых происходит событие или осуществляется деятельность (кабинет

начальника цеха, зал заседаний). Исполнение ролей (ролевые игры) - в этих играх отрабатывается тактика поведения, действий, выполнение функций и обязанностей конкретного лица. Для проведения игр с исполнением роли разрабатывается модель-пьеса ситуации, между студентами распределяются роли с «обязательным содержанием», характеризующиеся различными интересами; в процессе их взаимодействия должно быть найдено компромиссное решение. «Деловой театр» (метод инсценировки) - в нем разыгрывается какая-либо ситуация, поведение человека в этой обстановке, обучающийся должен вжиться в образ определенного лица, понять его действия, оценить обстановку и найти правильную линию поведения. Основная задача метода инсценировки - научить ориентироваться в различных обстоятельствах, давать объективную оценку своему поведению, учитывать возможности других людей, влиять на их интересы, потребности и деятельность, не прибегая к формальным атрибутам власти, к приказу.

- Игровое проектирование - является практическим занятием или циклом занятий, суть которых состоит в разработке инженерного, конструкторского, технологического и других видов проектов в игровых условиях, максимально воссоздающих реальность. Этот метод отличается высокой степенью сочетания индивидуальной и совместной работы обучающихся.

- Познавательно-дидактические игры не относятся к деловым играм. Они предполагают лишь включение изучаемого материала в необычный игровой контекст и иногда содержат лишь элементы ролевых игр. Такие игры могут проводиться в виде копирования научных, культурных, социальных явлений (конкурс знатоков, «Поле чудес», КВН и т.д.) и в виде предметно-содержательных моделей, (например, игры-путешествия, когда надо разработать рациональный маршрут, пользуясь различными картами).

- Анализ конкретных ситуаций. Конкретная ситуация – это любое событие, которое содержит в себе противоречие или вступает в противоречие с окружающей средой. Ситуации могут нести в себе как позитивный, так и отрицательный опыт. Все ситуации делятся на простые, критические и экстремальные.

- Кейс-метод (от английского case – случай, ситуация) – усовершенствованный метод анализа конкретных ситуаций, метод активного проблемно-ситуационного анализа, основанный на обучении путем решения конкретных задач – ситуаций (решение кейсов). Непосредственная цель метода case-study - обучающиеся должны проанализировать ситуацию, разобраться в сути проблем, предложить возможные решения и выбрать лучшее из них. Кейсы делятся на практические (отражающие реальные жизненные ситуации), обучающие (искусственно созданные, содержащие значительные элементы условности при отражении в нем жизни) и исследовательские (ориентированные на проведение исследовательской деятельности посредством применения метода моделирования). Метод конкретных ситуаций (метод case-study) относится к неигровым имитационным активным методам обучения.

- Тренинг (англ. training от train — обучать, воспитывать) – метод активного обучения, направленный на развитие знаний, умений и навыков и социальных установок. Тренинг – форма интерактивного обучения, целью которого является развитие компетентности межличностного и профессионального поведения в общении. Достоинство тренинга заключается в том, что он обеспечивает активное вовлечение всех участников в процесс обучения. Можно выделить основные типы тренингов по критерию направленности воздействия и изменений – навыковый, психотерапевтический, социально-психологический, бизнес-тренинг.

- Метод Сократа (Майевтика) – метод вопросов, предполагающих критическое отношение к догматическим утверждениям, называется еще как метод «сократовской иронии». Это умение извлекать скрытое в человеке знание с помощью искусных наводящих вопросов, подразумевающего короткий, простой и заранее предсказуемый ответ.

- Интерактивная лекция – выступление ведущего обучающего перед большой аудиторией с применением следующих активных форм обучения: дискуссия, беседа, демонстрация слайдов или учебных фильмов, мозговой штурм.

- Групповая, научная дискуссия, диспут. Дискуссия — это целенаправленное обсуждение конкретного вопроса, сопровождающееся обменом мнениями, идеями между двумя и более лицами. Задача дискуссии - обнаружить различия в понимании вопроса и в споре установить истину. Дискуссии могут быть свободными и управляемыми. К технике управляемой дискуссии относятся: четкое определение цели, прогнозирование реакции оппонентов, планирование своего поведения, ограничение времени на выступления и их заданная очередность. Разновидностью свободной дискуссии является форум, где каждому желающему дается неограниченное время на выступление, при условии, что его выступление вызывает интерес аудитории. Каждый конкретный форум имеет свою тематику — достаточно широкую, чтобы в её пределах можно было вести многоплановое обсуждение.

- Дебаты — это чётко структурированный и специально организованный публичный обмен мыслями между двумя сторонами по актуальным темам. Это разновидность публичной дискуссии участников дебатов, направляющая на переубеждение в своей правоте третьей стороны, а не друг друга. Поэтому вербальные и невербальные средства, которые используются участниками дебатов, имеют целью получения определённого результата — сформировать у слушателей положительное впечатление от собственной позиции.

- Метод работы в малых группах. Групповое обсуждение кого-либо вопроса направлено на достижение лучшего взаимопонимания и нахождения истины. Групповое обсуждение способствует лучшему усвоению изучаемого материала. Оптимальное количество участников - 5-7 человек. Перед обучающимися ставится проблема, выделяется определенное время, в течение которого они должны подготовить аргументированный обдуманный ответ. Педагогический работник может устанавливать правила проведения группового обсуждения — задавать определенные рамки обсуждения, ввести алгоритм выработки общего мнения, назначить лидера и др.

- Круглый стол - общество, собрание в рамках более крупного мероприятия (съезда, симпозиума, конференции). Мероприятие, как правило, на которое приглашаются эксперты и специалисты из разных сфер деятельности для обсуждения актуальных вопросов. Данная модель обсуждения, основываясь на соглашениях, в качестве итогов даёт результаты, которые, в свою очередь, являются новыми соглашениями.

- Коллоквиум - (лат. colloquium — разговор, беседа) - одна из форм учебных занятий в системе образования, имеющая целью выяснение и повышение знаний обучающихся. На коллоквиумах обсуждаются: отдельные части, разделы, темы, вопросы изучаемого курса (обычно не включаемые в тематику семинарских и других практических учебных занятий), рефераты, проекты и др. работы обучающихся. Это научные собрания, на которых заслушиваются и обсуждаются доклады. Коллоквиум — это и форма контроля, массового опроса, позволяющая преподавателю в сравнительно небольшой срок выяснить уровень знаний студентов по данной теме дисциплины. Коллоквиум проходит обычно в форме дискуссии, в ходе которой обучающимся предоставляется возможность высказать свою точку зрения на рассматриваемую проблему, учиться обосновывать и защищать ее. Аргументируя и отстаивая свое мнение, обучающийся в то же время демонстрирует, насколько глубоко и осознанно он усвоил изученный материал.

- Метод «мозговой штурм» (мозговой штурм, мозговая атака, англ. brainstorming) — оперативный метод решения проблемы на основе стимулирования творческой активности, при котором участникам обсуждения предлагают высказывать как можно большее количество вариантов решения, в том числе самых фантастических. Затем из общего числа высказанных идей отбирают наиболее удачные, которые могут быть использованы на практике. Является методом экспертного оценивания.

- Метод проектов - это способ достижения дидактической цели через детальную разработку проблемы (технологию), которая должна завершиться вполне реальным, осязаемым практическим результатом, оформленным тем или иным образом; это совокупность приёмов, действий обучающихся в их определённой последовательности для достижения поставленной задачи — решения проблемы, лично значимой для учащихся и оформленной в виде некоего конечного продукта. Основное предназначение метода проектов

состоит в предоставлении учащимся возможности самостоятельного приобретения знаний в процессе решения практических задач или проблем, требующего интеграции знаний из различных предметных областей.

- Брифинг - (англ. briefing от англ. brief – короткий, недолгий) – краткая пресс-конференция, посвященная одному вопросу. Основное отличие: отсутствует презентационная часть. То есть практически сразу идут ответы на вопросы журналистов.

- Метод портфолио (итал. portfolio — 'портфель, англ. - папка для документов) - современная образовательная технология, в основе которой используется метод аутентичного оценивания результатов образовательной и профессиональной деятельности. Портфолио как подборка сертифицированных достижений, наиболее значимых работ и отзывов на них.

Вопросы для самоподготовки к практическим (семинарским) занятиям по разделам (темам) дисциплины (модуля)

РАЗДЕЛ 1. Теория баз данных

Тема 1.1. Введение в базы данных. Общая характеристика основных понятий

Вопросы для самоподготовки:

1. Назначение БД
2. Виды связей между таблицами
3. Технология ввода и редактирования данных.
4. Сортировка данных
5. Поиск и замена данных.
6. Технология применения Автофильтра
7. Технология применения Расширенного фильтра.
8. Технология создания запроса на выборку
9. Групповые операции в запросах
10. Технология создания перекрестного запроса

Тема 1.2. Системы управления базами данных

Вопросы для самоподготовки:

1. Назначение БД
2. Виды связей между таблицами
3. Технология ввода и редактирования данных.
4. Сортировка данных
5. Поиск и замена данных.
6. Технология применения Автофильтра
7. Технология применения Расширенного фильтра.
8. Технология создания запроса на выборку
9. Групповые операции в запросах
10. Технология создания перекрестного запроса

РАЗДЕЛ 2. Общие принципы построения (архитектура) баз данных

Тема 2.1. Различные архитектурные решения, используемые при реализации многопользовательских СУБД. Краткий обзор СУБД

Вопросы для самоподготовки:

1. Типы данных в БД
2. Понятие ключевого поля.
3. Создание схемы данных
4. Средства контроля ввода данных
5. Средства автоматизации ввода данных.
6. Создание списков.
7. Виды стандартных автоформ.
8. Создание подчиненных форм.
9. Технология создания запроса на добавление.
10. Технология создания запроса на удаление данных

Тема 2.2. Краткий обзор СУБД

Вопросы для самоподготовки:

1. Типы данных в БД
2. Понятие ключевого поля.
3. Создание схемы данных
4. Средства контроля ввода данных
5. Средства автоматизации ввода данных.
6. Создание списков.
7. Виды стандартных автоформ.
8. Создание подчиненных форм.
9. Технология создания запроса на добавление.
10. Технология создания запроса на удаление данных

РАЗДЕЛ 3. Модели данных

Тема 3.1. Различные представления о данных в базах данных. Основные этапы проектирования баз данных

Вопросы для самоподготовки:

1. Технология построения запроса на создание таблиц.
2. Технология создания запроса на обновление данных.
3. Виды соединения таблиц в запросах
4. Свойства запроса.
5. Запросы с параметром.
6. Построитель выражений.
7. Страницы доступа данных
8. Виды стандартных отчетов
9. Группировка в отчетах
10. Макросы

Тема 3.2. Первая стадия концептуального проектирования базы данных (концептуальное моделирование. Вторая стадия концептуального проектирования (Модели данных СУБД. Представление концептуальной модели средствами модели данных СУБД)

Вопросы для самоподготовки:

1. Технология построения запроса на создание таблиц.
2. Технология создания запроса на обновление данных.
3. Виды соединения таблиц в запросах
4. Свойства запроса.
5. Запросы с параметром.
6. Построитель выражений.
7. Страницы доступа данных

8. Виды стандартных отчетов
9. Группировка в отчетах
10. Макросы

РАЗДЕЛ 4. Базисные операции с реляционными данными

Тема 4.1. Формализация реляционной модели

Вопросы для самоподготовки:

1. Формат команды на выборку SELECT.
2. Основные опции команды SELECT.
3. Формат команды редактирования данных INSERT
4. Форматы команды редактирования данных UPDATE.
5. Форматы команды редактирования данных DELETE
6. Формат команды создания таблиц SELECT INTO.
7. Формат команды создания таблиц CREATE TABLE.
8. Опции соединения таблиц в запросах.
9. Формат команды объединения данных UNION
10. Формат команды перекрестного запроса TRANSFORM

РАЗДЕЛ 5. Нормальные формы в реляционных базах данных

Тема 5.1. Использование формального аппарата для оптимизации схем отношений

Вопросы для самоподготовки:

1. Информация и данные, база данных, система управления базами данных (СУБД).
2. Эволюция концепции обработки данных, СУБД.
3. Требования к СУБД, основные особенности СУБД, составные части СУБД.
4. Системы быстрой разработки приложений. Модели данных.
5. Реляционная БД, история появления, принципы организации данных, достоинства и недостатки.
6. Базовые понятия реляционных БД: тип данных, домен, атрибут, кортеж, отношение, схема отношений.
7. Проектирование баз данных.
8. Нормализация БД, цели нормализации, 1НФ.
9. Нормализация БД, определение 1НФ, 2НФ, 3НФ.
10. Разработка приложений в среде MS Windows

РАЗДЕЛ 6. Проектирование баз данных с использованием семантического подхода

Тема 6.1. Физические модели данных (внутренний уровень)

Вопросы для самоподготовки:

1. Архитектура Microsoft Access.
2. Назначение объектов MS Access
3. Построение таблиц в MS Access.
4. Формы ввода-вывода данных.
5. Основные операции реляционной алгебры.
6. Дополнительные операции реляционной алгебры.
7. Запросы в MS Access.
8. Параметры запросов на выборку данных.
9. Перекрестные запросы.

10. Многотабличные запросы и схема данных.

РАЗДЕЛ 7. Проектирование баз данных

Тема 7.1. Программное обеспечение работы с современными базами данных

Вопросы для самоподготовки:

1. Понятие технологии "клиент-сервер".
2. Общие сведения о языке запросов SQL.
3. Сетевые БД, архитектура «файл-сервер», «клиент-сервер».
4. Язык SQL: общие сведения о языке, роль и место в современных СУБД, стандарт ANSI.
5. Запрос выборки данных в SQL, простейшая выборка из одной таблицы.
6. Специальные операторы SQL IN, BETWEEN, LIKE, IS NULL.
7. Соединение таблиц с использованием операции JOIN.
8. SQL: запрос выборки данных, функции агрегирования AVG, SUM, MAX, MIN.
9. Форматирование выходных данных запроса, секции GROUP BY и HAVING.
10. Соединение таблиц.

1.3. Учебно-наглядные пособия по разделам (темам) дисциплины (модуля)¹

РАЗДЕЛ 1. Теория баз данных

Тема 1.1. Введение в базы данных. Общая характеристика основных понятий

1. Гутгарц, Р. Д. Проектирование автоматизированных систем обработки информации и управления : учебное пособие для вузов / Р. Д. Гутгарц. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 351 с. — (Высшее образование). — ISBN 978-5-534-15761-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509638> (дата обращения: 07.03.2023).
2. Грекул, В. И. Проектирование информационных систем : учебник и практикум для вузов / В. И. Грекул, Н. Л. Коровкина, Г. А. Левочкина. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-9916-8764-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511889> (дата обращения: 07.03.2023).

Тема 1.2. Системы управления базами данных

1. Гутгарц, Р. Д. Проектирование автоматизированных систем обработки информации и управления : учебное пособие для вузов / Р. Д. Гутгарц. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 351 с. — (Высшее образование). — ISBN 978-5-534-15761-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509638> (дата обращения: 07.03.2023).
2. Грекул, В. И. Проектирование информационных систем : учебник и практикум для вузов / В. И. Грекул, Н. Л. Коровкина, Г. А. Левочкина. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-9916-8764-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511889> (дата обращения: 07.03.2023).

РАЗДЕЛ 2. Общие принципы построения (архитектура) баз данных

Тема 2.1. Различные архитектурные решения, используемые при реализации многопользовательских СУБД. Краткий обзор СУБД

1. Гутгарц, Р. Д. Проектирование автоматизированных систем обработки информации и управления : учебное пособие для вузов / Р. Д. Гутгарц. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 351 с. — (Высшее образование). — ISBN 978-5-534-15761-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509638> (дата обращения: 07.03.2023).
2. Грекул, В. И. Проектирование информационных систем : учебник и практикум для вузов / В. И. Грекул, Н. Л. Коровкина, Г. А. Левочкина. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-9916-8764-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511889> (дата обращения: 07.03.2023).

Тема 2.2. Краткий обзор СУБД

1. Гутгарц, Р. Д. Проектирование автоматизированных систем обработки информации и управления : учебное пособие для вузов / Р. Д. Гутгарц. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 351 с. — (Высшее образование). — ISBN 978-5-534-15761-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509638> (дата обращения: 07.03.2023).
2. Грекул, В. И. Проектирование информационных систем : учебник и практикум для вузов / В. И. Грекул, Н. Л. Коровкина, Г. А. Левочкина. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-9916-8764-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511889> (дата обращения: 07.03.2023).

РАЗДЕЛ 3. Модели данных

Тема 3.1. Различные представления о данных в базах данных. Основные этапы проектирования баз данных

1. Гутгарц, Р. Д. Проектирование автоматизированных систем обработки информации и управления : учебное пособие для вузов / Р. Д. Гутгарц. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 351 с. — (Высшее образование). — ISBN 978-5-534-15761-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509638> (дата обращения: 07.03.2023).
2. Грекул, В. И. Проектирование информационных систем : учебник и практикум для вузов / В. И. Грекул, Н. Л. Коровкина, Г. А. Левочкина. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-9916-8764-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511889> (дата обращения: 07.03.2023).

Тема 3.2. Первая стадия концептуального проектирования базы данных (концептуальное моделирование. Вторая стадия концептуального проектирования (Модели данных СУБД. Представление концептуальной модели средствами модели данных СУБД)

1. Гутгарц, Р. Д. Проектирование автоматизированных систем обработки информации и управления : учебное пособие для вузов / Р. Д. Гутгарц. — 2-е изд., перераб. и доп. —

Москва : Издательство Юрайт, 2023. — 351 с. — (Высшее образование). — ISBN 978-5-534-15761-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509638> (дата обращения: 07.03.2023).

2. Грекул, В. И. Проектирование информационных систем : учебник и практикум для вузов / В. И. Грекул, Н. Л. Коровкина, Г. А. Левочкина. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-9916-8764-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511889> (дата обращения: 07.03.2023).

РАЗДЕЛ 4. Базисные операции с реляционными данными

Тема 4.1. Формализация реляционной модели

1. Гутгарц, Р. Д. Проектирование автоматизированных систем обработки информации и управления : учебное пособие для вузов / Р. Д. Гутгарц. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 351 с. — (Высшее образование). — ISBN 978-5-534-15761-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509638> (дата обращения: 07.03.2023).
2. Грекул, В. И. Проектирование информационных систем : учебник и практикум для вузов / В. И. Грекул, Н. Л. Коровкина, Г. А. Левочкина. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-9916-8764-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511889> (дата обращения: 07.03.2023).

РАЗДЕЛ 5. Нормальные формы в реляционных базах данных

Тема 5.1. Использование формального аппарата для оптимизации схем отношений

1. Гутгарц, Р. Д. Проектирование автоматизированных систем обработки информации и управления : учебное пособие для вузов / Р. Д. Гутгарц. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 351 с. — (Высшее образование). — ISBN 978-5-534-15761-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509638> (дата обращения: 07.03.2023).
2. Грекул, В. И. Проектирование информационных систем : учебник и практикум для вузов / В. И. Грекул, Н. Л. Коровкина, Г. А. Левочкина. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-9916-8764-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511889> (дата обращения: 07.03.2023).

РАЗДЕЛ 6. Проектирование баз данных с использованием семантического подхода

Тема 6.1. Физические модели данных (внутренний уровень)

1. Гутгарц, Р. Д. Проектирование автоматизированных систем обработки информации и управления : учебное пособие для вузов / Р. Д. Гутгарц. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 351 с. — (Высшее образование). — ISBN 978-5-534-15761-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509638> (дата обращения: 07.03.2023).
2. Грекул, В. И. Проектирование информационных систем : учебник и практикум для вузов / В. И. Грекул, Н. Л. Коровкина, Г. А. Левочкина. — Москва : Издательство

Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-9916-8764-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511889> (дата обращения: 07.03.2023).

РАЗДЕЛ 7. Проектирование баз данных

Тема 7.1. Программное обеспечение работы с современными базами данных

1. Гутгарц, Р. Д. Проектирование автоматизированных систем обработки информации и управления : учебное пособие для вузов / Р. Д. Гутгарц. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 351 с. — (Высшее образование). — ISBN 978-5-534-15761-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509638> (дата обращения: 07.03.2023).
2. Грекул, В. И. Проектирование информационных систем : учебник и практикум для вузов / В. И. Грекул, Н. Л. Коровкина, Г. А. Левочкина. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-9916-8764-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511889> (дата обращения: 07.03.2023).

РАЗДЕЛ 7. Проектирование баз данных

Тема 7.1. Программное обеспечение работы с современными базами данных

3. Гутгарц, Р. Д. Проектирование автоматизированных систем обработки информации и управления : учебное пособие для вузов / Р. Д. Гутгарц. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 351 с. — (Высшее образование). — ISBN 978-5-534-15761-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509638> (дата обращения: 07.03.2023).
4. Грекул, В. И. Проектирование информационных систем : учебник и практикум для вузов / В. И. Грекул, Н. Л. Коровкина, Г. А. Левочкина. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-9916-8764-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511889> (дата обращения: 07.03.2023).

Тема 7.2. Основные операторы языка SQL. Интерактивный SQL. Использование языка SQL в прикладных программах

1. Гутгарц, Р. Д. Проектирование автоматизированных систем обработки информации и управления : учебное пособие для вузов / Р. Д. Гутгарц. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 351 с. — (Высшее образование). — ISBN 978-5-534-15761-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509638> (дата обращения: 07.03.2023).
2. Грекул, В. И. Проектирование информационных систем : учебник и практикум для вузов / В. И. Грекул, Н. Л. Коровкина, Г. А. Левочкина. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-9916-8764-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511889> (дата обращения: 07.03.2023).

РАЗДЕЛ 8. Физическое проектирование БД

Тема 8.1. Направления развития баз данных

1. Гутгарц, Р. Д. Проектирование автоматизированных систем обработки информации и управления : учебное пособие для вузов / Р. Д. Гутгарц. — 2-е изд., перераб. и доп. —

Москва : Издательство Юрайт, 2023. — 351 с. — (Высшее образование). — ISBN 978-5-534-15761-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509638> (дата обращения: 07.03.2023).

2. Грекул, В. И. Проектирование информационных систем : учебник и практикум для вузов / В. И. Грекул, Н. Л. Коровкина, Г. А. Левочкина. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-9916-8764-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511889> (дата обращения: 07.03.2023).

1.4. Методические материалы по подготовке к лабораторным занятиям по дисциплине (модулю)

Лабораторные занятия - одна из форм учебного занятия, направленная на развитие самостоятельности обучающихся и приобретение умений и навыков. Данные учебные занятия углубляют, расширяют, детализируют полученные ранее знания. Лабораторное занятие предполагает выполнение обучающимися по заданию и под руководством преподавателей одной или нескольких работ.

Цель лабораторных занятий состоит в развитии познавательных способностей, самостоятельности мышления и творческой активности обучающихся; углублении, расширении, детализировании знаний, полученных на лекции в обобщенной форме, и содействии выработке навыков профессиональной деятельности. В отдельных случаях на лабораторных занятиях руководителем занятия сообщаются дополнительные знания.

Для достижения поставленных целей и решения требуемого перечня задач лабораторные занятия проводятся традиционными технологиями или с использованием активных и интерактивных образовательных технологий.

При подготовке и работе во время проведения лабораторных занятий следует обратить внимание на следующие моменты: на процесс предварительной подготовки, на работу во время занятия, обработку полученных результатов, исправление полученных замечаний.

Предварительная подготовка к лабораторному занятию заключается в изучении теоретического материала в отведенное для самостоятельной работы время, ознакомление с инструктивными материалами с целью осознания задач занятия, техники безопасности при работе с приборами, веществами.

Работа во время проведения занятия включает:

- консультирование студентов преподавателями и вспомогательным персоналом с целью предоставления исчерпывающей информации, необходимой для самостоятельного выполнения предложенных преподавателем задач.

- самостоятельное выполнение заданий согласно обозначенной рабочей программой дисциплины (модуля) тематики.

2. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ) И САМОСТОЯТЕЛЬНОЙ РАБОТЕ

Освоение обучающимся дисциплины (модуля) «*Наименование дисциплины (модуля)*» предполагает изучение материалов дисциплины на аудиторных занятиях и в ходе самостоятельной работы. Аудиторные занятия проходят в форме лекций, семинаров, практических и лабораторных занятий. Самостоятельная работа включает разнообразный комплекс видов и форм работы обучающихся.

Для успешного освоения дисциплины (модуля) и достижения поставленных целей необходимо внимательно ознакомиться с рабочей программы дисциплины (модуля), доступной в электронной информационно-образовательной среде РГСУ.

Следует обратить внимание на списки основной и дополнительной литературы, на предлагаемые преподавателем ресурсы информационно-телекоммуникационной сети Интернет. Эта информация необходима для самостоятельной работы обучающегося.

При подготовке к аудиторным занятиям необходимо помнить особенности каждой формы его проведения.

Подготовка к учебному занятию лекционного типа заключается в следующем.

С целью обеспечения успешного обучения обучающийся должен готовиться к лекции, поскольку она является важнейшей формой организации учебного процесса, поскольку:

- знакомит с новым учебным материалом;
- разъясняет учебные элементы, трудные для понимания;
- систематизирует учебный материал;
- ориентирует в учебном процессе.

С этой целью:

- внимательно прочитайте материал предыдущей лекции;
- ознакомьтесь с учебным материалом по учебнику и учебным пособиям с темой прочитанной лекции;
- внесите дополнения к полученным ранее знаниям по теме лекции на полях лекционной тетради;
- запишите возможные вопросы, которые вы зададите лектору на лекции по материалу изученной лекции;
- постарайтесь уяснить место изучаемой темы в своей подготовке;
- узнайте тему предстоящей лекции (по тематическому плану, по информации лектора) и запишите информацию, которой вы владеете по данному вопросу.

Подготовка к занятию семинарского типа.

При подготовке и работе во время проведения занятий семинарского типа следует обратить внимание на следующие моменты: на процесс предварительной подготовки, на работу во время занятия, обработку полученных результатов, исправление полученных замечаний.

Предварительная подготовка к учебному занятию семинарского типа заключается в изучении теоретического материала в отведенное для самостоятельной работы время, ознакомление с инструктивными материалами с целью осознания задач практического занятия, техники безопасности при работе с приборами, веществами.

Работа во время проведения учебного занятия семинарского типа включает:

- консультирование студентов преподавателями и вспомогательным персоналом с целью предоставления исчерпывающей информации, необходимой для самостоятельного выполнения предложенных преподавателем задач.
- самостоятельное выполнение заданий согласно обозначенной рабочей программой дисциплины (модуля) тематики.

Самостоятельная работа.

Самостоятельная работа - планируемая учебная, учебно-исследовательская, научно-исследовательская работа студентов, выполняемая во внеаудиторное (аудиторное) время по заданию и при методическом руководстве преподавателя, но без его непосредственного участия (при частичном непосредственном участии преподавателя, оставляющем ведущую роль за работой студентов).

Для более углубленного изучения темы задания для самостоятельной работы рекомендуется выполнять параллельно с изучением данной темы. При выполнении заданий по возможности используйте наглядное представление материала.

Самостоятельная работа студентов в ВУЗе является важным видом учебной и научной деятельности студента. Самостоятельная работа студентов играет значительную роль в рейтинговой технологии обучения.

К современному специалисту в области медицины общество предъявляет достаточно широкий перечень требований, среди которых немаловажное значение имеет наличие у выпускников определенных навыков (компетенций) и умения самостоятельно добывать знания из различных источников, систематизировать полученную информацию, давать оценку конкретной профессиональной ситуации. Формирование такого умения происходит в течение всего периода обучения.

В процессе самостоятельной работы студент приобретает необходимые для будущей специальности компетенции, навыки самоорганизации, самоконтроля, самоуправления, саморефлексии и становится активным самостоятельным субъектом учебной деятельности.

Самостоятельная работа студентов должна оказывать важное влияние на формирование личности будущего специалиста, его компетентность. Каждый студент самостоятельно определяет режим своей работы и меру труда, затрачиваемого на овладение учебным содержанием по каждой дисциплине. Он выполняет внеаудиторную работу по личному индивидуальному плану, в зависимости от его подготовки, времени и других условий.

Виды самостоятельной работы.

Работа с литературой.

При работе с книгой необходимо подобрать литературу, научиться правильно ее читать, вести записи. Для подбора литературы в библиотеке используются алфавитный и систематический каталоги. Важно помнить, что рациональные навыки работы с книгой - это всегда большая экономия времени и сил. Правильный подбор учебников рекомендуется преподавателем, читающим лекционный курс. Необходимая литература может быть также указана в методических разработках по данному курсу. Изучая материал по учебнику, следует переходить к следующему вопросу только после правильного уяснения предыдущего, описывая на бумаге все выкладки и вычисления (в том числе те, которые в учебнике опущены или на лекции даны для самостоятельного вывода). При изучении любой дисциплины большую и важную роль играет самостоятельная индивидуальная работа. Особое внимание следует обратить на определение основных понятий курса. Студент должен подробно разбирать примеры, которые поясняют такие определения, и уметь строить аналогичные примеры самостоятельно. Нужно добиваться точного представления о том, что изучаешь. Полезно составлять опорные конспекты. При изучении материала по учебнику полезно в тетради (на специально отведенных полях) дополнять конспект лекций. Там же следует отмечать вопросы, выделенные студентом для консультации с преподавателем. Выводы, полученные в результате изучения, рекомендуется в конспекте выделять, чтобы они при перечитывании записей лучше запоминались. Опыт показывает, что многим студентам помогает составление листа опорных сигналов, содержащего важнейшие и наиболее часто употребляемые формулы и понятия. Такой лист помогает запомнить формулы, основные положения лекции, а также может служить постоянным справочником для студента. Различают два вида чтения: первичное и вторичное. Первичное - это внимательное, неторопливое чтение, при котором можно остановиться на трудных местах.

Задача вторичного чтения - полное усвоение смысла целого (по счету это чтение может быть и не вторым, а третьим или четвертым). Самостоятельная работа с учебниками и книгами (а также самостоятельное теоретическое исследование проблем, обозначенных преподавателем на лекциях) – это важнейшее условие формирования у себя научного способа познания.

Чтение научного текста является частью познавательной деятельности. Ее цель – извлечение из текста необходимой информации. От того на сколько осознанна читающим собственная внутренняя установка при обращении к печатному слову (найти нужные сведения, усвоить информацию полностью или частично, критически проанализировать материал и т.п.) во многом зависит эффективность осуществляемого действия.

Методические рекомендации по составлению конспекта:

1. Внимательно прочитайте текст. Уточните в справочной литературе непонятные слова. При записи не забудьте вынести справочные данные на поля конспекта;
2. Выделите главное, составьте план;
3. Кратко сформулируйте основные положения текста, отметьте аргументацию автора;
4. Законспектируйте материал, четко следуя пунктам плана. При конспектировании старайтесь выразить мысль своими словами. Записи следует вести четко, ясно.

5. Грамотно записывайте цитаты. Цитируя, учитывайте лаконичность, значимость мысли. В тексте конспекта желательно приводить не только тезисные положения, но и их доказательства. При оформлении конспекта необходимо стремиться к емкости каждого

предложения. Мысли автора книги следует излагать кратко, заботясь о стиле и выразительности написанного. Число дополнительных элементов конспекта должно быть логически обоснованным, записи должны распределяться в определенной последовательности, отвечающей логической структуре произведения. Для уточнения и дополнения необходимо оставлять поля. Овладение навыками конспектирования требует от студента целеустремленности, повседневной самостоятельной работы.

Методические материалы по самостоятельному решению задач

При самостоятельном решении задач нужно обосновывать каждый этап решения, исходя из теоретических положений курса. Если студент видит несколько путей решения проблемы (задачи), то нужно сравнить их и выбрать самый рациональный. Полезно до начала вычислений составить краткий план решения проблемы (задачи). Решение проблемных задач или примеров следует излагать подробно, вычисления располагать в строгом порядке, отделяя вспомогательные вычисления от основных. Решения при необходимости нужно сопровождать комментариями, схемами, чертежами и рисунками. Следует помнить, что решение каждой учебной задачи должно доводиться до окончательного логического ответа, которого требует условие, и по возможности с выводом. Полученный ответ следует проверить способами, вытекающими из существа данной задачи. Полезно также (если возможно) решать несколькими способами и сравнить полученные результаты. Решение задач данного типа нужно продолжать до приобретения твердых навыков в их решении.

Методические материалы к выполнению реферата

Реферат (от лат. *refere* – сообщать) – краткое изложение в письменном виде или в форме публикации доклада, содержания научного труда (трудов), литературы по теме. Работа над рефератом условно разделяется на выбор темы, подбор литературы, подготовку и защиту плана; написание теоретической части и всего текста с указанием библиографических данных используемых источников, подготовку доклада, выступление с ним. Тематика рефератов полностью связана с основными вопросами изучаемого курса.

Список литературы к темам не дается, и обучающиеся самостоятельно ведут библиографический поиск, причем им не рекомендуется ограничиваться университетской библиотекой.

Важно учитывать, что написание реферата требует от обучающихся определенных усилий. Особое внимание следует уделить подбору литературы, методике ее изучения с целью отбора и обработки собранного материала, обоснованию актуальности темы и теоретического уровня обоснованности используемых в качестве примеров фактов какой-либо деятельности.

Выбрав тему реферата, начав работу над литературой, необходимо составить план. Изучая литературу, продолжается обдумывание темы, осмысливание прочитанного, делаются выписки, сопоставляются точки зрения разных авторов и т.д. Реферативная работа сводится к тому, чтобы в ней выделились две взаимосвязанные стороны: во-первых, ее следует рассматривать как учебное задание, которое должен выполнить обучаемый, а во-вторых, как форму научной работы, творческого воображения при выполнении учебного задания.

Наличие плана реферата позволяет контролировать ход работы, избежать формального переписывания текстов из первоисточников.

Оформление реферата включает титульный лист, оглавление и краткий список использованной литературы. Список использованной литературы размещается на последней странице рукописи или печатной форме реферата. Реферат выполняется в письменной или печатной форме на белых листах формата А4 (210 x 297 мм). Шрифт Times New Roman, кегель 14, через 1,5 интервала при соблюдении следующих размеров текста: верхнее поле – 25 мм, нижнее – 20 мм, левое – 30 мм, правое – 15 мм. Нумерация страниц производится вверху листа, по центру. Титульный лист нумерации не подлежит.

Рефераты должны быть написаны простым, ясным языком, без претензий на наукообразность. Следует избегать сложных грамматических оборотов, непривычных терминов и символов. Если же такие термины и символы все-таки приводятся, то необходимо разъяснять их значение при первом упоминании в тексте реферата.

Объем реферата предполагает тщательный отбор информации, необходимой для краткого изложения вопроса. Важнейший этап – редактирование готового текста реферата и подготовка к обсуждению. Обсуждение требует хорошей ориентации в материале темы, умения выделить главное, поставить дискуссионный вопрос, привлечь внимание слушателей к интересной литературе, логично и убедительно изложить свои мысли.

Рефераты обязательно подлежат защите. Процедура защиты начинается с определения оппонентов защищающего свою работу. Они стремятся дать основательный анализ работы обучающимся, обращают внимание на положительные моменты и недостатки реферата, дают общую оценку содержанию, форме преподнесения материала, характеру использованной литературы. Иногда они дополняют тот или иной раздел реферата. Последнее особенно ценно, ибо говорит о глубоком знании обучающимся-оппонентом изучаемой проблемы.

Обсуждение не ограничивается заслушиванием оппонентов. Другие обучающиеся имеют право уточнить или опровергнуть какое-либо утверждение. Преподаватель предлагает любому обучающемуся задать вопрос по существу доклада или попытаться подвести итог обсуждению.

Алгоритм работы над рефератом

1. Выбор темы

Тема должна быть сформулирована грамотно (с литературной точки зрения);

В названии реферата следует поставить четкие рамки рассмотрения темы;

Желательно избегать слишком длинных названий;

Следует по возможности воздерживаться от использования в названии спорных с научной точки зрения терминов, излишней наукообразности, а также чрезмерного упрощения формулировок.

2. Реферат следует составлять из пяти основных частей: введения; основной части; заключения; списка литературы; приложений.

3. Основные требования к введению:

Во введении не следует концентрироваться на содержании; введение должно включать краткое обоснование актуальности темы реферата, где требуется показать, почему данный вопрос может представлять научный интерес и есть ли связь представляемого материала с современностью. Таким образом, тема реферата должна быть актуальна либо с научной точки зрения, либо с современных позиций.

Очень важно выделить цель, а также задачи, которые требуется решить для выполнения цели.

Введение должно содержать краткий обзор изученной литературы, в котором указывается взятый из того или иного источника материал, кратко анализируются изученные источники, показываются их сильные и слабые стороны;

Объем введения составляет две страницы текста.

4. Требования к основной части реферата:

Основная часть содержит материал, отобранный для рассмотрения проблемы;

Также основная часть должна включать в себя собственно мнение обучающихся и сформулированные самостоятельные выводы, опирающиеся на приведенные факты;

Материал, представленный в основной части, должен быть логически изложен и распределен по параграфам, имеющим свои названия;

В изложении основной части необходимо использовать сноски (в первую очередь, когда приводятся цифры и чьи-то цитаты);

Основная часть должна содержать иллюстративный материал (графики, таблицы и т. д.);

Объем основной части составляет около 10 страниц.

5. Требования к заключению:

В заключении формулируются выводы по параграфам, обращается внимание на выдвинутые во введении задачи и цели;

Заключение должно быть четким, кратким, вытекающим из содержания основной части.

6. Требования к оформлению списка литературы (по ГОСТу):

Необходимо соблюдать правильность последовательности записи источников: сначала следует писать фамилию, а после инициалы; название работы не ставится в кавычки; после названия сокращенно пишется место издания; затем идет год издания; наконец, называется процитированная страница.

Критерии оценки реферата

Обучающийся, защищающий реферат, должен рассказать о его актуальности, поставленных целях и задачах, изученной литературе, структуре основной части, сделанных в ходе работы выводах.

По окончании выступления ему может быть задано несколько вопросов по представленной проблеме.

Оценка складывается из соблюдения требований к реферату, грамотного раскрытия темы, умения четко рассказывать о представленном реферате, способности понять суть задаваемых по работе вопросов и найти точные ответы на них.

Методические материалы к выполнению эссе

Эссе – литературное произведение небольшого объема, обычно прозаическое, свободной композиции, передающее индивидуальные впечатления, суждения, соображения автора о той или иной проблеме, теме, о том или ином событии или явлении. Это вид самостоятельной исследовательской работы обучающихся, с целью углубления и закрепления теоретических знаний и освоения практических навыков. Цель эссе состоит в развитии самостоятельного творческого мышления и письменного изложения собственных мыслей. При написании эссе обучающийся должен представить развернутый письменный ответ на теоретический или практический актуальный вопрос, объявленный преподавателем в аудитории непосредственно перед ее написанием. В процессе написания эссе разрешается пользоваться нормативно-правовыми актами, конспектом лекций (в печатном виде). Использование интернет-ресурсов не допускается. Темы эссе преподаватель предлагает из числа тех, которые обучающиеся уже рассматривали на лекциях или семинарских занятиях, исходя из содержания заданий в составе оценочных средств. По решению преподавателя, в качестве темы эссе может быть выбрана одна или несколько тем, которые могут быть распределены между обучающимися по желанию.

Требования к выполнению эссе:

1. Проводится письменно.
2. Эссе выполняется на компьютере (гарнитура Times New Roman, шрифт 14) через 1,5 интервала с полями: верхнее, нижнее – 2; правое – 3; левое – 1,5. Отступ первой строки абзаца – 1,25. Сноски – постраничные. Таблицы и рисунки встраиваются в текст работы. При этом обязательный заголовок таблицы надо размещать над табличным полем, а рисунки сопровождать подрисуночными подписями. При включении в эссе нескольких таблиц и/или рисунков их нумерация обязательна. Обязательна и нумерация страниц. Их целесообразно проставлять внизу страницы – по середине или в правом углу. Номер страницы не ставится на титульном листе, но в общее число страниц он включается. Объем эссе, без учета приложений, не должен превышать 5 страниц. Значительное превышение установленного объема является недостатком работы и указывает на то, что обучающийся не сумел отобрать и переработать необходимый материал.

3. Работа должна содержать собственные умозаключения по сути поставленной проблемы, включать самостоятельно проведенный анализ по сути этой проблемы, выводы, обобщающие авторскую позицию по поставленной проблеме.

Критерии оценки эссе:

«Отлично» – исключительные знания материала, абсолютное понимание сути, безукоризненное знание основных понятий и положений, логически и лексически грамотно изложенный, содержательный, аргументированный, конкретный и исчерпывающий ответ.

«Хорошо» – глубокие знания материала, правильное понимание сути, знание основных понятий и положений, содержательный, полный и конкретный ответ.

«Удовлетворительно» – твердые, но недостаточно полные знания, верное понимание сути, в целом правильный ответ.

«Неудовлетворительно» – непонимание сущности задания, грубые ошибки в ответе.

Методические материалы по выполнению тестирования.

Тестовые задания содержат вопросы и 3-4 варианта ответа по базовым положениям изучаемой темы/раздела, составлены с расчетом на знания, полученные обучающимся в процессе изучения темы/раздела.

Тестовые задания выполняются в письменной форме и сдаются преподавателю, ведущему дисциплину (модуль). На выполнение тестовых заданий обучающимся отводится 45 минут.

При обработке результатов оценочной процедуры используются: критерии оценки по содержанию и качеству полученных ответов, ключи, оценочные листы.

Критерии оценки теста:

«Зачтено» - если обучающийся правильно выполнил не менее 2/3 всей работы или допустил не более одной грубой ошибки и двух недочетов, не более одной грубой и одной негрубой ошибки, не более трех негрубых ошибок, одной негрубой ошибки и трех недочетов, при наличии четырех-пяти недочетов.

«Не зачтено» - если число ошибок и недочетов превысило норму для оценки 3 или правильно выполнено менее 2/3 всей работы.

Методические материалы по выполнению доклада.

Рекомендуется следующая структура доклада:

1. титульный лист, содержание доклада;
2. краткое изложение;
3. цели и задачи;
4. изложение характера исследований и рассмотренных проблем, гипотезы, спорные вопросы;
5. источники информации, методы сбора и анализа данных, степень их полноты и достоверности;
6. анализ и толкование полученных в работе результатов;
7. выводы и оценки;
8. библиография и приложения.

Время выступления докладчика не должно превышать 10 минут.

Основные требования к оформлению доклада:

- титульный лист должен включать название доклада, наименование предметной (цикловой) комиссии, фамилию обучающегося;
- все использованные литературные источники сопровождаются библиографическим описанием;
- приводимая цитата из источника берется в кавычки (оформляются сноски);
- единицы измерения должны применяться в соответствии с действующими стандартами;
- все названия литературных источников следует приводить в соответствии с новейшими изданиями;
- рекомендуется включение таблиц, графиков, схем, если они отражают основное содержание или улучшают ее наглядность;
- названия фирм, учреждений, организаций и предприятий должны именоваться так, как они указываются в источнике;

Критерии оценки доклада

При выполнении доклада обучающийся должен продемонстрировать умение кратко излагать прочитанный материал, а также умение обобщать и анализировать материал по теме доклада.

Презентация

Методические материалы к презентациям

1. Объем презентации 10 -20 слайдов.

2. На титульном слайде должно быть отражено:
 - наименование факультета;
 - тема презентации;
 - фамилия, имя, отчество, направление подготовки/ специальность, направленность (профиль)/ специализация, форма обучения, номер группы автора презентации;
 - фамилия, имя, отчество, степень, звание, должность руководитель работы;
 - год выполнения работы.
3. В презентации должны быть отражены обоснование актуальности представляемого материала, цели и задачи работы.
4. Содержание презентации должно включать наиболее значимый материал доклада, а также, при необходимости, таблицы, диаграммы, рисунки, фотографии, карты, видео – вставки, звуковое сопровождение.
5. Заключительный слайд должен содержать информацию об источниках информации для презентации.

Критерии оценки презентации

1. Объём презентации 10 -20 слайдов.
2. Правильность оформления титульного слайда.
3. Актуальность отобранного материала, обоснованность формулировки цели и задач работы.
4. Наглядность и логичность презентации, обоснованность использования таблиц, диаграмм, рисунков, фотографий, карт, видео – вставок, звукового сопровождения; правильный выбор шрифтов, фона, других элементов дизайна слайда.
5. Объём и качество источников информации (не менее 2-х интернет – источников и не менее 2-х литературных источников).

Методические материалы по подготовке к опросу

Самостоятельная работа обучающихся включает подготовку к опросу на практических занятиях. Для этого обучающийся изучает лекции, основную и дополнительную литературу, публикации, информацию из Интернет-ресурсов.

Тема и вопросы к практическим занятиям, вопросы для самоконтроля содержатся в рабочей программе и доводятся до студентов заранее.

Для подготовки к опросу обучающемуся необходимо ознакомиться с материалом, посвященным теме практического занятия, в учебнике или другой рекомендованной литературе, конспекте лекции, обратить внимание на усвоение основных понятий дисциплины, выявить неясные вопросы и подобрать дополнительную литературу для их освещения.

Критерии оценки опроса

- «Отлично»:
- дан полный, развернутый ответ на поставленный вопрос;
 - в ответе прослеживается четкая структура, логическая последовательность, отражающая сущность раскрываемых понятий, теорий, явлений;
 - знание по предмету демонстрируется на фоне понимания его в системе данной науки и междисциплинарных связей;
 - свободное владение терминологией;
 - ответы на дополнительные вопросы четкие, краткие;
- «Хорошо»:
- дан полный, развернутый ответ на поставленный вопрос, показано умение выделять существенные и несущественные признаки, причинно-следственные связи;
 - ответ недостаточно логичен с единичными ошибками в частности, исправленные студентом с помощью преподавателя;
 - единичные ошибки в терминологии;
 - ответы на дополнительные вопросы правильные, недостаточно полные и четкие.
- «Удовлетворительно»:

– ответ не полный, с ошибками в деталях, умение раскрыть значение обобщённых знаний не показано, речевое оформление требует поправок, коррекции;

– логика и последовательность изложения имеют нарушения, студент не способен самостоятельно выделить существенные и несущественные признаки и причинно-следственные связи;

– ошибки в раскрываемых понятиях, терминах;

– студент не ориентируется в теме, допускает серьезные ошибки;

– студент не может ответить на большую часть дополнительных вопросов.

«Неудовлетворительно»:

– ответ представляет собой разрозненные знания с существенными ошибками по вопросу;

– присутствуют фрагментарность, нелогичность изложения, студент не осознает связь обсуждаемого вопроса с другими объектами дисциплины, речь неграмотная;

– незнание терминологии;

– ответы на дополнительные вопросы неправильные.

Методические материалы по выполнению практического задания

При выполнении практического задания обучающийся придерживается следующего алгоритма:

1. Записать дату, тему и цель задания;

2. Ознакомиться с правилами и условия выполнения практического задания;

3. Повторить теоретические задания, необходимые для рациональной работы и других практических действий, используя конспекты лекций и рекомендованную литературу, представленной в программе;

4. Выполнить работу по предложенному алгоритму действий;

5. Обобщить результаты работы, сформулировать выводы / дать ответы на контрольные вопросы;

Работа должна быть выполнена грамотно, с соблюдением культуры изложения. При использовании данных из учебных, методических пособий и другой литературы, периодических изданий, Интернет-источников должны иметься ссылки на вышеперечисленные.

Критерии оценки практического задания:

«Отлично» – правильный ответ, дается четкое обоснование принятому решению; рассуждения четкие последовательные логические; используются ссылки на полученные при изучении дисциплины знания; правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Хорошо» – правильный ответ, дается обоснование принятому решению; но с не существенными ошибками, в рассуждениях отсутствует логическая последовательность; используются ссылки на полученные при изучении дисциплины знания, правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Удовлетворительно» – правильный ответ, допускаются грубые ошибки в обосновании принятого решения; рассуждения не последовательные сумбурные; используются ссылки на полученные при изучении дисциплины знания; используются формулы, процедуры, понятия, имеющие прямое значение для подтверждения принятого решения, однако, при обращении к ним допускаются серьезные ошибки, студент не может правильно ими воспользоваться.

«Неудовлетворительно, не зачтено» – ответ неверный, отсутствует обоснование принятому решению; студент демонстрирует полное непонимание сути вопроса.

Для оценки решения ситуационной задачи (аналитического задания):

Оценка «отлично» выставляется, если задача решена грамотно, ответы на вопросы сформулированы четко. Эталонный ответ полностью соответствует решению студента, которое хорошо обосновано теоретически.

Оценка «хорошо» выставляется, если задача решена, ответы на вопросы сформулированы не достаточно четко. Решение студента в целом соответствует эталонному ответу, но не достаточно хорошо обосновано теоретически.

Оценка «удовлетворительно» выставляется, если задача решена не полностью, ответы не содержат всех необходимых обоснований решения.

Оценка «неудовлетворительно» выставляется, если задача не решена или имеет грубые теоретические ошибки в ответе на поставленные вопросы.

Методические материалы по выполнению лабораторного задания

При выполнении лабораторного задания обучающийся придерживается следующего алгоритма:

1. Записать дату, тему и цель задания;
2. Ознакомиться с правилами и условия выполнения задания;
3. Повторить теоретические задания, необходимые для рациональной работы и других практических действий, используя конспекты лекций и рекомендованную литературу, представленной в программе;
4. Выполнить работу по предложенному алгоритму действий;
5. Обобщить результаты работы, сформулировать выводы / дать ответы на контрольные вопросы;

Работа должна быть выполнена грамотно, с соблюдением культуры изложения. При использовании данных из учебных, методических пособий и другой литературы, периодических изданий, Интернет-источников должны иметься ссылки на вышеперечисленные.

Критерии оценки лабораторного задания:

«Отлично» – правильный ответ, дается четкое обоснование принятому решению; рассуждения четкие последовательные логические; используются ссылки на полученные при изучении дисциплины знания; правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Хорошо» – правильный ответ, дается обоснование принятому решению; но с не существенными ошибками, в рассуждениях отсутствует логическая последовательность; используются ссылки на полученные при изучении дисциплины знания, правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Удовлетворительно» – правильный ответ, допускаются грубые ошибки в обосновании принятого решения; рассуждения не последовательные сумбурные; используются ссылки на полученные при изучении дисциплины знания; используются формулы, процедуры, понятия, имеющие прямое значение для подтверждения принятого решения, однако, при обращении к ним допускаются серьезные ошибки, студент не может правильно ими воспользоваться.

«Неудовлетворительно, не зачтено» – ответ неверный, отсутствует обоснование принятому решению; студент демонстрирует полное непонимание сути вопроса.

Методические указания для подготовки к промежуточной аттестации.

Изучение учебных дисциплин (модулей) завершается зачетом/зачетом с оценкой или экзаменом. Подготовка к промежуточной аттестации способствует закреплению, углублению и обобщению знаний, получаемых, в процессе обучения, а также применению их к решению практических задач. Готовясь к экзамену, студент ликвидирует имеющиеся пробелы в знаниях, углубляет, систематизирует и упорядочивает свои знания. На зачете или экзамене студент демонстрирует то, что он освоил в процессе обучения по дисциплине (модулю).

Вначале следует просмотреть весь материал по дисциплине (модулю), отметить для себя трудные вопросы. Обязательно в них разобраться. В заключение еще раз целесообразно повторить основные положения, используя при этом листы опорных сигналов. Систематическая подготовка к занятиям в течение семестра позволит использовать время промежуточной аттестации для систематизации знаний.

3. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНИВАНИЯ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ И (ИЛИ) ОПЫТА ДЕЯТЕЛЬНОСТИ, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ

3.1. Организационные основы применения балльно-рейтинговой системы оценки успеваемости обучающихся по дисциплине (модулю)

Оценка качества освоения обучающимися дисциплины (модуля) реализуется в формате балльно-рейтинговой системы оценки успеваемости обучающихся (БРСО).

БРСО в ходе текущего контроля успеваемости и промежуточной аттестации осуществляется по 100-балльной шкале.

Академический рейтинг обучающегося по дисциплине (модулю) складывается из результатов:

- текущего контроля успеваемости (максимальный текущий рейтинг обучающегося 80 рейтинговых баллов);
- промежуточной аттестации (максимальный рубежный рейтинг обучающегося 20 рейтинговых баллов).

Условия оценки освоения обучающимся дисциплины (модуля) в формате БРСО доводятся преподавателем до сведения обучающихся на первом учебном занятии, а также размещены в свободном доступе в электронной информационно-образовательной среде Университета.

3.2. Проведение текущего контроля успеваемости обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося

В течение учебного семестра до промежуточной аттестации на основании утвержденной рабочей программы дисциплины (модуля) формируется текущий рейтинг обучающегося. Текущий рейтинг обучающегося складывается как сумма рейтинговых баллов, полученных им в течение учебного семестра по всем видам учебных занятий по учебной дисциплине.

В процессе текущего контроля оцениваются следующие действия обучающегося, направленные на освоение компетенций в рамках изучения учебной дисциплины:

- академическая активность (посещаемость учебных занятий, самостоятельное изучение содержания учебной дисциплины в электронной информационно-образовательной среде, соблюдение сроков сдачи практических заданий и текущих контрольных мероприятий и др.);
- выполнение и сдача текущих и итогового практических заданий (эссе, рефераты, творческие задания, кейс-задания, лабораторные работы, расчетные задания и др., активное участие в групповых интерактивных занятиях (дискуссии, WiKi-проекты и др.), защита проектов и др.);
- прохождение рубежей текущего контроля, включая соблюдение графика их прохождения в электронной информационно-образовательной среде.

Для планирования расчета текущего рейтинга обучающегося используются следующие пропорции:

Вид учебного действия	Максимальная рейтинговая оценка, баллов
академическая активность	10
практические задания	40
<i>из них: текущие практические задания</i>	20
<i>итоговое практическое задание</i>	20
рубежи текущего контроля	30
ИТОГО:	80

В течение учебного семестра по дисциплине (модулю) обучающимся должен быть накоплен текущий рейтинг не менее 52 рейтинговых баллов (65% от максимального значения текущего рейтинга).

Необходимыми условиями допуска обучающегося к промежуточной аттестации по дисциплине являются положительное прохождение обучающимся не менее 65% рубежей текущего контроля с накоплением не менее 65% максимального рейтингового балла за каждый рубеж текущего контроля и положительное выполнение итогового практического задания с накоплением не менее 65% максимального рейтингового балла, установленного за итоговое практическое задание.

Невыполнение вышеуказанных условий является текущей академической задолженностью, которая должна быть ликвидирована обучающимся до контрольного мероприятия промежуточной аттестации.

Сведения о наличии у обучающихся текущей академической задолженности, сроках и порядке добора рейтинговых баллов для её ликвидации доводятся до обучающихся педагогическим работником.

В случае неликвидации текущей академической задолженности, педагогический работник обязан во время контрольного мероприятия промежуточной аттестации поставить обучающемуся 0 рейтинговых баллов. В этом случае ликвидация текущей академической задолженности возможна в периоды проведения повторной промежуточной аттестации.

3.3. Проведение промежуточной аттестации обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося

Промежуточная аттестация по дисциплине (модулю) проводится в соответствии с Положением о промежуточной аттестации обучающихся по основным профессиональным образовательным программам высшего образования – программ специалитета в Российском государственном социальном университете и Положением о балльно-рейтинговой системе оценки успеваемости обучающихся по основным профессиональным образовательным программам высшего образования - программам бакалавриата, программам специалитета, программам магистратуры в Российском государственном социальном университете.

На промежуточную аттестацию отводится 20 рейтинговых баллов.

Ответы обучающегося на контрольном мероприятии промежуточной аттестации оцениваются педагогическим работником по 20 - балльной шкале, а итоговая оценка по дисциплине (модулю) выставляется по пятибалльной системе для дифференцированного зачета.

Критерии выставления оценки определяются Положением о балльно-рейтинговой системе оценки успеваемости обучающихся по основным профессиональным образовательным программам высшего образования - программам бакалавриата, программам специалитета, программам магистратуры в Российском государственном социальном университете.

В процессе определения рубежного рейтинга обучающегося используется следующая шкала:

Рубежный рейтинг	Критерии оценки освоения обучающимся учебной дисциплины в ходе контрольных мероприятий промежуточной аттестации
19-20 рейтинговых баллов	обучающийся глубоко и прочно усвоил программный материал, исчерпывающе, последовательно, грамотно и логически стройно его излагает, тесно увязывает с задачами и будущей деятельностью, не затрудняется с ответом при видоизменении задания, свободно справляется с задачами и практическими заданиями, правильно обосновывает принятые решения, умеет самостоятельно обобщать и излагать материал, не допуская

16-18 рейтинговых баллов	обучающийся твердо знает программный материал, грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос, может правильно применять теоретические положения и владеет необходимыми умениями и навыками при выполнении практических
13-15 рейтинговых баллов	обучающийся освоил основной материал, но не знает отдельных деталей, допускает неточности, недостаточно правильные формулировки, нарушает последовательность в изложении программного материала и испытывает затруднения в выполнении практических заданий
1-12 рейтинговых баллов	обучающийся не знает значительной части программного материала, допускает существенные ошибки, с большими затруднениями выполняет практические задания
0 рейтинговых баллов	не аттестован

Если результат контроля успеваемости в рамках проведения контрольных мероприятий промежуточной аттестации (рубежный рейтинг обучающегося) неудовлетворительный (получено менее 13 рейтинговых баллов), то промежуточная аттестация по учебной дисциплине (модулю) невозможна даже при наличии высокого текущего рейтинга, полученного по итогам текущего контроля по учебной дисциплине (модулю).

КОНСПЕКТЫ ЛЕКЦИОННЫХ ЗАНЯТИЙ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)

Проектирование баз данных

РАЗДЕЛ 1. Теория баз данных

Цели занятия.

Сформировать представление об основных понятиях объект, сущность, параметр, атрибут, таблица, запись, поле, домен, кортеж, первичный ключ и т.д. Три вида архитектур: локальная, «файл-сервер» и «клиент-сервер» (два варианта). Способы физического хранения данных и обмена данными. Сравнительный анализ и рекомендации по использованию каждого вида архитектур. Подробно рассматривается базовая технология СОМ. В краткой форме с указанием преимуществ и особенностей работы рассматриваются технологии ADO, CORBA и проч. Необходимость создания журнала аудита. Доступность и приемы заполнения журнала аудита. Назначение и использование просмотров. Просмотры редактируемые и не редактируемые. Область применения и преимущества каждого вида просмотра. Подробно рассматривается пять этапов проектирования базы данных. Состав работ, выполняемых на каждом этапе.

4. Структура лекционного занятия.

№ п/п	Содержание (кратко)	Методы и средства обучения
1	Основные понятия теории баз данных	1. Персональные компьютеры;
2	Описание целей и задач СУБД.	2. Средства доступа в Интернет; 3. Проектор.

5. Содержание лекционного занятия и взаимодействие с аудиторией.

Введение.

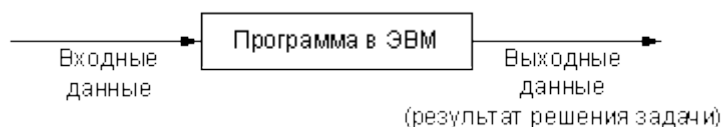
Формулирование темы занятия. Использование вводного примера, описание ситуации, демонстрация какого-либо процесса и т. д. (в зависимости от вида, замысла лекции и её структуры)

Тема 1.1. Введение в базы данных. Общая характеристика основных понятий Текст лекции.

Цель 1 лекции: показать, что с изменением вида решаемых на ЭВМ задач в программировании возникают новые виды представления данных, в том числе такой вид, как *базы данных*.

1.1. Развитие основных понятий представления данных

Любой вычислительный процесс представляет собой *отображение* (по определенному алгоритму) некоторых входных данных в выходные.



Соотношение сложности *представления* *обрабатываемых* *данных* и алгоритма вычислений определяет два класса задач:

- вычислительные задачи – достаточно простое представление данных и сложный, многооперационный процесс вычислений;
- задачи обработки данных (невывислительные задачи) – простой алгоритм обработки данных и сложное *представление обрабатываемых данных*.

На начальной стадии обучения программированию основное внимание уделяется разработке алгоритма решения задачи. Однако часто оказывается, что возможность (или невозможность) решения конкретной задачи зависит не только от выбранного алгоритма, но и от того, какие понятия используются для представления обрабатываемых данных.

Рассмотрим простейший пример вычисления по формуле:

$$Y=X^2+5X,$$

где **X** и **Y** – определенные числа, которые являются здесь элементарными единицами данных (*элементами данных*).

При программировании алгоритма решения этой задачи (*программирование формулы*) используется простейший вид данных – простая *переменная* (**X** и **Y** представляются в программе простыми переменными). Заметим, что простая *переменная* в системах программирования характеризуется определенным типом ее значений, которые должны выбираться при программировании. Даже в этом простейшем случае необходимо правильно выбрать *тип переменной*, причем от этого выбора может зависеть возможность или невозможность решения конкретной прикладной задачи (например, для представления конкретных данных не хватит отведенных разрядов).

Рассмотрим другой пример:

$$S=a_1+a_2+...+a_n.$$

Решение этой задачи в общем случае невозможно получить используя только простые переменные. Здесь обрабатывается не отдельное число, а последовательность чисел. В этом случае при программировании используется такой вид данных, как *массив* – совокупность элементов, с каждым из которых связан упорядоченный набор целых чисел, называемых индексами. Все элементы должны иметь одинаковый тип их значений, который и будет *типом массива*. В этом случае числа **a₁, a₂, ..., a_n** представляются в программе массивом **A(1), A(2), ..., A(N)**.

Приведенные примеры показывают, что изменение вида задач обуславливает необходимость использования других видов данных.

Ранние языки программирования (ФОРТРАН, АЛГОЛ-60) были предназначены для решения научно-технических вычислительных задач. В этих языках использовались только вышеуказанные виды данных (простые переменные и массивы) что было вполне достаточно.

Начиная с конца 60-х годов компьютеры начинают интенсивно использоваться для решения так называемых невывислительных задач, связанных с обработкой различного рода документов. Рассмотрим появление новых видов данных на примере упрощенных задач обработки данных.

Задача 1. Начисление заработной платы.

Рассматриваем задачу при двух упрощающих предположениях:

- сотруднику начисляется заработная плата на основе его оклада;
- никакие налоги и вычеты не учитываются.

Необходимые для решения этой задачи сведения о сотруднике представлены в следующей карточке НАЧИСЛЕНИЕ:

Фамилия, имя, отчество	Оклад	Количество отработанных дней в месяц	Начисленная сумма
FI0	0	K ₀	S

Для каждого работника начисленная сумма за определенный месяц рассчитывается по следующей формуле:

$$S = K_o O / K_r,$$

где K_r – количество рабочих дней в данном месяце.

Для каждого сотрудника соответствующие данные имеют конкретное значение, например:

Иванов Иван Иванович	1800	24	1800

Эти значения имеют смысл только во взаимосвязи друг с другом. Отдельно выбранное число 1800 теряет свой содержательный смысл, поэтому использовать такой вид данных, как простая *переменная*, здесь нельзя. В то же время набор соответствующих значений, характеризующих конкретного сотрудника, имеет разные типы (символьный и числовой), т.е. использовать для его представления такой вид данных, как *массив*, также нельзя. Таким образом, понятий "простая *переменная*" и "*массив*" недостаточно, чтобы представить соответствующую карточку.

Для описания аналогичных представлений данных в *предметной области* невычислительных задач вводится ряд новых понятий [[1.1]].

Элемент данных (поле) – наименьшая единица поименованных данных.

Для данного примера элементами данных являются FIO, O, K_o, S .

Для описания карточки сотрудника используется понятие " *Логическая запись* ".

Логическая запись – поименованная совокупность элементов данных (полей).

Экземпляр логической записи – текущее значение элементов записи.

Для представления всего набора карточек сотрудников используется понятие " *Логический файл* "

Логический файл - поименованная совокупность всех экземпляров записей заданного типа.

Пример *логического файла* НАЧИСЛЕНИЕ:

Иванов Иван Иванович	1800	24	1800

Петров Петр Петрович	2200	20	1830

Сидоров Сидор Сидорович	2500	24	2500

Таким образом, с помощью введенных понятий можно описывать соответствующие данные. Для отображения этих понятий в современных языках программирования, предназначенных как для вычислительных задач, так и для задач обработки данных, введены новые виды данных.

В алгоритмическом языке *Паскаль* вводится такой вид данных, как *запись* (RECORD) – сложная *переменная* с несколькими компонентами, которые могут иметь разные типы. Кроме того, *доступ* к компонентам записи (полям) осуществляется не по индексу, а по имени. При программировании задачи 1 на языке *Паскаль* *логическая запись* НАЧИСЛЕНИЕ представляется видом данных RECORD, набор экземпляров логических записей сотрудников (*логический файл*) представляется "физическим" файлом, формируемым средствами языка *Паскаль* и операционной системы.

Salary = RECORD

FIO: string;

O: real;

Ko: real;

S: real;

END;

Отметим важную специфику таких невычислительных задач. Для этих задач характерны большие объемы данных (большое количество сотрудников, большое количество производимых изделий и т. п.). Указанные данные, как правило, используются для решения задачи многократно (зарплата начисляется постоянно каждый месяц), поэтому данные должны достаточно долго храниться в памяти ЭВМ. Для длительного хранения всегда используется внешняя *память*.

В связи с этим решение задачи 1 состоит из двух этапов.

1. Ввод исходных данных и занесение их во внешнюю *память*.

```
type
Salary = RECORD
FIO:      string;
O:        real;
Ko:       real;
S:real;
END;
FSalary = File of Salary;
var
  F: FSalary;
...
{ Ввод исходных данных }
repeat
  write('Введите количество сотрудников (не более',
    MaxN, '): ');
  readln(N);
until (N>0) AND (N<=MaxN);
For I := 1 to N do
Begin
  Write('Введите фамилию сотрудника с номером ', I, ': ');
  ReadLn(Sotr[i].FIO);
  Write('Введите оклад сотрудника с номером ', I, ': ');
  ReadLn(Sotr[i].O);
  Write('Введите кол-во отработанных дней сотрудника с
    номером ', I, ': ');
  ReadLn(Sotr[i].Ko);
End;

{ Занесение данных во внешнюю память }
```

```
Assign(F, 'MyFile.fsf');
Rewrite(F);
For I := 1 to N do
  Write(F, Sotr[i]);
Close(F);
```

2. Чтение исходных данных из внешней памяти, расчет начисленных сумм и вывод на печать.

```
...
{ Чтение данных из внешней памяти }
Assign(F, 'MyFile.fsf');
Reset(F);
For I := 1 to N do
  Read(F, Sotr[i]);
Close(F);
```

```

{ Расчет и печать начисленных сумм }
For I := 1 to N do
Begin
  Sotr[i].S := Sotr[i].O * Sotr[i].Ko / Kr;
  WriteLn(Sotr[i].FIO, ' ', Sotr[i].S);
End;

```

...

Представленные программы решают поставленную задачу при сделанных предположениях. Необходимые для этого данные хранятся в файле *MyFile.fsf*, предназначенном только для решения этой задачи. Отметим, что в этом случае описание данных включено в прикладную программу. При изменении формата записей файла необходимо изменение прикладной программы. Таким образом, программная система, решающая поставленную задачу, определяет свои собственные данные и управляет ими. Такие программные системы называются файловыми системами [[1.2]], [[1.3]].

Задача 2. Учет кадрового состава.

Здесь обрабатываются сведения о сотруднике, представленные в карточке СОТРУДНИК:

Фамилия, имя, отчество	Должность	Год рождения	Оклад	Место жительства
FIO	D	G	O	M

Решение задачи состоит из следующих этапов:

Ввод исходных данных и занесение их во внешнюю *память*.

Чтение исходных данных из внешней памяти с целью удаления, корректировки или добавления записи.

...

```

{ Чтение данных из внешней памяти }
Assign(F, 'MyFile.fsf');
Reset(F);
IsFound := False;
For I := 1 to N do
Begin
  Read(F, Sotr);
  If Sotr.FIO = KeyFio Then
  Begin
    IsFound := True;
    Sotr.D := 'Начальник отдела';
    Seek(F, FilePos(F)-1);
    Write(F, Sotr);
    Break;
  End;
End;
If IsFound Then
WriteLn('Корректировка успешно произведена')
Else WriteLn('Сотрудника ', KeyFio, ' не обнаружено');
Close(F);

```

...

В рассматриваемом случае задача 2 решается независимо от задачи 1.

Задача 3. Учет экономии фонда оплаты труда (ФОТ) в связи с болезнью сотрудников.

Обрабатываются сведения, представленные записями ЭКОНОМИЯ ФОТ:

Фамилия, имя, отчество	Оклад	Количество дней на больничном листе	Невыплаченная сумма
FIO	O	K _{дв}	SN

$SN = K_{дв} O / K_r$.

Программа решения задачи 3 аналогична программе решения задачи 1.

Рассмотрим типичный случай, когда все три вышеуказанные программные системы **функционируют в одной организации**. Отметим следующие принципиальные эксплуатационные недостатки:

Информация дублируется. В трех файлах присутствуют поля **FIO, O**, что приводит к существенному перерасходу памяти. При внесении изменений (например, изменении фамилии) приходится вносить одно и то же *значение* несколько раз в разные файлы, что приводит к увеличению затрат машинного времени. Существует потенциальная возможность противоречивости данных (в один *файл* изменения внесены, в другой – нет).

Устранить перечисленные недостатки можно, объединив соответствующие записи и создав единую информационную базу для всех вышеназванных задач. На первый взгляд наиболее естественно объединить все записи в одну, убрав дублирующие поля. Получаем возможный вариант объединения:

FIO	D	O	G	K _о	M	K _{дв}	S	SN

Дублирование информации полностью убрано. Расход памяти минимален. Недостатки устранены. Рассмотрим, как в этом случае изменится время решения задач 1–3. Время решения задачи прямо пропорционально объему считываемых из внешней памяти данных.

Обозначим T_i , l_i , N_i соответственно время решения, длину записи, число записей i -й задачи ($i = 1, 2, 3$) при использовании отдельных файлов для каждой задачи:

$$T_i \approx C \times l_i \times N_i$$

где C – некоторый коэффициент пропорциональности.

Обозначим R_i , d , N соответственно время решения i -й задачи ($i = 1, 2, 3$) при использовании файла объединенных записей, длину записи, число записей:

$$R_i \approx C \times d \times N$$

Заметим, что $N_1 = N_2 = N$, $N_3 \ll N$.

Тогда время решения i -й задачи ($i = 1, 2$) при использовании объединенного файла увеличится в $R_i/T_i \approx d/l_i$ раз.

Для нашего примера время решения задач в зависимости от выбранной длины полей может изменяться в 2–3 раза. Таким образом, платой за *исключение* дублирования информации является увеличение времени решаемых задач. Заметим, что такое увеличение, как правило, допустимо.

Время решения задачи 3 увеличится в $R_3/T_3 \approx d \times N/l_3 \times N_3$ раз. Так как для данного примера $N_3 \ll N$, то $R_3 \gg T_3$. Время решения задачи 3 может увеличиться на несколько порядков, что совершенно недопустимо.

Рассмотрим другой вариант построения единой информационной базы. Объединим записи задач 1 и 2, *запись* задачи 3 оставим отдельно. Получим два типа записей:

FIO	D	O	G	K _о	S	M
FIO	O	K _{дв}	SN			

В этом случае дублирование остается (дублируются поля **FIO, O**). Но так как $N_3 \ll N$, то общий объем дублирования незначителен.

Время решения задачи 1 и 2 в этом случае незначительно возрастет по сравнению с вариантом отдельных файловых систем, время решения задачи 3 такое же, как и в начальном варианте отдельного файла. Такое *объединение* позволяет значительно уменьшить влияние недостатков и в то же время существенно увеличивает время решения всех задач. Все три задачи можно решать, используя общую информационную базу из двух типов записей. Отметим, что два приведенных типа записей связаны друг с другом по полю **FIO** (находятся в некотором отношении). Отметим, что приведенные варианты интеграции не исчерпывают все возможные способы *интеграции данных* для приведенных задач и к вопросу выбора наилучшего варианта вернемся в последующих лекциях.

Здесь очень важно, что в этом случае для решения вышеуказанных задач используется некоторый новый вид данных, формируемый на основе интеграции записей.

Для описания этого вида данных вводится новое понятие "База данных" [[1.1]].

База данных – совокупность экземпляров различных типов записей и отношений между записями и элементами.

Базу данных можно определить как совокупность взаимосвязанных хранящихся вместе данных при наличии такой минимальной избыточности, которая допускает их использование оптимальным образом для одного или нескольких приложений.

Таким образом, появление понятия "*Базы данных*" обусловлено возникновением нового класса невычислительных задач, при решении которых используются общие данные. В качестве основного критерия оптимальности функционирования *базы данных*, как правило, используются временные характеристики реализации запросов пользователей прикладными программами.

Краткие итоги. Рассмотрено развитие основных понятий представления данных. Описаны классические понятия программирования, связанные с данными (*переменная, массив*) и появление новых понятий программирования (*поле, запись, файл*) как следствие расширения круга решаемых задач и их отражения в системах программирования. Поставлена задача интегрирования данных при использовании несколькими задачами общих данных. Определено понятие *базы данных*.

Тема 1.2. Системы управления базами данных

Цель 2 лекции: показать необходимость создания программного интерфейса между прикладными программами и базой данных, определить понятие системы управления базами данных и сформулировать основные *функции СУБД*, вытекающие из задачи взаимодействия многих пользователей с базой данных.

В прикладной программе, использующей при решении задачи один или несколько отдельных файлов, за сохранность и *достоверность* данных отвечал программист, работающий с этой задачей. *Использование базы данных* предполагает работу с ней нескольких прикладных программ, решающих задачи разных пользователей.

Естественно, что за сохранность и *достоверность* интегрированных данных программист, решающий одну из прикладных задач, отвечать уже не может. Кроме того, расширение круга решаемых с использованием *базы данных* задач может приводить к появлению новых типов записей и отношений между ними. Такое изменение структуры *базы данных* не должно вести к изменению *множества* ранее разработанных и успешно функционирующих прикладных программных систем, работающих с базой данных. С другой стороны, возможное изменение любой из прикладных программ, в свою *очередь*, не должно приводить к изменению структуры данных. Все вышесказанное обуславливает необходимость отделения данных от прикладных программ.

Роль интерфейса между прикладными программами и базой данных, обеспечивающего их независимость, играет программный комплекс – *система управления базами данных (СУБД)* ([рис. 2.1](#)).

СУБД – программный комплекс поддержки интегрированной совокупности данных, предназначенный для создания, ведения и использования *базы данных* многими пользователями (прикладными программами).

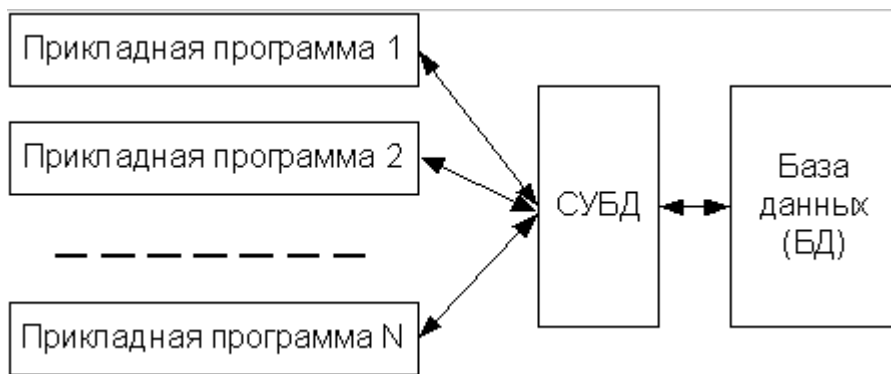


Рис. 2.1. Обеспечение независимости прикладных программ и базы данных
Определим еще одно понятие.

Банк данных – система языковых, алгоритмических, программных, технических и организационных средств поддержки интегрированной совокупности данных, а также сами эти данные, представленные в виде баз данных.

Перечислим основные *функции системы управления базами данных*.

1. *Определение* структуры создаваемой базы данных, ее инициализация и проведение начальной загрузки.

Как правило, создание структуры **базы данных** происходит в режиме диалога. *СУБД* последовательно запрашивает у пользователя необходимые данные. В большинстве современных *СУБД* **база данных** представляется в виде совокупности таблиц. Рассматриваемая *функция* позволяет описать и создать в памяти структуру таблицы, провести начальную загрузку данных в таблицы. Примеры таких действий для *СУБД* MS Access приведены на [рисунке 2.2](#).

Имя поля	Тип данных	Описание
Номер	Счетчик	
Фамилия	Текстовый	
Имя	Текстовый	
Отчество	Текстовый	
Номер_аттестата	Текстовый	
Дата_аттестата	Дата/время	

Свойства поля

Общие | Подстановка

Размер поля: Длинное целое

Новые значения: Последовательные

Формат поля:

Подпись:

Индексированное поле: Да (Совпадения не допускаются)

Имя поля может состоять из 64 знаков с учетом пробелов. Для справки по именам полей нажмите клавишу F1.

Рис. 2.2. Формирование структуры базы данных в СУБД Access

2. Предоставление пользователям возможности манипулирования данными (*выборка* необходимых данных, выполнение вычислений, разработка интерфейса ввода/вывода, *визуализация*).

Такие возможности в *СУБД* представляются либо на основе использования специального языка программирования, входящего в состав *СУБД*, либо с помощью графического интерфейса.

В MS Access реализация данной функции может быть реализована созданием запросов и форм ввода с помощью графического интерфейса (рис. 2.3).

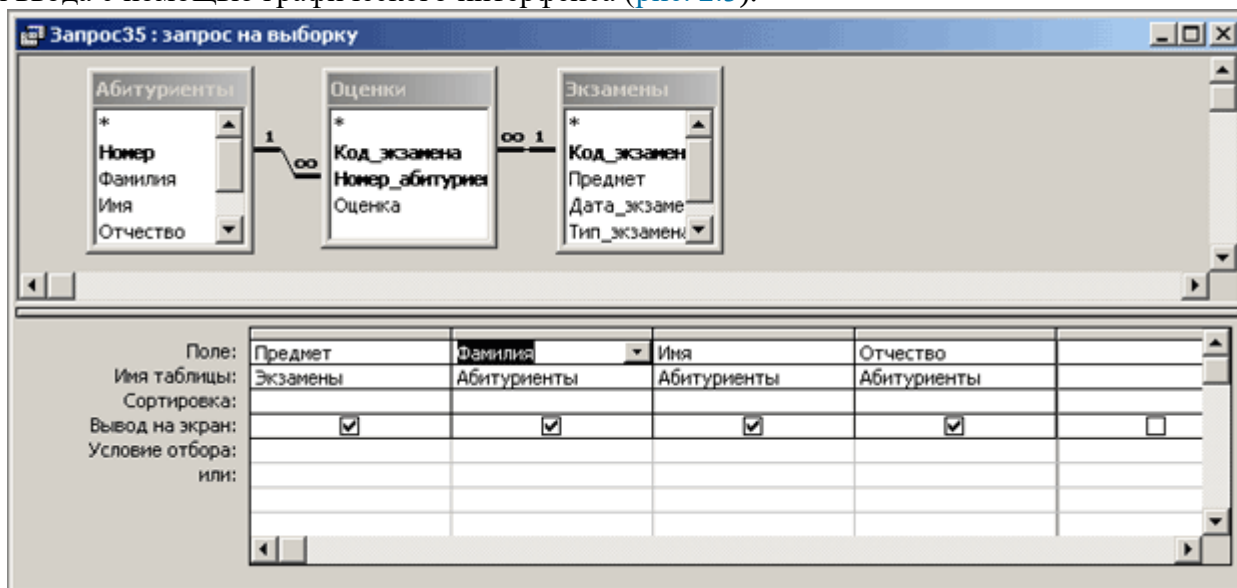


Рис. 2.3. Формирование запроса на выборку в СУБД Access

Для клиент-серверных СУБД существуют средства, позволяющие выполнять запросы, и *программные средства*, позволяющие создавать графический *интерфейс* пользователя.

3. Обеспечение независимости прикладных программ и данных (логической и физической независимости).

Важнейшим свойством СУБД является возможность поддерживать два независимых взгляда на базу данных – "взгляд пользователя", воплощаемый в логическом представлении данных, и его отражения в прикладных программах; и "взгляд системы" – физическое представление данных в памяти ЭВМ. Обеспечение логической независимости данных предоставляет возможность изменения (в определенных пределах) логического представления *базы данных* без необходимости изменения физических структур хранения данных. Таким образом, изменение логического представления данных в прикладных программах не приводит к изменению структур хранения данных. Обеспечение физической независимости данных предоставляет возможность изменять (в определенных пределах) способы организации *базы данных* в памяти ЭВМ не вызывая необходимости изменения "логического" представления данных. Таким образом, изменение способов организации *базы данных* не приводит к изменению прикладных программ.

4. Защита логической целостности **базы данных**.

Основной целью реализации этой функции является повышение достоверности данных в базе данных. *Достоверность* данных может быть нарушена при их вводе в БД или при неправомерных действиях процедур обработки данных, получающих и заносящих в БД неправильные данные. Для повышения достоверности данных в системе объявляются так называемые ограничения целостности, которые в определенных случаях "отлавливают" неверные данные. Так, во всех современных СУБД проверяется соответствие вводимых данных их типу, описанному при создании структуры. Система не позволит ввести символ в *поле* числового типа, не позволит ввести недопустимую дату и т.п. В развитых системах ограничения целостности описывает программист, исходя из содержательного смысла задачи, и их проверка осуществляется при каждом обновлении данных. Более подробно разные аспекты логической целостности *базы данных* будут рассматриваться в последующих разделах.

5. Защита физической целостности.

При работе ЭВМ возможны сбои в работе (например, из-за отключения электропитания), повреждение машинных носителей данных. При этом могут быть нарушены связи между данными, что приводит к невозможности дальнейшей работы.

Развитые СУБД имеют средства восстановления *базы данных*. Важнейшим используемым понятием является понятие "*транзакции*". *Транзакция* – это единица действий, производимых с базой данных. В состав транзакции может входить несколько операторов изменения *базы данных*, но либо выполняются все эти *операторы*, либо не выполняется ни один. СУБД, кроме ведения собственно *базы данных*, ведет также журнал *транзакций*.

Необходимость использования *транзакций* в базах данных проиллюстрируем на упрощенном примере. Предположим, что *база данных* используется в некотором банке и один из клиентов желает перевести деньги на счет другого клиента банка. В базе данных хранится информация о количестве денег у каждого из клиентов. Нам нужно сделать два изменения в базе данных – уменьшить сумму денег на счете одного из клиентов и, соответственно, увеличить сумму денег на другом счете. Конечно, реальный перевод денег в банке представляет собой гораздо более сложный процесс, затрагивающий много таблиц, а возможно, и много баз данных. Однако суть остается та же – нужно либо совершить все действия (увеличить счет одного клиента и уменьшить счет другого), либо не выполнить ни одно из этих действий. Нельзя уменьшить сумму денег на одном счете, но не увеличить сумму денег на другом. Предположим также, что после выполнения первого из действий (уменьшения суммы денег на счете первого клиента) произошел сбой. Например, могла прерваться *связь* клиентского компьютера с базой данных или на клиентском компьютере мог произойти системный сбой, что привело к перезагрузке операционной системы. Что в этом случае стало с базой данных? *Команда* на уменьшение денег на счете первого клиента была выполнена, а вторая *команда* – на увеличение денег на другом счете – нет, что привело бы к противоречивому, неактуальному состоянию *базы данных*.

Использование механизма *транзакций* позволяет находить решение в этом и подобных случаях. Перед выполнением первого действия выдается *команда* начала транзакции. В транзакцию включается операция снятия денег на одном счете и увеличения суммы на другом счете. Оператор завершения транзакций обычно называется COMMIT. Поскольку после выполнения первого действия *транзакция* не была завершена, изменения не будут внесены в базу данных. Изменения вносятся (фиксируются) только после завершения транзакции. До выдачи данного оператора сохранения данных в базе не произойдет.

В нашем примере, поскольку оператор фиксации транзакции не был выдан, *база данных* "откатится" в первоначальное состояние – иными словами, суммы на счетах клиентов останутся те же, что и были до начала транзакции. Администратор базы данных может отслеживать состояние транзакций и в необходимых случаях вручную "откатывать" транзакции. Кроме того, в очевидных случаях СУБД самостоятельно принимает решение об "откате" транзакции.

Транзакции не обязательно могут быть короткими. Бывают транзакции, которые длятся несколько часов или даже несколько дней. Увеличение количества действий в рамках одной транзакции требует увеличения занимаемых системных ресурсов. Поэтому желательно делать транзакции по возможности короткими. В журнал *транзакций* заносятся все транзакции – и зафиксированные, и завершившиеся "откатом". Ведение журнала транзакций совместно с созданием резервных копий *базы данных* позволяет достичь высокой надежности *базы данных*.

Предположим, что *база данных* была испорчена в результате аппаратного сбоя компьютера, на котором был установлен сервер СУБД. В этом случае нужно использовать последнюю сделанную резервную копию *базы данных* и журнал *транзакций*. Причем применить к базе данных нужно только те транзакции, которые были зафиксированы после создания резервной копии. Большинство современных СУБД позволяют администратору воссоздать базу данных исходя из резервной копии и журнала транзакций. В таких системах в определенный момент БД копируется на резервные носители. Все обращения к БД записываются программно в журнал изменений. Если *база данных* разрушена, запускается процедура восстановления, в процессе которой в резервную копию из журнала изменений вносятся все произведенные изменения.

6. Управление полномочиями пользователей на доступ к базе данных.

Разные пользователи могут иметь разные полномочия по работе с данными (некоторые данные должны быть недоступны; определенным пользователям не разрешается обновлять данные и т.п.). В СУБД предусматриваются *механизмы* разграничения полномочий доступа, основанные либо на принципах паролей, либо на описании полномочий.

7. Синхронизация работы нескольких пользователей.

Достаточно часто может иметь *место* ситуация, когда несколько пользователей одновременно выполняют операцию обновления одних и тех же данных. Такие коллизии могут привести к нарушению логической целостности данных, поэтому система должна предусматривать меры, не допускающие обновление данных другим пользователям, пока работающий с этими данными *пользователь* полностью не закончит с ними работать. Основным используемым здесь понятием является понятие "блокировка". Блокировки необходимы для того, чтобы запретить различным пользователям возможность одновременно работать с базой данных, поскольку это может привести к ошибкам.

Для реализации этого запрета СУБД устанавливает блокировку на объекты, которые использует *транзакция*. Существуют разные типы блокировок – табличные, страничные, строчные и другие, которые отличаются друг от друга количеством заблокированных записей. Чаще других используется строчная *блокировка* – при обращении транзакции к одной строке блокируется только эта строка, остальные строки остаются доступными для изменения.

Таким образом, процесс внесения изменений в базу данных состоит из следующей последовательности действий: выдается оператор начала транзакции, выдается оператор изменения данных, СУБД анализирует оператор и пытается установить блокировки, необходимые для его выполнения, в случае успешной блокировки оператор выполняется, затем процесс повторяется для следующего оператора транзакции. После успешного выполнения всех операторов внутри транзакции выполняется оператор фиксации транзакции. СУБД фиксирует изменения, сделанные транзакцией, и снимает блокировки. В случае неуспеха выполнения какого-либо из операторов *транзакция* "откатывается", данные получают прежние значения, блокировки снимаются.

8. Управление ресурсами среды хранения.

БД располагается во внешней памяти ЭВМ. При работе в БД заносятся новые данные (занимается *память*) и удаляются данные (освобождается *память*). СУБД выделяет ресурсы памяти для новых данных, перераспределяет освободившуюся *память*, организует ведение очереди запросов к внешней памяти и т.п.

9. Поддержка деятельности системного персонала.

При эксплуатации *базы данных* может возникать необходимость изменения параметров СУБД, выбора новых методов доступа, изменения (в определенных пределах) структуры хранимых данных, а также выполнения ряда других общесистемных действий. СУБД предоставляет возможность выполнения этих и других действий для поддержки деятельности БД обслуживающему БД системному персоналу, называемому администратором БД.

Краткие итоги. Рассмотрено понятие системы управления базами данных как интерфейса между прикладными программами и базами данных. Введено понятие банка данных. Дана характеристика основных функций систем управления базами данных, вытекающих из задачи взаимодействия многих пользователей с базой данных:

- Определение структуры создаваемой базы данных, ее инициализация и проведение начальной загрузки
- Предоставление пользователям возможности манипулирования данными (выборка необходимых данных, выполнение вычислений, разработка интерфейса ввода/вывода, визуализация).
- Обеспечение независимости прикладных программ (логической и физической независимости).
- Защита логической целостности базы данных.
- Защита физической целостности.

- Управление полномочиями пользователей на доступ к базе данных.
- Синхронизация работы нескольких пользователей.
- Управление ресурсами среды хранения.
- Поддержка деятельности системного персонала.

РАЗДЕЛ 2. Общие принципы построения (архитектура) баз данных

Цели занятия.

Сформировать представление о способах создания файла базы данных: на уровне команд и диалоговый режим работы с утилитой IVConsol. Приемы модификации структуры базы данных на уровне команд, с рассмотрением последствий не корректных действий при выполнении данной операции. Команда удаления файла базы данных. Команды по созданию, модификации и удалению таблиц, с указанием особенностей работы каждой из команд. Допустимые типы полей и способы их задания и приемы указания дополнительных характеристик полей. Понятие целостности и достоверности данных. Опция СНЕСК в командах создания и модификации таблицы. Приемы создания вычисляемых (виртуальных) полей, установка и удаление условий ссылочной целостности данных. Понятия индекса и ключа, с указанием общих свойств и различий. Типы и виды ключей и два способа создания ключей. Приемы создания именованных и не именованных условий ссылочной целостности и особенностей их работы. Понятие набора данных, его размера. Методы перемещения курсора по записям набора данных. Четыре метода доступа к полям и их значениям. Методы поиска данных на стороне клиента, как по произвольному полю (полям), так и по индексному полю (полям), с указанием особенностей каждого вида поиска. Поиск на частичное совпадение и поиск без учета регистра. Поиск по индексному полю: на точное совпадение и пошаговое приближение (инкрементальный локатор). Варианты формирования запросов на добавление, редактирование и удаление записи: Особенности и преимущества каждого из вариантов. Методы сортировки данных на стороне сервера и клиента. Сортировка по-простому и сложному индексу. Запросы на извлечение записей с группировкой. Запросов с фильтрацией записей. Комбинированные запросы с сортировкой, группировкой и фильтрацией. Понятие отчета. Процедура формирования отчета. Отчеты с вычислениями, группировкой и сортировкой. Вывод отчета на экран и печать. Понятие и назначение генератора. Команды создания, запуска, модификации, управления и удаления генератора. Понятие хранимой процедуры, её назначение. Создание различных типов хранимых процедур и их модификации. Назначение хранимых процедур, преимущества использования хранимых процедур. Вызов хранимых процедур. Понятие и назначение триггера. Создание, изменение и удаление триггера. Понятие отчета и его назначение. Компоненты, методы и свойства, с помощью которых формируется отчет. Редактирование отчета. Предварительный просмотр и вывод отчета на печать. Понятие транзакции. Излагаются методы создания и управления транзакциями (два способа). Сравнительная характеристика двух способов управления транзакциями. Состояния, в которых могут находиться транзакции. Конкурирующие транзакции. Понятие изоляции транзакций. Конфликты и приемы их разрешения.

4. Структура лекционного занятия.

№ п/п	Содержание (кратко)	Методы и средства обучения
	Типы архитектур СУБД	
1	Проектирование баз данных. Первая стадия	1. Персональные компьютеры;
2	концептуального проектирования базы данных (концептуальное моделирование). Вторая стадия	2. Средства доступа в Интернет;
	концептуального проектирования (Модели данных СУБД. Представление концептуальной модели средствами модели данных СУБД).	3. Проектор.

5. Содержание лекционного занятия и взаимодействие с аудиторией.

Введение.

Формулирование темы занятия. Использование вводного примера, описание ситуации, демонстрация какого-либо процесса и т. д. (в зависимости от вида, замысла лекции и её структуры)

Тема 2.1. Различные архитектурные решения, используемые при реализации многопользовательских СУБД.

Цель 3 лекции: показать основные варианты технологии работы нескольких пользователей с одной базой данных, связанные как с основными свойствами вычислительной техники, так и с развитием программного обеспечения.

Как уже отмечалось, понятие *базы данных* изначально предполагало возможность решения многих задач несколькими пользователями. В связи с этим, важнейшей характеристикой *современных СУБД* является наличие многопользовательской технологии работы. Разная реализация таких технологий в разное время была связана как с основными свойствами вычислительной техники, так и с развитием программного обеспечения. Дадим краткую характеристику этих технологий в хронологическом порядке.

3.1. Централизованная архитектура

При использовании этой технологии *база данных, СУБД* и *прикладная программа (приложение)* располагаются на одном компьютере (мэйнфрейме или персональном компьютере) ([рис.3.1.](#)). Для такого способа организации не требуется поддержки сети и все сводится к автономной работе. Работа построена следующим образом:

- База данных в виде набора файлов находится на жестком диске компьютера.
- На том же компьютере установлены СУБД и приложение для работы с БД.
- Пользователь запускает приложение. Используя предоставляемый приложением пользовательский интерфейс, он инициирует обращение к БД на выборку/обновление информации.
 - Все обращения к БД идут через СУБД, которая инкапсулирует внутри себя все сведения о физической структуре БД.
 - СУБД инициирует обращения к данным, обеспечивая выполнение запросов пользователя (осуществляя необходимые операции над данными).
 - Результат СУБД возвращает в приложение.
 - Приложение, используя пользовательский интерфейс, отображает результат выполнения запросов.

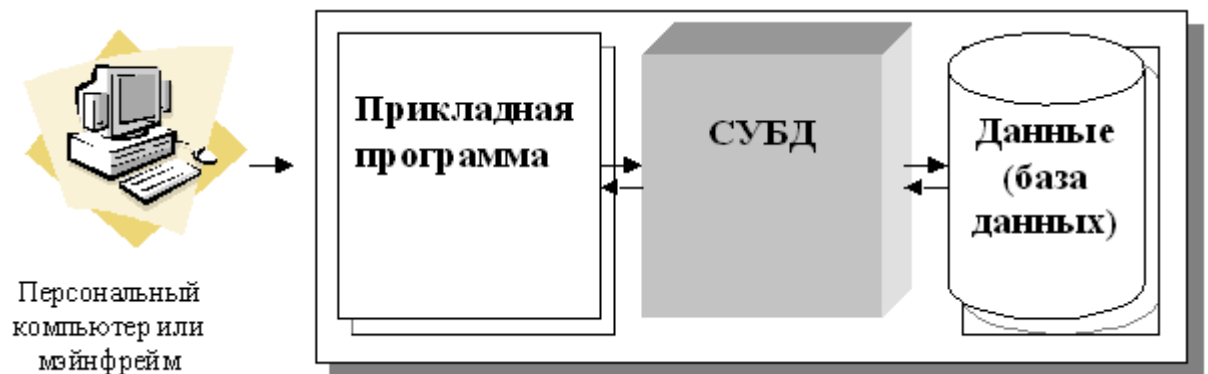


Рис. 3.1. Централизованная архитектура

Подобная *архитектура* использовалась в первых версиях *СУБД DB2, Oracle, Ingres* [[3.1](#)].

Многопользовательская технология работы обеспечивалась либо режимом мультипрограммирования (одновременно могли работать *процессор* и внешние устройства – например, пока в прикладной программе одного пользователя шло считывание данных из внешней памяти, *программа* другого пользователя обрабатывалась процессором), либо режимом разделения времени (пользователям по очереди выделялись кванты времени на выполнение их программ). Такая технология была распространена в период "господства" больших ЭВМ (IBM-370, ЕС-1045, ЕС-1060). Основным недостатком этой модели является резкое снижение производительности при увеличении числа пользователей.

3.2. Технология с сетью и файловым сервером (архитектура "файл-сервер")

Увеличение сложности задач, появление персональных компьютеров и локальных вычислительных сетей явились предпосылками появления новой архитектуры *файл-сервер*. Эта *архитектура* баз данных с сетевым доступом предполагает назначение одного из компьютеров сети в качестве выделенного сервера, на котором будут храниться файлы *базы данных* [[3.2]]. В соответствии с запросами пользователей файлы с *файл-сервера* передаются на рабочие станции пользователей, где и осуществляется основная часть обработки данных. Центральный *сервер* выполняет в основном только роль хранилища файлов, не участвуя в обработке самих данных (*рис. 3.2.*).

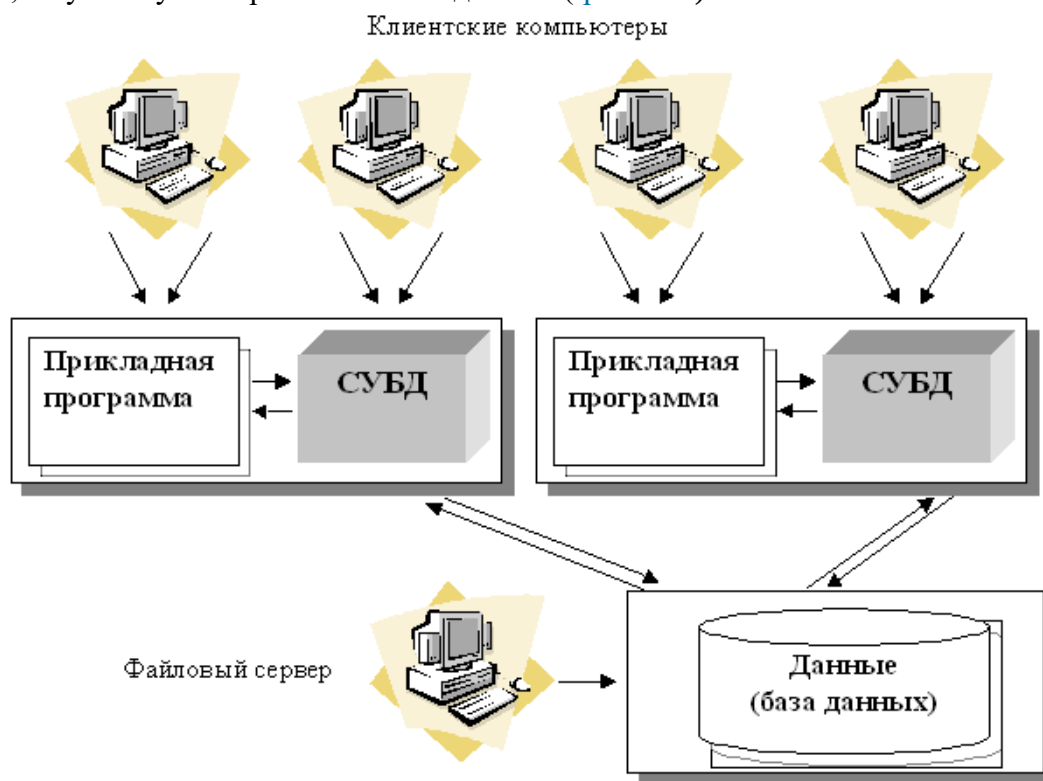


Рис. 3.2. Архитектура "файл-сервер"

Работа построена следующим образом:

- База данных в виде набора файлов находится на жестком диске специально выделенного компьютера (файлового сервера).
- Существует локальная сеть, состоящая из клиентских компьютеров, на каждом из которых установлены СУБД и приложение для работы с БД.
- На каждом из клиентских компьютеров пользователи имеют возможность запустить приложение. Используя предоставляемый приложением пользовательский интерфейс, он инициирует обращение к БД на выборку/обновление информации.
- Все обращения к БД идут через СУБД, которая инкапсулирует внутри себя все сведения о физической структуре БД, расположенной на файловом сервере.
- СУБД инициирует обращения к данным, находящимся на файловом сервере, в результате которых часть файлов БД копируется на клиентский компьютер и

обрабатывается, что обеспечивает выполнение запросов пользователя (осуществляются необходимые операции над данными).

- При необходимости (в случае изменения данных) данные отправляются назад на файловый сервер с целью обновления БД.
- Результат СУБД возвращает в приложение.
- Приложение, используя пользовательский интерфейс, отображает результат выполнения запросов.

В рамках архитектуры " *файл-сервер* " были выполнены первые версии популярных так называемых настольных *СУБД*, таких, как dBase и Microsoft Access.

В литературе [[3.2]] указываются следующие основные недостатки данной архитектуры:

- При одновременном обращении множества пользователей к одним и тем же данным производительность работы резко падает, т.к. необходимо дожидаться пока пользователь, работающий с данными, завершит свою работу. В противном случае возможно затирание исправлений, сделанных одними пользователями, изменениями других пользователей.
- Вся тяжесть вычислительной нагрузки при доступе к БД ложится на приложение клиента, так как при выдаче запроса на выборку информации из таблицы вся таблица БД копируется на клиентскую машину и выборка осуществляется на клиенте. Таким образом, неоптимально расходуются ресурсы клиентского компьютера и сети. В результате возрастает сетевой трафик и увеличиваются требования к аппаратным мощностям пользовательского компьютера.
- Как правило, используется навигационный подход, ориентированный на работу с отдельными записями.
- В БД на *файл-сервере* гораздо проще вносить изменения в отдельные таблицы, минуя приложения, непосредственно из инструментальных средств (например, из утилиты Database Desktop фирмы Borland для файлов Paradox и dBase); подобная возможность облегчается тем обстоятельством, что фактически у таких СУБД база данных – понятие более логическое, чем физическое, поскольку под БД понимается набор отдельных таблиц, сосуществующих в отдельном каталоге на диске. Все это позволяет говорить о низком уровне безопасности – как с точки зрения хищения и нанесения вреда, так и с точки зрения внесения ошибочных изменений.
- Недостаточно развитый аппарат транзакций служит потенциальным источником ошибок в плане нарушения смысловой и ссылочной целостности информации при одновременном внесении изменений в одну и ту же запись.

3.3. Технология "клиент – сервер"

Использование технологии " *клиент – сервер* " предполагает наличие некоторого количества компьютеров, объединенных в *сеть*, один из которых выполняет особые *управляющие* функции (является сервером сети).

Так, *архитектура* " *клиент – сервер* " разделяет функции приложения пользователя (называемого клиентом) и сервера. *Приложение-клиент* формирует *запрос* к серверу, на котором расположена *БД*, на структурном языке *запросов SQL* (Structured Query Language), являющемся промышленным стандартом в мире реляционных *БД*. Удаленный *сервер* принимает *запрос* и переадресует его *SQL-серверу БД*. *SQL-сервер* – специальная *программа*, управляющая удаленной базой данных. *SQL-сервер* обеспечивает интерпретацию запроса, его выполнение в базе данных, формирование результата выполнения запроса и выдачу его приложению-клиенту. При этом ресурсы клиентского компьютера не участвуют в физическом выполнении запроса; клиентский *компьютер* лишь отправляет *запрос* к серверной *БД* и получает результат, после чего интерпретирует его необходимым образом и представляет пользователю. Так как клиентскому приложению посылается результат выполнения запроса, по сети "путешествуют" только те данные, которые необходимы клиенту. В итоге снижается нагрузка на *сеть*. Поскольку выполнение запроса происходит там же, где хранятся данные (на сервере), нет необходимости в пересылке больших пакетов

данных. Кроме того, *SQL-сервер*, если это возможно, оптимизирует полученный *запрос* таким образом, чтобы он был выполнен в минимальное время с наименьшими накладными расходами [[3.2] , [3.3]]. *Архитектура* системы представлена на [рис. 3.3](#).

Все это повышает *быстродействие* системы и снижает *время ожидания* результата запроса. При выполнении запросов сервером существенно повышается степень безопасности данных, поскольку правила целостности данных определяются в базе данных на сервере и являются едиными для всех приложений, использующих эту *БД*. Таким образом, исключается возможность определения противоречивых правил поддержания целостности. Мощный аппарат транзакций, поддерживаемый *SQL-серверами*, позволяет исключить одновременное изменение одних и тех же данных различными пользователями и предоставляет возможность откатов к первоначальным значениям при внесении в *БД* изменений, закончившихся аварийно [[3.2] , [3.3]].

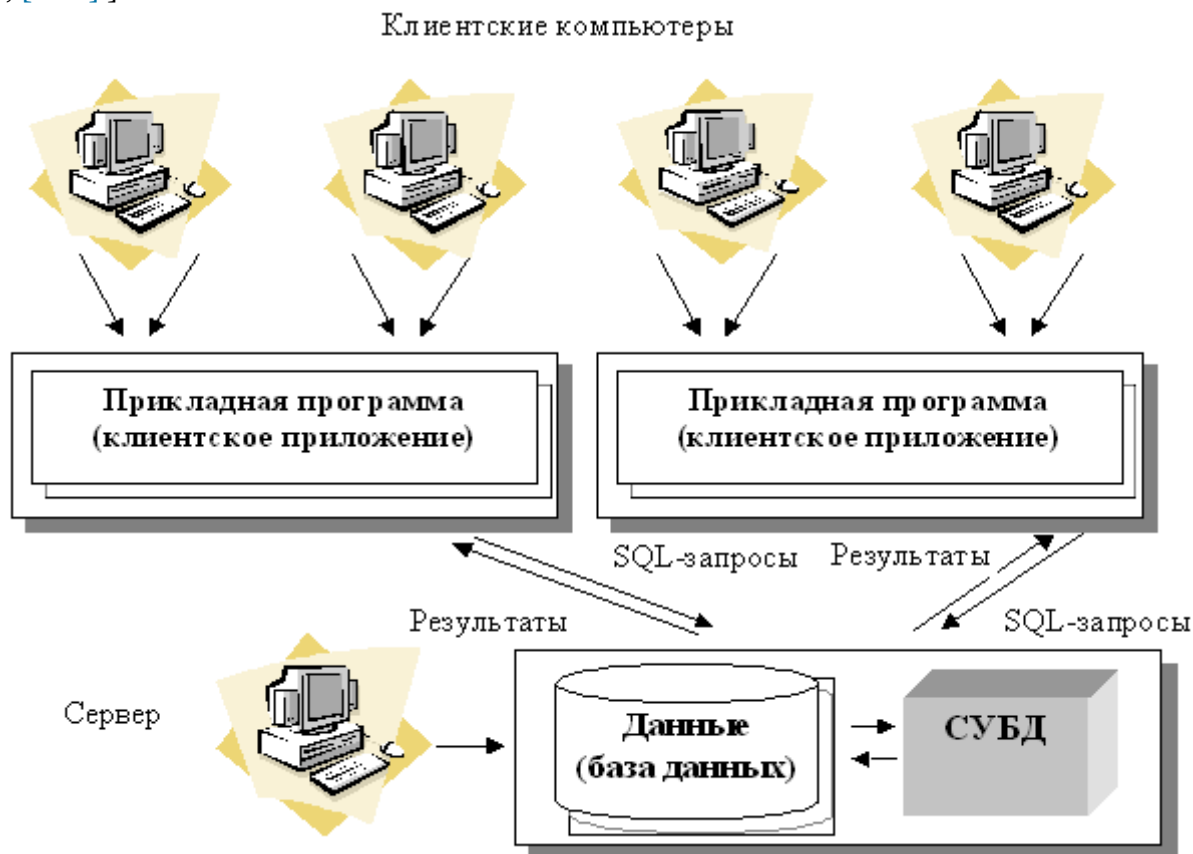


Рис. 3.3. Архитектура "клиент – сервер"

Итак, в результате работа построена следующим образом:

- База данных в виде набора файлов находится на жестком диске специально выделенного компьютера (сервера сети).
- СУБД располагается также на сервере сети.
- Существует локальная сеть, состоящая из клиентских компьютеров, на каждом из которых установлено клиентское приложение для работы с БД.
- На каждом из клиентских компьютеров пользователи имеют возможность запустить приложение. Используя предоставляемый приложением пользовательский интерфейс, он инициирует обращение к СУБД, расположенной на сервере, на выборку/обновление информации. Для общения используется специальный *язык запросов SQL*, т.е. по сети от клиента к серверу передается лишь текст запроса.
- СУБД инкапсулирует внутри себя все сведения о физической структуре БД, расположенной на сервере.
- СУБД инициирует обращения к данным, находящимся на сервере, в результате которых на сервере осуществляется вся обработка данных и лишь результат выполнения

запроса копируется на клиентский компьютер. Таким образом СУБД возвращает результат в приложение.

- Приложение, используя пользовательский интерфейс, отображает результат выполнения запросов.

Рассмотрим, как выглядит разграничение функций между сервером и клиентом.

- Функции приложения-клиента:
 - Посылка запросов серверу.
 - Интерпретация результатов запросов, полученных от сервера.
 - Представление результатов пользователю в некоторой форме (интерфейс пользователя).
- Функции серверной части:
 - Прием запросов от приложений-клиентов.
 - Интерпретация запросов.
 - Оптимизация и выполнение запросов к БД.
 - Отправка результатов приложению-клиенту.
 - Обеспечение системы безопасности и разграничение доступа.
 - Управление целостностью БД.
 - Реализация стабильности многопользовательского режима работы.

В архитектуре "клиент – сервер" работают так называемые "промышленные" СУБД. Промышленными они называются из-за того, что именно СУБД этого класса могут обеспечить работу информационных систем масштаба среднего и крупного предприятия, организации, банка. К разряду промышленных СУБД принадлежат MS SQL Server, Oracle, Gupta, Informix, Sybase, DB2, InterBase и ряд других [[3.2]].

Как правило, SQL-сервер обслуживается отдельным сотрудником или группой сотрудников (администраторы SQL-сервера). Они управляют физическими характеристиками баз данных, производят оптимизацию, настройку и *переопределение* различных компонентов БД, создают новые БД, изменяют существующие и т.д., а также выдают привилегии (разрешения на *доступ* определенного уровня к конкретным БД, SQL-серверу) различным пользователям [[3.2]].

Рассмотрим основные достоинства данной архитектуры по сравнению с архитектурой "файл-сервер":

- Существенно уменьшается сетевой трафик.
- Уменьшается сложность клиентских приложений (большая часть нагрузки ложится на серверную часть), а, следовательно, снижаются требования к аппаратным мощностям клиентских компьютеров.
- Наличие специального программного средства – SQL-сервера – приводит к тому, что существенная часть проектных и программистских задач становится уже решенной.
- Существенно повышается целостность и безопасность БД.

К числу недостатков можно отнести более высокие финансовые *затраты* на аппаратное и *программное обеспечение*, а также то, что большое количество клиентских компьютеров, расположенных в разных местах, вызывает определенные трудности со своевременным обновлением клиентских приложений на всех компьютерах-клиентах. Тем не менее, архитектура "клиент – сервер" хорошо зарекомендовала себя на практике, в настоящий момент существует и функционирует большое количество БД, построенных в соответствии с данной архитектурой.

3.4. Трехзвенная (многозвенная) архитектура "клиент – сервер".

Трехзвенная (в некоторых случаях *многозвенная*) архитектура (N-tier или multi-tier). представляет собой дальнейшее совершенствование технологии "клиент – сервер". Рассмотрев архитектуру "клиент – сервер", можно заключить, что она является 2-звенной: первое звено – клиентское *приложение*, второе звено – сервер БД + сама БД. В *трехзвенной* архитектуре вся бизнес-логика (деловая логика), ранее входившая в клиентские приложения, выделяется в отдельное звено, называемое сервером приложений. При этом клиентским

приложениям остается лишь пользовательский *интерфейс*. Так, в качестве клиентского приложения в описанном выше примере выступает Web-браузер.

Что улучшается при использовании *трехзвенной архитектуры*? Теперь при изменении бизнес-логики более нет необходимости изменять клиентские приложения и обновлять их у всех пользователей. Кроме того, максимально снижаются требования к аппаратуре пользователей.

Итак, в результате работа построена следующим образом:

- База данных в виде набора файлов находится на жестком диске специально выделенного компьютера (сервера сети).
- СУБД располагается также на сервере сети.
- Существует специально выделенный сервер приложений, на котором располагается программное обеспечение (ПО) делового анализа (бизнес-логика) [[3.1]].
- Существует множество клиентских компьютеров, на каждом из которых установлен так называемый "тонкий клиент" – клиентское приложение, реализующее интерфейс пользователя.
- На каждом из клиентских компьютеров пользователи имеют возможность запустить приложение – тонкий клиент. Используя предоставляемый приложением пользовательский интерфейс, он инициирует обращение к ПО делового анализа, расположенному на сервере приложений.
- Сервер приложений анализирует требования пользователя и формирует запросы к БД. Для общения используется специальный *язык запросов SQL*, т.е. по сети от сервера приложений к серверу БД передается лишь текст запроса.
- СУБД инкапсулирует внутри себя все сведения о физической структуре БД, расположенной на сервере.
- СУБД инициирует обращения к данным, находящимся на сервере, в результате которых результат выполнения запроса копируется на сервер приложений.
- Сервер приложений возвращает результат в клиентское приложение (пользователю).
- Приложение, используя пользовательский интерфейс, отображает результат выполнения запросов.

Тема 2.2. Краткий обзор СУБД

Многие авторы классифицируют *СУБД* на две большие категории: так называемые "настольные" и "серверные".

3.5.1. Настольные СУБД

Настольные СУБД используются для сравнительно небольших задач (небольшой объем обрабатываемых данных, малое количество пользователей). С учетом этого, указанные СУБД имеют относительно упрощенную архитектуру, в частности, функционируют в режиме файл-сервер, поддерживают не все возможные *функции СУБД* (например, не ведется журнал транзакций, отсутствует возможность автоматического восстановления базы данных после сбоев и т. п.). Тем не менее, такие системы имеют достаточно обширную область применения. Прежде всего, это государственные (муниципальные) учреждения, сфера образования, сфера обслуживания, малый и средний бизнес. Специфика возникающих там задач заключается в том, что объемы данных не являются катастрофически большими, частота обновлений не бывает слишком высокой, организация территориально обычно расположена в одном небольшом здании, количество пользователей колеблется от одного до 10–15 человек. В подобных условиях использование настольных СУБД для управления информационными системами является вполне оправданным, и они с успехом применяются.

Одними из первых СУБД были так называемые dBase-совместимые программные системы, разработанные разными фирмами. Первой широко распространенной системой такого рода была система dBase III – PLUS (фирма Achton-Tate). Развитый язык программирования, удобный интерфейс, доступный для массового пользователя,

способствовали широкому распространению системы. В то же время работа системы в режиме интерпретации обуславливала низкую производительность на стадии выполнения. Это привело к появлению новых систем-компиляторов, близких к системе dBase III – PLUS: *Clipper* (фирма Nantucket Inc.), *FoxPro* (фирма Fox Software), *FoxBase+* (фирма Fox Software), *Visual FoxPro* (фирма Microsoft). Одно время достаточно широко использовалась СУБД PARADOX (фирма Borland International).

В последние годы очень широкое распространение получила система управления базами данных Microsoft Access, которая входит в целый ряд версий пакета Microsoft Office (фирма Microsoft).

3.5.2. Серверные СУБД

Для крупных организаций ситуация принципиально меняется. Там использование *файл-серверных* технологий является неудовлетворительным по описанным выше причинам. Поэтому на передний край борьбы за автоматизацию выходят так называемые серверные СУБД.

Основными производителями таких систем обработки и хранения данных являются 3 корпорации: Oracle, Microsoft и IBM. Диаграмма соотношения объемов продаж соответствующих систем (источник: IDC Report, Май 2006) приводится на [рис. 3.4](#).

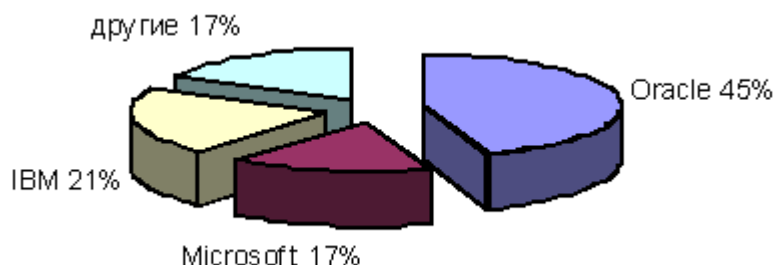


Рис. 3.4. Продажи ПО систем хранения данных в мире

Наиболее распространенными *клиент-серверными* системами здесь соответственно являются системы Oracle (разработчик компания Oracle), MS SQL Server (разработчик компания Microsoft), DB2, *Informix Dynamic Server* (компания IBM).

Дадим краткую характеристику этим системам.

MS SQL Server

К настоящему времени разработано несколько версий систем: MS SQL Server-2000, MS SQL Server -2005, MS SQL Server-2008. Приведем информацию о системе MS SQL Server-2008 с сервера Microsoft (<http://www.microsoft.com/rus/SQL/2008/default.mspx>)

Microsoft [®] SQL Server [™] 2008 - это законченное предложение в области баз данных и анализа данных для быстрого создания *масштабируемых решений* электронной коммерции, бизнес-приложений и хранилищ данных. Оно позволяет значительно сократить время выхода этих решений на рынок, одновременно обеспечивая масштабируемость, отвечающую самым высоким требованиям. В SQL Server включена поддержка языка XML и протокола HTTP, средства повышения быстродействия и доступности, позволяющие распределить нагрузку и обеспечить бесперебойную работу, функции для улучшения управления и настройки, снижающие совокупную стоимость владения.

Платформа бизнес-анализа SQL Server 2008, тесно интегрированная с Microsoft Office, предоставляет развитую масштабируемую инфраструктуру для внедрения мощных возможностей бизнес-анализа в рабочий процесс всех бизнес-подразделений вашей компании, открывая доступ к нужной бизнес-информации через знакомый интерфейс MS Excel и MS Word.

MS SQL Server-2008 поддерживает создание и работу с корпоративным хранилищем данных, объединяющим информацию со всех систем и приложений, позволяющим получить единую комплексную картину бизнеса вашей компании.

MS SQL Server-2008 предоставляет масштабируемый и высокопроизводительный "процессор данных" - для самых ответственных и требовательных бизнес-приложений, тем,

кому необходим высочайший уровень надежности и защиты, позволяя при этом снизить совокупную стоимость владения за счет расширенных возможностей по управлению серверной инфраструктурой.

MS SQL Server-2008 предлагает разработчикам развитую, удобную и функциональную среду программирования, включая средства работы с веб службами, инновационные технологии доступа к данным – все, что необходимо для эффективной работы с данными любых типов и форматов.

Отдельные аспекты MS SQL Server – 2008 будут описаны в лекциях "[Структура современной СУБД на примере Microsoft SQL Server 2008](#)" и "[Направления развития баз данных](#)" 14.

Oracle

К настоящему времени разработано несколько версий систем, каждая из которых включает целую линейку продуктов, например Oracle 8, Oracle 9i, Oracle 10g.

Соответствующие линейки продуктов включают как собственно СУБД (например Oracle Database 10g, Oracle Database 11g) , так и средства разработки и анализа данных.

Приведем информацию о системе с сервера Oracle (http://www.oracle.com/global/ru/mid/oracle_products/database.html).

Oracle предлагает комплексные, открытые, доступные и удобные в использовании технологические решения. Готовые пакетируемые решения автоматически включают в свою стоимость базу данных, сервер приложений, интеграционную платформу, инструменты аналитики и управления неструктурированными данными. Масштабируемые бизнес-приложения Oracle могут быть легко интегрированы с ИТ-инфраструктурой предприятия без потери уже вложенных в ИТ инвестиций.

СУБД Oracle Database 11g обеспечивает улучшенные характеристики за счет автоматизации задач администрирования и обеспечения лучших в отрасли возможностей по безопасности и соответствию нормативно-правовым актам в области защиты информации. Появилось больше функций автоматизации, самодиагностики и управления. Среди характеристик системы можно отметить управление большими объемами данных с использованием распределенных таблиц и компрессии, эффективную защиту данных, возможность полного восстановления, возможность интеграции геофизических данных медиа-контента в бизнес-процесс и т.д.

Серверы баз данных компании IBM

К настоящему времени разработаны линейки продуктов DB2 и *Informix*, включающие как собственно СУБД так и средства разработки и анализа данных (DB2 Universal Database DB2 Personal Edition, DB2 Enterprise 9 и др., а также *Informix* Dynamic Server, *Informix* Dynamic Server Express, *Informix* Extended Parallel Server и др.

Приведем информацию о части таких систем с сервера (<http://www-01.ibm.com/software/ru/data/?pgel=ibmhzn>)

Универсальный сервер баз данных DB2 Universal Database - это масштабируемая, объектно-реляционная система управления базами данных с интегрированной поддержкой мультимедиа и Web, работающая на системах от персональных компьютеров и серверов на процессорах Intel до Unix, от однопроцессорных систем до симметричных многопроцессорных систем (SMP) и систем с массовым параллелизмом (MPP), на хостах AS/400 и мейнфреймах. DB2 Universal Database объединяет в себе высокую производительность систем обработки транзакций в режиме on-line, объектно-реляционные расширения, усовершенствованные средства оптимизации с возможностями параллельной обработки и поддержкой очень больших баз данных. DB2 Universal Database также имеет новые встроенные средства для облегчения переноса на свою базу приложений, разработанных на других системах управления базами данных, таких как Oracle, Microsoft, Sybase и *Informix*. Помимо этого, DB2 Universal Database включает в себя дополнительные средства поддержки систем аналитической обработки в реальном времени (OLAP) и систем поддержки принятия решений, множество простых в использовании

расширений (DB2 extenders). DB2 Universal Database доступна на абсолютном большинстве ключевых платформ, что дает заказчикам ту гибкость, которая им необходима.

Кроме вышеуказанных зарубежных систем отметим и отечественную разработку – СУБД НИКА, преемницу широко распространенной в Советском Союзе СУБД ИНЕС для ЕС ЭВМ.

Краткие итоги. В лекции рассмотрены различные архитектурные решения, используемые при реализации *многопользовательских СУБД. Централизованная архитектура*. Технология с сетью и файловым сервером (*архитектура "файл-сервер"*). *Архитектура "клиент – сервер"* (распределенная модель вычислений). Трехзвенная (многозвенная) архитектура клиент – сервер. Дан обзор современных СУБД (настольные СУБД, серверные СУБД).

РАЗДЕЛ 3. Модели данных

Цели занятия.

Понятие модели данных. Объектные модели данных. Общая классификация моделей данных. Уровни моделирования баз данных. Общие и специальные критерии оценки качества логической и физической моделей данных. Основные принципы построения БД - 12 правил Кодда. Отношения в РБД. Их основные понятия. Соотношение основных понятий реляционного подхода. Ключи переменной отношения. Целостность реляционных данных. Функциональные зависимости между атрибутами в отношениях РБД. Связи в реляционных БД. Универсальное отношение. Избыточность данных. Аномалии.

4. Структура лекционного занятия.

№ п/п	Содержание (кратко)	Методы и средства обучения
1	Проектирование баз данных.	
2	Первая стадия концептуального проектирования базы данных (концептуальное моделирование). Вторая стадия концептуального проектирования (Модели данных СУБД. Представление концептуальной модели средствами модели данных СУБД).	1. Персональные компьютеры; 2. Средства доступа в Интернет; 3. Проектор.

5. Содержание лекционного занятия и взаимодействие с аудиторией.

Введение.

Формулирование темы занятия. Использование вводного примера, описание ситуации, демонстрация какого-либо процесса и т. д. (в зависимости от вида, замысла лекции и её структуры)

Тема 3.1. Различные представления о данных в базах данных. Основные этапы проектирования баз данных

Цель лекции: Показать существование различных представлений о данных (различных моделей) у разных групп лиц, работающих с данными. Рассмотреть отражение этих представлений в трехуровневой архитектуре *базы данных* (внешний уровень, *концептуальный уровень*, внутренний уровень), сформулировать достоинство трехуровневой архитектуры. Выделить основные *этапы проектирования базы данных* как процесса построения вышеуказанных моделей.

4.1. Различные представления о данных в базах данных

Создание *базы данных* предполагает интеграцию данных, предназначенных для решения нескольких прикладных задач разных пользователей. Соответственно, при интеграции данных должны учитываться требования к данным каждого пользователя, основанные на его представлении о данных и связях между ними. Далее эти требования должны обобщаться в единое *представление*, которое и будет служить основой для построения единой *базы данных* (рис. 4.1).

Обобщение представлений всех пользователей о данных называется **концептуальной моделью** (схемой) БД. **Концептуальная модель** представляет информационное описание предметной области с учетом логических взаимосвязей, поэтому её еще называют инфологической (информационно-логической) моделью. В модели отсутствуют какие-либо понятия, связанные с ЭВМ, памятью ЭВМ, способами размещения данных в памяти ЭВМ, и, по сути, это модель только *предметной области*.

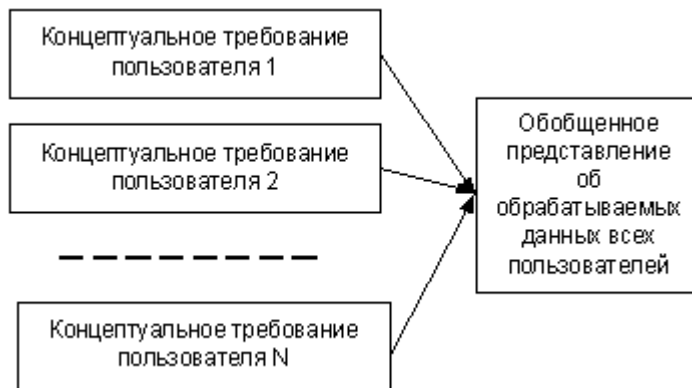


Рис. 4.1. Обобщение представления пользователей о данных

Как уже отмечалось, для создания *базы данных* и работы с ней используется система управления базами данных. Каждая конкретная СУБД поддерживает определенный вид данных (форматов записей и отношений), называемый *моделью данных СУБД*.

Следующий этап разработки *базы данных* предполагает выбор представления *концептуальной модели* с помощью модели данных конкретной СУБД. Полученное таким образом представление *концептуальной модели* называется *логической моделью БД*. Или другими словами, *логическая модель* – это **концептуальная схема, специфицированная в языке конкретной СУБД**. *Логическая модель* представляет данные и элементы данных вне зависимости от их содержания и среды хранения. Далее разработчик системы средствами СУБД отображает полученную логическую модель БД в *память ЭВМ* и определяет методы доступа. Полученное *представление данных* в памяти ЭВМ называется *внутренним представлением* или *структурой хранения*. Прикладные программы работают с логической моделью, причем каждому пользователю представляется *подмножество* этой логической модели (подсхема), отражающее его *представление о предметной области*. Каждая *прикладная программа* "видит" и обрабатывает только те данные, которые необходимы именно ей.

Соответствующее "видение" данных прикладными программами (пользователями) представляет собой **внешние представления**. Взаимосвязь вышеуказанных моделей изображена на рис. 4.2.

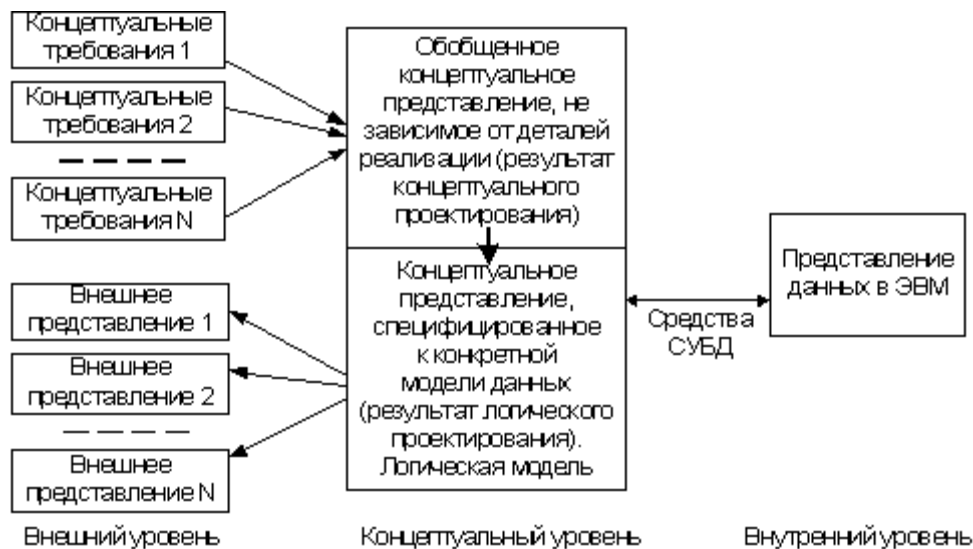


Рис. 4.2. Различные представления о данных в БД

На данной схеме выделены три различных уровня описания данных (внешний, концептуальный, внутренний). Эти уровни формируют так называемую *трехуровневую архитектуру ANSI/SPARC*, предложенную в 1975 г. Комитетом планирования стандартов и норм *SPARC* (Standards Planning and Requirements Committee) Национального института стандартизации США (American National Standards Institute – ANSI). Основная цель этой архитектуры состоит в отделении пользовательского представления о данных в базе данных от их физического представления. Использование таких представлений о данных позволяет обеспечить выполнение основного требования к БД – независимости программ и данных. При изменении прикладных программ может измениться соответствующее *внешнее представление*, *логическая модель* данных не изменяется и, соответственно, не будут изменяться другие прикладные программы. При изменении внутреннего представления (структур хранения) *логическая модель* не изменяется, соответственно, не изменяются прикладные программы.

Использование соответствующих представлений также позволяет четко разграничить полномочия различных лиц, работающих с базой данных.

Соответствующие представления позволяют описать "видение" *базы данных* разными лицами, работающими с ней:

- *внешнее представление* – представление специалиста предметной области (пользователя);
- *внешнее представление* и *логическая модель* – представление прикладного программиста, разрабатывающего конкретное приложение для пользователя;
- *логическая модель* и внутреннее представление – представление системного программиста, администрирующего базу данных.

4.2. Основные этапы проектирования базы данных

Проектирование данных (базы данных) представляет собой процесс последовательного отображения исследуемых явлений реального мира в виде данных в памяти ЭВМ (рис. 4.3).

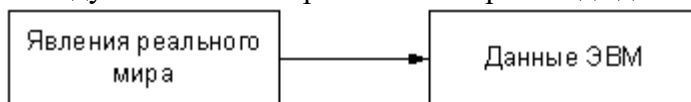


Рис. 4.3. Общая схема проектирования

Конкретные явления реального мира, представляющие интерес для проводимого исследования, будем называть *предметной областью*.

Проектирование (моделирование) базы данных представляет собой многоэтапный процесс. Основные этапы этого процесса приведены на рис. 4.4.).

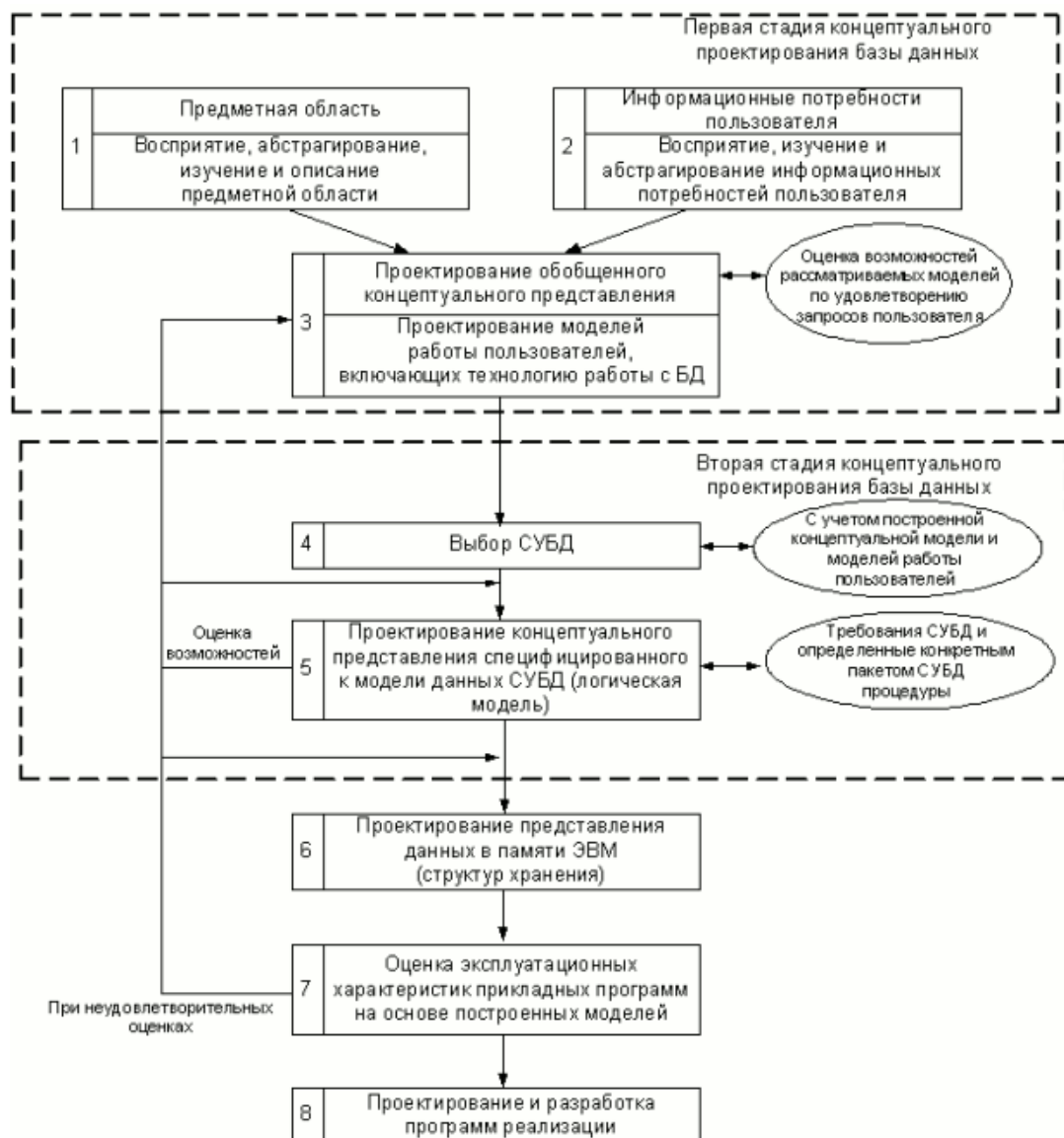


Рис. 4.4. Этапы проектирования базы данных

Подробно действия, отраженные на приведенном рисунке, будут рассмотрены в следующих лекциях. Здесь дадим лишь краткие комментарии к соответствующим блокам.

В блоках 1,2 необходимо особое внимание обратить на слово "абстрагирование". Имеется ввиду, что *проектирование базы данных* нужно вести не под конкретный документ, обрабатываемый пользователем, и не под конкретные действия пользователя с этим документом, а под обобщенный (абстрактный) образ документов и обобщенные (абстрактные) действия пользователей. Например, рассматривать документ не с конкретными числами строк и столбцов, а с абстрактными числами n и m ; вместо требуемого пользователем поиска по конкретному полю (например, фамилии) рассматривать *поиск* по любому полю и т.д. Это очень важно, так как конкретные формы документов и действия пользователей при работе с ними достаточно часто изменяются. В этом случае при *проектировании базы данных* под конкретные формы документов и конкретные действия придется перепроектировать базу данных, что связано с существенными временными и стоимостными затратами.

Очень важным является выбор *СУБД* (блок 4), от которого в значительной степени зависит работоспособность построенной *базы данных*. Проблема выбора *СУБД* уже обсуждалась в лекции 3. Заметим здесь, что выбор *СУБД* зависит от количества форм документов, от сложности связей между данными, от объема обрабатываемых данных, от количества пользователей, работающих с *БД* и т.д.

Ранее отмечалось, что *отображение* логической модели базы данных в структуру хранения (*представление данных* в памяти компьютера) осуществляется системой управления базой данных. Тем не менее, во многих СУБД для повышения эффективности функционирования *базы данных* представляется возможность выбора ряда параметров, управляющих представлением данных в памяти компьютера. Выбор таких параметров и подразумевается в блоке 6.

Заметим, что очень важно при *проектировании базы данных* делать оценки ее возможной работоспособности. Так, по завершении проектирования обобщенного концептуального представления нужно попытаться оценить необходимое число производимых операций с элементами моделей при реализации возможных запросов пользователей. При невозможности в рамках построенной модели ответить на какой-то *запрос* пользователя или при значительном числе производимых при этом операций (что приведет к невозможности реализации соответствующего запроса в реальном масштабе времени) необходим возврат по схеме [рис. 4.4.](#) на шаг назад (построение более эффективного обобщенного концептуального представления). Аналогичные оценки необходимо делать и при завершении других этапов проектирования (блоки 5, 7). При этом возможен возврат назад на один или несколько шагов. Так, например, при проектировании логической модели (блок 5) не удастся достичь адекватного представления *концептуальной модели* средствами *модели данных СУБД*. В этом случае необходимо либо вернуться на шаг назад и выбрать другую СУБД, либо вернуться к блоку 3 и изменить вид *концептуальной модели*. Так же, если полученные при реализации блока 7 оценки эксплуатационных характеристик не отвечают требованиям пользователя, возможны пересмотры всех ранее полученных решений (блоки 7, 6, 5, 4, 3). Кроме этого, необходим возврат на проектирование обобщенного концептуального представления при изменении внешних требований пользователей, а также при выявленных ошибках проектирования.

Краткие итоги: Рассмотрены различные представления о данных в базах данных - модели обрабатываемых данных (*внешнее представление, концептуальная модель, структура хранения*). Представлено отражение этих представлений в трехуровневой архитектуре *базы данных* (*внешний уровень, концептуальный уровень, внутренний уровень*), сформулировано достоинство трехуровневой архитектуры. Описаны основные *этапы проектирования базы данных* как процесса построения вышеуказанных моделей и *жизненный цикл проектирования базы данных* (создание, апробация, исправление ошибок и *улучшение характеристик, опытная эксплуатация*).

Тема 3.2. Первая стадия концептуального проектирования базы данных (концептуальное моделирование. Вторая стадия концептуального проектирования (Модели данных СУБД. Представление концептуальной модели средствами модели данных СУБД)

Цель лекции: показать, как описывается предметная область при концептуальном моделировании (с помощью каких понятий, средств представления и приемов построения) и как обеспечивается *достоверность* информации в базе данных за счет ограничений целостности концептуальной модели.

5.1. Описание информационного представления предметной области. ER-диаграмма

Иллюстрацию вводимых понятий и этапов проектирования *базы данных* будем проводить на примере близкой для читателя конкретной *предметной области: представление данных о студентах вуза*. Дадим краткое описание рассматриваемой *предметной области*. В вузе имеется несколько факультетов, на каждом из которых ведется подготовка по нескольким специальностям или направлениям. Для каждой специальности на факультете есть свой учебный план, в котором приводится перечень изучаемых учебных курсов с указанием количества часов занятий. Студенты изучают соответствующие дисциплины, сдают экзамены и зачеты, получают оценки.

Чаще всего концептуальная модель представляется в виде диаграммы сущностей – связей (entity – relationship) или ER-диаграммы. Процесс построения ER-диаграммы называется ER-моделированием.

Введем основные понятия, с помощью которых описывается предметная область.

Сущность (Entity) или объект – то, о чем будет накапливаться информация в информационной системе (нечто такое, за чем пользователь хотел бы наблюдать).

Если в системе обрабатывается информация о факультетах, сущностью будет являться факультет, если о студентах, сущность – студент и т.п.

Имя сущности при ER-моделировании, как правило, записывается заглавными буквами. Каждая сущность обладает определенным набором свойств (рассматриваем только свойства, представляющие интерес для пользователей в рамках проводимого исследования), которые запоминаются в информационной системе. Так, например, в качестве свойств сущности ФАКУЛЬТЕТ можно указать номер факультета, название факультета, в качестве свойств сущности СТУДЕНТ можно указать фамилию, дату рождения, место рождения, в качестве свойств сущности ЭКЗАМЕН – предмет, дату проведения экзамена, экзаменаторов.

Для информационного описания сущности вводится понятие атрибута.

Атрибут – поименованное свойство (характеристика) сущности. Атрибут представляет собой информационное отображение свойства сущности и принимает конкретное значение из множества допустимых значений. Так, например, для сущности ФАКУЛЬТЕТ атрибут "название" у конкретного экземпляра сущности принимает конкретное значение "вычислительной математики и кибернетики". Таким образом, атрибут представляет информационное описание количественных или качественных свойств сущности, описывает состояние сущности, позволяет идентифицировать сущность. Информация о сущности представляется совокупностью атрибутов. Такую совокупность атрибутов часто называют записью об объекте.

Совокупность сущностей, характеризующихся в информационной системе одним и тем же перечнем свойств, называется классом сущностей (набором объектов). Так, например, совокупность всех сущностей СТУДЕНТ составляет класс сущностей СТУДЕНТ, совокупность всех сущностей ФАКУЛЬТЕТ составляет класс сущностей ФАКУЛЬТЕТ. Класс сущностей описывается перечнем свойств сущностей, составляющих этот класс.

Экземпляр сущности будем называть конкретной сущностью (сущность с конкретными значениями соответствующих свойств). Выше мы определили сущность как то, о чем будет накапливаться информация в информационной системе. Это только одна сторона. Информация должна не просто храниться сама по себе, а использоваться для удовлетворения информационных потребностей пользователя. Для реализации подавляющего числа запросов пользователю прежде всего необходимо найти интересующий его экземпляр сущности (с целью обработки, корректировки, удаления). Поэтому важнейшим свойством сущности является однозначная идентификация ее экземпляров по одному или группе атрибутов (уникальному идентификатору). У сущности ФАКУЛЬТЕТ это, например, номер факультета, у сущности СТУДЕНТ это может быть атрибут "фамилия", если у всех студентов разные фамилии, группа атрибутов "фамилия", "имя", "отчество", или специально введенный уникальный идентификатор, например дополнительно введенный атрибут "код студента".

Наиболее распространенным способом представления концептуальной модели является так называемая ER-диаграмма. В разных источниках используются разные системы обозначений в ER-диаграммах. На практике использование различных способов записи ER-диаграмм не представляет особой сложности – беглое ознакомление с соответствующим разделом документации позволяет быстро освоить используемую систему обозначений. В данном пособии в ER-диаграмме класс сущностей будем представлять в виде четырехугольника. В четырехугольнике записано уникальное имя класса сущности (прописными буквами) и имена атрибутов строчными буквами.

Пример класса сущностей СТУДЕНТ и конкретного экземпляра сущности показан на рис. 5.1

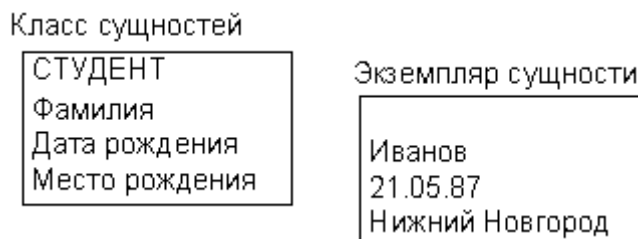


Рис. 5.1. Класс сущностей и экземпляр сущности

Для реализации информационных потребностей пользователя недостаточно найти интересующий его *экземпляр сущности*. Информационные потребности тесно связаны с функциональными взаимоотношениями, существующими в организации (например, необходимо определить, на каком факультете учиться конкретный студент). Для реализации таких запросов (информационных потребностей пользователя) используются существующие в *предметной области* взаимоотношения между сущностями. **Соответствующие взаимоотношения сущностей выражаются связями (Relationships)**. Различают классы связей и экземпляры связей. **Классы связей – это взаимоотношения между классами сущностей, а экземпляры связи – взаимоотношения между экземплярами сущностей.**

Класс связей может затрагивать несколько *классов сущностей*. Число *классов сущностей*, участвующих в связи, называется *степенью связи* $n = 2, 3, \dots$. Так, например, *класс сущностей* СТУДЕНТ связан с *классом сущностей* ФАКУЛЬТЕТ связью "учится на факультете". Степень этой связи равна двум. При $n=2$ связь называется бинарной. Заметим, что *связь* нужно рассматривать как двустороннюю: "студент учится на факультете" и "на факультете учатся студенты". Рассмотрим классификацию *бинарных связей*. В зависимости от того, сколько *экземпляров сущности* одного класса связаны со сколькими *экземплярами сущности* другого класса, различают следующие *типы связей*:

- **Связь 1:1.** Одиночный экземпляр сущности одного класса связан с одиночным экземпляром сущности другого класса. Примером является *связь* между классами сущностей ФАКУЛЬТЕТ и УЧЕБНЫЙ ПЛАН ПО СПЕЦИАЛЬНОСТИ ДЛЯ ФАКУЛЬТЕТА (каждому факультету соответствует свой учебный план по специальности или направлению).
- **Связь 1:M.** Единый экземпляр сущности одного класса связан со многими *экземплярами сущности* другого класса. Примером является *связь* между классами сущностей ФАКУЛЬТЕТ и СТУДЕНТ (на одном факультете учатся много студентов).
- **Связь M:N.** Несколько *экземпляров сущности* одного класса связаны с несколькими *экземплярами сущности* другого класса. Примером является *связь* между классами сущностей ФАКУЛЬТЕТ и СПЕЦИАЛЬНОСТЬ (на факультете может быть несколько специальностей и одна и та же специальность может быть на нескольких факультетах).

Числа, описывающие *типы бинарных связей* (1:1, 1:M, M:N), обозначают максимальное количество сущностей на каждой стороне связи. Эти числа называются *максимальными кардинальными числами*, а соответствующая пара чисел называется *максимальной кардинальностью*.

В данном пособии на *ER-диаграммах* связи между сущностями будем обозначать стрелками, рядом со стрелками указываем *имя связи*, а также *тип связи*. Пример *ER-диаграммы*, представляющей сущности СТУДЕНТ, ФАКУЛЬТЕТ, СПЕЦИАЛЬНОСТЬ и их взаимосвязи приводится на [рис. 5.2](#).

Напомним, что каждый *экземпляр сущности* должен уникально идентифицироваться (иметь *уникальный идентификатор*). Так как могут быть несколько студентов с одинаковой фамилией, введем дополнительный *атрибут* "код студента". У сущностей ФАКУЛЬТЕТ и СПЕЦИАЛЬНОСТЬ *атрибут* "номер" является *уникальным идентификатором*.



Рис. 5.2. Пример фрагмента ER-диаграммы

Заметим, что по этой ER-диаграмме можно указать последовательность действий, производимых при реализации запроса пользователей. Например, для реализации запроса "на каком факультете учится студент Иванов" необходимо выполнить следующие действия: найти среди экземпляров сущности СТУДЕНТ экземпляр с фамилией Иванов, перейти по связи "Студент учится на факультете" к экземпляру сущности ФАКУЛЬТЕТ, значение атрибута "Название" этого экземпляра и есть искомое название факультета. Отметим также, что иногда на ER-диаграммах две связи между сущностями изображают одной двухсторонней стрелкой или просто линией. Заметим, что на приведенной ER-диаграмме не представлены какие-либо способы реализации этих связей (на логическом и, тем более, на физическом уровнях). Соответствующие способы реализации связей зависят от возможностей модели данных конкретной СУБД и будут рассмотрены в следующей лекции ("Вторая стадия концептуального проектирования (Модели данных СУБД. Представление концептуальной модели средствами модели данных СУБД)") на второй стадии концептуального проектирования при представлении концептуальной модели средствами модели данных СУБД.

5.2. Построение концептуальной модели в виде ER-диаграммы

5.2.1 Основные этапы построения

Как уже отмечалось, концептуальная модель представляет собой обобщение представлений разных пользователей о данных. В связи с этим построение концептуальной модели, как правило, происходит в два этапа. На первом этапе производится сбор и анализ характеристик данных и строятся так называемые модели локальных представлений (локальные модели). Чаще всего локальная модель отражает представление отдельного пользователя (отдельной функциональной задачи). Иногда такая модель может описывать и некоторую независимую область данных нескольких функциональных задач (нескольких приложений). Здесь необходимо отметить, что моделирование представлений отдельных пользователей приводит к снижению уровня интеграции данных, а моделирование совместных представлений группы пользователей – к повышению сложности проектирования. В связи с этим при выборе области данных для локального моделирования приходится выбирать компромиссное решение между вышеуказанными вариантами.

При разработке концептуальной модели, прежде всего, следует определить сущности. С этой целью нужно сделать следующее:

- необходимо понять, какая информация должна храниться и обрабатываться и можно ли это определить как сущность;

- присвоить этой сущности имя;
- выявить *атрибуты сущности* и присвоить им имя;
- определить *уникальный идентификатор сущности*.

Выявив сущности, необходимо определить, какие связи имеются между ними.

При определении связей (естественно, рассматриваем только те связи, которые имеют отношение к решаемым задачам обработки данных) необходимо учитывать следующее:

- то, как экземпляр одной сущности связан с экземпляром другой сущности;
- то, как должны быть установлены связи, чтобы была возможность ответа на все запросы пользователей (исходя из их информационных потребностей).

Далее необходимо присвоить связям имена и определить тип связей.

На втором этапе построенные локальные модели объединяются в обобщенную концептуальную модель.

5.2.2. Моделирование локальных представлений

Прежде всего, необходимо отметить, что построенная модель должна удовлетворять ряду требований:

- адекватно отражать представление пользователя о данных;
- давать возможность ответа на возможные запросы пользователя, причем делать это с минимальными затратами по количеству просматриваемых сущностей;
- представлять данные с минимальным дублированием.

Процесс построения модели, удовлетворяющей указанным требованиям, является творческим, и формализовать его, как правило, невозможно. Тем не менее можно указать некоторые способы порождения вариантов при моделировании. Выбор одного из таких вариантов на основе оценок объемов дублирования и числа просматриваемых объектов при ответах на запросы пользователей позволяет улучшить эксплуатационные характеристики проектируемой базы данных.

Вариативность моделирования обуславливается неоднозначностью выбора сущностей, атрибутов и связей. В одном варианте можно что-то взять за сущность, в другом варианте это же можно взять за атрибут (несколько атрибутов), в третьем варианте это можно определить как *связь*. Так, например, ранее мы определили сущность ФАКУЛЬТЕТ с атрибутами "номер факультета", "название факультета". Введем сущность КАФЕДРА с атрибутами "номер кафедры", "название кафедры". Между этими сущностями есть *связь* "факультет состоит из кафедр". Возможен другой вариант, в котором вышеуказанная *связь* представляется через *атрибуты сущности* (у сущности ФАКУЛЬТЕТ введем дополнительные атрибуты, представляющие номера и названия всех кафедр этого факультета).

После того как выбран рациональный вариант локальной модели, производится редактирование введенных наименований сущностей, атрибутов и связей. Здесь выполняются следующие действия:

- устраняются расплывчатые наименования (все наименования должны однозначно пониматься каждым пользователем);
- устраняются синонимы (различные наименования одного и того же понятия);
- устраняются омонимы (одно и то же наименование разных понятий).

Эти действия, вообще говоря, носят итерационный характер, т.к. после их выполнения вновь могут возникать и расплывчатые наименования, и синонимы, и омонимы.

5.2.3. Объединение локальных моделей

На этом этапе ранее построенные модели *локальных представлений* отдельных пользователей (или групп пользователей) объединяются в единую концептуальную модель. Объединение локальных моделей производится следующими путями:

- слияние идентичных элементов;
- установление связей между наборами сущностей разных моделей;

- введение новых агрегированных элементов для представления связей между элементами разных моделей;
- обобщение различных подобных *типов сущностей*, позволяющее трактовать эти сущности как одну обобщенную сущность.

Рассмотрим каждый из этих путей.

Слияние идентичных элементов

Два или более элементов модели идентичны, если они имеют одинаковое смысловое значение.

Объединение моделей с идентичными элементами осуществляется путем "слияния" этих элементов в один. Два набора сущностей СПЕЦИАЛЬНОСТЬ в модели 1 и 2 имеют одинаковое смысловое значение ([рис. 5.3.](#)), и могут быть заменены одним набором сущностей ([рис. 5.4.](#)).

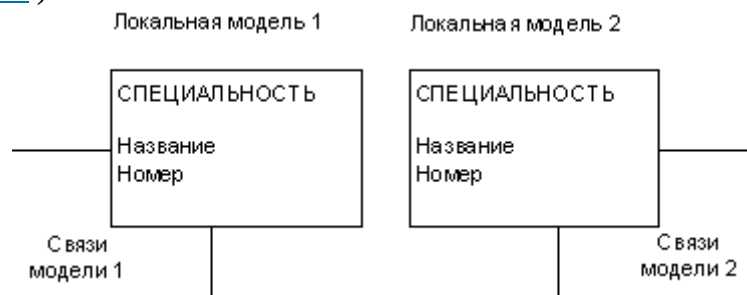


Рис. 5.3. Модели с идентичным элементом

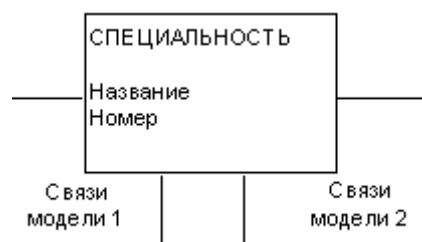


Рис. 5.4. Объединенная модель

Установление связей между наборами сущностей разных моделей

При рассмотрении наборов сущностей объединяемых моделей необходимо выявление связей между ними, т.к. именно эти связи и определяют в конечном итоге интегрированную базу данных.

Введение агрегированных элементов

При объединении моделей связь между элементами разных моделей может рассматриваться как новый элемент.

Рассмотрим в качестве примера моделирование информационного представления сдачи студентом экзаменов. Можно выделить ряд *локальных представлений* ([рис. 5.5.](#)).

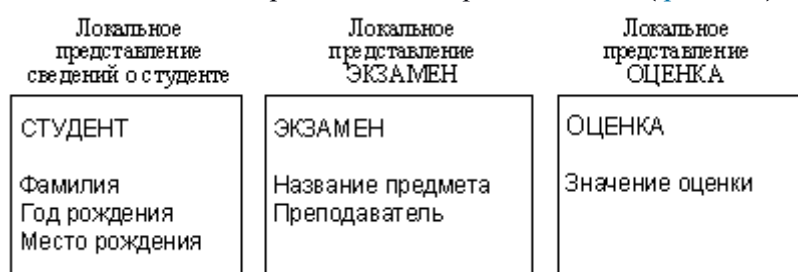


Рис. 5.5. Локальные представления

Объединяя *локальные представления*, устанавливаем новые связи ([рис. 5.6.](#)).

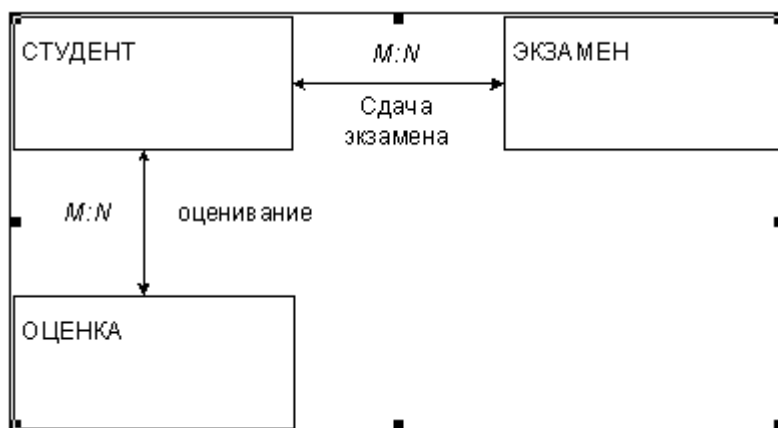


Рис. 5.6. Объединение локальных представлений

Как уже отмечалось, одним из показателей "зрелости" модели является возможность ответа на запросы пользователей, и установление связей преследует именно эту цель. Нетрудно видеть, что какие бы связи в рассматриваемой модели ни вводились, невозможно ответить на запрос "какую оценку получил студент А по дисциплине В". В таком случае необходимо использовать принцип агрегации – необходимую *связь* между элементами модели ввести как некоторый новый элемент. В данном примере можно определить этот новый агрегированный элемент как ЭКЗАМЕН СТУДЕНТА (рис. 5.7.).

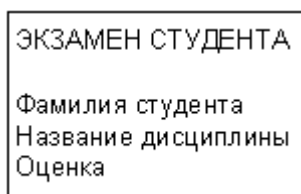


Рис. 5.7. Агрегированный элемент

Далее процесс объединения локальных моделей продолжается обычным образом. Обобщение подобных типов сущностей

Рассмотрим локальные модели разных факультетов, например – модель факультета вычислительной математики и кибернетики (ВМК), модель экономического факультета и так далее. В локальную модель факультета ВМК входят сущности СПЕЦИАЛЬНОСТИ ФАКУЛЬТЕТА ВМК и СТУДЕНТЫ ФАКУЛЬТЕТА ВМК, в локальную модель экономического факультета входят, соответственно, сущности СПЕЦИАЛЬНОСТИ ЭКОНОМИЧЕСКОГО ФАКУЛЬТЕТА и СТУДЕНТЫ ЭКОНОМИЧЕСКОГО ФАКУЛЬТЕТА (рис. 5.8.).

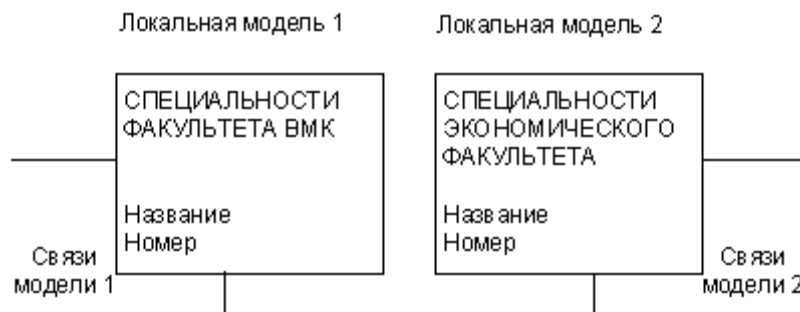


Рис. 5.8. Модели с подобным элементом

Два набора сущностей СПЕЦИАЛЬНОСТИ ФАКУЛЬТЕТА ВМК и СПЕЦИАЛЬНОСТИ ЭКОНОМИЧЕСКОГО ФАКУЛЬТЕТА в моделях 1 и 2 имеют одинаковое смысловое значение и могут быть заменены одним набором сущностей с добавлением нового атрибута – название факультета (рис. 5.9.).

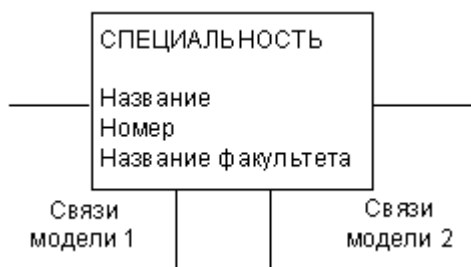


Рис. 5.9. Пример обобщенной сущности

Отметим, что в данном случае подобным образом можно слить и все остальные сущности локальных моделей факультетов, так как сущности **СТУДЕНТЫ ЭКОНОМИЧЕСКОГО ФАКУЛЬТЕТА** и **СТУДЕНТЫ ВМК** также имеют одинаковое смысловое значение и их также можно объединить. Однако в общем случае каждая локальная модель может содержать сущности и связи, которых нет в других локальных моделях.

Рассмотрим другой пример. Предположим, что мы храним данные о студентах (фамилия, имя, отчество, курс, группа) и о преподавателях (фамилия, имя, отчество, кафедра, должность). Соответственно, в предметной области выделяем две сущности – **СТУДЕНТ** и **ПРЕПОДАВАТЕЛЬ**.

Эти разные сущности можно в некоторых случаях трактовать как подобные. Для обобщения соответствующих сущностей необходимо, прежде всего, обобщить их атрибуты. Заметим, что атрибуты "Фамилия, Имя, Отчество" у обеих сущностей совпадают, атрибуты "Кафедра" и "Курс", "Группа" показывают место работы (учебы) и их можно заменить обобщенным атрибутом "Место работы (учебы)". Атрибут "Должность" можно использовать и у сущности **СТУДЕНТ**, если в качестве значения соответствующего атрибута использовать значение "студент". Тогда две сущности **ПРЕПОДАВАТЕЛЬ** и **СТУДЕНТ** можно трактовать как подобные и заменить их на обобщенную сущность. Дадим этой обобщенной сущности название **КАДРОВАЯ ЕДИНИЦА** (рис. 5.10.).

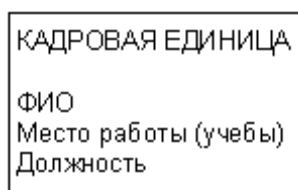


Рис. 5.10. Пример обобщенной сущности

У студента атрибут "Место работы (учебы)" будет принимать значение соответствующее атрибутам "Курс. Группа", у преподавателя – название кафедры. Обобщенная модель представлена на рис. 5.11.

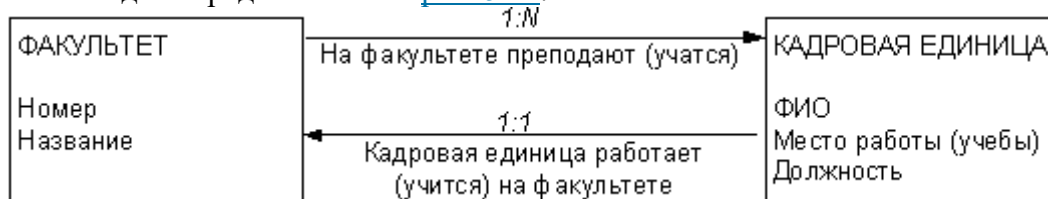


Рис. 5.11. Обобщенная модель

В этом случае почти в два раза упрощается структура концептуальной модели, и соответственно, структура базы данных. Для работы с данными о преподавателях и студентах достаточно одного набора программ. Таким образом, обобщение подобных типов объектов может существенно сократить последующие затраты на программирование.

В процессе объединения локальных представлений, как и при локальном моделировании, производится редактирование наименований (т.к. здесь появляются новые наименования). Процесс объединения также носит итерационный характер и продолжается до тех пор, пока не будут интегрированы все представления, согласованы и устранены все противоречия, отредактированы все наименования. Полученное в результате

объединения *локальных представлений* обобщенное представление и является концептуальной моделью.

5.3. Ограничения целостности

Под целостностью *базы данных* понимается то, что в ней содержится полная, непротиворечивая и адекватно отражающая предметную область (правильная) *информация*.

Огромный объем данных, вводимых в базу данных (причем разные данные могут вводиться разными пользователями), обуславливает большое число ошибок ввода (занесения). Заметим, что при традиционной "бумажной" обработке информации также достаточно часто встречаются данные, записанные неверно. Но человек, работая с определенными данными, неявно использует для контроля имеющиеся у него представления об этих данных. Например, сотрудник отдела кадров, увидев в карточке работника год рождения 1693, сразу заметит эту ошибку и предположит, что просто переставлены две цифры и реальный год рождения 1963. То есть в представлениях сотрудника заключены некоторые логические ограничения на данные. Очевидно, что для контроля правильности вводимых данных при работе с базой данных целесообразно сформировать и использовать ограничения.

Соответствующие ограничения обычно разделяют на 3 группы: внешние, специально конструируемые и внутренние. К *предметной области* относятся первые две группы, которые мы кратко охарактеризуем в этом подразделе. Внутренние ограничения относятся уже к модели данных и будут рассматриваться в разделе, посвященном модели данных.

Внешние ограничения

Эти ограничения связаны с адекватностью отражения предметной области. Например, сотрудник организации не может быть моложе 17 и старше 90 лет. Соответствующее ограничение на год рождения (GR) можно записать следующим образом:

Текущий год – 17 > GR > Текущий год – 90.

Одним из способов задания таких ограничений является перечисление конечного множества допустимых значений какого-либо атрибута (так называемый "перечислимый" тип данных). Например, должность преподавателя в вузе может принимать одно из следующих значений: профессор, доцент, старший преподаватель, преподаватель, ассистент. Вводимое значение должности для конкретного экземпляра, не совпадающее с одним из перечисленных значений, является ошибкой.

Ограничения, описанные с помощью специальных конструкций

Например, в базу данных вуза вводятся данные о числе студентов и преподавателей. По нормативным документам задано конкретное *значение отношения* числа студентов к числу преподавателей. Проверку этого отношения можно использовать для контроля достоверности данных. Такие конструкции строятся исходя из специфики данных рассматриваемой предметной области. Можно, например, построить много конструкций следующего вида: сумма значений по заданному атрибуту по всем *экземплярам сущностей* должна совпадать со значением определенного атрибута в экземпляре другой сущности.

Таким образом, на стадии ER-моделирования для повышения достоверности данных необходимо сформулировать соответствующие ограничения на данные. В идеальном случае каждое значение атрибута должно каким-то образом контролироваться. Использование этих ограничений позволяет существенно повысить достоверность данных в базе данных.

Краткие итоги. Рассмотрен процесс моделирования *предметной области*. Определены используемые при этом основные понятия (сущность, *атрибут*, *идентификатор*, *связь*, *типы связей*, *ER-диаграмма*).

Рассмотрены основные этапы моделирования сущностей и связей (*моделирование локальных представлений*, *объединение* локальных моделей с использованием понятий *идентичность*, *агрегация*, *обобщение*).

Дано понятие *ограничений целостности*, имеющих непосредственное *отношение* к *предметной области* (*внешние ограничения*; ограничения, описанные с помощью специальных конструкций).

Более подробно с материалом этой лекции можно познакомиться в [[1.1](#)], - [[5.10](#)].

РАЗДЕЛ 4. Базисные операции с реляционными данными

Цели занятия.

Специальные подходы к выполнению операций над множествами. Реляционная алгебра. Операции над отношениями. Теоретико-множественные операции над отношениями. Специальные реляционные операции. Реляционное исчисление.

4. Структура лекционного занятия.

№ п/п	Содержание (кратко)	Методы и средства обучения
1	Правила применения и принципы организации РБД. Использование формального аппарата для оптимизации схем отношений.	1. Персональные компьютеры; 2. Средства доступа в Интернет; 3. Проектор.

5. Содержание лекционного занятия и взаимодействие с аудиторией.

Введение.

Формулирование темы занятия. Использование вводного примера, описание ситуации, демонстрация какого-либо процесса и т. д. (в зависимости от вида, замысла лекции и её структуры)

Тема 5.1. Использование формального аппарата для оптимизации схем отношений

Тема 4.1. Формализация реляционной модели

Цель лекции: дать общее представление о модели данных СУБД как средства для представления концептуальной модели при создании базы данных, рассмотреть типовые модели данных (*сетевая модель, иерархическая модель, реляционная модель, многомерная модель*), показать как представляется *концептуальная модель* в разных СУБД, рассмотреть основные принципы работы средств *автоматизированного проектирования баз данных*.

6.1. Представление концептуальной модели средствами модели данных СУБД

Общие представления о моделях данных СУБД

В соответствии с основными этапами проектирования базы данных после построения концептуальной модели выбирается *система управления базой данных*, с помощью которой будет организована база данных и работа с ней. Каждая СУБД поддерживает определенные виды и типы данных, а также средства представления связей между данными, составляющими *модель данных СУБД*. Вторая стадия проектирования базы данных состоит в представлении построенной на предыдущей стадии концептуальной модели средствами *модели данных СУБД* или в отображении концептуальной модели в *модель данных СУБД*. Этот этап часто называют *логическим проектированием базы данных*. Полученная при этом модель часто также называется концептуальной моделью или схемой (но специфицированной к понятиям *модели данных СУБД*). В некоторых источниках полученную модель называют *логической структурой данных* или *моделью данных базы данных*.

Можно по-разному характеризовать понятие *модели данных СУБД*. С одной стороны, *модель данных СУБД* – это способ структурирования данных, которые рассматриваются как некоторая абстракция в отрыве от предметной области. С другой

стороны, модель данных СУБД – это инструмент представления концептуальной модели предметной области и динамики ее изменения в виде базы данных.

Учитывая обе вышеуказанные стороны, определим основные структуры моделей данных СУБД, используемые для представления концептуальной модели предметной области (сущностей, атрибутов, связей).

Элемент данных (поле) – наименьшая поименованная единица данных. Используется для представления значения атрибута.

С элементом данных неразрывно связано понятие "тип данных", который может принимать соответствующее поле. В разных СУБД могут использоваться разные типы данных, наиболее распространенными из которых (используемые во многих СУБД) являются следующие: числовой (*numeric*), символьный (*char*), дата (*date*) и т.д.

Запись – поименованная совокупность полей. Используется для представления совокупности атрибутов сущности (записи о сущности).

Экземпляр записи – запись с конкретными значениями полей.

Первичный ключ – минимальный набор полей записи, однозначно идентифицирующий экземпляр записи файла.

Файл – поименованная совокупность экземпляров записей одного типа. Используется для представления однородного набора сущностей.

Набор файлов – поименованная совокупность файлов, обрабатываемых в системе. Используется для представления нескольких наборов сущностей.

Введем понятие "группа", обобщающее понятия "файл" и "запись".

Группа – это поименованная совокупность элементов данных и других групп.

Важнейшим понятием концептуальной модели является понятие связи между сущностями (наборами сущностей). В моделях данных СУБД соответствующее понятие отражается понятием " **групповое отношение** ".

Групповое отношение – поименованное бинарное отношение, заданное на двух множествах экземпляров рассматриваемых групп. По характеру бинарных связей различают групповые отношения вида 1:1, 1:M, M:1, M:N. Пары чисел называют коэффициентами группового отношения. В групповом отношении один член группы назначается владельцем отношения, другой – членом.

База данных – поименованная совокупность экземпляров групп и групповых отношений.

Для представления группового отношения используется две формы:

а) **Графовая**. Группы изображаются вершинами графа, связи между группами – дугами, направленными от группы-владельца к группе-члену с указанием имени отношения и коэффициента.

По типу графов различают:

- иерархическую модель (граф без циклов – дерево) ;
- сетевую модель (ориентированный граф общего вида).

б) **Табличная**. Связь между группами изображается таблицей, столбцы которой представляют ключи соответствующих групп. Для формального описания таблицы используется математическое (теоретико-множественное) понятие отношения. Соответствующая модель данных называется реляционной моделью.

Модель данных СУБД описывается следующим образом:

- определены возможные типы и характеристики логических структур данных (полей, записей, файлов);
- заданы правила составления структур более общего типа из структур более простых типов (например, записей из полей, файлов из записей и т.д.);
- определен способ представления связей (отношений) между файлами и записями с помощью дополнительных полей ;
- определены возможные действия над структурами и правила их выполнения, включающие:
 - основные элементарные операции над данными;

- обобщенные операции (процедуры);
- средства контроля относительно простых условий корректности операций добавления, обновления или удаления данных (ограничения), реализуемые автоматически запускаемыми при выполнении вышеуказанных операций специальными процедурами (триггерами);
- средства контроля сколь угодно сложных условий корректности выполнения определенных действий (правила);
- специальный класс процедур (триггеры).

В качестве основных элементарных операций обычно рассматриваются следующие: *поиск* записи с заданным значением ключа, чтение нужной записи, добавление записи, корректировка, удаление. В *моделях данных СУБД* также предусматриваются специальные *операции* для установления групповых отношений.

Обобщенные *операции* или процедуры – последовательность операций, реализующая определенный *алгоритм* обработки данных. Процедуры могут инициироваться *СУБД* автоматически, а также могут запускаться пользователем. Примерами процедур являются процедуры копирования *БД*, восстановления *БД*, процедуры, вычисляющие значения определенных атрибутов в *БД* по значениям других атрибутов, и т.п.

Средства контроля используются для реализации ограничений целостности концептуальной модели. Простейшие средства контроля – ограничения – используются для реализации как *внешних ограничений* концептуальной модели, так и внутренних ограничений модели данных. В качестве последних ограничений, в частности, реализованы ограничения на ввод данных *несоответствующего типа*, несоответствующей характеристики (*по числу битов, по числу полей, по количеству записей* и т.п.). Более сложные средства контроля (правила) позволяют вызывать выполнение определенной последовательности операций (сколь угодно сложной) при изменении или добавлении данных в *БД* и тем самым реализовывать ограничения целостности, описанные с помощью специальных конструкций.

Построение модели данных базы данных (отображение концептуальной модели в модель данных СУБД)

После первой стадии концептуального проектирования у нас сформировано обобщенное *представление* пользователей о данных, как правило, представленное в виде *ER-диаграммы*. На следующей стадии (после того, как выбрана определенная *СУБД* с конкретной моделью данных) необходимо записать концептуальную схему в терминах и понятиях выбранной *СУБД*. На этой стадии каждая сущность концептуальной модели описывается как *запись*, состоящая из полей. Каждый *атрибут* описывается как *поле* с типом и характеристиками, возможными в выбранной *СУБД*. Описываются связи концептуальной модели в понятиях, соответствующих выбранной *СУБД*, определяется порядок реализации запросов пользователей к базе данных с помощью типовых операций *СУБД* и т. д.

Результатом этой стадии проектирования будет *концептуальная модель*, специфицированная к конкретной *СУБД*.

6.2 Типовые модели данных СУБД и представление концептуальной модели

6.2.1. Сетевая модель данных

Это одна из наиболее ранних моделей данных СУБД. Типовая *сетевая модель* данных была предложена рабочей группой по базам данных (Data Base Task Group – DBTG) системного комитета CODASYL (*Conference of Data System Languages*), основными функциями которого были анализ известных фирменных систем обработки управленческих данных с единых позиций и в единой терминологии, обобщение опыта организации таких систем и разработка рекомендаций по созданию соответствующих систем. Структура данных *сетевой модели* определяется в терминах раздела 6.1 (элемент, запись, группа, групповое отношение, файл, база данных).

Реализация групповых отношений в *сетевой модели* осуществляется с использованием специально вводимых дополнительных полей – указателей (адресов связи или ссылок), которые устанавливают связь между владельцем и членом группового отношения. Запись может состоять в отношениях разных типов (1:1, 1:N, M:N). Заметим, что если один из вариантов установления связи 1:1 очевиден (в запись – владелец отношения, поля которой

соответствуют *атрибутам сущности*, включается дополнительное поле – указатель на запись – член отношения), то возможность представления связей 1:N и M:N таким же образом весьма проблематична. Поэтому наиболее распространенным способом организации связей в сетевых СУБД является введение дополнительного типа записей (и соответственно, дополнительного файла), полями которых являются указатели.

Рассмотрим для примера представление группового отношения M:N. В модель вводится *дополнительная группа* (дополнительный вид записей). Элементы этой записи представляют собой указатели на две исходные группы и указатели на экземпляры рассматриваемой дополнительной записи, связывающие их в список (цепь), соответствующий M и (или) N членам группового отношения ([рис. 6.1.](#)).

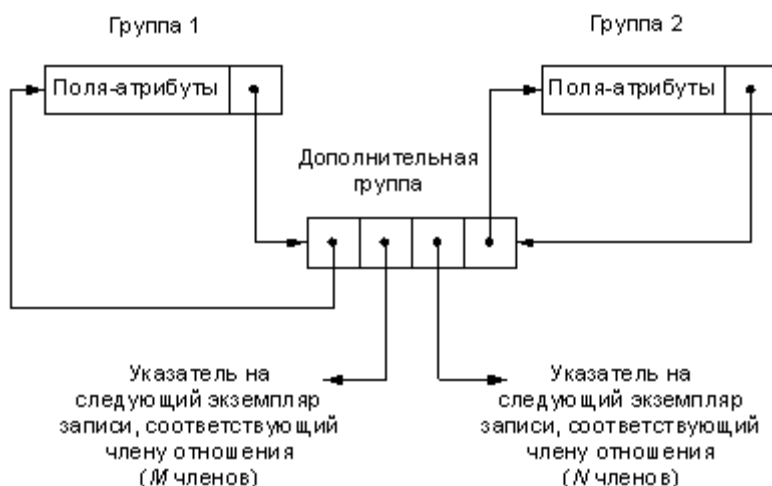


Рис. 6.1. Представление связей типа M:N

Представление связей 1:1, 1:M, N:1 является частным случаем связи типа M:N и осуществляется аналогично рассмотренному выше.

Заметим, что группа может быть членом более чем одного группового отношения. В этом случае вводится несколько *дополнительных групп-указателей*, а в группе – владельце отношений вводится несколько полей – указателей на *дополнительные группы*. Тогда множество записей (групп) и связей между ними образует некую сетевую структуру (ориентированный граф общего вида). Вершинами графа являются группы; дугами графа, направленными от владельца к члену группового отношения, – связи между группами.

Сетевая модель данных поддерживает все необходимые операции над данными, реализованные как действия со *списковыми структурами*. *Сетевая модель* данных является, вероятно, наиболее общей по возможностям *представления концептуальной модели*. По сути, любая ER-диаграмма без каких-либо изменений представляется средствами *сетевой модели*. К недостаткам *сетевой модели* обычно относят сложность получаемой на её основе концептуальной схемы и большую трудоемкость понимания соответствующей схемы внешним пользователем.

Рассмотрим пример записи части *ER-диаграммы* (СТУДЕНТ и ФАКУЛЬТЕТ) из предыдущей лекции в терминах сетевой СУБД. Для примера рассмотрим несколько *экземпляров сущности* СТУДЕНТ и сущности ФАКУЛЬТЕТ ([рис. 6.2.](#)).

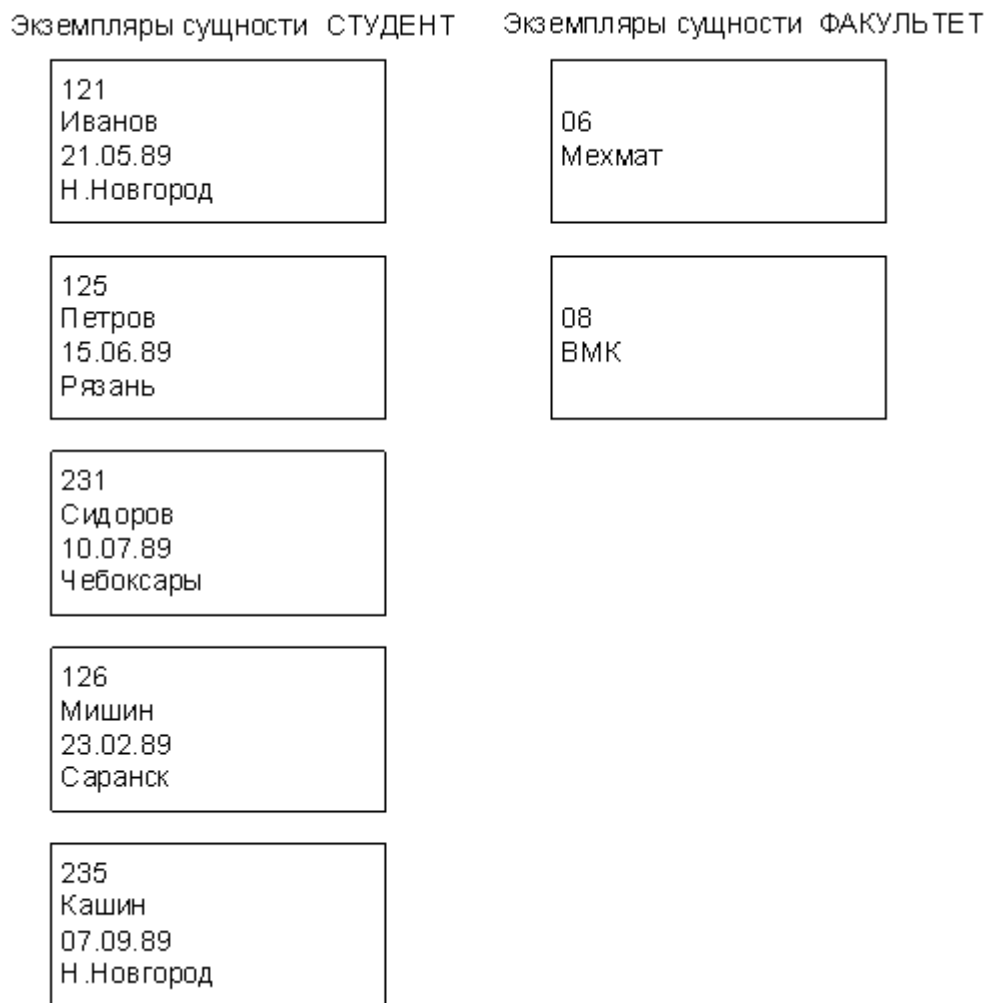


Рис. 6.2. Примеры экземпляров сущностей

Пусть студенты Иванов, Петров, Мишин учатся на факультете ВМК, Сидоров и Кашин на механико-математическом факультете. Тогда *сетевая модель* соответствующего фрагмента *ER-диаграммы* будет выглядеть следующим образом ([рис. 6.3](#)).

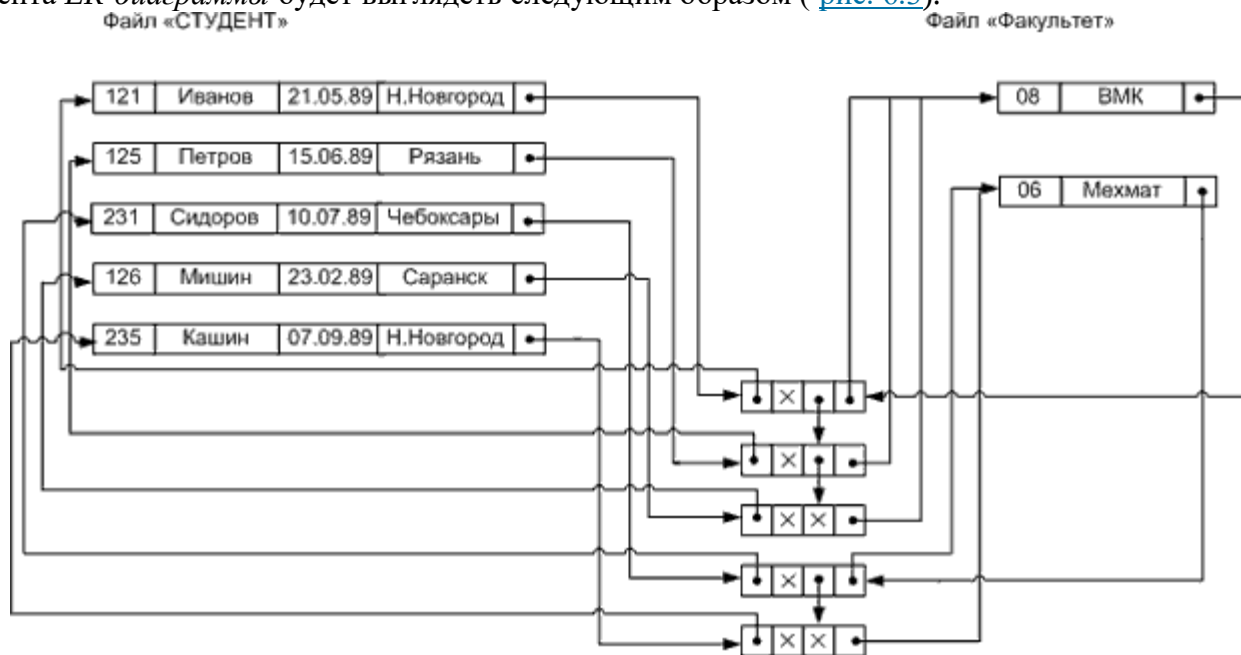


Рис. 6.3. Пример сетевой модели концептуального представления

Заметим, что в дополнительном файле один из указателей не потребовался, так как рассматриваемая связь имеет тип 1:N, а не M:N. Значок x обозначает отсутствие дальнейшей связи.

Наиболее существенным недостатком *сетевой модели* является "жесткость" получаемой концептуальной схемы. Связи закреплены в записях в виде указателей. При появлении новых аспектов использования этих же данных может возникнуть необходимость установления новых связей между ними. Это требует введения в записи новых указателей, т.е. изменения структуры БД, и, соответственно, переформирования всей базы данных.

СУБД, поддерживающие сетевую модель, широко использовались на вычислительных системах серии IBM 360/370 (ЕС ЭВМ). В качестве примеров таких систем можно указать *IDMS*, *UNIBAD* (БАНК), и их аналоги *СЕДАН*, *СЕТОР*. На персональных компьютерах сетевые СУБД не получили широкого распространения. Примером сетевой СУБД для персонального компьютера является *db_VISTA III*. Отметим, что система *db_VISTA* реализована на языке C и поэтому является переносимой. Система может эксплуатироваться на ПЭВМ типа IBM PC, SUN, Macintosh.

6.2.2. Иерархическая модель данных

Это также одна из наиболее ранних моделей данных. Реализация групповых отношений в иерархической модели, как и в сетевой, может осуществляться с помощью указателей и представляется в виде графа. Однако, в отличие от сетевой модели, здесь существует ряд принципиальных особенностей.

Групповые отношения являются отношениями соподчиненности. Группа (запись) – владелец отношения имеет подчиненные группы – члены отношений. Исходная группа называется предком, подчиненная – потомком.

1. Групповые отношения образуют иерархическую структуру, которую можно описать как ориентированный граф следующего вида:

- имеется единственная особая вершина (соответствующая группе), называемая корнем, в которую не заходит ни одно ребро (группа не имеет предков);
- во все остальные вершины входит только одно ребро (все остальные группы имеют одного предка), а исходит произвольное количество ребер (группы имеют произвольное количество потомков);
- отсутствуют циклы.

2. *Иерархическая модель* данных может представлять совокупность нескольких деревьев. В терминологии *иерархической модели* деревья, описывающие структуру данных, называются деревьями описания данных, а сами структурированные данные (база данных) – деревьями данных.

Особенностью реализации операций поиска в *иерархической модели* является то, что операция всегда начинает поиск с корневой вершины и специфицирует иерархический путь (последовательность связанных вершин) от корня до вершины, экземпляры которой удовлетворяют условиям поиска.

Необходимо отметить, что программы, реализующие операции *иерархической модели*, существенно проще, чем аналогичные программы для *сетевой модели*, т.к. здесь много легче осуществлять навигацию по структуре. Целесообразность появления *иерархической модели* обусловлена, конечно, тем, что большинство организационных систем реального мира имеют иерархическую структуру (административное деление страны, организационная структура предприятия и т.п.). Соответствующее концептуальное представление также будет иметь иерархическую структуру и естественным образом может быть описано в терминах *иерархической модели*. В качестве недостатков *иерархической модели* можно назвать вышеуказанные недостатки сетевой.

СУБД, поддерживающие *иерархическую модель*, достаточно широко использовались на вычислительных системах IBM 360/370 (ЕС ЭВМ). В качестве примеров таких систем можно указать *IMS*, *ОКА* и широко тиражируемую в СССР отечественную разработку ИНЕС. Примером иерархической СУБД для персональных ЭВМ является отечественная система НИКА (адаптация системы ИНЕС к IBM PC).

6.2.3. Реляционная модель данных

Учитывая отмеченные в предыдущих разделах недостатки сетевых и иерархических моделей, можно сформулировать желательные требования к модели данных:

модель должна быть понятна пользователю, не имеющему особых навыков в программировании;

- появление новых аспектов использования данных и необходимость введения новых связей не должны приводить к реструктуризации всей модели данных и базы данных в целом.

Моделью данных, удовлетворяющей вышеуказанным требованиям, является *реляционная модель*, часто называемая также **табличной**.

Основным используемым понятием здесь является понятие отношения, представляемого в виде таблицы, столбцы которой соответствуют *атрибутам сущности* (структура строки таблицы аналогична структуре записи). Каждый атрибут может принимать определенное множество значений, называемое доменом. Строка таблицы с конкретными значениями полей здесь называется кортежем (соответствует понятию "экземпляр записи"). Поля таблицы предполагаются элементарными (неделимыми). Таким образом, понятие "таблица" здесь соответствует понятию "файл" модели данных. Первичный ключ здесь – минимальный набор атрибутов, однозначно идентифицирующий кортеж в отношении.

Групповое отношение может представляться двумя способами. При первом способе в таблицы, соответствующие группам – членам отношения, добавляются столбцы ключевых полей (атрибутов) другого члена отношения (связь описывается через ключевые атрибуты). При втором способе групповое отношение определяется как *дополнительная группа* (дополнительная таблица). Столбцами этой дополнительной таблицы являются ключи групп – членов отношения. Таким образом, при любом способе соответствующая модель данных представляет собой совокупность структур таблиц.

Рассмотрим пример записи *ER-диаграммы* (см. [рис. 5.2.](#)) в терминах реляционных баз данных.

Сначала представим таблицы, соответствующие сущностям.

Таблица СТУДЕНТ

Код	Фамилия	Дата рождения	Место рождения

Таблица ФАКУЛЬТЕТ

Номер	Название

Таблица СПЕЦИАЛЬНОСТЬ

Номер	Название

Представим таблицы, описывающие связи.

Таблица "Студент учится на факультете"

Код студента	Номер факультета

Таблица "Студент учится по специальности"

Код студента	Номер специальности

Таблица "На факультете имеются специальности"

Номер факультета	Номер специальности

Заметим, что здесь реализован вышеописанный второй способ представления групповых отношений. Очевидно, что можно построить много вариантов таблиц, представляющих соответствующую *ER-диаграмму*.

Для приведенных таблиц не указаны домены атрибутов. Отсутствие указания доменов приводит к неоднозначной интерпретации содержания таблицы. Например, две таблицы с одинаковыми атрибутами

Код студента	Фамилия	Дата рождения	Место рождения

могут иметь разное смысловое значение и, соответственно, разное содержание (в одной таблице содержатся сведения о всех студентах факультета, в другой таблице сведения только о старостах факультета). Для устранения неоднозначной интерпретации в случае отсутствия указания доменов используют имя таблицы (СТУДЕНТ; СТАРОСТА)

Для формального описания таблицы используется теоретико-множественное понятие отношения.

Схемой отношения R называется перечень имен атрибутов отношения (соответствующих столбцам таблицы) с указанием доменов этих атрибутов и обозначается $R(A_1, A_2, \dots, A_n); \{A_i\} \subseteq D_i$, где $\{A_i\}$ – множество значений, принимаемых атрибутом $A_i (i=1, n)$.

Совокупность схем отношений, используемых для представления концептуальной модели, называется схемой реляционной базы данных, а текущие значения соответствующих отношений – реляционной базой данных.

В качестве основного недостатка реляционной модели можно указать дублирование информации при представлении связей.

Необходимо отметить, что большинство СУБД для *персональных ЭВМ* поддерживают именно реляционную модель данных. В качестве примеров таких наиболее распространенных СУБД можно указать все dBase-подобные системы, DB2, Paradox, Access, FoxPro, Oracle, MS SQL Server.

Более подробно *реляционная модель* данных будет рассмотрена в ["Формализация реляционной модели"](#).

6.2.4. Многомерная модель данных

Вернемся к понятию "сущность" концептуальной модели.

Сущность – это то, о чем накапливается информация в информационной системе. Часто оказывается, что информация об определенной сущности зависит еще от ряда параметров. Рассмотрим, например, сущность УСПЕВАЕМОСТЬ СТУДЕНТОВ со следующими атрибутами: число двоек, число троек, число четверок, число пятерок.

Значение атрибутов зависит от параметров "курс", "учебный год". Если использовать для описания соответствующей концептуальной схемы реляционную модель, то необходимо вводить множество таблиц УСПЕВАЕМОСТЬ СТУДЕНТОВ по каждому году для каждого курса. Так, при 5 курсах и необходимости анализировать данные за 10 лет число таблиц будет равно пятидесяти. Дублируются аналогичные структуры всех таблиц, достаточно сложна обработка данных, связанная с анализом однотипных данных при изменении значения одного из параметров и т.д.

Наиболее подходящей моделью данных для этого случая является так называемая *многомерная модель*, используемая в технологии OLAP (OnLine Analytical Processing – оперативная аналитическая обработка). Отметим, что многомерность модели данных означает здесь многомерное логическое представление структуры информации и, вообще говоря, не связана с многомерностью визуализации.

Многомерные структуры представляются как гиперкубы данных. Каждая грань куба является размерностью. Основными понятиями, используемыми в многомерных моделях данных, являются "измерение" (dimension) и "ячейка" (cell).

Измерение – упорядоченный набор значений, принимаемых конкретным параметром, соответствующий одной из граней гиперкуба. Для нашего примера можно указать в качестве измерений: учебный год – 2006-2007, 2007-2008, 2008-2009; курсы – 1,2,3 и т.д.

Ячейка или показатель – это поле, соответствующее атрибуту сущности, значение которого однозначно определяется фиксированным набором значений параметров (значениями "измерений", например, 2008-2009 учебный год, первый курс).

В многомерной модели данных определяется ряд дополнительных операций, среди которых можно выделить операции "формирование среза" и "агрегация".

При формировании среза пользователю по его запросу предоставляется некоторое подмножество гиперкуба, полученное в результате фиксации пользователем одного или

нескольких значений параметров. Операция "агрегация" обеспечивает переход к более общему представлению информации из гиперкуба пользователю, например суммируя значения показателей по всем значениям одного из параметров, допустим, по всем курсам.

Такая модель позволяет легко сравнивать данные при разных значениях параметров, строить графики зависимости значений конкретных атрибутов от значений определенных параметров (например, изменение атрибута по годам) и т.п. Поэтому основное назначение технологии OLAP – обработка информации для проведения анализа и принятия решения.

Массовое использование СУБД, поддерживающих многомерную модель данных, только начинается. В качестве наиболее известных СУБД такого типа можно указать Oracle Express Server.

6.3. Средства автоматизированного проектирования концептуальной модели

Средства автоматизированного проектирования концептуальной модели привлекают к себе в настоящее время большой интерес и используются в процессе создания структуры *базы данных* и интерфейса пользователя для доступа к данным.

Причина применения этих средств состоит в использовании в подавляющем большинстве реальных разработок баз данных *спиральной модели жизненного цикла* программного обеспечения, что предусматривает последовательное создание нескольких версий программного обеспечения. Каждая следующая версия включает в себя предыдущую (возможно, не полностью) и является ответом на замечания пользователя, полученные в результате тестирования предыдущей версии. При создании баз данных первая модель программного обеспечения, к сожалению, очень редко является удачной. Чаще всего заказчик отвергает первую версию, так как она недостаточно полно отвечает его требованиям. Причина такой ситуации заключается в том, что заказчик не может сразу, до создания начальной версии программы, четко и полно сформулировать свои требования. Обычно после получения первого варианта программного обеспечения заказчик выдвигает дополнительные требования, которые нельзя реализовать в рамках созданной *базы данных*. Это вынуждает разработчиков вносить изменения в структуру *базы данных*, а также, соответственно, в *интерфейс* пользователя для доступа к базе данных. Таких итераций может быть несколько до момента получения решения, адекватного запросам заказчика. Но даже после получения удовлетворительного решения процесс разработки *базы данных* не завершается. Жизнь не стоит на месте, и запросы заказчика меняются с течением времени. Часть этих изменений можно реализовать без изменения структуры *базы данных*, изменяя только *интерфейс* пользователя, другие же требуют изменения и интерфейса, и структуры *базы данных*. Надо заметить, что подобные изменения являются очень болезненными – работа по их внесению может оказаться трудоемкой и, что самое неприятное, потребовать замены большого количества отлаженного программного кода. Иными словами, замененный код был написан впустую, на самом деле его не нужно было писать.

Таким образом, создание работоспособной *базы данных* можно условно разделить на три этапа – проектирование *базы данных*, в процессе которого создаются рабочие прототипы, *кодирование* – создание структур баз данных и законченного интерфейса пользователя и сопровождение готовой *базы данных*.

Основная идея применения средств *автоматизированного проектирования баз данных* заключается в том, что процесс ручного кодирования начинается только после окончания процесса проектирования. На стадии проектирования *схема базы данных* и *интерфейс* пользователя для доступа к базе данных создаются автоматически, исходя из описания концептуальной модели, с помощью так называемых CASE-средств (*Computer Aided Software/System Engineering*). Конечно, созданный таким образом *интерфейс* не является законченным программным продуктом, однако он позволяет заказчику оценить возможности конечного продукта и внести свои коррективы. Только после одобрения заказчиком рабочего прототипа разработчики приступают к ручному кодированию – созданию законченного приложения.

При сопровождении все повторяется, за тем исключением, что генерируется не все *приложение* целиком, а только часть, которую надо изменять.

На практике чаще всего CASE-средства используются для создания схемы базы данных в виде ER-диаграмм и генерации структур баз данных для конкретной СУБД. После получения от заказчика изменений разработчики вносят соответствующие исправления в диаграмму "сущность – связь" и заново генерируют *структуры баз данных*. Средства автоматической генерации интерфейсов используются реже.

В настоящее время практически каждый производитель СУБД предлагает собственный *программный продукт* автоматизированного проектирования. Это Oracle Designer (Oracle), Power Designer (Sybase) и другие. Демонстрационные версии данных программных продуктов можно загрузить с соответствующих сайтов (<http://www.oracle.com>, <http://www.sybase.com>).

Кроме того, на рынке представлены решения третьих фирм, не производящих СУБД. Одними из самых распространенных являются программные продукты фирмы AllFusion – AllFusion ERwin Data Modeler и AllFusion Process Modeler (ранее – BPwin) и другие. На российском рынке данные программы предлагает фирма Interface Ltd. (<http://www.interface.ru>). Создание диаграммы "сущность – связь" осуществляется с помощью AllFusion ERwin Data Modeler, дальнейшее *моделирование*, включая генерацию программного кода создания базы данных производится с помощью программы AllFusion Process Modeler.

Создав наглядную модель базы данных можно оптимизировать структуру БД и добиться её полного соответствия требованиям и задачам организации. Визуальное *моделирование* повышает качество создаваемой базы данных, продуктивность и скорость её разработки.

На сайте Interface Ltd. доступна для загрузки демонстрационная версия AllFusion ERwin Data Modeler, которая представляет собой полнофункциональную версию, ограниченную по времени.

Краткие итоги. В лекции рассмотрены вопросы, относящиеся ко второй стадии концептуального проектирования – представлению концептуальной модели в терминах модели данных определенной СУБД. Дано описание общего представления о модели данных (основные используемые понятия - элемент, запись, файл; основные составляющие описания). Рассмотрены модели данных СУБД как инструмент представления концептуальной модели (сетевая модель данных, иерархическая модель данных, реляционная модель данных, многомерная модель данных и OLAP-технология). Приведены примеры записи концептуальной модели в терминах конкретной модели данных СУБД. Приводятся сведения о средствах автоматизированного проектирования концептуальной модели.

Вопросы данной лекции рассматриваются в [[1.1](#)] - [[6.8](#)]].

Цель лекции: рассмотреть формализованное описание реляционной модели и операций манипулирования данными как основу для использования математических методов проектирования баз данных и основу создания языков запросов к базе данных.

7.1. Формализованное описание отношений и схемы отношений

Как уже отмечалось в п. 6.2.3, реляционная модель описывает *представление* данных в виде двумерной таблицы, называемой *отношением*. Наименованиями столбцов этой таблицы служат имена *атрибутов*. Рассмотрим формализованное описание соответствующих понятий.

Пусть $A_1, A_2, \dots, A_n$ имена атрибутов. Каждому имени атрибута A_i соответствует допустимое множество значений, которые может принимать атрибут A_i . Это множество значений D_i называется доменом атрибута A_i , $i=1, n$. По определению, домены являются непустыми конечными или счетными множествами. Уточним, что в теории реляционных баз данных домен рассматривается как множество значений одного (причем простого) типа данных. Понятию домена D_i соответствует множество значений, стоящих в столбце A_i рассматриваемой таблицы.

Схемой отношения $R \{A_1, A_2, \dots, A_n\}$ называется конечное множество имен атрибутов $\{A_1, A_2, \dots, A_n\}$, причем атрибут A_i принимает значение из множества D_i ($i=1, 2, \dots, n$), где n – *арность* отношения.

Понятию "схема отношения" соответствует описание структуры двумерной таблицы (имена столбцов и допустимые множества значений).

Пусть $D = D_1 \cup D_2 \cup \dots \cup D_n$.

Отношением r со схемой R называется конечное множество отображений $\{t_1, t_2, \dots, t_p\}$ из множества $R: \{A_1, A_2, \dots, A_n\}$ в множество $D: \{D_1 \cup D_2 \cup \dots \cup D_n\}$, таких, что $t_k(A_i) \in D_i, k = \overline{1, p}; i = \overline{1, n}$.

Отображение t_k называется k -м кортежем, p – размерность кортежа.

Понятию k -го кортежа соответствует множество значений, стоящих в k -й строке рассматриваемой таблицы.

Понятию отношения r соответствует множество значений, стоящих во всех строках рассматриваемой таблицы.

Ключом отношения r со схемой R называется минимальное подмножество $K = \{A_{i1}, A_{i2}, \dots, A_{im}\} \subseteq \{A_1, A_2, \dots, A_n\}$, где $\{i_1, i_2, \dots, i_m\} \subseteq \{1, 2, \dots, n\}$, такое, что любые два различных кортежа $t_1, t_2 \in r (t_1 \neq t_2)$ не совпадают по значениям множества $K = \{A_{i1}, A_{i2}, \dots, A_{im}\}$.

Возможны случаи, когда отношение r имеет несколько ключей. Такие ключи называются потенциальными (возможными). **Выбранный из них ключ для идентификации кортежей называется первичным ключом.** Таким образом, достаточно знать значение кортежа на множестве K , чтобы однозначно его идентифицировать. Ключ используется для представления связей между отношениями. С этой целью **первичный** ключ одного отношения включается в структуру (набор атрибутов) связанного с ним отношения. Для второго отношения соответствующий **ключ** называется **внешним ключом**.

Совокупность схем отношений, используемых для представления концептуальной модели, называется схемой реляционной базы данных (реляционной моделью данных). Текущие значения соответствующих отношений называются реляционной базой данных.

Выпишем реляционную модель данных примера из "Вторая стадия концептуального проектирования (Модели данных СУБД. Представление концептуальной модели средствами модели данных СУБД)" (см. рис. 6.3.). Введем обозначения **атрибутов** всех соответствующих сущностей. Пусть A_1 – код студента, A_2 – фамилия, A_3 – дата рождения, A_4 – место рождения, A_5 – номер факультета, A_6 – название факультета, A_7 – номер специальности, A_8 – название специальности. Обозначим схему отношения СТУДЕНТ как $R1$, ФАКУЛЬТЕТ как $R2$, СПЕЦИАЛЬНОСТЬ как $R3$, СТУДЕНТ УЧИТСЯ НА ФАКУЛЬТЕТЕ как $R4$, СТУДЕНТ УЧИТСЯ ПО СПЕЦИАЛЬНОСТИ как $R5$, НА ФАКУЛЬТЕТЕ ИМЕЮТСЯ СПЕЦИАЛЬНОСТИ как $R6$.

Тогда реляционная модель соответствующего примера описывается следующей совокупностью схем отношений:

$R1(A_1, A_2, A_3, A_4)$

$R2(A_5, A_6)$

$R3(A_7, A_8)$

$R4(A_1, A_5)$

$R5(A_1, A_7)$

$R6(A_5, A_7)$

Напомним, что понятие "схема отношения" соответствует описанию структуры таблицы. Таблице с заполненными значениями (заполненными строками) соответствует понятие "отношение". Для данного примера отношения, соответствующие вышеуказанным схемам отношений будем обозначать

$r1, r2, r3, r4, r5, r6$,

Отметим следующие свойства отношения:

1. *Отношение* имеет имя, которое отличается от имен всех других отношений.
2. Каждое значение элементов кортежей представляется простым (атомарным) типом данных.
3. Каждый *атрибут* имеет уникальное имя.
4. Значения всех атрибутов являются атомарными (неделимыми). Это следует из определения *домена* как множества значений простого типа данных, т.е. среди значений *домена* не могут содержаться множества.
5. Порядок рассмотрения атрибутов в схеме отношения (отношении) не имеет значения, т.к. для ссылки на значение атрибута в *кортеже отношения* всегда используется имя атрибута.
6. Порядок рассмотрения кортежей в отношении не имеет значения, т.к. *отношение* представляет собой множество кортежей, а элементы множества, по определению теории множеств, неупорядочены.

7.2. Манипулирование данными в реляционной модели

Для манипулирования данными в реляционной модели используются два формальных аппарата:

реляционная алгебра, основанная на теории множеств;

реляционное исчисление, базирующееся на исчислении предикатов первого порядка.

Механизмы реляционной алгебры и реляционного исчисления эквивалентны, т.е. для любого допустимого выражения *реляционной алгебры* можно построить эквивалентную формулу *реляционного исчисления* и наоборот.

Отличаются два этих формальных аппарата уровнем процедурности. Выражения *реляционной алгебры* строятся на основе алгебраических операций (высокого уровня), и подобно тому, как интерпретируются арифметические и логические выражения, выражение *реляционной алгебры* также имеет процедурную интерпретацию. Другими словами, *запрос*, представленный на языке *реляционной алгебры*, может быть реализован как последовательность элементарных алгебраических операций с учетом их старшинства и возможного наличия скобок.

Для формулы *реляционного исчисления* однозначная интерпретация (соответствующая однозначная последовательность действий), вообще говоря, отсутствует. Формула только устанавливает условия, которым должны удовлетворять кортежи результирующего отношения. Поэтому языки *реляционного исчисления* являются более непроцедурными или декларативными.

Операции, реализуемые с помощью указанных аппаратов, обладают важным свойством: они замкнуты на множестве отношений. Это означает, что выражения *реляционной алгебры* и формулы *реляционного исчисления* определяются над отношениями *реляционных БД* и результатом вычисления также являются отношения. В результате любое выражение или формула могут интерпретироваться как *отношение*, что позволяет использовать их в других выражениях или формулах.

Как мы увидим, *алгебра* и исчисление обладают большой выразительной мощностью, очень сложные запросы к базе данных могут быть выражены с помощью одного выражения *реляционной алгебры* или одной формулы *реляционного исчисления*. Именно по этой причине такие механизмы включены в *реляционную модель данных*. Конкретный язык манипулирования *реляционными БД* называется *реляционно полным*, если любой запрос, выражаемый с помощью одной операции *реляционной алгебры* или одной формулы *реляционного исчисления*, может быть выражен с помощью одного оператора этого языка.

Заметим, что крайне редко *алгебра* или исчисление принимаются в качестве полной основы какого-либо языка *БД*. Обычно (как, например, в случае языка *SQL*) язык основывается на некоторой смеси алгебраических и логических конструкций. Тем не менее знание алгебраических и логических основ языков баз данных часто бывает полезно на практике.

7.3. Операции реляционной алгебры

Операции реляционной алгебры определены на множестве отношений и являются замкнутыми относительно этого множества (образуют алгебру). Оказывается, что любой

произвольный запрос к БД можно представить в виде последовательности, составленной из пяти основных операций реляционной алгебры. Рассмотрим эти операции.

Объединение $r \cup s$

Объединением отношений r и s называется множество кортежей, которые принадлежат или r , или s , или им обоим. Для операции объединения требуется одинаковая арность отношений.

Для примера, пусть

r			s		
a	b	a	b	g	a
d	a	f	d	a	f
c	b	d			

тогда

$r \cup s$		
a	b	a
d	a	f
c	b	d
b	g	a

Заметим, что с помощью операции объединения может быть реализовано добавление нового кортежа к имеющемуся отношению. В этом случае r – исходное отношение, s – отношение, содержащее один добавляемый кортеж.

Разность $r - s$

Разностью отношений r и s называется множество кортежей, принадлежащих r , но не принадлежащих s . Для этой операции также требуется одинаковая арность отношений.

$r - s$		
a	b	a
c	b	d

Заметим, что с помощью операции разности может быть реализовано удаление кортежа из имеющегося отношения. В этом случае r – исходное отношение, s – отношение, содержащее один удаляемый кортеж.

Декартово произведение $r \times s$

Пусть r и s – отношения арности k_1 и k_2 соответственно. Декартовым произведением $r \times s$ называется множество кортежей длины $k_1 + k_2$, первые k_1 компонентов которых образуют кортежи, принадлежащие r , а последние k_2 – кортежи, принадлежащие s .

$r \times s$					
a	b	a	b	g	a
a	b	a	d	a	f
d	a	f	b	g	a
d	a	f	d	a	f
c	b	d	b	g	a
c	b	d	d	a	f

Проекция $\pi_{A_{i1}, A_{i2}, \dots, A_{im}}(r)$

Проекция $\pi_{A_{i1}, A_{i2}, \dots, A_{im}}(r)$ есть множество кортежей, получаемых из кортежей отношения r выбором столбцов с именами $A_{i1}, A_{i2}, \dots, A_{im}$.

Другими словами, это операция построения "вертикального" подмножества, получаемого путем выбора определенных атрибутов и исключения остальных. Повторяющиеся кортежи исключаются.	
$\pi_{1,3}(r)$	
a	a

d	f
c	d

Выбор (селекция) $\sigma_F(r)$

Пусть **F** – формула, образованная: операндами, являющимися константами или именами атрибутов, арифметическими операторами сравнения, логическими операторами (и, или, не), тогда выбором (селекцией) σ_F называется множество кортежей, компоненты которого удовлетворяют условию, заданному формулой **F**.

$$\sigma_{(1)=(3)}(r)$$

Здесь **F:(1)=(3)** – содержимое первого столбца равно содержимому третьего столбца.

Приведем ряд примеров представления запросов с помощью формальных операций для реляционной модели (СТУДЕНТ, ФАКУЛЬТЕТ, СПЕЦИАЛЬНОСТЬ), рассмотренной выше.

Пример 1.

Сформировать список студентов (фамилия).

Рассмотрим схему отношения СТУДЕНТ.

Атрибут "Фамилия" обозначен здесь **A₂**. Для ответа на запрос необходимо взять проекцию отношения **r1** на столбец **A₂**.

$$\pi_{A_2}(r1)$$

Пример 2.

Выдать список фамилий и дат рождений студентов, которым на текущую дату (**date**) больше 35 лет.

Рассмотрим то же отношение **r1**. Сначала выбираем студентов, которым больше 35 лет:

$$\sigma_{(A_3)+35 < date}(r1)$$

Затем берем проекцию полученного отношения на столбцы

$$\pi_{A_1, A_3}(\sigma_{(A_3)+35 < date}(r1))$$

Заметим, что можно было бы выполнить эти две операции в другой последовательности – сначала проекция, а затем селекция. Предлагается оценить, какой из этих вариантов лучше по оценке числа выполняемых элементарных действий и объему требуемой памяти.

Пример 3.

Выдать список фамилий студентов, обучающихся по специальности "Информационные технологии". Название специальности является атрибутом отношения **r3**. Если бы в этом отношении присутствовал атрибут "фамилия", то задача решалась бы аналогично примеру 2. В отношении **r5** присутствует атрибут "код студента", а "фамилия" присутствует в отношении **r1**. Для ответа на этот запрос необходимо связывать по "код студента" отношение **r3** и отношение **r1**.

Сначала выберем из отношения **r3** кортежи с названием специальности "Информационные технологии". Обозначим полученное отношение **rp1**. (Дальнейшие промежуточные отношения будем обозначать последовательно **rp1**, **rp2**, **rp3** и т.д.).

$$rp1 = \sigma_{(A_8) = \text{"Информационные технологии"}}(r3)$$

Далее нас будет интересовать только атрибут **A₁** – "код студента". Поэтому возьмем проекцию на эти столбцы.

$$rp2 = \pi_{A_1}(rp1)$$

Далее необходимо связать отношения **r1** и **rp2** (склеить таблицы). Для склейки таблиц используется операция "декартово произведение":

$$rp3 = r1 \times rp2$$

В отношении **r3** присутствуют два одинаковых столбца: **A₁** из отношения **r1** и **A₁** из отношения **rp2**. Выбирая из отношения **rp3** строки, в которых значения в соответствующих

столбцах совпадают, получим сведения о студентах, обучающихся по специальности "Информационные технологии"

$$rp4 = \sigma_{(A1*r1)=(A1*rp2)}(rp3),$$

где $A1 * r1$ и $A1 * rp2$ обозначают соответственно столбец A_1 соответствующей первой и второй составной части декартова произведения. Теперь осталось только выбрать фамилии соответствующих студентов

$$rp5 = \pi_{A1}(rp4)$$

Получаем требуемый результат. Заметим, что для экономии действий и памяти, перед тем как склеивать таблицы, целесообразно было сделать операцию проекции отношения $r1$ на столбцы A_1, A_2 . (чтобы не включать в декартово произведение лишние столбцы).

Введенные пять основных операций реляционной алгебры позволяют реализовать любой запрос к реляционной базе данных. Однако наряду с основными операциями достаточно часто удобно использовать так называемые дополнительные операции реляционной алгебры (которые могут быть выражены через основные).

Пересечение $r \cap s$

Пересечением отношений r и s называется множество кортежей, принадлежащих как r , так и s . Пересечение может быть выражено через операции разности

$$r \cap s = r - (r - s).$$

$$r \bowtie s$$

θ -соединение $i \theta j$

θ -соединение r и s по столбцам A_i и A_j представляет собой множество таких кортежей в декартовом произведении r и s , что i -й компонент r находится в отношении θ с j -м компонентом s , где θ – арифметический оператор сравнения. Если θ является оператором равенства, то эта операция называется эквисоединением

$$r \bowtie_{i \theta j} s = \sigma_{i \theta(j)}(r \times s),$$

где l – арность отношения r .

Пример. r			s		
1	2	3	3	1	
4	5	6	6	2	
7	8	9			
$r \bowtie_{(2) < (1)} s$					
1	2	3	3		
1	2	3	6		
4	5	6	6		

Заметим, что в примере 3 две последовательно идущие операции (декартово произведение и селекция) вместе как раз представляют операцию соединения. Причем использование декартова произведения для соединения таблиц обязательно обуславливает использование селекции как следующей операции для установления связи между таблицами. Поэтому целесообразно использовать такую объединенную операцию и программно реализовывать в СУБД именно операцию соединения.

Естественное соединение $r \bowtie s$

Операция применима тогда и только тогда, когда столбцы имеют имена (являются атрибутами). Операция применима к отношениям, у которых есть одинаковые атрибуты.

Пусть

$$r = (A_1, \dots, A_k, B_1, \dots, B_n), s = (A_1, \dots, A_k, C_1, \dots, C_m),$$

имена $A_1, \dots, A_k$ совпадают.

Тогда $r \triangleright \triangleleft s$ определяется следующим образом

$$r \triangleright \triangleleft s = \pi_{B_1, \dots, B_n, A_1, \dots, A_k, C_1, \dots, C_m} (\sigma_{r.A_1=s.A_1 \& r.A_2=s.A_2 \& \dots \& r.A_k=s.A_k} (r \times s)).$$

Для подчеркивания важности приведенных *операций реляционной алгебры*, а также для уточнения понятия реляционной *СУБД* приведем следующее *определение* одного из ведущих специалистов в области реляционных баз данных К.Дж. Дейта: " Будем называть систему реляционной, если она поддерживает, по крайней мере, реляционные базы данных, т.е. базы данных, которые могут восприниматься пользователем как таблицы и только как таблицы, операции селекции, проекции и соединения реляционной алгебры, не требуя при этом, чтобы каким-то образом были предопределены физические пути доступа для поддержки этих операций ".

Краткие итоги: В лекции рассматриваются вопросы, связанные с формализацией наиболее распространенной в настоящее время *модели данных СУБД* – реляционной модели. Формальное описание реляционной модели и полученные на этой основе математические методы и алгоритмы позволяют формализовать ряд шагов проектирования реляционной *базы данных*, получить оптимальную (по определенным критериям) структуру *базы данных* и эффективные алгоритмы обработки. Здесь рассматривается формализованное описание отношений, формальные средства манипулирования данными в реляционной модели (дано понятие *реляционного исчисления* и *реляционной алгебры*, приводятся основные *операции реляционной алгебры*). Приводятся примеры представления запросов как последовательность формальных *операций реляционной алгебры*.

РАЗДЕЛ 5. Нормальные формы в реляционных базах данных

Цели занятия.

Нормальные формы в РБД. Нормализация. Функциональные зависимости атрибутов в отношениях. Первая нормальная форма (1НФ). Вторая нормальная форма (2НФ). Третья нормальная форма (3НФ). Алгоритм нормализации (приведение к 3НФ). Корректность процедуры нормализации. Теорема Хеза. Нормальная форма Бойса-Кодда, четвертая и пятая нормальные формы. Пример логического моделирования БД при помощи нормальных форм. Области применения и проблемы логического моделирования БД при помощи нормальных форм.

4. Структура лекционного занятия.

№ п/п	Содержание (кратко)	Методы и средства обучения
1	Использования формальных методов построения оптимальной (по определенным показателям) структуры реляционной <i>базы данных</i> путем нормализации схем отношений.	1. Персональные компьютеры; 2. Средства доступа в Интернет; 3. Проектор.

5. Содержание лекционного занятия и взаимодействие с аудиторией.

Введение.

Формулирование темы занятия. Использование вводного примера, описание ситуации, демонстрация какого-либо процесса и т. д. (в зависимости от вида, замысла лекции и её структуры)

Тема 5.1. Использование формального аппарата для оптимизации схем отношений

Цель лекции: показать возможность эффективного использования формальных методов построения оптимальной (по определенным показателям) структуры реляционной *базы данных* путем нормализации схем отношений.

8.1. Проблема выбора рациональных схем отношений

При представлении концептуальной схемы в виде реляционной модели возможны различные варианты выбора схем отношений. Одни варианты выбора рассматривались в предыдущих разделах (п. 6.2.3), другие получаются объединением (или разбиением) некоторых схем отношений. От правильного выбора схем отношений, представляющих концептуальную схему, в значительной степени будет зависеть эффективность функционирования *базы данных*.

Рассмотрим для примера конкретную схему отношений и проанализируем её недостатки. Предположим, что данные о студентах, факультетах, специальностях, включены в таблицу со следующей схемой отношения: СТУДЕНТ (Код студента, Фамилия, Название факультета, Название специальности).

Эта схема отношений обуславливает следующие недостатки соответствующей *базы данных*:

- Дублирование информации (избыточность). У студентов, обучающихся на одном факультете, будет повторяться название факультета. Для разных факультетов будут повторяться специальности.
- Потенциальная противоречивость (*аномалии обновления*). Если, например, изменится название специальности, то изменяя её в одном кортеже (у одного студента), необходимо изменять и во всех других кортежах, где она присутствует.
- Потенциальная возможность потери сведений (*аномалии удаления*). При удалении информации о всех студентах, поступающих на определенную специальность, мы теряем все сведения об этой специальности.
- Потенциальная возможность невключения информации в базу данных (*аномалии включения*). В базе данных будут отсутствовать сведения о специальности, если на ней нет обучающихся студентов.

В теории реляционных баз данных существуют формальные методы построения реляционной модели *базы данных*, в которой отсутствует *избыточность* и *аномалии обновления, удаления и включения*.

Нормализация. Первая нормальная форма.

Построение рационального варианта схем отношений (обладающего лучшими свойствами при операциях включения, модификации и удаления данных, чем все остальные наборы схем) осуществляется путем так называемой нормализации схем отношений. Нормализация производится в несколько этапов. На начальном этапе схема отношений должна находиться в первой нормальной форме (1НФ).

Отношение находится в первой нормальной форме, если все атрибуты отношения принимают простые значения (атомарные или неделимые), не являющиеся множеством или кортежем из более элементарных составляющих.

Рассмотрим следующий пример.

Таблица представляет сущность ЭКЗАМЕНАЦИОННАЯ ВЕДОМОСТЬ

Код студента	Фамилия	Код экзамена	Предмет и дата	Оценка
1	Сергеев	1	Математика 5.06.08	4
2	Иванов	1	Математика 5.06.08	5
1	Сергеев	2	Физика 9.06.08	5
2	Иванов	2	Физика 9.06.08	5

Теперь на пересечении любой строки и любого столбца находится одно значение и, следовательно, данная таблица находится в первой нормальной форме.

Далее отношение, представленное в первой нормальной форме, последовательно преобразуется во вторую и третью нормальные формы. Процесс построения второй и третьей нормальных форм будет описан в следующих подразделах. При некоторых предположениях о данных третья нормальная форма является искомым наилучшим вариантом.

Если эти предположения не выполняются, то процесс нормализации продолжается и отношение преобразуется в четвертую и пятую нормальные формы. Построение соответствующих форм описано в литературе и в данной книге не рассматривается.

Прежде чем перейти к построению второй нормальной формы, необходимо определить ряд формальных понятий.

8.2. Функциональные зависимости (зависимости между атрибутами отношения)

Пусть $R(A_1, A_2, \dots, A_n)$ – схема отношения, а X и Y – подмножества $\{A_1, A_2, \dots, A_n\}$.

Функциональная зависимость на отношении R – это утверждение вида "Если два кортежа R совпадают по атрибутам множества $X \subset \{A_1, A_2, \dots, A_n\}$ (т.е. эти кортежи имеют в соответствующих друг другу компонентах одни и те же значения для каждого атрибута множества X), то они должны совпадать и по атрибутам множества $Y \subset \{A_1, A_2, \dots, A_n\}$ ". Формально эта зависимость записывается выражением $X \rightarrow Y$, причем говорится, что X функционально определяет Y . Часто используется другое утверждение: X функционально определяет Y или Y функционально зависит от X (обозначается $X \rightarrow Y$) тогда и только тогда, когда каждое значение множества X отношения R связано с одним значением множества Y отношения R . Иначе говоря, если два кортежа R совпадают по значению X , они совпадают и по значению Y .

Замечание. Вообще говоря, под термином "отношение" могут подразумеваться два понятия:

- отношение как переменная, которая может принимать разные значения (таблица, в строки и столбцы которой могут быть вписаны разные значения);
- отношение, как набор конкретных значений (таблица с заполненными элементами).

Функциональные зависимости характеризуют все отношения, которые могут быть значениями схемы отношения R в принципе. Поэтому единственный способ определить функциональные зависимости – внимательно проанализировать семантику (смысл) атрибутов.

Функциональные зависимости являются, в частности, ограничениями целостности, поэтому целесообразно проверять их при каждом обновлении базы данных.

Пример функциональных зависимостей для отношения ЭКЗАМЕНАЦИОННАЯ ВЕДОМОСТЬ

Код студента $\rightarrow$ Фамилия

Код студента, Код экзамена $\rightarrow$ Оценка

Пример функциональных зависимостей для отношения СТУДЕНТ, приведенного в начале настоящей лекции

Код студента $\rightarrow$ Фамилия,

Код студента $\rightarrow$ Факультет

Заметим, что последняя зависимость существует при условии, что один студент не может обучаться на нескольких факультетах.

Полное множество функциональных зависимостей

Для каждого отношения существует вполне определенное множество функциональных зависимостей между атрибутами данного отношения. Причем из одной или более функциональных зависимостей, присущих рассматриваемому отношению, можно вывести другие функциональные зависимости, также присущие этому отношению.

Заданное множество функциональных зависимостей для отношения R обозначим F , полное множество функциональных зависимостей, которые логически можно получить из F , называется замыканием F и обозначается F^+ .

Если множество функциональных зависимостей совпадает с замыканием данного множества, то такое множество функциональных зависимостей называется полным.

Введенные понятия позволяют формально определить понятие ключа.

Пусть существует некоторая схема R с атрибутами $A_1 A_2 \dots A_n$, F – некоторое множество функциональных зависимостей и X – некоторое подмножество R . Тогда X называется ключом, если, во-первых, в F^+ существует зависимость $X \rightarrow A_1 A_2 \dots A_n$ и, во-вторых, ни для какого подмножества Y , входящего в X , зависимость $Y \rightarrow A_1 A_2 \dots A_n$ не принадлежит F^+ .

Полной функциональной зависимостью называется зависимость неключевого атрибута от всего составного ключа.

Частичной функциональной зависимостью будем называть зависимость неключевого атрибута от части составного ключа.

Для вычисления замыкания множества функциональных зависимостей используются следующие **правила вывода** (аксиомы Армстронга):

Пусть известна некоторая схема отношения $R\{A_1, A_2, \dots, A_n\}$ с множеством атрибутов $U=\{A_1, A_2, \dots, A_n\}$ и множество функциональных зависимостей F , заданных на множестве U .

Аксиома рефлексивности. Если Y входит в X , а X входит в $U(Y \subseteq X \subseteq U)$, то $X \rightarrow Y$ логически следует из F . Это правило дает тривиальные зависимости, так как в них правая часть содержится в левой части.

Аксиома пополнения. Если $X \rightarrow Y$ и Z есть подмножество U , то $XZ \rightarrow YZ$. В данном случае функциональная зависимость $X \rightarrow Y$ либо содержалась в исходном множестве F , либо может быть выведена из F с использованием описываемых аксиом.

Аксиома транзитивности. Если $X \rightarrow Y$ и $Y \rightarrow Z$, то $X \rightarrow Z$.

Справедлива следующая **теорема**. Аксиомы Армстронга являются полными и надежными.

Это значит, что используя их мы выведем все возможные функциональные зависимости, логически следующие из F , и не выведем никаких лишних зависимостей.

Существует несколько других правил вывода, которые следуют из аксиом Армстронга.

Правило самоопределения. $X \rightarrow X$.

Правило объединения. Если $X \rightarrow Y$ и $X \rightarrow Z$, то $X \rightarrow Y \cup Z$.

Правило псевдотранзитивности. Если $X \rightarrow Y$ и $W \cup Y \rightarrow Z$, то $X \cup W \rightarrow Z$.

Правило композиции. Если $X \rightarrow Y$ и $Z \rightarrow W$, то $X \cup Z \rightarrow Y \cup W$.

Правило декомпозиции. Если $X \rightarrow Y$ и Z входит в Y , то $X \rightarrow Z$.

Надо отметить, что вычисление замыкания множества функциональных зависимостей является трудоемкой задачей при достаточно большом количестве атрибутов (за счет выписывания большого количества тривиальных зависимостей).

8.3. Декомпозиция схемы отношения

Последовательный переход от одной нормальной формы к другой при нормализации схем отношений реализуется через декомпозицию. Основной операцией, с помощью которой осуществляется декомпозиция, является проекция.

Декомпозицией схемы отношения $R = \{A_1, A_2, \dots, A_n\}$ называется замена ее совокупностью подмножеств R , таких, что их объединение дает R . При этом допускается, чтобы подмножества были пересекающимися.

Алгоритм декомпозиции основан на следующей теореме.

Теорема о декомпозиции. Пусть $R(A, B, C)$ – отношение, A, B, C – атрибуты.

Если R удовлетворяет зависимости $A \rightarrow B$, то R равно соединению его проекций A, B и A, C

$R(A, B, C) = R(A, B), R(A, C)$

При нормализации необходимо выбирать такие декомпозиции, которые обладают свойством соединения без потерь. В этом случае, декомпозиция должна обеспечить то, что

запросы (выборка данных по условию) к исходному отношению и отношениям, получаемым в результате декомпозиции, дадут одинаковый результат. Соответствующее условие будет выполняться, если каждый *кортеж отношения* R может быть представлен как *естественное соединение* его проекций на каждое из подмножеств. Для проверки, обладает ли *декомпозиция* данным свойством, используются специальные алгоритмы, описанные в литературе (в данной книге не рассматриваются).

Вторым важнейшим желательным свойством декомпозиции является свойство сохранения функциональных зависимостей. Стремление к тому, чтобы *декомпозиция* сохраняла зависимости, естественно. *Функциональные зависимости* являются некоторыми ограничениями на данные. Если *декомпозиция* не обладает этим свойством, то для того чтобы проверить, не нарушаются ли при вводе данных условия целостности (*функциональные зависимости*), нам приходится соединять все проекции.

Таким образом, для правильно построенного проекта *базы данных* необходимо, чтобы декомпозиции обладали свойством соединения без потерь, и желательно, чтобы они обладали свойством сохранения функциональных зависимостей.

8.4. Выбор рационального набора схем отношений путем нормализации

Вторая нормальная форма (2НФ)

Отношение находится в 2НФ, если оно находится в 1НФ и каждый неключевой атрибут зависит от всего первичного ключа (не зависит от части ключа).

Для перевода отношения в 2НФ необходимо, используя *операцию проекции*, разложить его на несколько отношений следующим образом:

1. построить проекцию без атрибутов, находящихся в *частичной функциональной зависимости* от первичного ключа;
2. построить проекции на части *составного ключа* и атрибуты, зависящие от этих частей.

Третья нормальная форма (3НФ)

Отношение находится в 3НФ, если оно находится в 2НФ и каждый ключевой атрибут нетранзитивно зависит от первичного ключа.

Отношение находится в 3НФ в том и только том случае, если все неключевые атрибуты отношения взаимно независимы и полностью зависят от первичного ключа.

Оказывается, что любая схема отношений может быть приведена к 3НФ декомпозицией, обладающей свойствами соединения без потерь и сохраняющей зависимости.

Мотивировка третьей нормальной формы

Третья нормальная форма исключает избыточность и аномалии включения и удаления.

К сожалению, 3НФ не предотвращает все возможные аномалии.

Нормальная форма Бойса-Кодда (НФБК)

Если в R для каждой зависимости $X \rightarrow A$, где A не принадлежит X , X включает в себя некоторый ключ, то говорят, что данное отношение находится в нормальной форме Бойса-Кодда.

Детерминантом функциональной зависимости называется минимальная группа атрибутов, от которой зависит некоторый другой атрибут или группа атрибутов, причем эта зависимость нетривиальная.

Отношение находится в НФБК тогда и только тогда, когда каждый его детерминант является потенциальным ключом.

НФБК является более строгой версией 3НФ. Иными словами, любое отношение, находящееся в НФБК, находится в 3НФ. Обратное неверно.

Мотивировка нормальной формы Бойса-Кодда

В нормальной форме Бойса-Кодда не существует избыточности и аномалий включения, удаления и модификации. Оказывается, что любая схема отношения может быть приведена в нормальную форму Бойса-Кодда таким образом, чтобы декомпозиция обладала свойством соединения без потерь. Однако схема отношения может быть неприводимой в

НФБК с сохранением зависимостей. В этом случае приходится довольствоваться третьей *нормальной формой*.

8.5. Пример нормализации до 3НФ

Для улучшения структуры реляционной *базы данных* (устранения возможных аномалий) необходимо привести все таблицы *базы данных* к третьей *нормальной форме* или в более высокой форме (если это возможно). Таким образом, задача сводится к проверке *нормализации* всех сущностей, отображающихся в таблицы *базы данных*. Если *таблица*, получающаяся из некоторой сущности, не является таблицей в третьей *нормальной форме*, то она должна быть заменена на несколько таблиц, находящихся в третьей *нормальной форме*.

Продолжим рассмотрение примера с отношением ЭКЗАМЕНАЦИОННАЯ ВЕДОМОСТЬ

В начале этой лекции мы привели *отношение* к первой *нормальной форме*.

Код студента	Фамилия	Код экзамена	Предмет	Дата	Оценка
1	Сергеев	1	Математика	5.08.03	4
2	Иванов	1	Математика	5.08.03	5
1	Сергеев	2	Физика	9.08.03	5
2	Иванов	2	Физика	9.08.03	5

Ключом данного отношения будет совокупность атрибутов – Код студента и Код экзамена.

Для более краткой записи процесса нормализации введем следующие обозначения:

КС – код студента, КЭ – код экзамена, Ф – фамилия, П – предмет, Д – дата, О – оценка.

Выпишем *функциональные зависимости*

КС, КЭ $\rightarrow$ Ф, П, Д, О

КС, КЭ $\rightarrow$ Ф

КС, КЭ $\rightarrow$ П

КС, КЭ $\rightarrow$ Д

КС, КЭ $\rightarrow$ О

КЭ $\rightarrow$ П

КЭ $\rightarrow$ Д

КС $\rightarrow$ Ф

В соответствии с определением, *отношение* находится во второй *нормальной форме* (2НФ), если оно находясь в 1НФ и каждый неключевой *атрибут* зависит от первичного ключа и не зависит от части ключа. Здесь атрибуты П, Д, Ф зависят от части ключа. Чтобы избавиться от этих зависимостей необходимо произвести декомпозицию отношения. Для этого используем теорему о декомпозиции.

Имеем *отношение* R(КС, Ф, КЭ, П, Д, О). Возьмем зависимость КС $\rightarrow$ Ф в соответствии с формулировкой теоремы исходное *отношение* равно соединению его проекций R1(КС, Ф) и R2(КС, КЭ, П, Д, О).

В отношении R1(КС, Ф) существует *функциональная зависимость* КС $\rightarrow$ Ф, ключ КС – составной, не ключевой *атрибут* Ф не зависит от части ключа. Это *отношение* находится в 2НФ. Так как в этом отношении нет транзитивных зависимостей, *отношение* R(КС, Ф) находится в 3НФ, что и требовалось.

Рассмотрим *отношение* R2(КС, КЭ, П, Д, О) с составным ключом КС, КЭ. Здесь есть зависимость КЭ $\rightarrow$ П, КЭ $\rightarrow$ Д, КЭ $\rightarrow$ П, Д. Атрибуты П, Д зависят от части ключа, следовательно *отношение* не находится в 2НФ. В соответствии с теоремой о декомпозиции исходное *отношение* (используем функциональную зависимость КЭ $\rightarrow$ П, Д) равно соединению проекций R3(КЭ, П, Д), R4(КС, КЭ, О). В отношении R3(КЭ, П, Д) существуют *функциональные зависимости* КЭ $\rightarrow$ П, КЭ $\rightarrow$ Д, КЭ $\rightarrow$ П, Д. Зависимости неключевых атрибутов от части ключа нет, следовательно *отношение* находится в 2НФ. Транзитивных зависимостей в этом отношении так же нет, следовательно *отношение* находится в 3НФ.

Таким образом, исходное *отношение* приведено в к трем отношениям, каждое из которых находится в третьей *нормальной форме* R1(КС, Ф), R3(КЭ, П, Д), R4(КС, КЭ, О).

Заметим, что в отношении R4 атрибуты КС, КЭ являются внешними ключами, используемыми для установления связей с другими отношениями. Представим полученную модель в виде диаграммы объектов-связей (ER-диаграммы). Для наглядности и возможности последующего программирования перейдем к английским названиям объектов (отношений) и атрибутов.

Отношение R1 представляет объект student с атрибутами id_st (*первичный ключ*), surname.

Отношение R3 представляет объект exam_st с атрибутами id_ex (*первичный ключ*), subject, date.

Отношение R4 представляет объект mark_st с атрибутами id_st (*внешний ключ*), id_ex (*внешний ключ*), mark. *Первичный ключ* здесь id_st, id_ex.

Соответствующая ER-диаграмма изображена на [рис. 8.1](#).

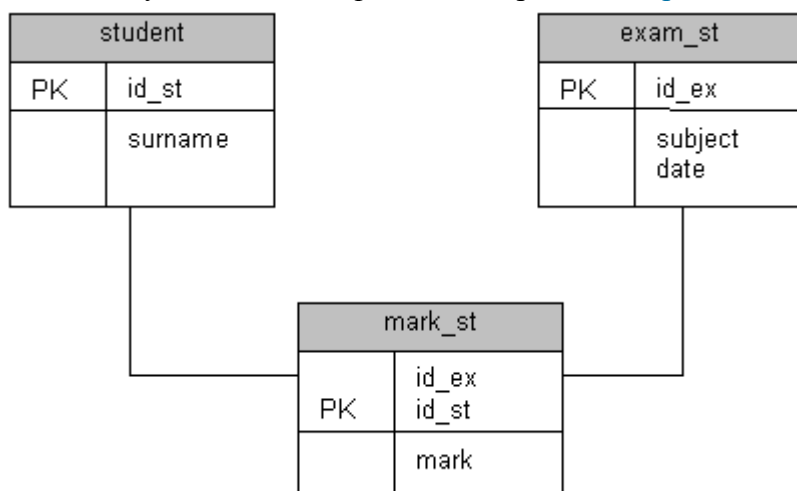


Рис. 8.1. ER-диаграмма, представляющая рассмотренный фрагмент предметной области 8.6. Целостная часть реляционной модели. Реализация условия целостности данных в современных СУБД

Напомним, что под целостностью *базы данных* понимается то, что в ней содержится полная, непротиворечивая и адекватно отражающая предметную часть (правильная) информация. Поддержка целостности в реляционных БД основана на выполнении следующих требований.

1. Первое требование называется **требованием целостности сущностей**. Объекту или сущности реального мира в реляционных БД соответствуют кортежи отношений. Конкретно требование состоит в том, что любой *кортеж* любого отношения отличим от любого другого кортежа этого отношения, т.е., другими словами, любое *отношение* должно обладать определенным первичным ключом. Это требование автоматически удовлетворяется, если в системе не нарушаются базовые свойства отношений.

2. Второе требование называется **требованием целостности по ссылкам**. Очевидно, что при соблюдении нормализованности отношений сложные сущности реального мира представляются в реляционной БД в виде нескольких кортежей нескольких отношений. Связь между отношениями осуществляется с помощью миграции ключа.

Пример внешнего ключа.

СТУДЕНТ (Код студента, Фамилия) сдает ЭКЗАМЕН (Код студента, Предмет, Оценка).

Атрибут Код студента сущности ЭКЗАМЕН называется *внешним ключом*, поскольку его значения однозначно характеризуют сущности, представленные кортежами некоторого другого отношения – отношения Студент (мы предполагаем, что поле Код студента является ключом отношения Студент).

Говорят, что *отношение*, в котором определен *внешний ключ*, ссылается на соответствующее *отношение*, в котором такой же *атрибутируется* первичным ключом.

Требование целостности по ссылкам или требование внешнего ключа состоит в том, что для каждого значения внешнего ключа в ссылающемся отношении в отношении, на которое ведет ссылка, должен найтись кортеж с таким же значением первичного ключа либо значение внешнего ключа должно быть неопределенным (т.е. ни на что не указывать).

Ограничения целостности сущности и по ссылкам должны поддерживаться СУБД. Для соблюдения целостности сущности достаточно гарантировать отсутствие в любом отношении кортежей с одним и тем же значением первичного ключа. (В Access для этого предназначена специальная реализация целочисленного поля – *поле* типа "Счетчик".) С целостностью по ссылкам дела обстоят несколько более сложно.

Понятно, что при обновлении ссылающегося отношения (вставке новых кортежей или модификации значения внешнего ключа в существующих кортежах) достаточно следить за тем, чтобы не появлялись некорректные значения внешнего ключа.

Но как быть при удалении кортежа из отношения, на которое ведет *ссылка*?

Здесь существуют три подхода, каждый из которых поддерживает *целостность* по ссылкам. Первый подход заключается в том, что запрещается производить удаление кортежа, на который существуют ссылки (т.е. сначала нужно либо удалить ссылающиеся кортежи, либо соответствующим образом изменить значения их внешнего ключа). При втором подходе при удалении кортежа, на который имеются ссылки, во всех ссылающихся кортежах *значение* внешнего ключа автоматически становится неопределенным. Наконец, третий подход (*каскадное удаление*) состоит в том, что при удалении кортежа из отношения, на которое ведет *ссылка*, из ссылающегося отношения автоматически удаляются все ссылающиеся кортежи.

В развитых реляционных СУБД обычно можно выбрать способ поддержания целостности по ссылкам для каждой отдельной ситуации определения внешнего ключа. Конечно, для принятия такого решения необходимо анализировать требования конкретной прикладной области.

Заметим, что все современные СУБД поддерживают и *целостность сущностей*, и *целостность* по ссылкам, но позволяют пользователям выключать данные ограничения и, таким образом, строить *базы данных*, не соответствующие реляционной модели. *Опыт* показывает, что отход от основных положений реляционной модели приводит к краткосрочному выигрышу – алгоритмы становятся проще, но впоследствии серьезно усложняют задачу, особенно ее сопровождение.

Краткие итоги: Лекция посвящена вопросам оптимизации схем отношений (структуры реляционной *базы данных*) на основе формальных методов *теории реляционных баз данных*. Здесь рассматривается ряд необходимых для этого понятий (*функциональная зависимость, нормальные формы, декомпозиция схем отношений*). Разбирается пример приведения таблицы к третьей нормальной форме, оптимальной по ряду показателей (исключающей *избыточность, аномалии включения и удаления*). Рассматриваются вопросы реализации целостности данных в реляционных СУБД.

В лекции рассматриваются вопросы использования формального аппарата для оптимизации схем отношений. Сформулирована *проблема выбора рациональных схем отношений* и пути реализации такого выбора путем *нормализации* (последовательного преобразования схемы отношения в ряд *нормальных форм*). Для формального описания соответствующего процесса определены понятие функциональной зависимости (зависимости между атрибутами отношения), ключа, сформулированы правила вывода *множества* функциональных зависимостей, понятие *декомпозиции* *схемы отношения*. Определены первая, вторая, третья *нормальные формы* и *нормальная форма* Бойса-Кодда. Приведен пример *нормализации* до 3НФ. Рассмотрены вопросы реализации условий целостности данных в реляционных СУБД.

Вопросы настоящей лекции рассматриваются в [[5.4](#)] - [[4.5](#)].

РАЗДЕЛ 6. Проектирование баз данных с использованием семантического подхода

Цели занятия.

Постановка задачи проектирования, предметная область будущей БД. Анализ предметной области, концептуальное (инфологическое) проектирование БД. Построение логической модели данных. Физическое проектирование БД. Синтез компьютерной модели объекта.

4. Структура лекционного занятия.

№ п/п	Содержание (кратко)	Методы и средства обучения
1	Основные типовые способы организации данных в памяти ЭВМ в СУБД с оценкой соответствующих моделей по времени доступа к данным в базе данных и по объему занимаемой памяти	1. Персональные компьютеры; 2. Средства доступа в Интернет; 3. Проектор.

5. Содержание лекционного занятия и взаимодействие с аудиторией.

Введение.

Формулирование темы занятия. Использование вводного примера, описание ситуации, демонстрация какого-либо процесса и т. д. (в зависимости от вида, замысла лекции и её структуры)

Тема 6.1. Физические модели данных (внутренний уровень)

Цель лекции: дать *представление* об основных типовых способах организации данных в памяти ЭВМ в СУБД с оценкой соответствующих моделей по времени доступа к данным в базе данных и по объему занимаемой памяти.

Как уже отмечалось, концептуальная схема, специфицированная к СУБД, автоматически отображается в *структуру хранения* программами СУБД. Внешний *пользователь* может ничего не знать о том, как его *представление* о данных физически организовано в памяти вычислительной системы. Тем не менее от физического *размещения данных в памяти ЭВМ* существенно зависит время решения прикладных задач. В связи с этим, даже на одном из начальных этапов проектирования *базы данных* – этапе выбора СУБД, желательно знать возможности физических *структур хранения*, представляемых конкретными СУБД, и оценивать временные характеристики проектируемой *базы данных* с учетом этих возможностей.

Способы физической организации данных в различных СУБД, как правило, различны и определяются типом используемой ЭВМ, инструментальными средствами разработки СУБД, а также критериями, которыми руководствуются разработчики СУБД при выборе методов размещения данных и способов доступа к этим данным. Заметим, что наиболее распространенным критерием служит *время доступа* к данным, однако в качестве критерия может выбираться, например, трудоемкость реализации соответствующих методов.

В настоящей лекции будут рассмотрены типовые физические модели организации данных в конкретных СУБД.

Физические модели данных служат для отображения моделей данных. Основными понятиями модели данных являются *поле, логическая запись, логический файл*. Слово "логический" введено, чтобы отличать понятия, относящиеся к *логической модели данных*, от понятий, относящихся к *физической модели данных*. Основными понятиями *физической модели данных*, используемыми для представления логической модели данных, являются *поле, физическая запись, физический файл*. В частности, *логическая запись*,

состоящая из полей, может быть представлена в виде физической записи (из тех же полей), *логический файл* – в виде физического файла. Прежде чем конкретизировать понятия, относящиеся к *физической модели данных*, рассмотрим структуру памяти ЭВМ.

9.1. Структура памяти ЭВМ

Важнейшей особенностью памяти ЭВМ, в значительной степени определяющей методы организации данных и доступа к ним, является её неоднородность. Существуют два разных типа памяти – оперативная (ОП) и внешняя (ВП), причем *процессор* работает только с данными из оперативной памяти (рис. 9.1.).

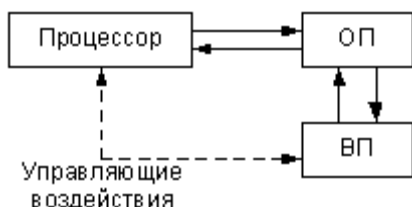


Рис. 9.1. Схема работы ЭВМ

Как уже многократно отмечалось, *базы данных* создаются для работы с большими объемами данных, что обуславливает необходимость использования внешней памяти. Поэтому организация данных и доступа к ним должна учитывать как специфику каждого типа памяти, так и способы их взаимодействия.

Отметим основные свойства оперативной памяти:

- единицей памяти является байт;
- память прямоадресуема (каждый байт имеет адрес);
- процессор выбирает для обработки нужные данные, непосредственно адресуясь к последовательности байтов, содержащих эти данные.

Отметим основные свойства внешней памяти:

- минимальной адресуемой единицей является *физическая запись* ;
- для последующей обработки (например, работы с полями) запись должна быть считана в оперативную память;
- время чтения записи в ОП на несколько порядков выше времени обработки процессором записи из ОП;
- организация обмена осуществляется порциями, т.к. невозможно считать сразу всю базу данных.

9.2. Представление экземпляра логической записи

Логическая запись представляется в оперативной памяти следующим образом:

Логическая запись					Последовательность байтов ОП			
Поле 1	Поле 2	..	Поле N		B1	B2	...	BN
Тип поля					B _i – последовательность байтов ОП, используемая для хранения поля i			
Характеристика поля								
Длина								

Прямая адресация байтов позволяет процессору выбирать для обработки нужное *поле*.

Заметим, что указанное *представление* не делает различий для записей в сетевой, иерархической и реляционных моделях. В случае сетевой и *иерархической моделей* некоторые поля могут являться указателями, тогда последовательность байтов, используемая для хранения этих полей, содержит *адрес* начала последовательности байтов, соответствующей записи – члену отношения.

В большинстве современных *СУБД* используется формат записей фиксированной длины. В этом случае все записи имеют одинаковую длину, определяемую суммарной длиной полей, составляющих *запись*. В *СУБД* другие форматы записей (переменной длины, неопределенной длины) встречаются гораздо реже, поэтому в данной книге эти форматы не рассматриваются. Заметим, что поля записи, принимающие значения существенно разной

длины в различных экземплярах записей, в *предметной области* встречаются достаточно часто. Примером может служить *поле резюме* в записи СОТРУДНИК. *Резюме* может составлять полстраницы текста, страницу и т.д. Возникает проблема – как эту информацию переменной длины представить в записи фиксированной длины. Возможным вариантом является установление размера соответствующего поля по максимальному значению. В этом случае у многих экземпляров записи указанное *поле* будет заполнено не полностью и, таким образом, *память* ЭВМ будет использоваться неэффективно. Более эффективный и часто используемый в *СУБД* прием организации таких записей состоит в следующем. Вместо поля (полей), принимающего *значение* существенно разной длины, в *запись* включается *поле-указатель* на область памяти, где будет размещаться *значение* исходного поля. Как правило, эта область является областью внешней памяти прямого доступа. В процессе ввода соответствующего значения в выделенной области занимает столько памяти, какова *длина* этого значения.

На [рис. 9.2](#) представлен пример вышеуказанного представления *экземпляров записей* из *N* полей, причем *поле N* принимает значения соответственно разной длины у разных *экземпляров записей*.



Рис. 9.2. Представление полей переменной длины

Конкретной реализацией такой схемы является *поле* типа МЕМО в *СУБД* (dBase III+, FoxPro, Access и т.д.).

9.3. Организация обмена между оперативной и внешней памятью

Единицей обмена данными между оперативной и внешней памятью является физическая запись. Физическая запись читается (записывается) за одно обращение к внешней памяти. В частности, *физическая запись* может соответствовать одному экземпляру логической записи. Число обращений к внешней памяти при работе с базой данных определяет время отклика системы. В связи с этим для уменьшения числа обращений к *БД* при работе с ней увеличивают длину физической записи (объединяют в одну *физическую запись* несколько экземпляров логических записей). В этом случае *физическую запись* называют также блоком, число *k* экземпляров логических записей, составляющих *физическую запись*, – коэффициентом блокировки.

Ввод исходных данных в *БД* осуществляется следующим образом:

- в ОП последовательно вводятся *k* экземпляров логических записей (кортежей);
- введенные *k* экземпляров объединяются в *физическую запись* (блок);
- *физическая запись* заносится во внешнюю память.

Ввод *k* экземпляров записей исходной таблицы, составляющих *i*-ю *физическую запись*, изображен на [рис. 9.3](#).

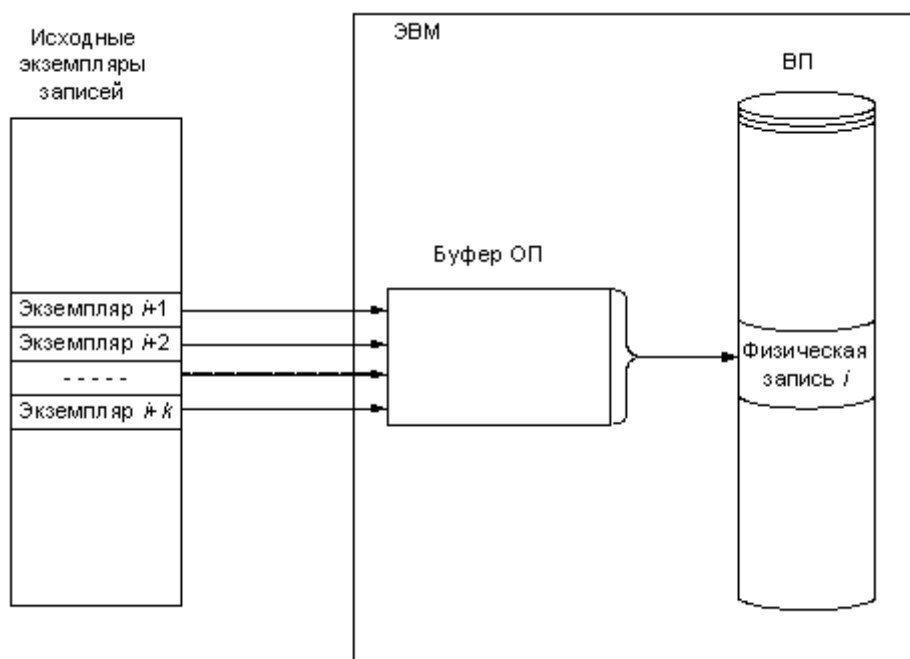


Рис. 9.3. Схема занесения записей во внешнюю память

Обработка данных, хранящихся во внешней памяти, осуществляется следующим образом:

- *физическая запись* (блок) считывается в оперативную память;
- обрабатываются экземпляры логических записей внутри блока (выбираются нужные поля, производится сравнение ключевого поля с заданным значением, осуществляется корректировка полей, выполняются операции удаления и т.п.).

В некоторых СУБД (например, *MS SQL Server*) единицей обмена между оперативной и внешней памятью является *страница* (вид физической записи, размер которой фиксирован и не зависит от длины логической записи). Организация обмена между оперативной и внешней памятью в этом случае аналогична описанной выше. Отличие здесь будет состоять в том, что экземпляры логических записей формируются в буфере, размером со страницу (если размер страницы не кратен длине логической записи, страница может быть заполнена неполностью, *физическая запись* на внешнем носителе, соответственно, будет заполнена не полностью).

9.4. Структуры хранения данных во внешней памяти ЭВМ

В современных СУБД наибольшее распространение получили *табличные модели* данных. В связи с этим, а также для большей определенности в настоящем разделе мы будем говорить о *структурах хранения* для табличной модели. Однако отметим, что некоторые из рассматриваемых ниже *структур хранения* могут использоваться и для представления сетевых и *иерархических моделей*.

В качестве внешней памяти мы рассматриваем наиболее распространенную в современных ЭВМ *память прямого доступа*. *Память прямого доступа* дает возможность обращения к любой записи, если известен её *адрес*. Для упрощения изложения мы не будем конкретизировать ряд служебных полей, которые содержит *физическая запись*, и их рассмотрение опускаем.

9.4.1. Последовательное размещение физических записей

В этой структуре хранения записи в памяти размещаются последовательно друг за другом. Как уже отмечалось, считаем, что все записи имеют равную длину. Физический адрес записи может быть легко вычислен по номеру записи (для вычисления необходимо знать формат соответствующей физической записи).

Физическая запись с номером **I** содержит логические записи с номерами

$$\begin{array}{l}
 (I-1)k + 1 \\
 (I-1)k + 2 \\
 \dots \\
 (I-1)k + k \\
 I = 1, 2, \dots, \lceil N/k \rceil;
 \end{array}$$

знаком $\lceil N/k \rceil$ обозначим ближайшее целое, большее или равное N/k , – целое сверху.

Рассмотрим, как реализуются основные элементарные операции модели данных в этой структуре хранения, и оценим число этих операций. Напомним, что с точки зрения пользователя в табличной модели данных эти операции являются операциями над строками (столбцами) таблицы.

Поиск записи с заданным значением ключа

При последовательной структуре хранения поиск может осуществляться только перебором. Читается первая *физическая запись*, в ОП она разбивается на k логических записей (разблокируется), заданное значение ключа сравнивается со значением ключа каждой логической записи. При несовпадении читается следующая *физическая запись* и процесс повторяется. В лучшем случае нужная запись будет найдена за одно обращение, в худшем – необходимо считать все физические записи. Среднее число обращений к внешней памяти для поиска нужной записи ТР определяется следующей формулой

$$ТР = (1 + \lceil N/k \rceil)/2,$$

где N – число логических записей, k – коэффициент блокировки, $\lceil N/k \rceil$ – число физических записей.

Чтение записи с заданным значением ключа

Сначала необходимо найти нужную запись (смотри операцию "поиск"). После окончания операции "поиск" нужная запись уже считана в ОП. Число обращений к ВП равно ТР.

Корректировка записи

Сначала необходимо найти нужную запись (смотри операцию "поиск"). После окончания операции "поиск" в ОП найденная *логическая запись* корректируется, формируется *физическая запись* (блок) и заносится во внешнюю память по тому адресу, откуда она была считана. Число обращений к ВП равно ТР+1.

Удаление записи

Аналогична операции корректировки. Служебное поле соответствующей логической записи помечается как "удаленная запись". Число обращений к ВП равно ТР+1.

Добавление записи

Рассмотрим два случая. В первом случае пользователь вводит новую *логическую запись* в конец таблицы. Тогда вводимая *логическая запись* добавляется в конец файла. Она заносится либо в последнюю *физическую запись* (если в ней меньше k логических записей – блок неполон), для чего эта запись должна быть считана в ОП, или формируется новая *физическая запись*, которая заносится в конец файла. Число обращений к ВП равно соответственно либо 2, либо 1.

Во втором случае пользователь вводит новую *логическую запись* в указываемую им i -ю строку таблицы ($i=1, 2, \dots, n$). В этом случае читается *физическая запись* с номером $\lceil (i-1)/k \rceil$, содержащая i -ю *логическую запись*. Если соответствующая *физическая запись* содержит пустые логические записи, то добавляемая запись вставляется в этот блок, блок записывается на свое место в ВП. Число обращений к ВП равно 2. Если указанная *физическая запись* содержит k экземпляров логических записей исходной таблицы, читается *физическая запись* с номером $\lceil i/k \rceil$. Если эта *физическая запись* содержит пустые логические записи, добавляемая запись вставляется в этот блок, блок

записывается на свое место в ВП. Суммарное число обращений в этом случае будет на единицу больше и равно 3.

Если физические записи с номерами $\lceil (i-1)/k \rceil$ и $\lceil i/k \rceil$ содержат по k экземпляров исходных логических записей, необходимо формировать дополнительную *физическую запись*. Соответствующий блок будет содержать добавляемую *логическую запись* и $k-1$ пустых логических записей. Блоки с номерами $\lceil i/k \rceil, \lceil (i+1)/k \rceil, \dots, \lceil N/k \rceil$ переписываются на одну позицию ниже (сдвигаются). Сформированная *физическая запись* заносится на освободившееся место (место записи с номером $\lceil i/k \rceil$).

В лучшем случае ($i = N$) ни один блок не сдвигается. В худшем случае ($i = 1$) сдвигаются все блоки. Среднее число обращений к ВП для перезаписи блоков (чтение + запись) составит $2\lceil N/k \rceil/2$. Тогда суммарное число обращений к ВП при добавлении записи в этом случае будет равно $3 + \lceil N/k \rceil$.

Заметим, что если записи упорядочены по значениям ключа поиск может производиться дихотомическим методом и число обращений к внешней памяти будет пропорционально не $(1 + \lceil N/k \rceil)/2$, а $\log_2 \lceil N/k \rceil$, т.е. существенно меньше. Однако добавление записи потребует для сохранения упорядоченности, как правило, сдвига большого числа записей. Поэтому размещение физических записей с упорядочением их по значениям ключа в СУБД не используется.

9.4.2. Размещение физических записей в виде списковой структуры

Основная проблема в использовании изложенного в п. 9.4.1 способа организации записей состоит в отображении добавления логической записи в произвольное место таблицы. При этом приходится переписывать в памяти (сдвигать на одну позицию) физические записи, соответствующие логическим записям таблицы, расположенным ниже места вставки добавляемой строки. Соответствующую проблему можно устранить, используя для представления физических записей связный список (рис. 9.4).

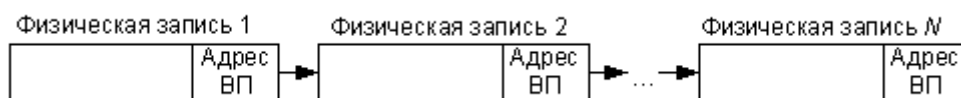


Рис. 9.4. Список физических записей

Кроме этого списка в ВП формируется список свободных элементов ("пустых" физических записей), элементы которого используются при вводе новой записи с данными (рис. 9.5).

Напомним, что каждая *физическая запись* состоит, как и ранее, из k логических записей.

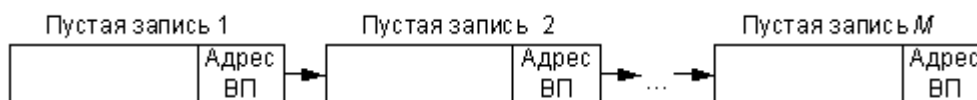


Рис. 9.5. Список свободных элементов

Рассмотрим, как реализуются основные элементарные операции модели данных в этой структуре хранения.

Поиск записи с заданным значением ключа

Заметим, что упорядочение записей по значениям ключа не дает здесь ускорения процедуры поиска. Это связано с тем, что после ряда добавлений новых записей и удаления каких-то имеющихся записей физическая и логическая последовательность записей в списке будут существенно различаться. При этом будет невозможно по номеру записи определить ее адрес и обращаться к записи, соответствующей середине таблицы, для реализации дихотомического метода поиска. Поэтому поиск можно вести только с помощью перебора. В ОП читается первая запись списка, разблокируется,

значения ключевых полей логических записей этой физической записи сравниваются с заданным значением. Если значения совпали, нужная запись найдена, если не совпали, из записи выбирается адрес следующей записи списка, читается эта запись. Далее процедура повторяется. Среднее число обращений к ВП будет равно, как и в 9.4.1, $(1 + \lceil N/k \rceil)/2$.

Чтение записи

После завершения предыдущей операции запись считана в ОП. Оценка числа обращений к ВП та же.

Корректировка записи

Считанная запись корректируется и заносится в ВП на свое место (по своему адресу). Число обращений к ВП на единицу больше, чем при чтении.

Удаление записи

Заметим, что мы говорим об операциях над логическими записями. Операция удаления логической записи аналогична операции корректировки. Служебное поле соответствующей логической записи помечается как "удаленная запись". Сформированная *физическая запись* заносится в ВП. Число обращений к ВП равно **ТР+1**.

Добавление записи

Для определенности будем считать, что задан ключ логической записи, после которой должна быть добавлена новая запись. Осуществляется операция поиска и чтения физической записи, в которой расположена запись с ключом **РК**. Если в этом блоке есть *логическая запись*, помеченная как удаленная, добавляемая запись заносится на ее место. Блок записывается в ВП. Число обращений к ВП равно **ТР+1**. Если в этом блоке нет логических записей, помеченных как удаленные, необходимо добавлять новую *физическую запись*, выбираемую из списка свободных элементов. С этой целью адрес связи найденной ранее физической записи заменяется на адрес начала списка свободных элементов.

Читается первая *физическая запись* списка свободных элементов. Адрес связи этой записи заменяет адрес начала пустого списка. В ОП формируется новая *физическая запись*, содержащая добавляемую *логическую запись*. В качестве ее адреса связи заносится адрес связи из физической записи, предшествующей добавляемой. Каждая из этих записей заносится в ВП. Число обращений к ВП при добавлении записи будет примерно равно **ТР+3**.

Рассмотренный метод организации *структуры хранения* достаточно эффективно решает проблемы добавления и удаления записей, но не уходит от перебора при поиске нужной записи.

9.4.3. Использование индексов (индексирование)

Как уже отмечалось, упорядочение записей позволяет использовать дихотомический метод поиска нужной записи и тем самым существенно сократить одну из основных составляющих времени поиска – число обращений к ВП. Однако при этом возникают проблемы с добавлением записей, связанные с необходимостью перезаписи части физических записей (сдвига).

Для того чтобы использовать дихотомический поиск и не перемещать физические записи при добавлении новых записей, используется так называемое логическое упорядочение физических записей (индексирование). Основная структура хранения содержит записи исходной таблицы и представлена в виде неупорядоченной последовательности физических записей (см. п. 9.4.1). Для возможной реализации дихотомического поиска по определенному ключу создается дополнительная структура хранения (так называемый индекс). Число записей в индексе равно числу записей исходной таблицы (числу физических записей в основной структуре хранения). Каждая запись индекса имеет два поля: ключевое поле записи основной структуры и указатель – адрес записи основной структуры с соответствующим значением ключа.

Записи индекса (индексного файла) упорядочены по значению ключа. Адреса связи этих записей определяют логическое упорядочение записей основной *структуры хранения*. Пример соответствующей *структуры хранения* приводится в предположении **k=1** на [рис. 9.6](#).

Рассматриваемую *структуру хранения* называют еще инвертированным списком. Смысл этого термина состоит в следующем. Можно было бы упорядочить записи основной *структуры хранения*, не переставляя их, а объединив в соответствующий упорядоченный список. В нашем случае адреса связи как бы удаляются из списка и

включаются в состав файла-индекса (инвертируются). Поэтому полученная структура интерпретируется как инвертированный список.

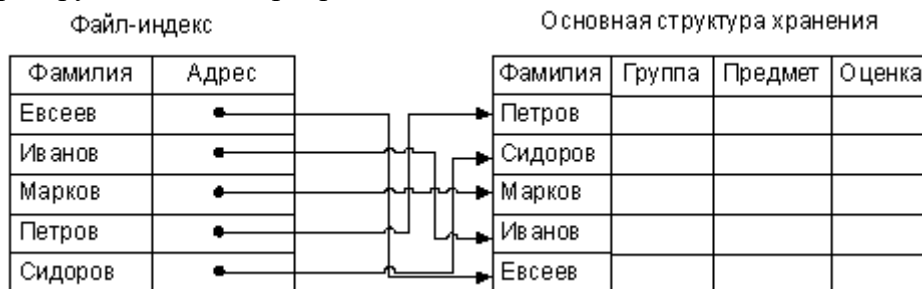


Рис. 9.6. Индексирование

Поиск нужной записи по заданному значению ключа осуществляется в индексном файле *методом половинного деления*. Заметим, что так как записи индекса содержат всего два поля, суммарный объем записей индекса невелик, поэтому индекс, как правило, целиком считывается для обработки в ОП за одно обращение к ВП. После того как в индексном файле обнаружена искомая запись, по адресу связи читается полная соответствующая запись основной *структуры хранения*. Если необходим поиск по другому ключу, строится еще один индекс по соответствующему ключу. Таким образом, по любому ключу поиск можно осуществлять дихотомическим методом.

Оценим число обращений к ВП при реализации элементарных операций. Соответствующие оценки сделаны для случая, когда *физическая запись* состоит из одной логической записи (коэффициент блокировки k равен 1). Расчет оценок для произвольного k производится по аналогии с расчетами пп. 9.4.1–9.4.2.

Поиск записи с заданным значением ключа

Из ВП читается индексный файл (число обращений к ВП для этого зависит от объема индексного файла, как правило, невелико и много меньше числа записей N). После нахождения нужной записи в индексном файле читается соответствующая запись основного файла (одно обращение к ВП).

Чтение записи

В ходе операции поиска искомая запись считана в ОП.

Корректировка записи

Считанная запись корректируется и заносится на свое место (еще одно обращение к ВП).

Удаление записи

Найденная запись помечается как удаленная в основном файле, соответствующая запись в индексном файле удаляется, измененный индекс записывается в ВП. Число обращений к ВП в этом случае по сравнению с числом обращений к ВП при поиске увеличивается на два.

Добавление записи

Добавляемая запись заносится в конец основного файла. Формируется новая запись индекса, соответствующая добавляемой записи. Записи индекса переупорядочиваются по значениям ключа, и индекс заносится в ВП. Число обращений к ВП в этом случае, в основном, определяется чтением-записью индекса.

Таким образом, использование индексов позволяет ценой некоторого увеличения объема используемой памяти (за счет индекса) существенно сократить время реализации основных операций. В связи с этим *индексирование* используется во многих современных СУБД.

9.4.4. В-дерево

Структура В-дерева (сбалансированное дерево) является следствием дальнейшего расширения концепции использования индексов (строится индекс над индексом) и представляет собой многоуровневые индексы.

В-дерево строится следующим образом. Последовательность записей, соответствующая записям исходной таблицы, упорядочивается по значениям первичного ключа. Логические записи объединяются в блоки (по k записей в блоках).

Значением ключа блока является минимальное значение ключа у записей, входящих в блок. Последовательность блоков представляет собой последний уровень *В-дерева*. Строится

индекс предыдущего уровня. Записи этого уровня содержат значение ключа блока следующего уровня и указатель-адрес связи соответствующего блока; записи этого уровня также объединяются в блоки (по k записей). Затем аналогично строится индекс более высокого уровня и т.д., пока количество записей индекса на определенном уровне будет не более k .

Рассмотрим процедуру работы с В-деревом на примере. Пусть имеется файл экземпляров логических записей, ключи которых принимают значения 2, 7, 8, 12, 15, 27, 28, 40, 43, 50. Для определенности возьмем $k=2$ (в блок объединяем по 2 экземпляра записей). Построенное для этого примера В-дерево изображено на рис. 9.7 (для упрощения рисунка на уровне 4 представлены только ключи логических записей и не представлены значения других полей этих записей).

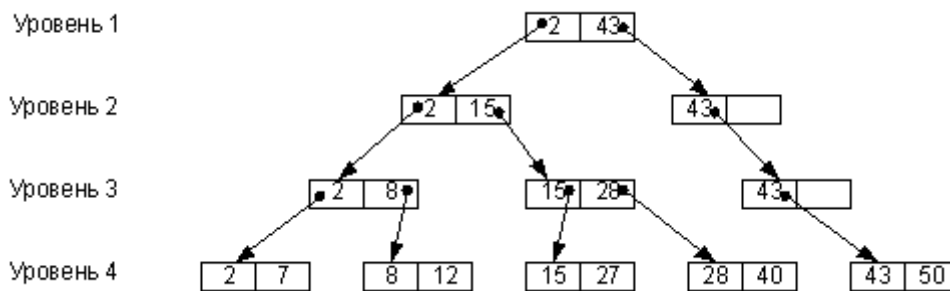


Рис. 9.7. В-дерево

В блоках указано значение ключа соответствующего блока. Значение k принято равным 2.

По построению В-дерева все исходные записи находятся на одном расстоянии от верхнего индекса (дерево является сбалансированным).

Рассмотрим реализацию основных операций.

Поиск и чтение записи с заданным значением ключа

Читается верхний индекс. Сравниваем заданное значение ключа со значением ключа последней записи индекса. Если заданное значение ключа больше, чем значение ключа очередной записи индекса (если такая запись имеется), или равно ему, то по адресу связи, указанному в текущей записи, читается блок записей индекса следующего уровня. Далее процесс повторяется.

Считаем, что все блоки расположены в ВП. Тогда число обращений к ВП при поиске информации будет равно числу уровней дерева. Число уровней дерева равно минимальному значению l , при котором выполняется условие $k^l \geq N$ (N – число логических записей).

Модификация (корректировка) записи

После поиска и чтения записи изменяются корректируемые поля. Если корректируется не ключ записи, то измененная запись заносится на свое место. Если изменено значение ключа, то старая запись удаляется (в соответствующем блоке появляется "пустая" запись), а измененная запись заносится так же, как вновь добавляемая.

Удаление записи

После поиска найденная запись удаляется (в соответствующий блок на место этой записи заносится "пустая" запись).

Добавление записи

Прежде всего определяется, где должна быть расположена добавляемая запись с заданным значением ключа. Процедура поиска блока, где должна быть расположена эта запись, аналогична вышеописанной процедуре поиска записей с заданным значением ключа. Если в найденном блоке низшего уровня есть "пустая" запись, добавляемая запись заносится в этот блок (с необходимым переупорядочением записей внутри блока).

Если в соответствующем блоке низшего уровня нет пустого места, блок делится на два блока. В первый из них заносится $\lceil k/2 \rceil$ записей, во второй заносятся остальные. Значением ключа каждого из указанных блоков будет являться, как и описано ранее, минимальное значение ключей у записей, входящих в блок. Добавляемая запись заносится в тот блок,

значение ключа которого меньше значения ключа добавляемой записи. Появление нового блока с новым значением ключа обуславливает необходимость формирования соответствующей новой записи в индексе на предыдущем уровне. Эта запись содержит новое значение ключа нового блока и указатель на его месторасположение. Процедура добавления такой записи аналогична описанной выше. Находится блок предыдущего уровня, куда должна быть помещена эта запись. Если в блоке есть пустое место, запись добавляется в блок, если блок полон, он делится на два блока, запись заносится в один из блоков, формируется запись индекса предыдущего уровня и т.д.

Возможен вариант, когда придется делить блок самого верхнего уровня и формировать еще один уровень дерева.

Рассмотрим для примера, изображенного на [рис. 9.7](#), добавление записи с ключом 10.

1. Сравнение на первом уровне.

$2 < 10 < 43$

Движение по левой ветви.

2. Сравнение на втором уровне.

$2 < 10 < 15$

Движение по левой ветви.

3. Сравнение на третьем уровне.

$2 < 8 < 10$

Движение по правой ветви.

Искомый блок

8	12

4. Блок заполнен.

Он делится на 2 блока

8	

12	

Сравнение $8 < 10 < 12$.

Запись с ключом 10 заносится в блок 1

8	10

12	

На низшем уровне появилась новая запись с значением ключа 12. Необходимо добавление новой записи с ключом 12 и указателем на запись низшего уровня к индексу предыдущего уровня.

5. Запись с ключом 12 уровня 3 должна добавляться в блок

2	8

. Блок полон, он делится на два блока

2	

8	

Сравнение $8 < 12$.

Запись добавляется во второй блок

8	12

6. На уровне 3 появился блок с новым ключом 8. Необходимо добавление новой записи с ключом 8 и указателем на соответствующий блок уровня 3 на уровне 2.

7. Запись с ключом 8 уровня 2 должна добавиться в блок

2	15

. Блок полон, он делится на два блока.

2	

15	

$2 < 8 < 15$

Запись добавляется в блок 1

2	8

.

8. На уровне 2 появился блок с новым ключом 15, необходимо добавление новой записи с ключом 15 и указателем на соответствующий блок уровня 2 на уровне 1.

9. Запись с ключом 15 уровня 1 должна добавляться в блок

2	43

. Блок полон, он делится на два блока.

2	

43	

$2 < 15 < 43$

Запись с ключом 15 добавляется в первый блок

2	15

43	

10. Необходимо сформировать еще один уровень дерева

2	43

.

Полученная структура будет иметь вид, представленный на [рис. 9.8](#).

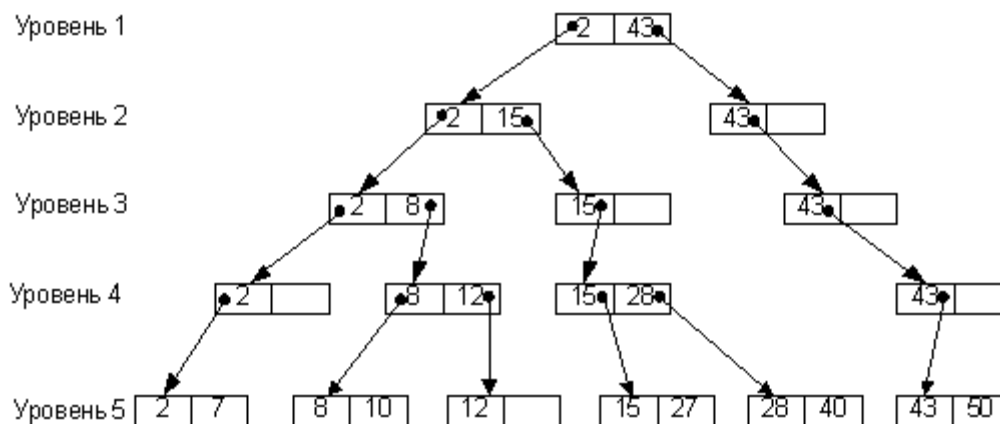


Рис. 9.8. В-дерево после добавления элемента

Необходимо заметить, что используемый прием деления пополам полностью заполненного блока при добавлении в него записи приведет к тому, что блоки будут заполнены, в среднем, наполовину. Тогда процедура добавления записи будет существенно менее трудоемкой (если в нужном блоке есть место, запись добавляется в этот блок и вышестоящие уровни не перестраиваются).

Структура хранения в виде В-дерева позволяет эффективно проводить операции поиска, чтения, удаления, модификации с оценкой числа обращений к внешней памяти числом уровней дерева $l \approx \log_k N$, что существенно меньше числа обращений при переборе $[N/k]$.

Процедура добавления записи тоже достаточно эффективна. Соответствующая структура хранения, в частности, используется в отечественной СУБД НИКА (ранее использовалась в системе ИНЕС) и на реальных задачах показала высокую эффективность.

9.4.5. Размещение записей с использованием хэширования

Как в любом другом способе организации структур хранения, логические записи группируются в физические записи (блоки) по k штук. Однако в отличие от всех других способов организации структур хранения здесь выбран особенный способ группировки. Определенным образом выбирается так называемая хэш-функция f . Аргументом этой функции является значение x первичного ключа логической записи. Тогда $f(x)$ указывает адрес расположения блока, в котором должна находиться логическая запись со значением ключа x .

Функция f должна, по возможности, равномерно распределять значения x по физическим блокам. Обсуждению возможных хэш-функций посвящено достаточно много литературы, поэтому здесь мы не будем касаться этого вопроса. Можно лишь добавить, что иногда, исходя из специфики множества значений x первичного ключа, можно построить функцию f , удовлетворяющую всем необходимым условиям. Таким образом, логическая запись таблицы со значением x первичного ключа размещается в блоке внешней памяти по адресу $f(x)$. В этом блоке может находиться не более k записей. Может оказаться, что выбранная функция отображает в один адрес памяти (один блок) более k записей. Возникает так называемая коллизия. Возможным способом разрешения коллизий является использование дополнительной области переполнения следующим образом. Если очередная запись распределяется с помощью функции хэширования в блок, а он полностью заполнен, то в области переполнения формируется список записей, соответствующих этому блоку, с включением в него указанной записи, а в сам блок заносится указатель – адрес связи на первую запись этого списка. Возможны и другие способы разрешения коллизий.

Рассмотрим реализацию основных операций и дадим оценку числа обращений к ВП при их выполнении.

Поиск записи с заданным значением ключа и чтение

По заданному значению ключа x подсчитывается значение функции $f(x)$. Далее из ВП считывается блок, находящийся по адресу $f(x)$. В ОП внутри этого блока перебором ищется нужная запись. Если

записей в блоке нет, то по указателю в блоке (адресу связи) читается первая запись списка переполнения, относящаяся к этому блоку. Далее необходимая запись ищется по этому списку. Число обращений к ВП при этом равно:

единице, если запись находится в блоке;

единице плюс число записей в соответствующем этому блоку списке области переполнения (как правило, небольшое число).

Модификации записи

Осуществляется поиск и чтение записи, затем в ОП модифицируются поля записи (не являющиеся первичным ключом), запись заносится на свое место. Число обращений к ВП в этом случае на единицу больше, чем при чтении записи. Если модифицируется значение ключа, то занесение записи осуществляется как ввод новой записи (добавление).

Удаление записи

Осуществляется поиск и чтение записи. Если удаляемая запись находилась в блоке основной памяти, на ее место заносится "пустая" запись (или признак "пустой" записи). Если удаляемая запись находилась в списке области переполнения, удаление ее производится по правилам удаления элемента списка. Число обращений к ВП при удалении находится примерно в тех же пределах, что и для предыдущих операций.

Добавление записи

При добавлении записи со значением ключа x подсчитывается адрес соответствующего блока $f(x)$. Блок считывается в ОП. Если в нем есть место, запись заносится в блок, блок записывается в ВП по своему адресу. Если блок заполнен, из него выбирается адрес начала списка записей, переполняющих блок. Далее добавление записи в список производится по правилам добавления элемента в список. Число обращений к ВП при добавлении записей находится примерно в тех же пределах, что и для предыдущих операций.

Таким образом, описанная *структура хранения* с использованием хэширования является наиболее эффективной (из рассмотренных выше) по критерию минимизации числа обращений к ВП при реализации основных операций.

9.4.6. Комбинированные структуры хранения

Необходимо заметить, что в СУБД могут использоваться как каждая из выше рассмотренных структур в отдельности, так и их комбинация. Так, например, в ряде промышленных систем UNIBAD, БАНК для ЭВМ типа IBM 360/370 (ЕС ЭВМ), PARADOX для персональных ЭВМ используются следующие комбинации методов:

размещение записей по первичному ключу организовано с использованием хэширования;

последовательность записей по вторичному ключу задается с помощью списковой структуры.

Краткие итоги: Лекция посвящена вопросам физической организации данных в памяти компьютера (организации структур хранения). Физические модели представления данных жестко заложены в структуру конкретной СУБД и различны в различных системах управления базами данных. Заметим, что в данной лекции рассматриваются не структуры хранения конкретной СУБД, а некоторые типовые структуры хранения, на основе которых и реализуются физические модели организации данных в конкретных СУБД. Здесь описывается двухуровневая структура памяти компьютера как среда размещения данных; организация обмена между внешней и оперативной памятью, определяющая специфику обработки данных. Представлены типовые физические модели (структуры хранения данных) во внешней памяти ЭВМ (последовательное размещение физических записей, размещение физических записей в виде списковой структуры, использование индексов, организация данных в виде В-дерева, размещение записей с использованием хэширования, а также комбинированные структуры хранения). Для основных структур хранения сделана оценка числа действий при выполнении операций поиска данных, чтения, занесения данных, модификации (корректировки), удаления.

РАЗДЕЛ 7. Проектирование баз данных

Цели занятия.

Постановка задачи проектирования, предметная область будущей БД. Анализ предметной области, концептуальное (инфологическое) проектирование БД. Построение логической модели данных. Физическое проектирование БД. Синтез компьютерной модели объекта.

4. Структура лекционного занятия.

№ п/п	Содержание (кратко)	Методы и средства обучения
1	Основные задачи программного обеспечения баз данных, существующих подходов к решению этих задач, в том числе и о структурированном языке запросов SQL.	1. Персональные компьютеры; 2. Средства доступа в Интернет; 3. Проектор.
2.	Общая характеристика операторов языка SQL и показать, как записываются основные запросы к базе данных на языке SQL (в интерактивном режиме).	

5. Содержание лекционного занятия и взаимодействие с аудиторией.

Введение.

Формулирование темы занятия. Использование вводного примера, описание ситуации, демонстрация какого-либо процесса и т. д. (в зависимости от вида, замысла лекции и её структуры)

Тема 7.1. Программное обеспечение работы с современными базами данных

Цель лекции: дать общее представление об основных задачах программного обеспечения баз данных, существующих подходов к решению этих задач, в том числе и о структурированном языке запросов SQL.

11.1. Основные задачи программного обеспечения баз данных

При работе с реляционными базами данных можно условно выделить две основные задачи: собственно работа с базой данных, включающая создание и ведение базы данных (создание структур таблиц, добавление записи в таблицу, удаление записи, обновление, выборка нужной записи); создание пользовательских приложений, включающих разработку пользовательского интерфейса по работе с базой данных.

Для решения указанных задач современные СУБД в своем составе могут содержать следующие программные средства: языки процедурного пошагового программирования, средства визуального программирования (графический интерфейс, диспетчер проекта, мастера и строители), средства создания объектно-ориентированных приложений. Кроме этого, при разработке пользовательских программ во многих СУБД допускается использование других языков программирования, а также использование библиотек разного рода. Так, например, при работе с СУБД ACCESS можно использовать язык программирования ACCESS, мастер ACCESS и язык программирования VISUAL BASIC.

При работе с клиент-серверными системами ситуация немного сложнее. Здесь в работе участвуют два типа компьютеров (сервер и клиент) и, соответственно, различают клиентское и серверное программное обеспечение. Серверное программное обеспечение включает язык программирования, поддерживающий создание и ведение базы данных, также реализацию

поступающих от клиентов запросов пользователей к базе данных. Пользовательские приложения создаются и работают на компьютерах-клиентах. Именно эти компьютеры должны иметь, наряду со средствами формирования запросов к базе данных, *средства разработки интерфейса*. В связи с этим, для клиент-серверных *СУБД* программное обеспечение разделяется на две части: *программное обеспечение – клиент* и *программное обеспечение – сервер*. Заметим, что наряду с программным обеспечением – клиент, при разработке пользовательских программ в конкретной *СУБД* могут использоваться другие языки программирования, специальные библиотеки, другие системы программирования (определенные для этой *СУБД*). В качестве примера в таблице приводятся возможные варианты использования программного обеспечения для организации клиент-серверного взаимодействия в *СУБД* Microsoft SQL Server.

Таблица 11.1. Возможные варианты использования программного обеспечения в СУБД MS SQL Server	
Средства ведения баз данных на сервере MS SQL	Средства разработки клиентских приложений
Службы SQL-сервер (MS SQL server и др.)	• Программное обеспечение клиента SQL-сервер (Transact SQL, SQL Server Query Analyzer и др.) • MS Access (ODBC) • MS Visual Basic • MS Visual Studio • MS Visual FoxPro • Java (JDBC) • Borland Delphi • Borland C++Builder и др. • Библиотеки функций (API, ODBC и др.)

Полное рассмотрение всего спектра программного обеспечения работы *СУБД* очень обширно и выходит за рамки данного пособия. Поэтому в данной работе будет рассмотрены только средства создания и ведения *базы данных*.

11.2. Проблемы создания и ведения реляционных баз данных

При создании *базы данных* и организации работы с ней возникают три основные проблемы: собственно создание базы данных (создание таблиц, индексов, ограничений целостности); обеспечение безопасности и разграничения доступа;

организация доступа к элементам таблицы (выборка, редактирование, удаление, добавление).

Первая проблема может быть решена посредством создания в каждой конкретной *СУБД* некоторой утилиты, позволяющей пользователю в определенный момент осуществлять все необходимые действия *по созданию базы данных*. Однако это не полностью решает проблему. Такая *утилита* не позволяет создать таблицу динамически во время работы прикладной программы, не позволяет, например, добавить в таблицу сформированный во время работы пользовательской программы столбец. Необходимы средства, дающие возможность формирования во время работы прикладной программы запроса на изменение структуры и содержания *базы данных*. То же самое можно сказать и о решении второй проблемы.

Более подробно рассмотрим возможный *путь* решения третьей проблемы. Организация доступа к базе данных является важнейшей функцией информационной системы. Пользователи постоянно работают с данными. Рассмотрим простой пример. Пусть у нас есть *таблица* СТУДЕНТ, хранящая информацию следующего рода:

СТУДЕНТ (Код студента, Фамилия, Имя, Отчество, Дата поступления).

Теперь мы хотим выполнить некоторый *запрос* к базе данных, результатом которого должны стать те строки таблицы СТУДЕНТ, для которых дата поступления окажется больше 01.06.2006. Рассмотрим последовательность действий для реализации данного запроса.

1. Получаем доступ к таблице СТУДЕНТ и устанавливаем указатель текущей строки на первую строку таблицы.

2. Анализируем поле "Дата поступления" в текущей строке.

3. Если значение "Дата поступления" > "01.06.2006", распечатываем на экране данные об абитуриенте.

4. Если таблица не кончилась, перемещаем указатель текущей строки на следующую строку и переходим к шагу 2, иначе заканчиваем работу.

Любой человек, знакомый с программированием, легко представит себе реализацию подобного алгоритма на любом языке программирования высокого уровня. Вот, в частности, пример реализации на *Object Pascal*:

```
Table.First;  
while (not Table.Eof) do  
begin  
  if FieldByName("Дата поступления").Value >  
    "01.06.2006"  
  then List.Add(FieldByName("Фамилия").AsString);  
  Table.Next;  
end;
```

В этом случае разработчик приложения сам организует работу по выборке данных, программируя каждое движение по таблице (осуществляет навигацию по таблице). Такой подход к обработке данных, ориентированный на последовательную работу с отдельными записями, называется *навигационным*. Здесь на конкретном языке программирования мы описываем процедуру - последовательность действий, необходимых для получения результата. Языки, в которых используется такой подход, называются *процедурными*. Очевидно, что с возрастанием сложности запроса существенно возрастает сложность процедуры и, соответственно, объем текста программы.

При работе с информационной системой *пользователь* реализует свои запросы к базе данных с помощью разработанных программистами прикладных программ. При навигационном подходе должны быть запрограммированы все возможные запросы. Очевидно, что заранее предугадать все запросы, потребность в которых может возникнуть и запрограммировать их, невозможно. С учетом того, что подавляющее большинство пользователей не владеет навыками программирования, это означает, что объем их действий будет ограничен рамками написанных программ, а именно, теми запросами, реализация которых предусмотрена заранее.

Кроме того, должен учитываться механизм взаимодействия в рамках архитектуры "клиент – сервер". Пользовательская программа выполняется на компьютере-клиенте. Запрос к базе данных реализуется компьютером – сервером. Необходим механизм формирования в пользовательской программе клиента запроса к базе данных сервера. В этом случае навигационный подход неприемлем. В связи с этим, для работы с базами данных разработан и используется другой подход, основанный на использовании так называемых языков запросов, которые задают не последовательность необходимых действий, а условия, которым должен удовлетворять результат (при добавлении столбца, выборке записи, добавлении записи и т. п.). Такой подход решает все три вышеперечисленные проблемы.

С этой целью разработан и активно используется во всех базах данных – специальный язык запросов *SQL*. Особо отметим, что основой языка являются операции реляционной алгебры.

Язык *SQL* (*Structured Query Language* – структурированный язык запросов) применяется для общения пользователя с реляционной базой данных и состоит из трех частей [[11.2]]:

- *DDL* (*Data Definition Language*) – язык определения данных. Предназначен для создания базы данных (таблиц, индексов и т.д.) и редактирования ее схемы.
- *DCL* (*Data Control Language*) – язык управления данными. Содержит операторы для разграничения доступа пользователей к объектам базы данных.
- *DML* (*Data Manipulation Language*) – язык обработки данных. Содержит операторы для внесения изменений в содержимое таблиц базы данных.

Как видно из написанного выше, *SQL* решает все рассмотренные ранее вопросы, предоставляя пользователю достаточно простой и понятный *механизм доступа к данным*, не связанный с конструированием алгоритма и его описанием на языке программирования высокого уровня. Так, вместо указания того, как необходимо действовать, *пользователь* при помощи операторов *SQL* объясняет *СУБД*, что ему нужно сделать. Далее *СУБД* сама анализирует текст запроса и определяет, как именно его выполнять.

В архитектуре "клиент – сервер" язык *SQL* занимает очень важное место. Именно он используется как язык общения клиентского программного обеспечения с серверной *СУБД*, расположенной на удаленном компьютере. Так, клиент посылает серверу *запрос* на языке *SQL*, а сервер разбирает его, интерпретирует, выбирает *план выполнения*, выполняет *запрос* и отправляет клиенту результат.

Посмотрим, как выглядит *запрос* на языке *SQL*, решающий задачу о выборке студентов *по* дате поступления.

```
SELECT Фамилия  
FROM Студент  
WHERE Дата поступления > "01.06.2006"
```

Может сложиться ложное впечатление, что появление языка *SQL* является альтернативой языков программирования высокого уровня. Это не соответствует действительности. Выполнение запроса средствами *SQL* все равно сводится к работе с отдельными записями, и от этого никуда не уйти. Важно понимать, что появление языка *SQL* дало, *по крайней мере*, две новые возможности.

1. Появился новый уровень абстракции между пользователем и *СУБД*. Этот уровень находится ближе к пользователю, чем уровень программирования на языке высокого уровня, что снижает требования к квалификации пользователей.

2. Многие типовые задачи, возникающие при работе с базами данных, и ранее решаемые каждым программистом по-своему (зачастую дублируя действия другого программиста) решены реализацией языка *SQL*. Таким образом, отпала необходимость самостоятельного решения многих проблем, решенных в *СУБД* соответствующим образом. Язык *SQL* предоставляет средства для доступа к этим типовым возможностям *СУБД*.

11.3. Понятие языка *SQL* и его основные части

11.3.1. История возникновения и стандарты языка *SQL*

История возникновения языка *SQL* восходит к 1970 году [[3.1](#)], когда доктор Е.Ф. Кодд предложил реляционную модель в качестве новой модели базы данных. Для доказательства жизнеспособности новой модели данных внутри компании IBM был создан мощный исследовательский проект, получивший название System/R. Проект включал разработку собственно реляционной *СУБД* и специального языка запросов к базе данных. Так в начале 70-х годов появился первый исследовательский прототип реляционной *СУБД*. Для этого прототипа разрабатывались и опробовались разные языки запросов, один из которых получил название *SEQUEL* (Structured English Query Language). С момента создания и до наших дней этот язык претерпел массу изменений, но идеология осталась неизменной.

Период с 1979 года (окончание проекта System/R) до настоящего времени характеризуется развитием и совершенствованием языка *SQL* и его постоянно увеличивающейся ролью в индустрии, связанной с созданием и эксплуатацией баз данных. Совершенно очевидно, что язык никогда не получил бы мирового признания, если бы на него не было никаких стандартов. Стандартизация – важная часть технологических процессов конца XX века. Именно наличие разработанных и официально признанных стандартов позволило утвердиться многим современным технологиям (не только в индустрии разработки программного обеспечения, но и в многих других сферах человеческой деятельности). Как обстоит дело со стандартами языка *SQL* и их поддержкой в распространенных *СУБД*?

Когда ведут речь о стандартах в области, связанной с разработкой программного обеспечения, обычно подразумевают две организации:

- ANSI (American National Standards Institute) – Американский национальный институт стандартов;
- ISO (*International Standards Organization*) – Международную организацию по стандартизации.

Работа над официальным стандартом языка SQL началась в 1982 году [8] в рамках комитета ANSI. В 1986 году (обратите внимание, сколько времени ушло на разработку стандарта и согласование деталей!) был утвержден первый вариант стандарта ANSI, а в 1987 году этот стандарт был утвержден и ISO. В 1989 году стандарт претерпел незначительные изменения, но именно этот вариант получил название SQL-1 или SQL-89. В чем особенность SQL-89? За время разработки стандарта (1982–1989 гг.) были созданы, представлены на рынке и активно использовались несколько различных СУБД, в которых в том или ином виде был реализован некоторый диалект языка SQL. С учетом того, что разработкой стандартов занимались те же люди, кто внедрял SQL в СУБД, *стандарт SQL-89* представлял собой плод множества компромиссов, приведших к наличию в нем большого количества "белых пятен", т.е. мест, которые не были описаны, а отданы на усмотрение разработчиков диалекта. В результате чуть ли не все имеющиеся диалекты стали совместимыми со стандартом, но особой пользы это не принесло.

Следующая реализация стандарта была призвана решить эту проблему. В результате длительных обсуждений и согласований в 1992 году был принят новый стандарт ANSI SQL-2 или SQL-92. SQL-92, который заполнил многие "белые пятна", впервые добавив в стандарт возможности, еще не реализованные в существующих коммерческих СУБД.

Работа над стандартизацией продолжается и далее. Появились *стандарты SQL-1999, SQL-2003*. Тем не менее, все эти стандарты не решили всех проблем, связанных с наличием нескольких диалектов языка. Как правило, разработчики как игнорировали, так и игнорируют некоторые положения стандарта, с одной стороны, отказываясь реализовывать некоторые его части и, с другой стороны, реализуя то, что отсутствует в стандарте. Несмотря на имеющиеся отличия, все коммерческие СУБД поддерживают некоторое ядро языка, описанное в стандарте, одинаково. Отличий не очень много, они не носят слишком принципиального характера. Хотя каждая СУБД по-прежнему поддерживает свой диалект языка.

В систему управления базами данных Microsoft SQL Server входит язык Transact-SQL, разработанный на основе одного из стандартов SQL.

11.3.2. Достоинства языка SQL

Для ознакомления с достоинствами языка обратимся к соответствующей литературе [[3.1](#)]. Вот некоторые из них:

- межплатформенная переносимость;
- наличие стандартов;
- одобрение и поддержка компанией IBM (СУБД DB2);
- поддержка со стороны компании Microsoft (СУБД SQL Server, протокол ODBC и технология ADO);
- реляционная основа;
- высокоуровневая структура;
- возможность выполнения специальных интерактивных запросов;
- обеспечение программного доступа к базам данных;
- возможность различного представления данных;
- полноценность как языка, предназначенного для работы с базами данных;
- возможность динамического определения данных;
- поддержка архитектуры клиент/сервер;
- поддержка корпоративных приложений;
- расширяемость и поддержка объектно-ориентированных технологий;
- возможность доступа к данным в Интернете;
- интеграция с языком Java (протокол JDBC);
- промышленная инфраструктура.

11.3.2. Общая характеристика SQL

Язык запросов SQL основан на операциях реляционной алгебры и, таким образом ориентирован на работу с множествами (отношениями), а не с отдельными записями. Как и в реляционной алгебре, операндами языка являются отношения (таблицы), результатами выполнения операции также являются отношения (таблицы). Таким образом, язык SQL предназначен для выполнения операций над таблицами, причем как над таблицами в целом (создание, удаление, изменение структуры), так и над данными таблиц (выборка, изменение, добавление и удаление). Отметим, что в явном виде язык SQL не является универсальным языком программирования в обычном понимании. В нем отсутствуют операторы условного перехода, организации циклов, позволяющие управлять ходом выполнения программы. Поэтому язык SQL относится к классу непроедурных языков программирования. Это именно язык запросов к базе данных, который служит исключительно для организации базы данных и работы с ней. Как уже отмечалось выше, для разработки прикладных программ необходимо использовать другие базовые средства программирования, в который операторы языка SQL будут встраиваться. Языку SQL посвящено большое количество литературы, в том числе и учебников. Подробное изучение языка SQL не входит в задачи настоящего курса, это может занимать отдельный курс. Заметим, что этому языку посвящено большое количество литературы, в том числе и учебников. В связи с этим, здесь будут изложены только общие сведения о языке, как фундаментальном инструменте работы с базами данных.

Терминология

Под запросом, реализуемым с помощью языка SQL- запросов к базе данных, понимается команда, предназначенная для выполнения (и выполняемая) системой управления базами данных определяемого этой командой действия с базой данных.

Запрос реализуется с помощью операторов языка SQL. Операторы состоят из отдельных логических частей, называемых предложениями. Стандарты языка SQL регламентируют синтаксис операторов. Несмотря на то, что язык SQL работает с реляционной базой данных, вместо термина "отношение" здесь используется термин "таблица", вместо терминов "кортеж" и "атрибут" используются соответственно термины "строка" и "столбец".

Разновидности SQL

Как отмечалось выше, в отличие от "обычных" языков программирования в SQL отсутствует возможность объявления переменных, нет инструкции IF, нет цикла FOR и т.д. Собственно программирование (разработка прикладных программ) на подобном языке практически невозможно. Поэтому к настоящему моменту используются следующие технологии (режимы) работы с базой данных на языке SQL (в некоторых источниках эти технологии называют разновидностями языка SQL):

- формирование непосредственно пользователем запроса на языке SQL в интерактивном режиме (**интерактивный SQL**);
- формирование запроса на языке SQL в прикладной программе (программный или встроенный SQL):
 - статическое формирование запроса (**статический SQL**);
 - динамическое формирование запроса (**динамический SQL**);
 - формирование запроса с помощью библиотек (**API – интерфейсы вызова подпрограмм**).

В интерактивном режиме работы с базой данных: пользователь работает с базой данных в прямом диалоге: вводит запрос на языке SQL – получает результат, вводит другой запрос – получает другой результат и т.д.

Встроенный SQL представляется операторами языка SQL, встроенные в прикладные программы, написанные на других языках программирования (в других программных средах). Это дает возможность работы с базой данных с помощью прикладных программ, написанных на других алгоритмических языках, но требует включения дополнительных средств, обеспечивающих интерфейс между операторами языка SQL и соответствующим языком программирования.

При статическом использовании языка (**статический SQL**) в текст прикладной программы включаются конкретные операторы SQL, и после компиляции исходной программы в выполняемый модуль жестко включаются соответствующие этим операторам функции SQL. Изменения в вызываемых функциях могут здесь определяться только изменениями параметров операторов SQL, иницируемых с помощью переменных языка программирования.

При динамическом использовании языка (**динамический SQL**) формирование SQL-запросов, соответствующие вызовы SQL-функций для обращения к базе данных осуществляется динамически в ходе выполнения программы.

Еще одним способом динамического формирования SQL-запросов в прикладной программе является обращение к соответствующим SQL-функциям с помощью специальных **интерфейсов программирования приложений** (библиотек функций, разработанных для связи прикладной программы и СУБД посредством SQL-запросов).

В настоящем пособии для всех указанных технологий (разновидностей SQL) будут приведены основные идеи и рассмотрены ключевые концепции. Интерактивный SQL будет рассмотрен более подробно, чем программный. Детальное рассмотрение статического, динамического SQL и различных API-интерфейсов (ODBC, JDBC, DB Library и др.) выходит за рамки нашего курса.

Краткие итоги: В лекции рассматриваются общие принципы организации программного обеспечения работы с реляционными базами данных, включающего

- создание и ведение базы данных;
- создание пользовательских приложений, включающих разработку пользовательского интерфейса по работе с базой данных.

Рассматриваются подходы к организации доступа к данным (навигационный подход и подход, основанный на использовании интерпретируемых языков запросов). Дается общее *представление* о языке SQL (история возникновения и стандарты языка SQL, достоинства языка SQL, основная терминология, технологии работы).

По языку SQL написано достаточно много литературы. Для более подробного знакомства можно указать, в частности [[3.1](#)] - [[5.4](#)].

Тема 7.2. Основные операторы языка SQL. Интерактивный SQL. Использование языка SQL в прикладных программах

Цель лекции: дать общую характеристику операторов языка SQL и показать, как записываются основные запросы к базе данных на языке SQL (в интерактивном режиме).

12.1. Общее представление об основных операторах языка SQL

Как уже отмечалось в "[Программное обеспечение работы с современными базами данных](#)", все *операторы* языка SQL разделяются на три составные части: DDL – язык определения данных, DCL – язык управления данными, DML – язык обработки данных.

Приведем примеры основных операторов из вышеуказанных частей (без описания синтаксиса). Описание синтаксиса операторов SQL можно посмотреть в многочисленных книгах по языку SQL, в меню "Справка" конкретных СУБД.

Операторы разграничения доступа пользователей к объектам базы данных (DCL).

GRANT – создание в системе безопасности записи, разрешающей пользователю работать с данными или выполнять определенные *операции SQL*.

DENY - создание в системе безопасности записи, запрещающей *доступ* для определенной учетной записи.

Операторы определения данных (язык DDL).

Соответствующие *операторы* предназначены для создания, удаления, изменения основных объектов модели данных реляционных СУБД: таблиц, представлений, индексов.

CREATE TABLE <имя> - создание новой таблицы в базе данных.

DROP TABLE <имя> - удаление *таблицы* из базы данных.

ALTER TABLE <имя> - изменение структуры существующей таблицы или ограничений целостности, задаваемых для данной таблицы.

При выполнении аналогичных операций с представлениями или индексами в указанных операторах вместо служебного слова **TABLE** записывается слово **VIEW** (*представление*) или слово **INDEX** (*индекс*).

Операторы манипулирования данными (язык DML).

Операторы DML работают с базой данных и используются для изменения данных и получения необходимых сведений.

SELECT – выборка строк, удовлетворяющих заданным условиям. Оператор реализует, в частности, такие операции реляционной алгебры как "селекция" и "проекция".

UPDATE – изменение значений определенных полей в строках таблицы, удовлетворяющих заданным условиям.

INSERT – вставка новых строк в таблицу.

DELETE – удаление строк таблицы, удовлетворяющих заданным условиям. Применение этого оператора учитывает принципы поддержки целостности, поэтому он не всегда может быть выполнен корректно.

12.2 Интерактивный режим работы с SQL (интерактивный SQL)

Соответствующий режим предусматривает непосредственную работу пользователя с базой данных по следующему алгоритму: используя прикладную программу (клиентское приложение) или стандартную утилиту, входящую в СУБД, пользователь:

- устанавливает соединение с БД (подтверждая наличие прав доступа);
- вводит соответствующий оператор SQL, при необходимости в режиме диалога вводит дополнительную информацию;
- инициирует выполнение команды.

Текст запроса поступает в СУБД, которая:

- осуществляет синтаксический анализ запроса (проверяет, является ли запрос корректным);
- проверяет, имеет ли пользователь право выполнять подобный запрос (например, пользователь, у которого определены права только на чтение, пытается что-то удалить);
- выбирает, каким образом осуществлять выполнение запроса – план выполнения запроса;
- выполняет запрос;
- результат выполнения отправляет пользователю.

Схема взаимодействия пользователя и СУБД с использованием интерактивного SQL приводится на рис. 12.1.

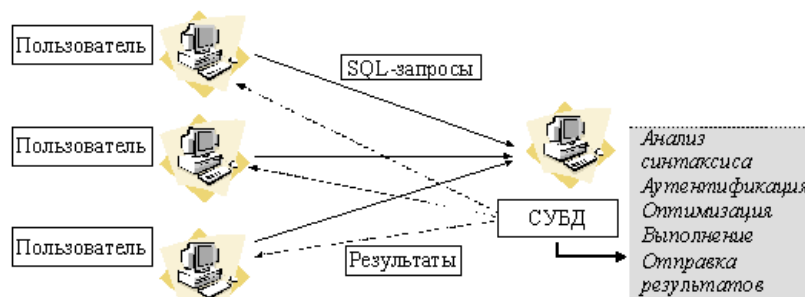


Рис. 12.1. Схема работы интерактивного SQL

12.3. Использование языка SQL для выбора информации из таблицы

Выборка данных осуществляется с помощью оператора **SELECT**, который является самым часто используемым оператором языка SQL. Синтаксис оператора **SELECT** имеет следующий вид:

```
SELECT [ALL/DISTINCT] <список атрибутов>/*
FROM <список таблиц>
[WHERE <условие выборки>]
[ORDER BY <список атрибутов>]
[GROUP BY <список атрибутов>]
```


[HAVING <условие>]

[UNION<выражение с оператором SELECT>]

В квадратных скобках указываются элементы, которые могут в запросе отсутствовать.

Ключевое слово **ALL** означает, что результатом будут все строки, удовлетворяющие условию запроса, в том числе и одинаковые строки. **DISTINCT** означает, что в результирующий набор не включаются одинаковые строки. Далее идет *список* атрибутов исходной таблицы, которые будут включены в таблицу-результат. Символ ***** означает, что в таблицу-результат включаются все атрибуты исходной таблицы.

Обязательным ключевым словом является слово **FROM**, за ним следуют имена таблиц, к которым осуществляется *запрос*.

В предложении с ключевым словом **WHERE** задаются условия выборки строк таблицы. В таблицу-результат включаются только те строки, для которых условие, указанное в предложении **WHERE**, принимает *значение истина*.

Ключевое слово **ORDER BY** задает операцию упорядочения строк таблицы-результата *по* указанному списку атрибутов.

В предложении с ключевым словом **GROUP BY** задается *список* атрибутов группировки (разъяснение этого и последующего ключевого слова будет представлено немного позднее).

В предложении **HAVING** задаются условия, накладываемые на каждую группу.

Отдельно отметим, что ключевые слова **FROM, WHERE, ORDER BY** используются аналогичным образом и в других операторах манипулирования данными языка **SQL**.

Рассмотрим реализацию запросов для конкретного примера, представленного в "[Использование формального аппарата для оптимизации схем отношений](#)" (см. [рис. 8.1](#))

Выдать список всех студентов.

SELECT *

FROM student

или

SELECT id_st, surname

FROM student

Заметим, что если добавить к данному запросу предложение **ORDER BY surname**, то *список* будет упорядочен *по* фамилии. *По* умолчанию подразумевается, что *сортировка* производится *по* возрастанию. Если необходимо упорядочение *по* убыванию, после имени атрибута добавляется слово **DESC**.

Выдать список оценок, которые получил студент с кодом "1".

SELECT id_st, mark

FROM mark_st

WHERE id_st = 1

Выдать список кодов студентов, которые получили на экзаменах хотя бы одну двойку или тройку.

В предложении **WHERE** можно записывать *выражение* с использованием *арифметических операторов* сравнения (<, >, и т.д.) и логических операторов (**AND, OR, NOT**) как и в обычных языках программирования.

SELECT id_st, mark

FROM mark_st

WHERE (MARK >= 2) AND (MARK <= 3)

Наряду с операторами сравнения и логическими операторами для составления условий в языке **SQL** (из-за специфики области применения) существуют ряд специальных операторов, которые, как правило, не имеют аналогов в других языках. Вот эти *операторы*:

- **IN** – вхождение в некоторое множество значений;
- **BETWEEN** – вхождение в некоторый диапазон значений;
- **LIKE** – проверка на совпадение с образцом;
- **IS NULL** – проверка на неопределенное значение.

Оператор **IN** используется для проверки вхождения в некоторое множество значений. Так, *запрос*

```
SELECT id_st, mark
FROM mark_st
WHERE mark IN (2,3)
```

дает тот же результат, что и вышеуказанный *запрос* (выведет идентификаторы всех абитуриентов, получивших хотя бы одну двойку или тройку на экзаменах).

Того же результата можно добиться, используя оператор **BETWEEN**:

```
SELECT id_st, mark
FROM mark_st
WHERE mark BETWEEN 2 AND 3
```

Выдать список всех студентов, фамилии которых начинаются с буквы А.

В этом случае удобно использовать оператор **LIKE**.

Оператор **LIKE** применим исключительно к символьным полям и позволяет устанавливать, соответствует ли *значение* поля образцу. Образец может содержать специальные символы:

- _** (символ подчеркивания) – замещает любой одиночный символ;
- %** (знак процента) – замещает последовательность любого числа символов.

```
SELECT id_st, surname
FROM student
WHERE surname LIKE 'A%'
```

Очень часто возникает необходимость произвести *вычисление* минимальных, максимальных или средних значений в столбцах. Так, например, может понадобиться вычислить средний балл. Для осуществления подобных вычислений **SQL** предоставляет специальные *агрегатные функции*:

- MIN** – минимальное значение в столбце;
- MAX** – максимальное значение в столбце;
- SUM** – сумма значений в столбце;
- AVG** – среднее значение в столбце;
- COUNT** – количество значений в столбце, отличных от NULL.

Следующий *запрос* считает среднее среди всех баллов, полученных студентами на экзаменах.

```
SELECT AVG(mark)
FROM mark_st
```

Естественно, можно использовать *агрегатные функции* совместно с предложением **WHERE**:

```
SELECT AVG(mark)
FROM mark_st
WHERE id_st = 100
```

Данный *запрос* вычислит средний балл студента с кодом 100 по результатам всех сданных им экзаменов.

```
SELECT AVG(mark)
FROM mark_st
WHERE id_ex = 10
```

Данный *запрос* вычислит средний балл студентов по результатам сдачи экзамена с кодом 10. В дополнение к рассмотренным механизмам язык **SQL** предоставляет мощный аппарат для вычисления агрегатных функций не для всей таблицы результатов запроса, а для разных значений по группам. Для этого в **SQL** существует специальная конструкция **GROUP BY**, предназначенная для указания того столбца, по значениям которого будет производиться группировка. Так, например, мы можем вычислить средний балл по всем экзаменам для каждого студента. Для этого достаточно выполнить следующий *запрос*:

```
SELECT id_st, AVG(mark)
FROM mark_st
```

GROUP BY id_st

Все это, как обычно, может быть совмещено с предложением **WHERE**. При этом, не вдаваясь в тонкости выполнения запроса внутри *СУБД*, можно считать, что сначала выполняется *выборка* тех строк таблицы, которые удовлетворяют условиям из предложения **WHERE**, а потом производится группировка и *агрегирование*.

Приведем *запрос*, который вычисляет средний балл *по* оценкам, полученным на экзамене с кодом 100, для каждого студента.

```
SELECT id_st, AVG(mark)
FROM mark_st
WHERE id_ex = 100
GROUP BY id_st
```

Заметим, что группировка может производиться более чем *по* одному полю.

Для запросов, содержащих секцию **GROUP BY** существует важное ограничение: такие запросы могут включать в качестве результата столбцы, *по* которым производится группировка, и столбцы, которые содержат собственно результаты агрегирования.

Для того чтобы форматировать *вывод*, существуют различные возможности **SQL**. Так, например, допустимым является включение текста в *запрос*. Рассмотрим пример того, как это делается:

```
SELECT 'Средний балл=', AVG(mark)
FROM mark_st
WHERE id_ex = 10
```

В результате данного запроса *пользователь* увидит не просто некоторое число, а число, сопровождаемое поясняющим текстом.

12.4. Использование SQL для выбора информации из нескольких таблиц

До сих пор мы рассматривали выбор информации из единственной таблицы. Можно запрашивать информацию из нескольких таблиц, реализуя описанные в соответствующем разделе учебника *реляционные операции*. Стоит упомянуть, что полное рассмотрение темы выходит за рамки данного учебника. Подробно этот вопрос можно изучить при помощи, например, [3.1], [11.2]. Рассмотрим некоторые примеры того, как это делается.

Как правило, в тех случаях когда возникает необходимость выбирать информацию из разных таблиц, они тем или иным образом связаны друг с другом, например отношениями один к многим или один к одному *по* некоторому полю.

Еще раз вернемся к примеру из "[Физические модели данных \(внутренний уровень\)](#)". Рассмотрим соответствующую *ER-диаграмму* ([рис. 12.2.](#)).

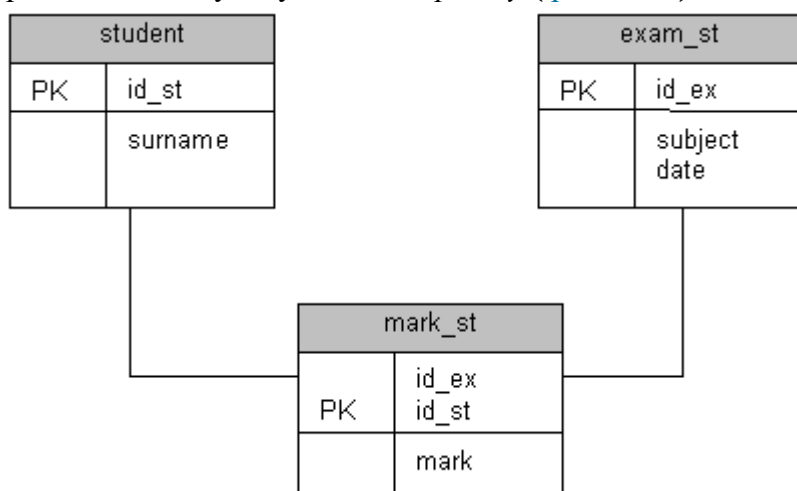


Рис. 12.2. Пример связанных таблиц

В этом примере тоже присутствуют связанные таблицы. Рассмотрим таблицы *student*, *mark_st* и *exam_st*.

Таблица *mark_st* связана с таблицей *exam_st* *по* полю **id_ex**.

Таблица mark_st связана с таблицей student по полю id_st.

Допустим, требуется распечатать список студентов с оценками, которые они получили на экзаменах. Для этого необходимо выполнить следующий запрос:

```
SELECT student.surname, mark_st.id_ex, mark_st.mark
FROM student, mark_st
WHERE student.id_st = mark_st.id_st
```

Отметим следующие изменения по сравнению с запросами к одной таблице.

1. В секции FROM указаны две таблицы.

2. Так как таблиц стало больше одной, появилась некоторая неоднозначность при упоминании полей. Так, во многих случаях неизвестно, из какой таблицы из списка FROM брать поле. Для устранения неоднозначности имена полей указываются с префиксом – именем таблицы. Имя таблицы от имени поля отделяется точкой.

3. В предложении WHERE указано условие соединения таблиц.

Нетрудно заметить, что использование префиксов-имен таблиц сильно загромождает запрос. Для того чтобы избежать подобного загромождения, используются псевдонимы. Так, можно переписать предыдущий запрос следующим образом:

```
SELECT E.surname, M.id_ex, M.mark
FROM student E, mark_st M
WHERE E.id_st = M.id_st
```

12.5. Использование SQL для вставки, редактирования и удаления данных в таблицах

Для добавления данных в таблицу в стандарте SQL предусмотрена команда INSERT. Рассмотрим ряд примеров запросов.

```
INSERT INTO mark_st
VALUES (1, 2, 5)
```

Данный запрос вставляет в таблицу mark_st строку, содержащую значения, перечисленные в списке VALUES. Если не нужно указывать значение какого-то поля, можно присвоить ему NULL:

```
INSERT INTO mark
VALUES (1, 2, NULL)
```

В случае если необходимо использование для некоторых полей значений по умолчанию, SQL позволяет явно указать, какие поля необходимо заполнить конкретными данными, а какие – значениями по умолчанию:

```
INSERT INTO mark_st (id_st, id_ex)
VALUES (1, 2)
```

Для удаления данных из таблицы существует команда DELETE:

```
DELETE
FROM student
```

Этот запрос удаляет все данные из таблицы student.

Можно ограничить диапазон удаляемой информации следующим образом:

```
DELETE
FROM student
WHERE surname > 'И'
```

Для обновления данных используется команда UPDATE.

```
UPDATE mark_st
SET mark = '5'
WHERE id_st = 100 AND id_ex = 10
```

При помощи этого запроса изменится на "5" оценка у студента с кодом 100 по экзамену с кодом 10.

12.5. Язык SQL и операции реляционной алгебры

Язык SQL является средством выражения мощного математического аппарата теории множеств и реляционной алгебры. В данном разделе рассматривается связь операторов языка SQL с операциями реляционной алгебры и теории множеств.

Операция объединения

Средствами языка SQL операция объединения представляется следующим образом:

```
SELECT *  
FROM A  
UNION  
SELECT *  
FROM B
```

Операция разности

Средствами языка SQL операция разности представляется следующим образом:

```
SELECT *  
FROM A  
EXCEPT  
SELECT *  
FROM B
```

Операция проекции

```
SELECT Fieldi1, ..., Fieldin  
FROM A
```

Операция выборки (селекции)

```
SELECT *  
FROM A  
WHERE (<condition>)
```

Операция пересечения

```
SELECT *  
FROM A  
INTERSECT  
SELECT *  
FROM B
```

Операция соединения, эквисоединения

```
SELECT A.Field1, ..., A.Fieldn, B.Field1, ..., B.Fieldm  
FROM A, B  
WHERE (A.Fieldi  $\Theta$  B.Field1)
```

Если Θ – операция "=", то это эквисоединение.

Операция естественного соединения

Пусть есть отношения $A(X_1, \dots, X_n, A_1, \dots, A_m)$ и $B(X_1, \dots, X_n, B_1, \dots, B_r)$.

```
SELECT A.X1, ..., A.Xn, A.A1, ..., A.Am, B.B1, ..., B.Br  
FROM A, B  
WHERE (A.X1 = B.X1) AND ... AND (A.Xn = B.Xn)
```

Краткие итоги: В лекции дается общая характеристика операторов языка *SQL*, используемых, в частности, для работы с базой данных в интерактивном режиме (создание таблиц, выбор информации из таблиц, добавление, удаление и модификация элементов). Дается понятие интерактивного режима работы с *SQL*. Рассматриваются основные *операторы SQL*, используемые для манипулирования данными (выбор информации из таблиц, добавление, удаление и модификация элементов). Приводятся примеры записи запросов к базе данных на языке *SQL* с использованием операторов *select, insert, update, delete*. Рассматривается *связь* между операциями *реляционной алгебры* и операторами языка *SQL*.

Более подробно материалы лекции рассматриваются в [[3.1](#)] - [[5.4](#)] .

Цель лекции: показать основные возможности формирования запросов к базе данных из прикладных программ.

13.1. Программный (встроенный) SQL

Основная работа с базой данных проводится с использованием прикладных программ, из которых и идут запросы к базам данных. В этом случае *интерактивный режим* работы не может быть использован, текст SQL-запроса должен быть либо включен в прикладную программу (если *запрос* полностью определен заранее), либо формироваться в процессе работы прикладной программы.

Программный SQL предназначен для того, чтобы встраивать SQL-запросы в прикладную программу, написанную на одном из языков программирования. При этом возникают следующие вопросы:

- компилятор с алгоритмического языка должен иметь возможность выделения в тексте прикладной программы последовательность операторов SQL.
- компилятор должен объединять возможности языка программирования высокого уровня (переменные, ветвления, циклы) и возможности SQL (запросы на языке, близком к естественному).

Решение этих проблем частично описано в стандарте SQL.

Рассмотрим *алгоритм* выполнения SQL-запросов в интерактивном режиме работы. Легко видеть, что *пользователь* вынужден ожидать результатов выполнения запроса в течение всего времени работы реализации SQL-запроса. Если через некоторое время пользователю снова нужно будет выполнить тот же самый *запрос*, СУБД вновь проделает те же самые действия, что и при предыдущем обращении. Налицо некоторое несовершенство механизма:

- одни и те же этапы выполняются каждый раз заново для одинаковых запросов;
- СУБД не может обрабатывать *интерактивные запросы* с опережением.

Решение подобных проблем очевидно – часть действий по обработке запроса необходимо выполнять один раз, сохранять результат в некотором виде, а потом использовать столько раз, сколько необходимо. Эта идея является одной из основных идей программного SQL. Таким образом, программный SQL позволяет:

- использовать операторы *интерактивного SQL* в тексте программы на языке программирования высокого уровня;
- наряду с операторами *интерактивного SQL* использовать новые специальные конструкции, дополняющие SQL и увеличивающие его возможности;
- для передачи параметров в запрос использовать в тексте запроса переменные, объявленные в программе;
- для возврата в программу результатов запроса использовать специальные конструкции, отсутствующие в интерактивном SQL;
- осуществлять компиляцию запросов совместно с программой, обеспечивая впоследствии согласованную работу программы и СУБД. Заранее (на этапе компиляции) выполнять действия по анализу и *оптимизации запросов*, экономя время, затрачиваемое на этапе выполнения программы.

На настоящий момент используются три варианта *встраивания* запросов на языке SQL в прикладную программу (программного SQL): *статический SQL*, *динамический SQL* и метод, основанный на различных *интерфейсах программирования приложений (API)*. Рассмотрим соответствующие варианты.

13.2. Статический SQL

Статический SQL – разновидность программного SQL, предназначенная для встраивания SQL-операторов в текст программы на языке программирования высокого уровня.

Основная особенность статического SQL определяется его названием: встраиваемые запросы должны быть четко определены на стадии написания прикладной программы, так как именно конкретный текст запросов вставляется в прикладную программу.

Рассмотрим два основных этапа, связанных с работой *статического SQL*, – *компиляция* программы и работа (выполнение) программы.

Схема компиляции и сборки программы выглядит следующим образом (рис. 13.1):

- Программа, включающая операторы языка программирования высокого уровня (ЯПВУ) вместе с операторами SQL, подается на вход специального препроцессора, который выделяет из нее части, связанные с SQL.
- Вместо инструкций *встроенного SQL* препроцессор подставляет вызовы специальных *функций СУБД*. Библиотеки таких функций для связи с языками программирования существуют для всех распространенных серверных СУБД. Стоит особо отметить, что эти библиотеки имеют "закрытый" интерфейс, т.е. разработчики библиотеки могут менять его по своему усмотрению, соответственно обновив препроцессор. Все это говорит о том, что программист не должен вмешиваться в этот процесс.
- Сами инструкции SQL препроцессор выделяет в отдельный файл.
- Программа поступает на вход обычного компилятора языка программирования, после чего получают объектные модули. Далее эти объектные модули вместе с *библиотеками СУБД* собираются в один исполняемый модуль – приложение.
- Наряду с этими операциями происходит работа с файлом, содержащим SQL-инструкции. В литературе этот модуль часто носит название "модуль запросов к базе данных" (Database Request Module, DBRM) [1]. Обработку этого модуля осуществляет специальная утилита, которая обычно носит название BIND. Для каждой инструкции SQL утилита выполняет следующие действия:
 - осуществляет синтаксический анализ запроса (проверяет, является ли запрос корректным);
 - проверяет, существуют ли в базе данных те объекты, на которые ссылается запрос;
 - выбирает, каким образом осуществлять выполнение запроса – *план выполнения* запроса;
- Все *планы выполнения* запросов сохраняются в СУБД для последующего использования.



Рис. 13.1. Схема компиляции программы с встроенными инструкциями статического SQL

Схема выполнения программы выглядит следующим образом (рис. 13.2.):

Программа запускается на выполнение обычным образом. При необходимости выполнить *запрос* программой осуществляется вызов специальной *функции СУБД*, которая

отыскивает уже сформированный ранее *план выполнения* запроса. *СУБД* выполняет *запрос* в соответствии с выбранным *планом*. Результат выполнения запроса поступает в *приложение*.

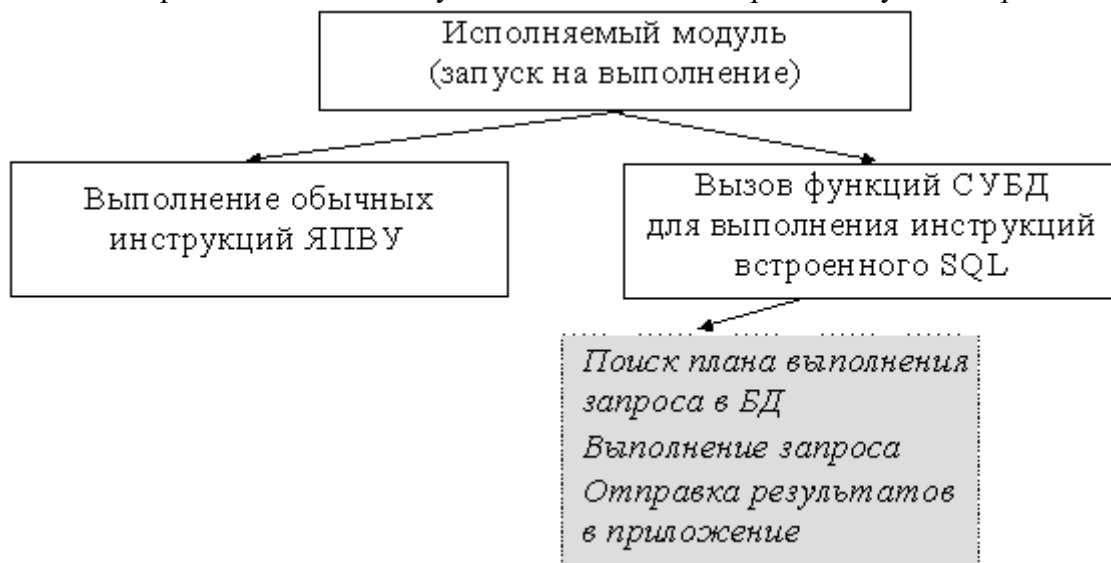


Рис. 13.2. Схема выполнения программы с встроенными инструкциями статического SQL

Для реализации вышеуказанных схем *статический SQL* должен содержать дополнительные *операторы* (по сравнению с *интерактивным SQL*), позволяющие компилятору выделить в тексте программы SQL-запросы, объявлять используемые в этих запросах таблицы, объявлять переменные для обработки ошибок, как результатов реализации запросов и т. п. Основные команды *статического SQL* приводятся в следующей таблице.

Таблица 13.1. Основные команды <i>статического SQL</i>	
EXEC SQL	Спецификатор, указывающий, что следующая за ним инструкция является инструкцией <i>встроенного SQL</i>
;	В языке C – признак окончания инструкции <i>встроенного SQL</i>
DECLARE TABLE	Объявляет таблицу, которая потом будет использоваться в инструкциях <i>встроенного SQL</i>
SQLCODE	Переменная для обработки ошибок
SQLSTATE	Переменная для обработки ошибок
GET DIAGNOSTICS	Инструкция для обработки ошибок
WHENEVER SQLERROR SQLWARNING NOT FOUND GOTO CONTINUE	Набор совместно используемых инструкций для упрощения обработки ошибок
BEGIN DECLARE SECTION END DECLARE SECTION	Инструкции для определения области, в которой будут объявлены переменные, впоследствии используемые в запросах SQL
INTO	Используется в операторе SELECT для указания переменной, в которую необходимо поместить результат выполнения запроса

DECLARE CURSOR	Курсор – специальный инструмент, предназначенный для обработки результатов запроса, содержащих более одной строки. Работа с курсором похожа на работу с файлами. Данная инструкция служит для создания курсора и связывания его с конкретным запросом
OPEN	Команда, открывающая курсор и побуждающая СУБД начать выполнение запроса. Устанавливает курсор перед первой строкой результата запроса
FETCH	Команда, перемещающая указатель текущей строки (курсor) на следующую строку. В некоторых СУБД <i>истандарте SQL-92</i> реализованы разные формы команды FETCH, перемещающие курсор на произвольную строку результатов запроса
CLOSE	Закрывает курсор и прекращает доступ к результатам запроса

Использование описанной выше схемы компиляции/сборки/выполнения программы позволяет:

- использовать SQL совместно с программой на языке программирования высокого уровня;
- заранее осуществлять проверку синтаксиса запросов и оптимизацию их выполнения (выбор плана). Понятно, что проверка синтаксиса выполняется быстро, но выбор *плана выполнения* – весьма трудоемкая процедура. Тот факт, что она выполняется один раз на этапе компиляции, позволяет говорить о существенном уменьшении накладных расходов.

Однако статическая разновидность программного SQL имеет некоторые существенные ограничения. Так, переменные в запросах могут использоваться только в тех местах, где в запросах обычно стоят *константы*. Например, нельзя задавать имя таблицы, из которой производится *выборка*, а также названия столбцов, как *параметр*. В связи с этим при использовании статического варианта вложенного (программного) SQL необходимо на этапе написания программы точно знать состав запросов, которые необходимо будет выполнять в прикладной программе. Во многих случаях это ограничение является существенным. Для его устранения была введена новая разновидность программного SQL – *динамический SQL*. Рассмотрим кратко основные идеи *динамического SQL*.

13.3. Динамический SQL

Динамический SQL – разновидность программного SQL, предназначенная для встраивания SQL-операторов в текст программы на языке программирования высокого уровня, допускающая динамическое формирование и выполнение запросов во время работы программы.

История возникновения *динамического SQL* во многом связана с компанией IBM, внедрившей этот мощный инструмент в свою СУБД DB2. Стандарты SQL, в частности SQL-1, не поддерживали *динамического SQL*. Лишь в 1992 году в стандарт SQL-2 были включены спецификации *динамического SQL*. Основной концепцией *динамического SQL* является следующее утверждение: *встроенная инструкция SQL* не записывается в исходный текст программы, вместо этого *программа* формирует текст инструкции во время выполнения в одной из своих областей данных, а затем передает сформированную инструкцию в СУБД для динамического выполнения [[3.1]].

Напомним, что при использовании *статического SQL* схема реализации подразумевала два этапа – компиляцию программы и выполнение программы. При этом на этап компиляции ложилась основная нагрузка. Именно здесь решались вопросы проверки, разбора и *оптимизации запросов*, поскольку *запрос* был заранее известен. Совершенно очевидно, что подобную двухэтапную схему нельзя реализовать для *динамического SQL*, так как на этапе компиляции программы *запрос* неизвестен. Поэтому проверку, разборку и *оптимизацию запросов* здесь приходится выполнять непосредственно во время работы программы. Таким образом, если эти *операции* в статическом SQL выполнялись *вовремя компиляции* один раз, то в *динамическом SQL* они будут выполняться столько раз для одного запроса, сколько раз он

будет сформирован в процессе работы прикладной программы. Это определяет существенный недостаток *динамического SQL* – низкую *производительность* по сравнению со статическим. Достоинство *динамического SQL* в том, что он позволяет формировать *запрос* к базе данных во время работы программы, реагируя на те или иные произошедшие события. Такая возможность является жизненно важной для клиент-серверной и *трехзвенной архитектур*, в которых структура *базы данных* и деловые правила имеют тенденцию к изменению, что требует определенной гибкости при организации процесса обработки данных.

Учитывая относительно низкую *производительность динамического SQL*, представляется правильным, там, где только возможно, рекомендовать использование статической разновидности *SQL*, применяя аппарат *динамического SQL* где это действительно необходимо.

Динамический SQL также должен содержать дополнительные *операторы* (по сравнению с *интерактивным SQL*). Основные *операторы динамического SQL* приводятся в следующей таблице.

Таблица 13.2. Основные команды <i>динамического SQL</i>	
EXECUTE IMMEDIATE	Немедленное выполнение инструкции
PREPARE	Подготовка инструкции к выполнению
EXECUTE	Выполнение подготовленной ранее инструкции
DESCRIBE	Специальная команда, участвующая при возврате результата выполнения инструкций <i>динамического SQL</i>
DECLARE CURSOR	Разновидность инструкции <i>DECLARE CURSOR</i> , применявшейся ранее в рамках <i>статического SQL</i> , содержащая вместо запроса его имя (связанное с запросом при помощи инструкции PREPARE)
OPEN FETCH CLOSE	Разновидности инструкций для работы с курсором в <i>динамическом SQL</i>

Рассмотрим схему функционирования *динамического SQL* (рис. 13.3). Схема предусматривает одноэтапное и двухэтапное выполнение инструкций.

Одноэтапное выполнение инструкций осуществляется командой **EXECUTE IMMEDIATE**.

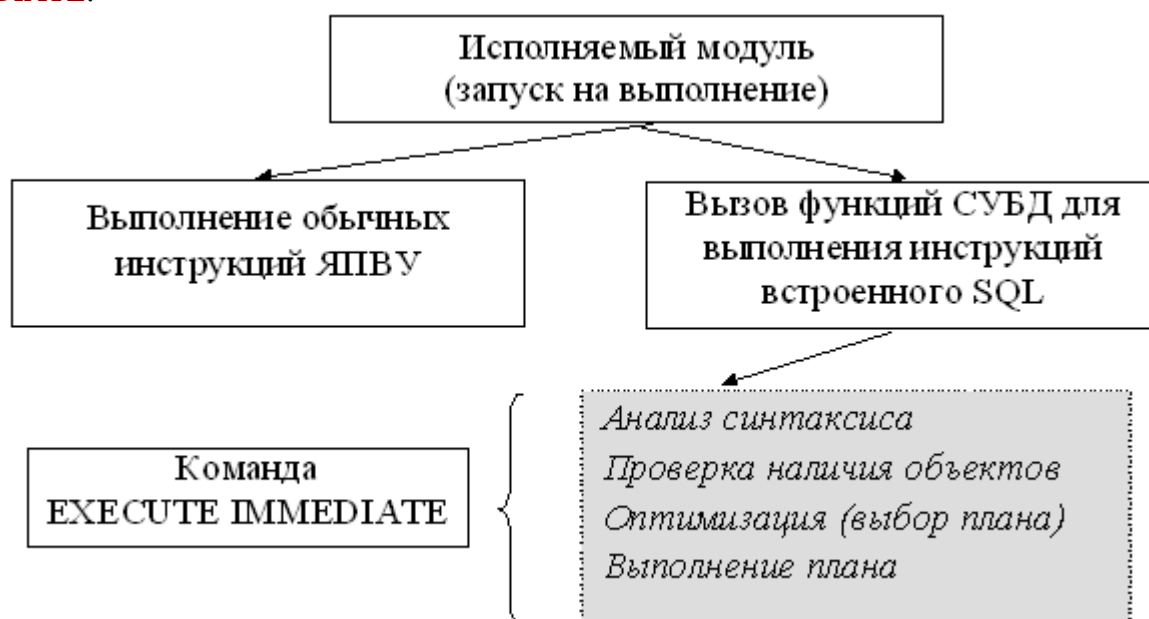


Рис. 13.3. Схема выполнения программы со встроенными инструкциями динамического SQL с применением одноэтапной схемы

Схема выполнения инструкции подразумевает:

- динамическое формирование команды SQL в строковом виде во время работы программы;
- передачу строкового вида инструкции в СУБД при помощи команды EXECUTE IMMEDIATE;
- выполнение инструкции системой управления БД, включающее синтаксический анализ, проверку параметров, оптимизацию (выбор плана) и выполнение этого плана.

Основные проблемы одноэтапной схемы заключаются в том, что она не позволяет выполнять инструкции SELECT (ибо нет средств для возврата в *приложение* результатов запроса) и приводит к нерациональному расходованию вычислительных ресурсов (т.к. при повторном выполнении той же инструкции вновь будет затрачено время на все те же действия по ее интерпретации и выполнению).

Двухэтапное выполнение инструкций основано на следующем соображении: скорее всего, команда динамического SQL в таком виде, как она поступает на выполнение, будет выполняться неоднократно. При этом могут меняться какие-то конкретные детали. А это значит, что инструкцию можно параметризовать. Использование параметризованных инструкций позволяет сделать схему выполнения двухэтапной, разделив процесс на "подготовку инструкции" и "выполнение инструкции" (рис. 13.4).



Рис. 13.4. Схема выполнения программы со встроенными инструкциями динамического SQL с применением двухэтапной схемы

На этапе подготовки можно осуществить синтаксический анализ инструкции, интерпретировать ее и подготовиться к выполнению, выбрав план выполнения.

На этапе выполнения СУБД подставляет значения параметров (полученные из программы) и использует сформированный ранее план выполнения для достижения результата.

При этом реализуется идея однократного выполнения тех действий, которые можно выполнить один раз. Так, подготовленная одна инструкция может быть выполнена десятки раз с разными параметрами.

13.4. Интерфейсы программирования приложений (API). DB-Library, ODBC, OCI, JDBC

Как замечено выше, программный SQL отличается от обычной, интерактивной формы наличием некоторых специальных инструкций, а также механизмом трансляции и выполнения запросов. Таким образом, для применения программного SQL в тексте своих программ программистам необходимо ознакомиться с некоторым специфическим набором инструкций. Стоит заметить, что в разных СУБД эти наборы инструкций, вообще говоря, могут несколько отличаться друг от друга. В результате возникает некоторая проблема, связанная с непереносимостью программы.

Наряду с описанным выше механизмом существует и активно применяется еще один подход, связанный с наличием специальных интерфейсов – API (application programming interface – интерфейс программирования приложений). Эти API представляют собой библиотеки функций, разработанные для обеспечения связи прикладной программы с СУБД посредством выполнения SQL-запросов. Прикладная программа вызывает специальные функции библиотеки для передачи в СУБД SQL-запроса в текстовом виде и для получения результатов выполнения запросов, а также различной служебной информации.

Применение подобного подхода приводит к тому, что программистам более не требуется изучать специальные наборы инструкций SQL, а необходимо лишь изучить специальную библиотеку функций. С учетом того, что механизм использования API является широко используемым и стандартным подходом (чего только стоит использование мощного аппарата Windows API), для специалистов нет ничего нового в изучении еще одной библиотеки, в данном случае – для общения с СУБД.

Кроме этого, программа, содержащая вызовы некоторых функций специализированной библиотеки, ничем не отличается по схеме компиляции и выполнения от обычной программы. Так, подобная программа не требует применения специализированного препроцессора с механизмом раздельной компиляции. Может показаться, что подход, связанный с использованием библиотек API, является наиболее прогрессивным, на самом же деле такой вывод вряд ли верен. Так, на настоящий момент очень активно используются и динамический SQL и библиотеки API. В каждом из этих подходов существуют свои достоинства, недостатки и границы разумной применимости. Как обычно, выбор того, каким из подходов воспользоваться, лежит на административной группе разработчиков базы данных, которая принимает решения в зависимости от особенностей конкретной задачи и имеющихся специалистов.

В данном разделе рассматривается подход, основанный на интерфейсе программирования приложений.

Посмотрим, как работают прикладные программы, использующие различные API. Принципы работы разных библиотек аналогичны. Схема работы приложения совместно с SQL API выглядит следующим образом [3.1]:

- программа получает доступ к базе данных путем вызова одной или нескольких API-функций, подключающих программу к СУБД и к конкретной базе данных;
- для пересылки инструкций SQL в СУБД программа формирует инструкцию в виде текстовой строки и затем передает эту строку в качестве параметра при вызове API-функции;
- программа вызывает выполнение API-функции для проверки состояния переданной в СУБД инструкции и обработки ошибок;
- если инструкция SQL представляет собой запрос на выборку, то, вызывая API-функции, программа записывает результаты запроса в свои переменные; обычно за один вызов возвращается одна строка или столбец данных;
- свое обращение к базе данных программа заканчивает вызовом API-функции, отключающей ее от СУБД.

Из имеющихся для реализации SQL-запросов интерфейсов API на настоящий момент выделилось несколько библиотек, "стандартных" в том смысле, что они активно применяются

множеством разработчиков по всему миру. Данное пособие не является подробным руководством по всем этим библиотекам. Более того, для их профессионального освоения необходимо серьезное изучение соответствующей литературы. В рамках данного курса мы лишь приведем обзор этих библиотек, рассмотрим основные заложенные в них идеи. Подробно эти *SQL API* описаны, например в [[3.1](#)].

Протокол ODBC

ODBC (Open Database Connectivity – открытый доступ к базам данных) – разработанный компанией Microsoft универсальный интерфейс программирования приложений для доступа к базам данных [[3.1](#)].

Основной целью разработки протокола ODBC считается стандартизация механизмов взаимодействия с различными СУБД. Основная проблема, связанная с разработкой приложений, взаимодействующих с базами данных на основе специальных SQL API, состояла в том, что каждая СУБД имела собственный программный интерфейс доступа, каждый из них имел свои особенности и функционировал не совсем так, как другие. В связи с этим разработка приложения существенно зависела от используемой СУБД. Компания Microsoft сделала важный шаг для решения этой проблемы. Основная идея заключалась в разработке универсального интерфейса на уровне семейства операционных систем Windows, который мог бы быть поддержан в разных СУБД.

Рассмотрим кратко структуру программного обеспечения ODBC [[3.1](#)]:

интерфейс вызовов функций ODBC: это так называемый верхний уровень ODBC, содержащий API, который и используется непосредственно приложениями. Данный API реализован в виде библиотеки динамической компоновки DLL и входит в состав операционной системы Windows;

драйверы ODBC: это так называемый нижний уровень ODBC, содержащий набор драйверов для СУБД, поддерживающих протокол ODBC. В рамках технологии для каждой СУБД может быть разработан соответствующий ODBC-драйвер, который будет являться промежуточным звеном между прикладной программой и СУБД, транслируя вызовы *функций СУБД* в вызовы внутренних специализированных *функций СУБД*. Таким образом решается проблема стандартизации. Для многих современных СУБД существуют специализированные *драйверы ODBC*, отдельно устанавливаемые в операционную систему;

диспетчер драйверов ODBC: данный программный механизм представляет средний уровень ODBC, управляя процессом загрузки необходимых драйверов.

Схема выполнения программы с использованием протокола ODBC для доступа к данным приводится на [рис. 13.5](#).

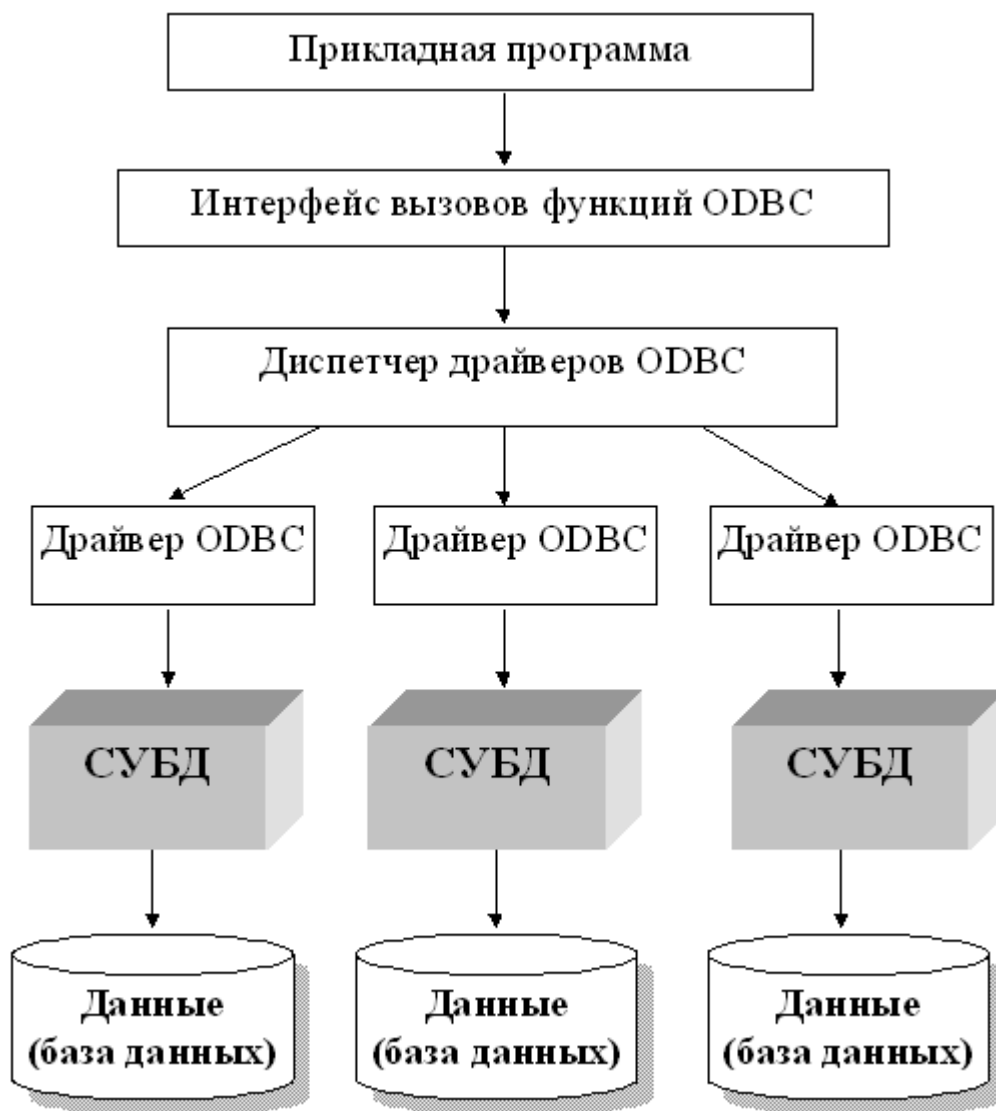


Рис. 13.5. Схема выполнения программы с использованием протокола ODBC для доступа к данным

Перечень некоторых базисных функций ODBC API приводится в следующей таблице.

Таблица 13.3. Базисные функции ODBC API			
Назначение	Функция	Описание	
Соединение с источником данных	SQLAllocEnv	Получает указатель окружения. Одно окружение может служить для создания нескольких соединений.	
	SQLAllocConnect	Получает указатель соединения.	
	SQLConnect	Соединяется с указанным драйвером, используя имя источника данных, идентификатор пользователя и пароль.	
Подготовка SQL запросов	SQLAllocStmt	Размещает указатель выражения.	
	SQLPrepare	Подготавливает SQL выражение для дальнейшего использования.	
	SQLGetCursorName	Возвращает имя, связанное с указателем выражения.	
	SQLSetCursorName	Устанавливает имя курсора.	

	CursorName	
Выполнение запросов	SQLExecute	Выполняет заранее подготовленный запрос.
	SQLExecuteDirect	Выполняет запрос.
Выборка результатов и информации о результатах	SQLRowCount	Возвращает количество записей, задействованных в операциях вставки, удаления, модификации.
	SQLNumResultCols	Возвращает количество колонок в выбранном наборе данных.
	SQLDescribeCol	Описывает колонку в выбранном наборе данных.
	SQLGetColAttributes	Описывает атрибуты колонки в выбранном наборе данных.
	SQLGetTypeInfo	Присваивает место в памяти для колонки в выбранном наборе данных и указывает ее тип данных.
	SQLFetch	Возвращает несколько наборов данных.

Протокол JDBC

JDBC (Java Database Connectivity) представляет собой *API* для выполнения SQL-запросов к базам данных из программ, написанных на языке Java [[3.1](#)].

Рассмотрим основные принципы JDBC.

С развитием глобальных сетей, в частности Интернета, и всех сопутствующих технологий стали появляться новые языки, специально предназначенные для работы в новых условиях. Одним из таких языков является *язык программирования Java*. В настоящее время Интернет-приложения занимают существенное место на рынке, работая в рамках 2-, 3- и многозвенной архитектуры. При этом значение языка Java как средства создания приложений, работающих с базами данных, существенно возрастает. Именно это и явилось одной из основных причин разработки нового программного интерфейса – JDBC. Первоначально интерфейс JDBC был разработан компанией Sun Microsystems, в настоящий момент этот *API* поддерживается всеми ведущими коммерческими СУБД.

Известно несколько различных версий JDBC. Так, версия 1.0 содержала некоторые средства доступа к данным:

- диспетчер драйверов (для подключения к разным СУБД);
- механизм управления сеансами (для одновременной работы с несколькими СУБД);
- механизм передачи инструкций SQL на выполнение в СУБД;
- механизм работы с курсорами (для передачи результатов выполнения запросов из СУБД в приложение).

Этот перечень определенным образом напоминает аналогичный функциональный аппарат протокола ODBC.

Версия JDBC 2.0 содержит существенные отличия. Так, вследствие увеличения возможностей интерфейса было проведено его идеологическое разделение на две основные части: Core API (основные возможности) и Extensions API (так называемые расширения).

В [[3.1](#)] указаны следующие возможности JDBC:

- **Пакетные операции.** Программа на Java может осуществить обновление базы данных в пакетном режиме, т.е. одна функция JDBC может добавить в базу данных несколько записей, что положительно сказывается на производительности программ.
- **Курсоры с произвольным доступом.** В JDBC 2.0 существует средство, позволяющее перемещаться по результатам запроса произвольным образом.

- **Обновляемые курсоры.** В JDBC 2.0 курсоры, наряду с функцией возврата результата запроса, используются и при обновлении базы данных. Обновления производятся при добавлении или изменении одной из строк в результатах запроса.

- **Организация связанного пула.** Несколько программ на языке Java могут пользоваться совместным доступом к базе данных, уменьшая затраты на подключения к базе данных и отключения от нее. Данный перечень можно продолжить (распределенные транзакции, поддержка JNDI и т.д.).

Версия JDBC 3.0 появилась совсем недавно и содержит такие новации, как объектно-реляционные расширения SQL и улучшенные механизмы обработки транзакций. Архитектура JDBC берет свое начало от ODBC и в существенной части повторяет ее, поэтому схема выполнения программы на Java с использованием протокола JDBC для доступа к данным полностью аналогична схеме на [рис. 13.5](#) (слова ODBC заменяются на слова JDBC). В отличие от ODBC, драйверы JDBC подразделяются на четыре типа. Основные отличия между этими типами связаны с местонахождением API СУБД (на клиентской или серверной СУБД) и способом доступа к базе данных (через собственный API СУБД или через ODBC).

Библиотека DB-Library

Библиотека DB-Library реализует *интерфейс программирования приложений* для совместной работы с широко распространенной СУБД Microsoft SQL Server. Данная библиотека является весьма обширной и содержит более 100 функций. Основными из них являются:

dblogin(); dbopen() – подключение к БД;

- **dbopen(); dbexit()** – установка/разрыв соединения с БД;
- **dbcmd()** – передача инструкции (пакета инструкций) SQL в СУБД в текстовом виде;
- **dbSQLexec()** – требование к СУБД выполнить текущий пакет инструкций;
- **dbcancel()** – прекращение выполнения пакета инструкций SQL;
- **dbresults()** – получение результатов выполнения очередной инструкции SQL в текущем пакете;
- **dbbind(), dbdata(), dbnextrow(), dbnumcols(), dbdatlen()** и др. – обработка результатов запросов на выборку данных.

Логика работы прикладной программы, обрабатывающей данные, хранящиеся в базе данных под управлением Microsoft SQL Server, выглядит следующим образом:

- при помощи указанных выше функций (**dblogin(), dbopen()**) прикладная программа формирует сведение об авторизации и пытается установить соединение с СУБД;
- при помощи СУБД программа открывает конкретную базу данных, с которой будет происходить работа (**dbopen()**);
- при помощи специальной функции (**dbcmd()**) программа передает в СУБД текст SQL-инструкции, которую далее необходимо будет выполнить; в библиотеке DB-Library поддерживается так называемый пакетный режим работы. Данный режим подразумевает возможность создания пакетов инструкций. Так, вызывая функцию **dbcmd()** несколько раз, вы можете передать в СУБД текст нескольких команд SQL, которые впоследствии будут выполнены как одна команда;
- используя функцию **dbSQLexec()**, программа вызывает выполнение инструкций, переданных ранее при помощи вызовов функций **dbcmd()** ;
- вызывая функцию **dbresults()**, программа может определить, удалось ли СУБД выполнить очередную инструкцию (как правило, число вызовов **dbresults()** соответствует числу инструкций в очередном пакете);
- в случае если мы имеем дело с запросом, возвращающим набор строк в качестве результата (запросом на выборку), программа при помощи вызовов функции **dbbind()** осуществляет связывание каждого поля результатов запроса с некоторой областью оперативной памяти. Далее при помощи функции **dbnextrow()** программа выполняет переход к следующей строке результатов запроса, что приводит к помещению в буфер новых данных;

- при помощи функции **dbexit()** программа разрывает соединение с базой данных. Библиотека DB-Library представляет собой большой и сложный механизм. Так, в библиотеке предусмотрены специальные механизмы обработки ошибок, разные способы передачи результатов выполнения запросов в прикладную программу и т.д.

Краткие итоги: В лекции рассматриваются разные технологии формирования запросов на языке *SQL* в прикладных программах (программный *SQL*). Дается понятие *статического SQL*, *динамического SQL* и приводятся соответствующие основные *операторы*. Рассматриваются *интерфейсы программирования приложений (API)* (протокол *ODBC*, протокол *JDBC*, библиотека DB-Library).

РАЗДЕЛ 8. Физическое проектирование БД

Цели занятия.

Постановка задачи проектирования, предметная область будущей БД. Анализ предметной области, концептуальное (инфологическое) проектирование БД. Построение логической модели данных. Физическое проектирование БД. Синтез компьютерной модели объекта.

4. Структура лекционного занятия.

№ п/п	Содержание (кратко)	Методы и средства обучения
1	Основные черты в новых направлениях развития теории и практики создания баз данных (новые свойства, присущие объектно-ориентированным и распределенным базам данных) и хранилищ данных	1. Персональные компьютеры; 2. Средства доступа в Интернет; 3. Проектор.

5. Содержание лекционного занятия и взаимодействие с аудиторией.

Введение.

Формулирование темы занятия. Использование вводного примера, описание ситуации, демонстрация какого-либо процесса и т. д. (в зависимости от вида, замысла лекции и её структуры)

Тема 8.1. Направления развития баз данных

Цель лекции: выделить основные черты в новых направлениях развития теории и практики создания баз данных (новые свойства, присущие объектно-ориентированным и распределенным базам данных) и хранилищ данных.

14.1. Объектно-ориентированный подход к организации баз данных

В начале 90-х годов XX века начались активные попытки по внедрению объектно-ориентированных технологий в отрасль проектирования и разработки баз данных. Бытовала точка зрения о том, что соответствующие технологии быстро вытеснят все остальные, так же как и во многих других программистских отраслях, но ничего подобного не произошло.

Объектно-ориентированное программирование

Рассмотрим термин "объектно-ориентированное программирование". Заметим, что это термин, принятый преимущественно в российской литературе. В западной литературе [[14.2](#)] под этим понимается сразу три аспекта:

- Объектно-ориентированный анализ – *OOA, object-oriented analysis*. **Объектно-ориентированный анализ – это методология, при которой требования к системе**

воспринимаются с точки зрения классов и объектов, выявленных в предметной области.

- *Объектно-ориентированное проектирование* – OOD, object-oriented design. **Объектно-ориентированное проектирование** – это методология проектирования, соединяющая в себе процесс объектной декомпозиции и приемы представления логической и физической, а также статической и динамической моделей проектируемой системы.

- *Объектно-ориентированное программирование* – OOP, object-oriented programming. **Объектно-ориентированное программирование** – это методология программирования, основанная на представлении программы в виде совокупности объектов, каждый из которых является экземпляром определенного класса, а классы образуют иерархию наследования.

Здесь и далее по тексту условимся не отступать от традиций и понимать под объектно-ориентированным программированием (ООП) сразу три указанных выше аспекта.

Основой объектно-ориентированной технологии является так называемая объектная модель, которая возникает как результат объектно-ориентированной декомпозиции. Она выделяет основные абстракции предметной области, определяет классы абстракций и выясняет, какими данными (атрибутами) описывается каждая абстракция, какую функциональность эти абстракции должны обеспечивать. В отличие от традиционных технологий программирования объектно-ориентированная технология представляет программу как совокупность классов объектов, взаимодействующих друг с другом.

Объект – конкретная материализация абстракции; сущность с хорошо определенными границами, в которой инкапсулированы состояние и поведение.

Объект ООП – инкапсулированная структура, имеющая атрибуты и методы.

Термин "инкапсулированная структура" означает, что объект является самодостаточным, программы, внешние по отношению к объекту, ничего "не знают" о его структуре и такое "знание" им не требуется. "Внешний" вид объекта называется его интерфейсом.

В таком понимании *объект* – это черный ящик, нам неизвестно, что у него внутри, мы лишь можем вызвать его методы и только через них взаимодействовать с ним. Кроме этого, объекты могут принадлежать иерархии "от общего к частному", которая реализуется путем наследования. Инкапсулированные состояния объекта могут быть как простыми типами данных, так и другими объектами, или даже массивами объектов. Каждый объект содержит определенную совокупность методов, классы взаимодействуют друг с другом посредством механизма сообщений. Объекты идентифицируются с помощью специальных указателей – дескрипторов. Методы объектов ООП представляют собой последовательности инструкций, выполняемых объектом. Например, у объекта может быть метод, отображающий данный объект, создающий данный объект и изменяющий его.

Предметная область моделируется как множество классов взаимодействующих объектов. Объект характеризуется набором свойств, которые являются как бы его пассивными характеристиками, и набором методов работы с этим объектом. Работать с объектом можно только с использованием его методов. Атрибуты объекта могут принимать множество допустимых значений, набор конкретных значений атрибутов определяет состояние объекта. Используя методы работы с объектом можно изменять значение его атрибутов и тем самым как бы изменить состояние самого объекта. Множество объектов с одним и тем же набором атрибутов и методов образует класс объектов. Класс, объекты которого могут служить значениями атрибута объектов другого класса, называется доменом этого атрибута.

К числу основных идей объектно-ориентированной технологии, как правило, относят [14.1]: абстрагирование, инкапсуляцию, модульность, иерархичность, типизацию, полиморфизм, наследование.

Инкапсуляция ограничивает область видимости имени атрибута пределами того объекта, в котором оно определено. Смысл этого атрибута будет определяться тем объектом, в котором оно инкапсулировано.

Полиморфизм – способность одного и того же программного кода работать с разнообразными данными. Другими словами, он допускает возможность в объектах разных типов иметь методы (процедуры или функции) с одинаковыми именами. Во время выполнения объектной программы одни и те же методы оперируют с разными объектами в зависимости от типа аргумента.

Наследование. Допускается порождение нового класса на основе уже существующего класса, и этот процесс называется наследованием. В этом случае новый класс, называемый подклассом существующего класса, наследует все атрибуты и методы класса. В подклассе, кроме того, могут быть определены дополнительные атрибуты и методы. Различают случаи простого и множественного наследования. В первом случае подкласс может определяться только на основе одного класса, во втором случае – на основе нескольких классов. Набор классов образует иерархическую структуру.

Объектно-ориентированные базы данных

К настоящему моменту терминология еще не устоялась, существует много разных определений и трактовок. *Представляется, что объектно-ориентированная база данных (ООБД) – база данных, основанная на принципах объектно-ориентированной технологии.* К основным описательным моментам, связанным с ООБД, в литературе [[14.2](#)] относят:

- *объекты* (в ООБД любая сущность – объект и обрабатывается как объект); отметим, что здесь используется понятие "объект" объектно-ориентированного программирования, которое отличается от понятия "объект", рассматриваемого ранее в данном учебном пособии.
- *классы* (понятие "тип данных" реляционной модели заменяется понятиями "класс" и "подкласс");
- *наследование* (классы образуют иерархию наследования, заимствуя свойства друг друга);
- *атрибуты* (характеристики объекта моделируются его атрибутами);
- *сообщения и методы* (каждый класс имеет определенную совокупность методов, классы взаимодействуют друг с другом посредством механизма сообщений);
- *инкапсуляция* (внутренняя структура объектов скрыта);
- *идентификаторы объектов* – дескрипторы.

Схема представления объекта приводится на [рис. 14.1](#)

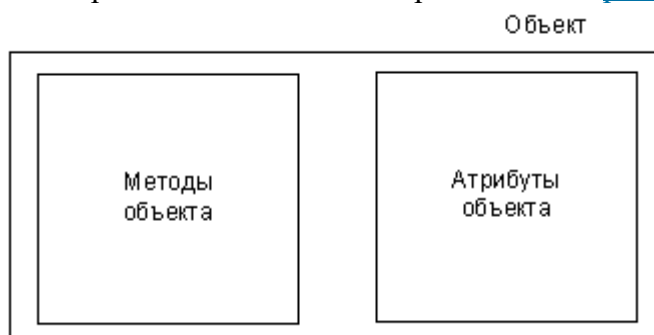


Рис. 14.1. Схема представления объекта

Система управления объектно-ориентированной базой данных называется объектно-ориентированной СУБД (ООСУБД). Цель ООСУБД – обеспечение постоянного хранения объектов, причем в отличие от традиционной СУБД ООСУБД должна хранить в составе объекта данные и программы.

Поскольку каждый объект данного класса имеет один и тот же набор методов, методы сохраняются только один раз – как *методы класса* (данные каждого экземпляра объекта хранятся отдельно).

Схема представления класса объектов приводится на [рис. 14.2](#)



Рис. 14.2. Схема представления класса объектов

Используя *наследование*, всем объектам ПОДРАЗДЕЛЕНИЕ можно приписать свойство объекта-родителя (ФАКУЛЬТЕТ) – название факультета, номер факультета. Схема представления объектов ФАКУЛЬТЕТ и ПОДРАЗДЕЛЕНИЕ приводится на [рис. 14.3](#).

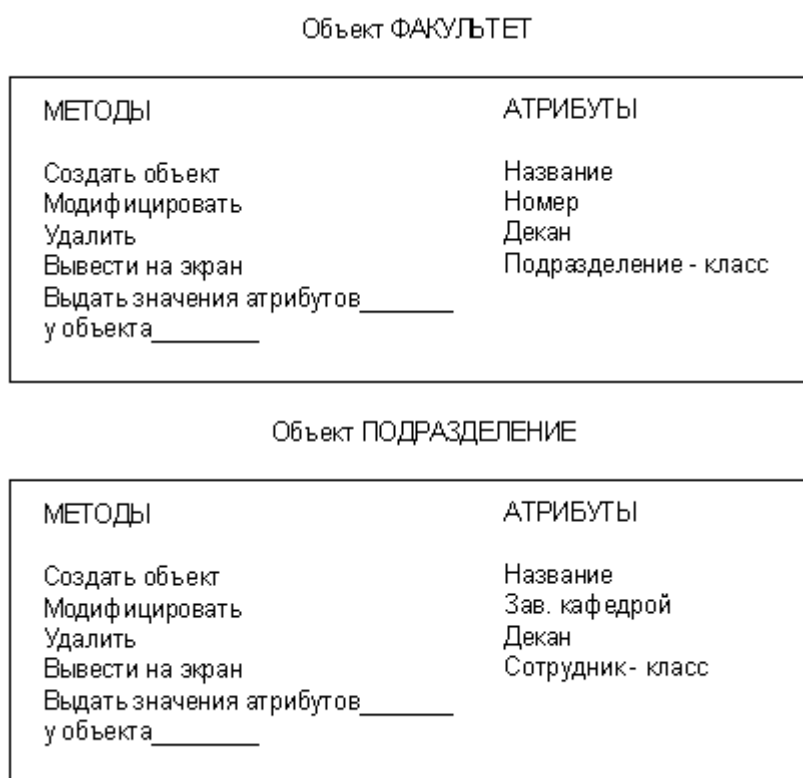


Рис. 14.3. Фрагменты представления конкретных объектов

Сравнивая объектно-ориентированный и реляционный подходы к БД, можно отметить следующие особенности. В реляционных БД (РБД) реальные объекты представляются как структуры, состоящие из набора *элементарных типов* данных. Такое представление имеет понятную интерпретацию – строка в плоской таблице. В том случае, когда специфика предметной области позволяет работать с такого рода приближением реальных объектов, РБД отлично справляются со своей задачей. Довольно часто реляционная модель и ее способ описания предметной области в виде набора плоских таблиц не отражают внутренней структуры для многих предметных областей, являются искусственными и становятся совершенно непонятными при увеличении количества таблиц. Основная причина несостоятельности реляционного подхода заключается в слишком сильной абстракции реального объекта, что ведет к потере семантики.

В отличие от реляционных баз данных *объектно-ориентированные базы данных* обладают простой и естественной связью с предметной областью, представляя ее структуру и состав, что облегчает проектирование и положительно сказывается на понимании

принципов функционирования программ. Так, в сложных неоднородных предметных областях использование ООБД (в частности, там, где разные объекты имеют разные методы) должно действительно упростить процесс проектирования и разработки.

К сожалению, в ООБД существуют свои проблемы. В ООБД отсутствует универсальная модель данных, и соответственно, отсутствует мощная математическая база, как, например, в реляционной модели. В связи с этим у ООБД нет языка запросов высокого уровня, аналогичного SQL, и при доступе к данным используется мало эффективный навигационный подход. ООСУБД отличаются от реляционных СУБД тем, что программный интерфейс создания приложения либо очень слаб, либо вообще отсутствует. Это означает, для написания приложения, работающего с ООБД, не существует мастеров и конструкторов (не считая, например, конструктора создания списка полей в объекте, который поставляется вместе с ООСУБД ObjectStore). Поэтому разработчик создает приложения на одном из алгоритмических языков.

По нашему мнению, существенным ограничением развития объектно-ориентированного подхода к созданию баз данных является то, что методы объекта содержатся внутри объекта и неразрывно связаны с ним. Это делает, по сути, невозможным создание для объектно-ориентированной базы данных соответствующей системы управления базой данных в традиционном понимании СУБД, функциями которой, в частности, является реализация операций обработки данных. Поэтому ООСУБД часто является не системой управления базами данных, а библиотекой программ, с помощью которой можно построить объектно-ориентированную базу данных. Примером такой библиотеки является ООСУБД ObjectStore. В связи с этим, возникает проблема реализации непредвиденных запросов.

Для перехода к объектно-ориентированным БД стандарт объектного программирования был дополнен стандартизованными средствами доступа к базам данных (стандарт ODMG 93; Object Database Management Group – группа управления объектно-ориентированными базами данных). К настоящему времени этот стандарт не реализован. Состояние проблемы подробно описано также в работах [[14.2](#)], [[2.1](#)], [[1.2](#)], [[6.5](#)], [[1.3](#)] и др.]. Отметим только, что ООБД используются, но пока не стали реальной альтернативой реляционным базам данных.

Объектно-ориентированные возможности появляются в ведущих современных СУБД, таких, как, например, Oracle. Предпринимаются попытки внесения изменений в стандарты языка SQL с целью его частичной адаптации к ООБД. Так, новый стандарт SQL-3 включает большой раздел, посвященный этому вопросу.

Объектно-реляционные СУБД

В настоящее время реляционные СУБД доминируют среди систем управления данными.

Преимущества объектно-ориентированного подхода для создания сложных специализированных приложений с одной стороны, и стремление разработчиков систем управления базами данных с другой стороны расширить границы применения соответствующих СУБД обусловили включение объектно-ориентированных компонент (расширяемая пользователем система типов, инкапсуляция, наследование, полиморфизм и т. п.) в модель данных реляционной СУБД. Соответствующие СУБД, называемые объектно-реляционными, соединяют в себе лучшие качества реляционных и объектно-ориентированных баз данных. Отметим, что в разных СУБД реализован разный набор из перечисленных объектно-ориентированных компонент. Таким образом, не существует общепринятой объектно-реляционной модели, а скорее имеется несколько таких моделей, поддерживающих определенный набор объектно-ориентированных компонент. Однако, основой всех таких моделей являются реляционные таблицы, используется язык запросов, включено понятие объекта, а в некоторых дополнительно реализована возможность сохранения методов в базе данных.

Соответствующие изменения реляционной модели обусловили необходимость расширения стандарта языка запросов SQL. Первый вариант такого стандарта получил название SQL3. Работа над стандартом продолжается и в настоящее время.

В качестве примера в максимальной степени объектно-ориентированной СУБД можно указать исследовательскую СУБД Postgres [[2.1](#)].

Отметим считающиеся объектными расширениями элементы СУБД Microsoft Server 2008.

- **Пользовательские расширения.** Пользователи имеют возможность вмешиваться в изначально предоставляемый СУБД инструментарий, создавая, в частности, новые *пользовательские типы данных*.
- **Хранение больших объемов данных.** Наряду с теми данными, которые хранились в БД традиционно, *Microsoft SQL Server 2008* позволяет хранить в столбцах таблицы данные больших размеров (поддерживаются соответствующие типы данных).
- **Новые, ориентированные на определенные классы объектов, типы данных.** В системе определены новые типы данных (*geometry, geography*), характерные для тех направлений, в которых объектно-ориентированный подход весьма эффективен и часто используется (картография и соответствующие приложения, геометрическое представление объектов самой разной природы).
- **Хранимые процедуры.** В определенном смысле хранимые процедуры также являются объектным расширением, осуществляя необходимые пользователю воздействия на данные (стандартный для ООП процедурный подход).

14.2. Распределенные базы данных

База данных – интегрированная совокупность данных, с которой работают много пользователей.

Изложение всех предыдущих разделов предполагало единую базу данных, размещаемую на одном компьютере. Напомним основные принципы, положенные в основу теории баз данных:

- централизованное хранение данных;
- централизованное обслуживание данных (ввод, корректировка, чтение, контроль целостности).

Заметим, что *базы данных* появились в период господства больших ЭВМ. *База данных* велась на одной ЭВМ, все пользователи работали именно на ЭВМ (возможные режимы работы описаны в "[Различные архитектурные решения, используемые при реализации многопользовательских СУБД. Краткий обзор СУБД](#)"). Других вариантов использования вычислительной техники в то время просто не существовало. Если проанализировать работу пользователей с данными в компаниях, организациях, предприятиях в "докомпьютерное" время, то нетрудно заметить, что на отдельных участках пользователи работали со "своими" данными (осуществляли сбор определенных данных, их хранение, обработку, передачу обработанных данных на другие участки или уровни управления).

У такой технологии были существенные недостатки, которые уже отмечались в предыдущих разделах: дублирование некоторых данных, отсутствие возможности сравнительного анализа данных всех участков. Однако у этой технологии были и существенные достоинства: данные вводились и хранились в местах их порождения; с этими данными работал *пользователь*, являющийся специалистом именно по этим данным, что позволяло ему вести эффективный *контроль* правильности данных на всех стадиях обработки; данные находились непосредственно у пользователя, что давало возможность их оперативной обработки. Централизация данных на одной ЭВМ, несомненно, дающая эффективные возможности хранения и обработки данных, не позволяла реализовывать вышеназванные достоинства.

Развитие вычислительных компьютерных сетей обусловило новые возможности в организации и ведении баз данных, позволяющие каждому пользователю иметь на своем компьютере свои данные и работать с ними и в то же время позволяющие работать всем пользователям со всей совокупностью данных как с единой централизованной базой данных. Соответствующая совокупность данных называется *распределенной базой данных*.

Термин "**распределенная база данных**" достаточно часто встречается в литературе [[14.2], [2.1], [1.2], [6.5], [1.3]]. Однако в разных источниках под этим термином понимаются совершенно разные вещи. Часть авторов понимают под *распределенной базой данных* то, что имеется удаленный *сервер*, на котором расположены данные, а также клиентские компьютеры, расположенные территориально в другом месте. Такая трактовка нам представляется неправильной. Настоящая **распределенная база данных** располагается на

нескольких компьютерах. При этом часть файлов расположена на одном компьютере, часть на другом и т.д. Более того, возможна и даже часто встречается ситуация, когда информация на этих компьютерах пересекается, дублируется.

Распределенная база данных – совокупность логически взаимосвязанных *разделяемых данных* (и описаний их структур), физически распределенных в компьютерной сети.

Система управления распределенной базой данных – программная система, обеспечивающая работу с распределенной базой данных и позволяющая пользователю работать как с его локальными данными, так и со всей базой данных в целом.

Система управления распределенной базой данных (РаСУБД) является распределенной системой. Каждый фрагмент *базы данных* работает под управлением отдельной *СУБД*, которая осуществляет *доступ* к данным этого фрагмента. Пользователи взаимодействуют с распределенной базой данных через локальные и глобальные приложения. Локальные приложения дают пользователю возможность работать со своими локальными данными и не требуют доступа к другим фрагментам. Глобальные приложения дают пользователю возможность работать с другими фрагментами *базы данных*, расположенными на других компьютерах сети. Общая схема распределенной *базы данных* представлена на [рис. 14.4](#).

Объединение данных организуется виртуально. Соответствующий подход, по сути, отражает организационную структуру предприятия (и даже общества в целом), состоящего из отдельных подразделений. Причем, хотя каждое подразделение обрабатывает свой набор данных (эти наборы, как правило, пересекаются), существует необходимость доступа к этим данным как к единому целому (в частности, для управления всем предприятием).

Одним из примеров реализации такой модели может служить *сеть Интернет*: данные вводятся и хранятся на разных компьютерах по всему миру, любой *пользователь* может получить *доступ* к этим данным, не задумываясь о том, где они физически расположены.

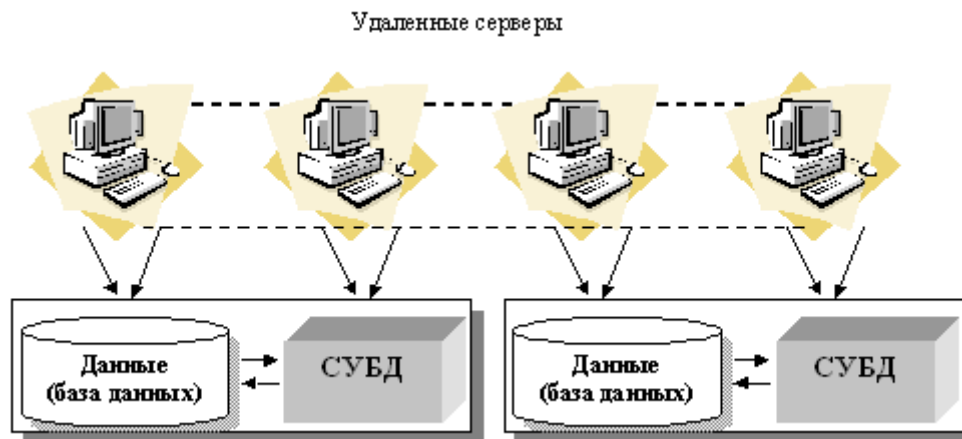


Рис. 14.4. Распределенная база данных

К.Дж. Дейт провозглашает следующий фундаментальный принцип распределенной *базы данных* [[2.1](#)]. Для пользователя распределенная система должна выглядеть точно так же, как нераспределенная. Из этого принципа следует ряд правил:

1. Локальная автономия.
2. Независимость от центрального узла.
3. Непрерывное функционирование.
4. Независимость от расположения.
5. Независимость от фрагментации.
6. Независимость от репликации.
7. Обработка распределенных запросов.
8. Управление *распределенными транзакциями*.
9. Независимость от аппаратного обеспечения.
10. Независимость от операционной системы.
11. Независимость от сети.
12. Независимость от СУБД.

Заметим, что понятие распределенной базы данных можно интерпретировать как следующий шаг в развитии понятий о данных (см. "Введение в базы данных. Общая характеристика основных понятий"), обусловленный распределенностью данных в реальных предметных областях, а также новым этапом развития средств вычислительной техники – широким использованием вычислительных сетей.

В этой интерпретации распределенную базу данных можно понимать как совокупность логически взаимосвязанных распределенных по разным компьютерам баз данных.

Перечислим основные проблемы создания распределенной базы данных.

1. Фрагментация данных и распределение по компьютерам.
2. Составление глобального каталога, содержащего информацию о каждом фрагменте БД и его местоположении в сети. (Каталог может храниться на одном узле или быть распределенным)
3. Организация обработки запросов (синхронизация нескольких запросов к одним и тем же данным, исключение аномалий удаления и обновления одних и тех же данных, расположенных на различных узлах, оптимизация последовательности шагов при обработке запроса и т.д.).

Значительным достоинством этой модели является приближение данных к месту их порождения, что позволяет существенно повысить их достоверность, недостатком – достаточно высокая сложность управления данными как единым целым.

К сожалению, процесс создания и обслуживания распределенных баз данных связан и с техническими трудностями, среди которых можно выделить жесткие требования к пропускной способности каналов связи, а также низкую производительность, обусловленную значительными затратами коммуникационных и вычислительных ресурсов при их синхронизации во время выполнения транзакций (особенно при интенсивных обращениях из разных узлов к одному фрагменту).

В задачу данного учебника не входит подробное изучение принципов построения распределенных баз данных. Интересующимся рекомендуем обратиться к соответствующей литературе, например [[2.1], [1.2], [6.5], [1.3]] Здесь мы хотим лишь обрисовать проблему и сделать некоторые выводы по перспективам ее решения. Технология, связанная с использованием распределенных баз данных, в наибольшей степени соответствует организационной человеческой деятельности (информация распределена по месту деятельности людей, и они обмениваются ей в процессе работы) и позволяет наиболее успешно решать важнейшие проблемы ведения баз данных:

- повысить достоверность информации (информация вводится в месте ее порождения лицом, которое лучше всех понимает ее смысловое значение);
- повысить оперативность локальной обработки информации (соответствующие вопросы решаются на локальном компьютере с фрагментом базы данных).

Поэтому очевидно, что задача проектирования, создания и функционирования распределенных баз данных является весьма существенной, активно изучается в настоящее время и будет решаться и далее.

14.3. Хранилища данных

Как уже неоднократно отмечалось, технологии баз данных предназначены, как правило, для решения текущих задач обработки данных организации. В базу данных постоянно вносятся изменения, то есть база данных отражает моментальный снимок определенной области деятельности предприятия. Для эффективного принятия решений руководством при управлении организацией важно не только знать текущее положение дел, но и иметь возможность анализировать динамику (изменение во времени) основных показателей, причем, зачастую из разных баз данных. Такую возможность дает технология так называемых хранилищ данных.

Приведем определение хранилища данных (Bill Inmon).

Хранилище данных – предметно-ориентированный, интегрированный, привязанный ко времени и неизменяемый набор данных, предназначенный для поддержки принятия решений.

Под **предметной ориентированностью** здесь понимается ориентированность на предметы (определенные группы данных), а не на конкретные приложения. Например, ориентация на данные о сотрудниках, а не только о расчете их заработной платы.

Под **интегрированностью** здесь понимается возможное *объединение* данных из разных источников (баз данных), имеющих разный формат и несогласованных.

Привязка ко времени предполагает, что для всех данных указан момент или промежуток времени, в который они корректны.

Данные в хранилище не изменяются, они лишь регулярно пополняются из оперативных баз данных.

Общая схема взаимодействия информационного хранилища и баз данных приводится на [рис. 14.5](#).

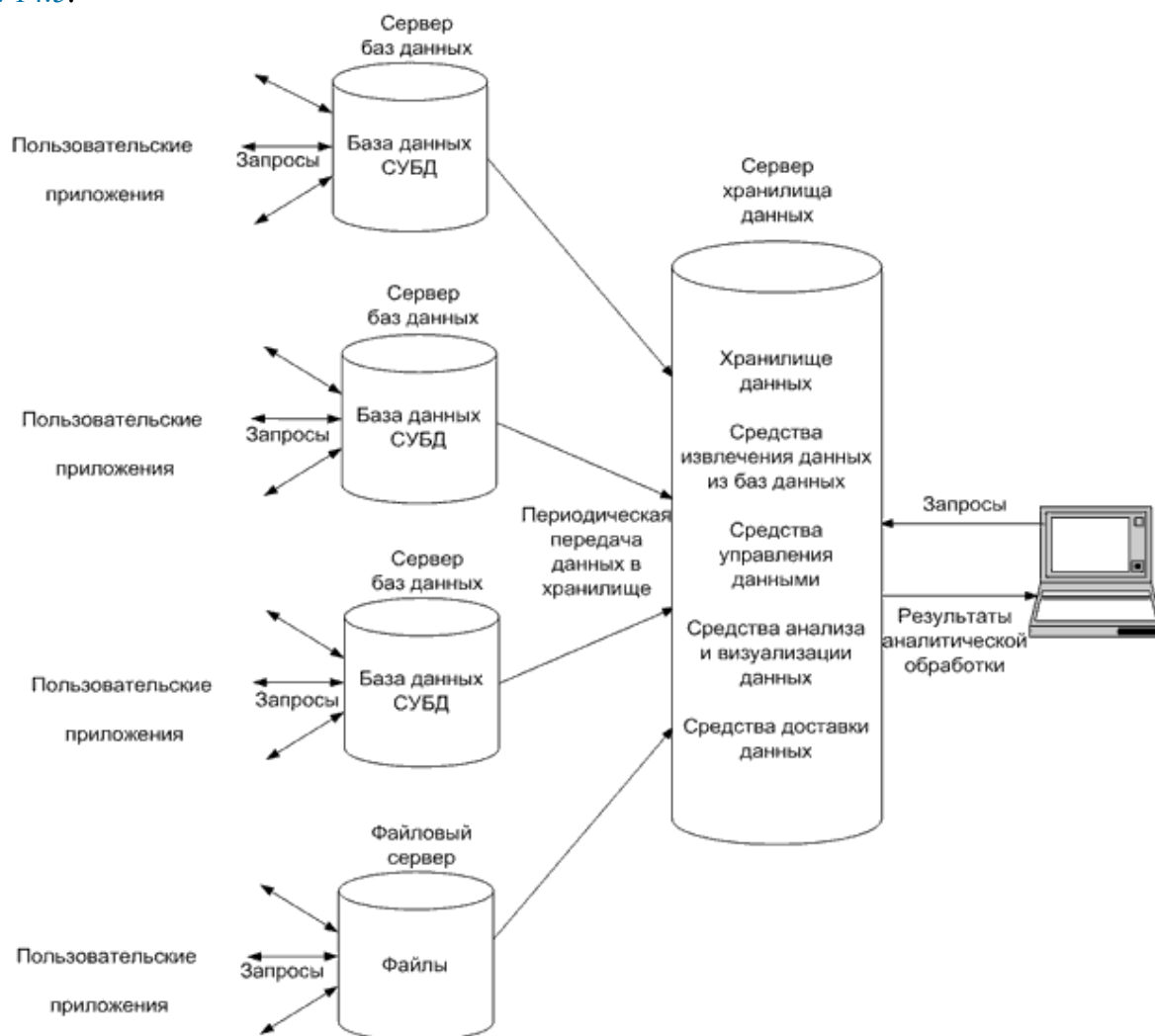


Рис. 14.5. Схема организации работы хранилища данных

Еще раз подчеркнем, что основной целью хранилищ данных является бизнес-анализ или информационная *поддержка* принятия управленческих решений.

Для реализации всей необходимой обработки информации в соответствии с этой схемой необходимы следующие *программные средства*:

- средства извлечения данных из баз данных;
- средства управления данными хранилища (система управления базой данных хранилища);
- средства анализа данных хранилища (используется OLAP-технология);
- средства доставки данных;
- средства визуализации результатов обработки для конечных пользователей.

Для работы соответствующих программных средств необходимо описание структуры содержимого информационного хранилища (метаописание).

Для самого общего случая, если данные берутся из баз данных, управляемых разными СУБД, из файлов разных типов, а данные разнородны, средства управления данными хранилища пока не созданы. Однако, если данные в информационное хранилище выбираются только из реляционных баз данных, то в качестве средств управления данными хранилища может быть взята мощная реляционная СУБД. Поэтому разработчики современных СУБД включают в состав программного обеспечения СУБД средства организации работы с хранилищами данных.

Рассмотрим в качестве **примера** *возможности СУБД Microsoft SQL Server 2008 для организации хранилищ данных.*

Microsoft SQL Server 2008 содержит в своем составе средства извлечения, преобразования и загрузки данных (SQL Server 2008 Integration Services), способные интегрировать данные из различных источников, проверять данные на допустимость и преобразовывать перед загрузкой в хранилище. Эти средства также способствуют перемещению данных, поддерживают текстовый анализ и нечеткий поиск. Нужно отметить также среду визуальной разработки (Business Intelligence Development Studio) для создания многомерных кубов, отчетов, пакетов извлечения, преобразования и загрузки данных.

Существенной особенностью хранилищ данных является их очень большой объем. Microsoft SQL Server 2008 как средство управления данными хранилища позволяет работать с большими объемами данных, причем для сокращения времени обработки предусмотрена поддержка параллельных вычислений (путем разделения таблиц и индексов на секции и обеспечение параллельной обработки секций). В системе предусмотрена возможность сжатия данных (таблиц), что позволяет уменьшить физический размер таблиц и существенно сокращает время обмена между оперативной и внешней памятью.

В качестве средств анализа данных хранилища используется SQL Server 2008 Analysis Services, применяемый для построения многомерных кубов (многомерных моделей данных). Это средство содержит семь эффективных алгоритмов анализа данных с целью поддержки принятия управленческих решений, в том числе анализ тенденций и статистический анализ данных.

В качестве средств представления аналитических данных пользователям предлагается использовать средство генерации отчетов SQL Server 2008 Reporting Services.

Таким образом, Microsoft SQL Server 2008 является эффективным средством реализации хранилищ данных на основе реляционных баз данных.

Краткие итоги: В лекции рассмотрены перспективные направления в теории и практике создания баз данных – объектно-ориентированные и распределенные базы данных. Здесь описываются основные идеи объектно-ориентированного программирования (объект, класс, методы класса, наследование) и их приложение к теории баз данных. Отмечены основные достоинства и недостатки объектно-ориентированных баз данных.

Рассматривается понятие распределенных баз данных как следующий шаг в развитии понятий о данных. Отмечены основные достоинства распределенных баз данных и проблемы, возникающие при их разработке.

Рассматривается понятие хранилища данных, в качестве примера системы управления данными хранилища приводится СУБД Microsoft SQL Server 2008.

**Приложение № 3 к методическим материалам
по дисциплине (модулю). Конспекты
лабораторных занятий по дисциплине
(модулю)**

КОНСПЕКТЫ ЛАБОРАТОРНЫХ ЗАНЯТИЙ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)

РАЗДЕЛ 1. ТЕОРИЯ БАЗ ДАННЫХ

ЗАДАНИЯ К ЛАБОРАТОРНЫМ ЗАНЯТИЯМ РАЗДЕЛА 1

Тема лабораторных занятий: Знакомство с Access

Перечень изучаемых элементов содержания

Назначение баз данных. Основные понятия теории баз данных: сущность, предметная область. Система управления базами данных (СУБД). Классификация баз данных по форме хранимой информации, по способу организации, по модели данных, по степени распределённости хранения и передачи данных, по содержанию. Классификация СУБД по используемой модели данных, по степени распределённости, по способу доступа к БД, по языкам общения, по числу уровней в архитектуре, по степени универсальности. Основные функции СУБД. Критерии качества баз данных.

Форма практического задания: лабораторный практикум.

Задания лабораторного практикума

- 1.1 «Основные понятия БД. Объекты Access»
- 1.2 «Ввод и редактирование данных»
- 1.3 «Сортировка, Поиск и Замена»
- 1.4 «Фильтрация»
- 1.5 «Фильтр по выделенному»
- 1.6 «Автофильтр»
- 1.7 «Расширенный фильтр»
- 1.8 «Запросы»
- 1.9 «Групповые операции»
- 1.10 «Перекрестные запросы»

РУБЕЖНЫЙ КОНТРОЛЬ К РАЗДЕЛУ 1

форма рубежного контроля – отчет к лабораторным работам с ответами на контрольные вопросы

1. Назначение БД
2. Виды связей между таблицами
3. Технология ввода и редактирования данных.
4. Сортировка данных
5. Поиск и замена данных.
6. Технология применения Автофильтра

7. Технология применения Расширенного фильтра.
8. Технология создания запроса на выборку
9. Групповые операции в запросах
10. Технология создания перекрестного запроса

РАЗДЕЛ 2. Общие принципы построения (архитектура) баз данных

Перечень изучаемых элементов содержания

Трехуровневая модель системы управления базой данных ANSI. Схемы баз данных. Внешний уровень представления информации в БД. Внутренний уровень представления информации в БД. Концептуальный уровень представления информации в БД. Независимость данных в БД. Процесс прохождения пользовательского запроса. Пользователи баз данных. Основные типы архитектуры баз данных с сетевым доступом.

ЗАДАНИЯ К ЛАБОРАТОРНЫМ ЗАНЯТИЯМ РАЗДЕЛА 2

Темы лабораторных занятий:

Проектирование БД

Форма практического задания: лабораторный практикум.

Задания лабораторного практикума

- 2.1 «Разработка инфологической модели и создание БД»
- 2.2 «Проектирование БД. Создание таблиц»
- 2.3 «Проектирование БД. Создание связей между таблицами»
- 2.4 «Средства контроля и автоматизации ввода данных»
- 2.5 «Создание экранных форм»
- 2.6 «Запросы на добавление данных»
- 2.7 «Запросы на удаление данных»

РУБЕЖНЫЙ КОНТРОЛЬ К РАЗДЕЛУ 2

форма рубежного контроля – отчет по лабораторной работе с ответами на контрольные вопросы.

1. Типы данных в БД
2. Понятие ключевого поля.
3. Создание схемы данных
4. Средства контроля ввода данных
5. Средства автоматизации ввода данных.
6. Создание списков.
7. Виды стандартных автоформ.
8. Создание подчиненных форм.
9. Технология создания запроса на добавление.
10. Технология создания запроса на удаление данных

РАЗДЕЛ 3. МОДЕЛИ ДАННЫХ

Перечень изучаемых элементов содержания

Понятие модели данных. Объектные модели данных. Общая классификация моделей данных. Уровни моделирования баз данных. Общие и специальные критерии оценки качества логической и физической моделей данных. Основные принципы построения БД - 12 правил Кодда. Отношения в РБД. Их основные понятия. Соотношение основных понятий реляционного подхода. Ключи переменной отношения. Целостность реляционных данных. Функциональные зависимости между атрибутами в отношениях РБД. Связи в реляционных БД. Универсальное отношение. Избыточность данных. Аномалии.

ЗАДАНИЯ К ЛАБОРАТОРНЫМ ЗАНЯТИЯМ РАЗДЕЛА 3

Темы лабораторных занятий:

Обработка данных

Форма практического задания: лабораторный практикум.

Задания лабораторного практикума

- 3.1 «Запросы на создание таблиц»
- 3.2 «Виды соединения таблиц в запросах»
- 3.3 «Запросы на обновление данных»
- 3.4 «Создание отчетов»
- 3.5 «Создание страниц доступа к данным»
- 3.6 «Макросы»
- 3.7 «Кнопочная форма»

РУБЕЖНЫЙ КОНТРОЛЬ К РАЗДЕЛУ 3

форма рубежного контроля – отчет по лабораторной работе с ответами на контрольные вопросы.

- 1. Технология построения запроса на создание таблиц.
- 2. Технология создания запроса на обновление данных.
- 3. Виды соединения таблиц в запросах
- 4. Свойства запроса.
- 5. Запросы с параметром.
- 6. Построитель выражений.
- 7. Страницы доступа данных
- 8. Виды стандартных отчетов
- 9. Группировка в отчетах
- 10. Макросы

РАЗДЕЛ 4. БАЗИСНЫЕ ОПЕРАЦИИ С РЕЛЯЦИОННЫМИ ДАННЫМИ

Перечень изучаемых элементов содержания

Специальные подходы к выполнению операций над множествами. Реляционная алгебра. Операции над отношениями. Теоретико-множественные операции над отношениями. Специальные реляционные операции. Реляционное исчисление.

ЗАДАНИЯ К ЛАБОРАТОРНЫМ ЗАНЯТИЯМ РАЗДЕЛА 4

Темы лабораторных занятий:

Конструкции языка SQL

Форма практического задания: лабораторный практикум.

Задания лабораторного практикума

- 4.1 «Команда запроса на выборку SELECT»
- 4.2 «Команды редактирования данных INSERT, UPDATE, DELETE»
- 4.3 «Команды создания таблиц»

РУБЕЖНЫЙ КОНТРОЛЬ К РАЗДЕЛУ 4

форма рубежного контроля – отчет по лабораторной работе с ответами на контрольные вопросы.

- 1. Формат команды на выборку SELECT.
- 2. Основные опции команды SELECT.
- 3. Формат команды редактирования данных INSERT
- 4. Форматы команды редактирования данных UPDATE.
- 5. Форматы команды редактирования данных DELETE
- 6. Формат команды создания таблиц SELECT INTO.
- 7. Формат команды создания таблиц CREATE TABLE.
- 8. Опции соединения таблиц в запросах.
- 9. Формат команды объединения данных UNION
- 10. Формат команды перекрестного запроса TRANSFORM

РАЗДЕЛ 5. НОРМАЛЬНЫЕ ФОРМЫ В РЕЛЯЦИОННЫХ БАЗАХ ДАННЫХ

Перечень изучаемых элементов содержания

Нормальные формы в РБД. Нормализация. Функциональные зависимости атрибутов в отношениях. Первая нормальная форма (1НФ). Вторая нормальная форма (2НФ). Третья нормальная форма (3НФ). Алгоритм нормализации (приведение к 3НФ). Корректность процедуры нормализации. Теорема Хеза. Нормальная форма Бойса-Кодда, четвертая и пятая нормальные формы. Пример логического моделирования БД при помощи нормальных форм. Области применения и проблемы логического моделирования БД при помощи нормальных форм.

ЗАДАНИЯ К ЛАБОРАТОРНЫМ ЗАНЯТИЯМ РАЗДЕЛА 5

Темы лабораторных занятий:

Конструкции языка SQL

Форма практического задания: лабораторный практикум.

Задания лабораторного практикума

5.1 «Соединение таблиц в запросах»

5.2 «Команда запроса объединения данных UNION»

РУБЕЖНЫЙ КОНТРОЛЬ К РАЗДЕЛУ 5

форма рубежного контроля – отчет по лабораторной работе с ответами на контрольные вопросы.

1. Информация и данные, база данных, система управления базами данных (СУБД).
2. Эволюция концепции обработки данных, СУБД.
3. Требования к СУБД, основные особенности СУБД, составные части СУБД.
4. Системы быстрой разработки приложений. Модели данных.
5. Реляционная БД, история появления, принципы организации данных, достоинства и недостатки.
6. Базовые понятия реляционных БД: тип данных, домен, атрибут, кортеж, отношение, схема отношений.
7. Проектирование баз данных.
8. Нормализация БД, цели нормализации, 1НФ.
9. Нормализация БД, определение 1НФ, 2НФ, 3НФ.
10. Разработка приложений в среде MS Windows

РАЗДЕЛ 6. ПРОЕКТИРОВАНИЕ БАЗ ДАННЫХ С ИСПОЛЬЗОВАНИЕМ СЕМАНТИЧЕСКОГО ПОДХОДА

Перечень изучаемых элементов содержания

Постановка задачи проектирования, предметная область будущей БД. Анализ предметной области, концептуальное (инфологическое) проектирование БД. Построение логической модели данных. Физическое проектирование БД. Синтез компьютерной модели объекта

ЗАДАНИЯ К ЛАБОРАТОРНЫМ ЗАНЯТИЯМ РАЗДЕЛА 6

Темы лабораторных занятий:

Конструкции языка SQL

Форма практического задания: лабораторный практикум.

Задания лабораторного практикума

6.1 «Групповые операции в запросах»

6.2 «Команда перекрестного запроса TRANSFORM»

РУБЕЖНЫЙ КОНТРОЛЬ К РАЗДЕЛУ 6

форма рубежного контроля – отчет по лабораторной работе с ответами на контрольные вопросы.

11. Архитектура Microsoft Access.
12. Назначение объектов MS Access

13. Построение таблиц в MS Access.
14. Формы ввода-вывода данных.
15. Основные операции реляционной алгебры.
16. Дополнительные операции реляционной алгебры.
17. Запросы в MS Access.
18. Параметры запросов на выборку данных.
19. Перекрестные запросы.
20. Многотабличные запросы и схема данных.

РАЗДЕЛ 7. ПРОЕКТИРОВАНИЕ БАЗ ДАННЫХ

Перечень изучаемых элементов содержания

Постановка задачи проектирования, предметная область будущей БД. Анализ предметной области, концептуальное (инфологическое) проектирование БД. Построение логической модели данных. Физическое проектирование БД. Синтез компьютерной модели объекта.

ЗАДАНИЯ К ЛАБОРАТОРНЫМ ЗАНЯТИЯМ РАЗДЕЛА 7

Темы лабораторных занятий:

Проектирование БД

Форма практического задания: решение задач.

1. Записать SQL-запрос, обеспечивающий ...
... извлечение кодов, названий и городов поставщиков со статусом 20 в алфавитном порядке названий городов, а для одинаковых городов — в порядке названий — из следующей таблицы:
Поставщик (Код, Название, Город, Статус)
2. Записать SQL-запрос, обеспечивающий ...
... извлечение кодов, названий и городов поставщиков, у которых название или город начинаются с буквы «А», из следующей таблицы:
Поставщик (Код, Название, Город, Статус)
3. Записать SQL-запрос, обеспечивающий ...
... извлечение кодов, названий и городов поставщиков, у которых статус находится в диапазоне 20–70, из следующей таблицы:
Поставщик (Код, Название, Город, Статус)
4. Записать SQL-запрос, обеспечивающий ...
... извлечение кодов, названий и городов поставщиков, которые находятся в городах Москва, Петербург, Уфа или Стерлитамак, из следующей таблицы:
Поставщик (Код, Название, Город, Статус)
5. Записать SQL-запрос, обеспечивающий ...
... извлечение кодов и названий поставщиков, а также значений их статуса, умноженных на 100 (в столбец по имени «МСтатус»), из следующей таблицы:
Поставщик (Код, Название, Город, Статус)
6. Записать SQL-запрос, обеспечивающий ...
... извлечение списка городов (без дубликатов), в которых находятся поставщики, из следующей таблицы:
Поставщик (Код, Название, Город, Статус)
7. Записать SQL-запрос, обеспечивающий ...
... извлечение списка городов с указанием среднего статуса поставщиков из этого города, из следующей таблицы:
Поставщик (Код, Название, Город, Статус)
8. Записать SQL-запрос, обеспечивающий ...
... извлечение списка городов с указанием суммарного статуса всех поставщиков из этого города, из следующей таблицы:

- Поставщик (Код, Название, Город, Статус)
9. Записать SQL-запрос, обеспечивающий ...
... извлечение списка городов с указанием минимального статуса среди поставщиков из этого города, из следующей таблицы:
- Поставщик (Код, Название, Город, Статус)
10. Записать SQL-запрос, обеспечивающий ...
... извлечение списка городов с указанием максимального статуса среди поставщиков из этого города, из следующей таблицы:
- Поставщик (Код, Название, Город, Статус)
11. Записать SQL-запрос, обеспечивающий ...
... извлечение списка городов с указанием числа поставщиков из этого города из следующей таблицы:
- Поставщик (Код, Название, Город, Статус)
12. Записать SQL-запрос, обеспечивающий ...
... извлечение списка городов с указанием числа различных значений статуса среди поставщиков из этого города из следующей таблицы:
- Поставщик (Код, Название, Город, Статус)
13. Записать SQL-запрос, обеспечивающий ...
... вставку полных сведений о новом поставщике: код «П007»; название «МММ»; город «Москва»; статус «20» в следующую таблицу:
- Поставщик (Код, Название, Город, Статус)
14. Записать SQL-запрос, обеспечивающий ...
... вставку частичных сведений о новом поставщике: код «П007»; город «Москва» в следующую таблицу:
- Поставщик (Код, Название, Город, Статус)
15. Записать SQL-запрос, обеспечивающий ...
... вставку полных сведений о *всех* новых поставщиках в таблицу
- Поставщик (Код, Название, Город, Статус)
из таблицы НовыйПоставщик, имеющей ту же самую структуру.
16. Записать SQL-запрос, обеспечивающий ...
... вставку полных сведений о новом товаре: код «Т007»; название «Ггг»; город «Москва»; вес «20», цвет «Желтый» в следующую таблицу:
- Товар (Код, Название, Город, Вес, Цвет)
17. Записать SQL-запрос, обеспечивающий ...
... вставку частичных сведений о новом товаре: код «Т007»; город «Москва»; цвет «Желтый» в следующую таблицу:
- Товар (Код, Название, Город, Вес, Цвет)
18. Записать SQL-запрос, обеспечивающий ...
... вставку полных сведений о *всех* новых товарах в таблицу
- Товар (Код, Название, Город, Вес, Цвет)
из таблицы НовыйТовар, имеющей ту же самую структуру.
19. Записать SQL-запрос, обеспечивающий ...
... удаление всех записей из таблицы Поставщик.
20. Записать SQL-запрос, обеспечивающий ...
... удаление всех записей из таблицы Товар.
21. Записать SQL-запрос, обеспечивающий ...
... удаление из таблицы Поставщик записей о поставщиках из Парижа:
- Поставщик (Код, Название, Город, Статус)
22. Записать SQL-запрос, обеспечивающий ...
... удаление из таблицы Товар записей о товарах из Парижа:
- Товар (Код, Название, Город, Вес, Цвет)
23. Записать SQL-запрос, обеспечивающий ...
... удаление таблицы Поставщик (как содержимого, так и структуры).

24. Записать SQL-запрос, обеспечивающий ...
... удаление таблицы Товар (как содержимого, так и структуры).
25. Записать SQL-запрос, обеспечивающий ...
... замену названия «Ленинград» на «С.-Петербург» в сведениях о поставщиках в таблице
Поставщик (Код, Название, Город, Статус)
26. Записать SQL-запрос, обеспечивающий ...
... замену цвета «голубой» на «циан» в сведениях о товарах в таблице
Товар (Код, Название, Город, Вес, Цвет)
27. Записать SQL-запрос, обеспечивающий ...
... увеличение в 2 раза значения статуса у всех поставщиков в таблице
Поставщик (Код, Название, Город, Статус)
28. Записать SQL-запрос, обеспечивающий ...
... переход от веса в граммах к весу в килограммах для всех товаров в таблице
Товар (Код, Название, Город, Вес, Цвет)
29. Записать SQL-запрос, обеспечивающий ...
... запись значения статуса «50» для существующего поставщика с кодом «П007» в таблице
Поставщик (Код, Название, Город, Статус)
30. Записать SQL-запрос, обеспечивающий ...
... запись значения веса «50» для существующего товара с кодом «Т007» в таблице
Товар (Код, Название, Город, Вес, Цвет)
31. Записать SQL-запрос, обеспечивающий ...
... создание структуры таблицы
Поставщик (Код, Название, Город, Статус) ,
где Код является первичным ключом.
32. Записать SQL-запрос, обеспечивающий ...
... создание структуры таблицы
Поставка (КодПоставщика, КодТовара, Количество) ,
где КодПоставщика и КодТовара составляют первичный ключ.
33. Записать SQL-запрос, обеспечивающий ...
... создание структуры таблицы
Поставщик (Код, Название, Город, Статус) ,
где Статус имеет по умолчанию значение 20.
34. Записать SQL-запрос, обеспечивающий ...
... удаление из структуры существующей таблицы
Поставщик (Код, Название, Город, Статус)
столбца Город.
alter table Поставщик drop column Город
35. Записать SQL-запрос, обеспечивающий ...
... предоставление пользователю Boss все привилегии доступа к таблице Поставщик.
36. Записать SQL-запрос, обеспечивающий ...
... предоставление пользователю Manager27 привилегии чтения таблицы Поставщик и
обновления в ней столбца Статус.
37. Записать SQL-запрос, обеспечивающий ...
... отменить все предоставленные пользователю Boss привилегии доступа к таблице
Поставщик.
38. Записать SQL-запрос, обеспечивающий ...
... отменить все предоставленные пользователю Boss привилегии доступа.
39. Записать SQL-запрос, обеспечивающий ...
... отменить предоставленные пользователю Manager27 привилегии обновления таблицы
Поставщик.

РУБЕЖНЫЙ КОНТРОЛЬ К РАЗДЕЛУ 7

форма рубежного контроля – отчет по лабораторной работе с ответами на контрольные вопросы

11. Понятие технологии "клиент-сервер".
12. Общие сведения о языке запросов SQL.
13. Сетевые БД, архитектура «файл-сервер», «клиент-сервер».
14. Язык SQL: общие сведения о языке, роль и место в современных СУБД, стандарт ANSI.
15. Запрос выборки данных в SQL, простейшая выборка из одной таблицы.
16. Специальные операторы SQL IN, BETWEEN, LIKE, IS NULL.
17. Соединение таблиц с использованием операции JOIN.
18. SQL: запрос выборки данных, функции агрегирования AVG, SUM, MAX, MIN.
19. Форматирование выходных данных запроса, секции GROUP BY и HAVING.
20. Соединение таблиц.

РАЗДЕЛ 8. ФИЗИЧЕСКОЕ ПРОЕКТИРОВАНИЕ БД

Перечень изучаемых элементов содержания

Постановка задачи проектирования, предметная область будущей БД. Анализ предметной области, концептуальное (инфологическое) проектирование БД. Построение логической модели данных. Физическое проектирование БД. Синтез компьютерной модели объекта.

ЗАДАНИЯ К ЛАБОРАТОРНЫМ ЗАНЯТИЯМ РАЗДЕЛА 8

Темы лабораторных занятий:

Проектирование БД

Форма практического задания: лабораторный практикум.

Задания лабораторного практикума

Описание вариантов предметной области по вариантам, согласно номеру зачетной книжки

Вариант № 1

Предметная область: Библиотека (учет читателей).

Основные предметно-значимые сущности: Книги, Читатели.

Основные предметно-значимые атрибуты сущностей:

- книги - автор книги, название, год издания, цена, является ли новым изданием, краткая аннотация;
- читатели - номер читательского билета, ФИО, адрес и телефон читателя.

Основные требования к функциям системы:

- выбрать книги, которые находятся у читателей или определенного читателя;
- выбрать читателей, которые брали ту или иную книгу с указанием даты выдачи книги и даты сдачи книги читателем;
- выбрать книги, пользующиеся наибольшим спросом.

Вариант № 2

Предметная область: Деканат (успеваемость студентов).

Основные предметно-значимые сущности: Студенты, Группы студентов, Дисциплины.

Основные предметно-значимые атрибуты сущностей:

- студенты – фамилия, имя, отчество, пол, дата рождения, адрес прописки, группа студентов;
- группы студентов – название, курс, семестр;
- дисциплины – название.

Основные требования к функциям системы:

- выбрать успеваемость студента по дисциплинам с указанием общего количества часов и вида контроля;
- выбрать успеваемость студентов по группам и дисциплинам;
- выбрать дисциплины, изучаемые группой студентов на определенном курсе или определенном семестре.

Вариант № 3

Предметная область: Отдел кадров (контингент сотрудников).

Основные предметно-значимые сущности: Сотрудники, Подразделения.

Основные предметно-значимые атрибуты сущностей:

-сотрудники – фамилия, имя, отчество, пол, дата рождения, адрес прописки, должность, подразделение;

-подразделения – название, вид подразделения.

Основные требования к функциям системы:

-выбрать список сотрудников по подразделениям или определенному подразделению;

-подсчитать средний возраст сотрудников по предприятиям;

-выбрать список сотрудников по составу (профессорско-преподавательский состав, учебно-вспомогательный состав, административно-хозяйственный состав и т.п.).

Вариант № 4

Предметная область: Приемная комиссия (абитуриенты).

Основные предметно-значимые сущности: Абитуриенты, Специальности, Предметы.

Основные предметно-значимые атрибуты сущностей:

-абитуриенты – фамилия, имя, отчество, пол, дата рождения, специальность;

-специальности – название специальности;

-предметы – название предмета, вид контроля.

Основные требования к функциям системы:

-выбрать всех абитуриентов по специальностям или определенной специальности;

-выбрать всех абитуриентов, сдавших вступительные экзамены, и их рейтинг (сумма баллов по всем сданным предметам) по специальностям или определенной специальности;

-подсчитать средний балл по дисциплинам и специальностям.

Вариант № 5

Предметная область: Учебно-методическое управление (учет площади помещений).

Основные предметно-значимые сущности: Помещения, Подразделения. Основные предметно-значимые атрибуты сущностей:

-помещения – название или номер помещения, вид помещения (аудитория, кабинет и т.п.), площадь, количество посадочных мест, подразделение;

-подразделения – название, вид подразделения.

Основные требования к функциям системы:

-выбрать названия или номера помещений по подразделениям;

-подсчитать общую площадь учебных аудиторий по помещениям и в целом по учебному заведению;

-подсчитать общее количество посадочных мест для сотрудников по подразделениям.

Вариант № 6

Предметная область: Поликлиника (учет пациентов).

Основные предметно-значимые сущности: Пациенты, Врачи.

Основные предметно-значимые атрибуты сущностей:

-пациенты – фамилия, имя, отчество, дата рождения;

-врачи – фамилия, имя, отчество, дата рождения, должность, специализация.

Основные требования к функциям системы:

-выбрать все диагнозы по пациентам или определенному пациенту;

-выбрать всех пациентов, записанных к определенному врачу на определенную дату;

-выбрать всех врачей, к которым записан определенный пациент.

Вариант № 7

Предметная область: Телефонный узел связи (учет абонентов).

Основные предметно-значимые сущности: Абоненты, Подразделения, Помещения.

Основные предметно-значимые атрибуты сущностей:

-абоненты – фамилия, имя, отчество, дата рождения, подразделение;

-помещения – название или номер помещения, вид помещения (аудитория, кабинет и т.п.), подразделение;

-подразделения – название, вид подразделения.

Основные требования к функциям системы:

- выбрать номера абонента по подразделениям;
- выбрать номера абонента по помещениям;
- подсчитать количество абонентов по подразделениям, помещениям.

Вариант № 8

Предметная область: Транспорт (движение общественного транспорта).

Основные предметно-значимые сущности: Станции, Маршруты.

Основные предметно-значимые атрибуты сущностей:

- станции – название;
- маршруты – название или номер маршрута.

Основные требования к функциям системы:

- выбрать все станции по маршрутам или определенному маршруту;
- выбрать все маршруты по станциям или определенной станции;
- подсчитать общее время движения по маршрутам.

Вариант № 9

Предметная область: Студенческое общежитие.

Основные предметно-значимые сущности: Студенты, Общежития, Комнаты.

Основные предметно-значимые атрибуты сущностей:

- студенты – фамилия, имя, отчество, группа студентов;
- общежития – название или номер общежития, адрес;
- комнаты – название или номер комнаты, этаж.

Основные требования к функциям системы:

- выбрать всех студентов, проживающих в общежитии, с указанием комнаты по общежитиям или определенному общежитию;
- выбрать всех студентов, проживающих в общежитии, с указанием комнаты по группам студентов или определенной группе;
- подсчитать количество проживающих студентов по комнатам с указанием общежития.

Вариант № 10 (цифра 0)

Предметная область: Учебно-методический отдел (расписание занятий).

Основные предметно-значимые сущности: Дисциплины, Аудитории, Группы студентов, Преподаватели.

Основные предметно-значимые атрибуты сущностей:

- дисциплины – название;
- аудитории – название или номер аудитории;
- группы студентов – название или номер группы;
- преподаватели – фамилия, имя, отчество.

Основные требования к функциям системы:

- выбрать все занятия с указанием аудитории по группам или определенной группе;
- выбрать все занятия с указанием аудиторий по преподавателям или определенному преподавателю;
- подсчитать общее количество часов занятий в неделю по аудиториям или определенной аудитории.

РУБЕЖНЫЙ КОНТРОЛЬ К РАЗДЕЛУ 8

форма рубежного контроля – отчет по лабораторным работам с ответами на контрольные вопросы

1. Вложенные подзапросы.
2. Связанные подзапросы. Оператор EXISTS.
3. Вложенные и связанные подзапросы. Операторы ANY, SOME, ALL.
4. Объединение запросов.
5. SQL: запрос выборки данных по нескольким таблицам, оператор JOIN, левое, правое и внутреннее соединение.
6. Запросы обновления таблиц INSERT, UPDATE, DELETE..

7. Создание, модификация и уничтожение таблиц. Ограничения на множество допустимых значений данных. Значение по умолчанию.
8. Создание и уничтожение индексов. Поддержка ссылочной целостности
9. Создание представлений.
10. Определение прав доступа к данным.
11. Определение синонимов объектов. Понятие транзакций. Управление параллелизмом
12. Сервер баз данных, базовые понятия.
13. СУБД DB2. Иерархия объектов базы данных.
14. Объекты DB2, их назначение.
15. SQL: хранимые процедуры, область применения.

**Приложение № 4 к методическим материалам
по дисциплине (модулю). Учебно-наглядные
пособия по дисциплине (модулю)**

УЧЕБНО-НАГЛЯДНЫЕ ПОСОБИЯ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)

РАЗДЕЛ 1. Теория баз данных

Тема 1.1. Введение в базы данных. Общая характеристика основных понятий

3. Гутгарц, Р. Д. Проектирование автоматизированных систем обработки информации и управления : учебное пособие для вузов / Р. Д. Гутгарц. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 351 с. — (Высшее образование). — ISBN 978-5-534-15761-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509638> (дата обращения: 07.03.2023).
4. Грекул, В. И. Проектирование информационных систем : учебник и практикум для вузов / В. И. Грекул, Н. Л. Коровкина, Г. А. Левочкина. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-9916-8764-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511889> (дата обращения: 07.03.2023).

Тема 1.2. Системы управления базами данных

3. Гутгарц, Р. Д. Проектирование автоматизированных систем обработки информации и управления : учебное пособие для вузов / Р. Д. Гутгарц. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 351 с. — (Высшее образование). — ISBN 978-5-534-15761-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509638> (дата обращения: 07.03.2023).
4. Грекул, В. И. Проектирование информационных систем : учебник и практикум для вузов / В. И. Грекул, Н. Л. Коровкина, Г. А. Левочкина. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-9916-8764-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511889> (дата обращения: 07.03.2023).

РАЗДЕЛ 2. Общие принципы построения (архитектура) баз данных

Тема 2.1. Различные архитектурные решения, используемые при реализации многопользовательских СУБД. Краткий обзор СУБД

3. Гутгарц, Р. Д. Проектирование автоматизированных систем обработки информации и управления : учебное пособие для вузов / Р. Д. Гутгарц. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 351 с. — (Высшее образование). — ISBN 978-5-534-15761-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509638> (дата обращения: 07.03.2023).
4. Грекул, В. И. Проектирование информационных систем : учебник и практикум для вузов / В. И. Грекул, Н. Л. Коровкина, Г. А. Левочкина. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-9916-8764-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511889> (дата обращения: 07.03.2023).

Тема 2.2. Краткий обзор СУБД

3. Гутгарц, Р. Д. Проектирование автоматизированных систем обработки информации и управления : учебное пособие для вузов / Р. Д. Гутгарц. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 351 с. — (Высшее образование). — ISBN 978-5-534-15761-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509638> (дата обращения: 07.03.2023).
4. Грекул, В. И. Проектирование информационных систем : учебник и практикум для вузов / В. И. Грекул, Н. Л. Коровкина, Г. А. Левочкина. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-9916-8764-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511889> (дата обращения: 07.03.2023).

РАЗДЕЛ 3. Модели данных

Тема 3.1. Различные представления о данных в базах данных. Основные этапы проектирования баз данных

3. Гутгарц, Р. Д. Проектирование автоматизированных систем обработки информации и управления : учебное пособие для вузов / Р. Д. Гутгарц. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 351 с. — (Высшее образование). — ISBN 978-5-534-15761-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509638> (дата обращения: 07.03.2023).
4. Грекул, В. И. Проектирование информационных систем : учебник и практикум для вузов / В. И. Грекул, Н. Л. Коровкина, Г. А. Левочкина. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-9916-8764-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511889> (дата обращения: 07.03.2023).

Тема 3.2. Первая стадия концептуального проектирования базы данных (концептуальное моделирование. Вторая стадия концептуального проектирования (Модели данных СУБД. Представление концептуальной модели средствами модели данных СУБД)

3. Гутгарц, Р. Д. Проектирование автоматизированных систем обработки информации и управления : учебное пособие для вузов / Р. Д. Гутгарц. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 351 с. — (Высшее образование). — ISBN 978-5-

534-15761-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509638> (дата обращения: 07.03.2023).

4. Грекул, В. И. Проектирование информационных систем : учебник и практикум для вузов / В. И. Грекул, Н. Л. Коровкина, Г. А. Левочкина. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-9916-8764-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511889> (дата обращения: 07.03.2023).

РАЗДЕЛ 4. Базисные операции с реляционными данными

Тема 4.1. Формализация реляционной модели

3. Гутгарц, Р. Д. Проектирование автоматизированных систем обработки информации и управления : учебное пособие для вузов / Р. Д. Гутгарц. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 351 с. — (Высшее образование). — ISBN 978-5-534-15761-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509638> (дата обращения: 07.03.2023).
4. Грекул, В. И. Проектирование информационных систем : учебник и практикум для вузов / В. И. Грекул, Н. Л. Коровкина, Г. А. Левочкина. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-9916-8764-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511889> (дата обращения: 07.03.2023).

РАЗДЕЛ 5. Нормальные формы в реляционных базах данных

Тема 5.1. Использование формального аппарата для оптимизации схем отношений

3. Гутгарц, Р. Д. Проектирование автоматизированных систем обработки информации и управления : учебное пособие для вузов / Р. Д. Гутгарц. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 351 с. — (Высшее образование). — ISBN 978-5-534-15761-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509638> (дата обращения: 07.03.2023).
4. Грекул, В. И. Проектирование информационных систем : учебник и практикум для вузов / В. И. Грекул, Н. Л. Коровкина, Г. А. Левочкина. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-9916-8764-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511889> (дата обращения: 07.03.2023).

РАЗДЕЛ 6. Проектирование баз данных с использованием семантического подхода

Тема 6.1. Физические модели данных (внутренний уровень)

3. Гутгарц, Р. Д. Проектирование автоматизированных систем обработки информации и управления : учебное пособие для вузов / Р. Д. Гутгарц. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 351 с. — (Высшее образование). — ISBN 978-5-534-15761-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509638> (дата обращения: 07.03.2023).
4. Грекул, В. И. Проектирование информационных систем : учебник и практикум для вузов / В. И. Грекул, Н. Л. Коровкина, Г. А. Левочкина. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-9916-8764-5. — Текст :

электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511889> (дата обращения: 07.03.2023).

РАЗДЕЛ 7. Проектирование баз данных

Тема 7.1. Программное обеспечение работы с современными базами данных

5. Гутгарц, Р. Д. Проектирование автоматизированных систем обработки информации и управления : учебное пособие для вузов / Р. Д. Гутгарц. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 351 с. — (Высшее образование). — ISBN 978-5-534-15761-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509638> (дата обращения: 07.03.2023).
6. Грекул, В. И. Проектирование информационных систем : учебник и практикум для вузов / В. И. Грекул, Н. Л. Коровкина, Г. А. Левочкина. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-9916-8764-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511889> (дата обращения: 07.03.2023).

Тема 7.2. Основные операторы языка SQL. Интерактивный SQL. Использование языка SQL в прикладных программах

3. Гутгарц, Р. Д. Проектирование автоматизированных систем обработки информации и управления : учебное пособие для вузов / Р. Д. Гутгарц. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 351 с. — (Высшее образование). — ISBN 978-5-534-15761-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509638> (дата обращения: 07.03.2023).
4. Грекул, В. И. Проектирование информационных систем : учебник и практикум для вузов / В. И. Грекул, Н. Л. Коровкина, Г. А. Левочкина. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-9916-8764-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511889> (дата обращения: 07.03.2023).

РАЗДЕЛ 8. Физическое проектирование БД

Тема 8.1. Направления развития баз данных

3. Гутгарц, Р. Д. Проектирование автоматизированных систем обработки информации и управления : учебное пособие для вузов / Р. Д. Гутгарц. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 351 с. — (Высшее образование). — ISBN 978-5-534-15761-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/509638> (дата обращения: 07.03.2023).
4. Грекул, В. И. Проектирование информационных систем : учебник и практикум для вузов / В. И. Грекул, Н. Л. Коровкина, Г. А. Левочкина. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-9916-8764-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511889> (дата обращения: 07.03.2023).

ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

№ п/п	Содержание изменения	Реквизиты документа об утверждении изменения	Дата введения изменения
1.			
2.			
3.			
4.			



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Российский государственный социальный университет»

УТВЕРЖДАЮ

Заведующий кафедрой комплекса естественно-
научных дисциплин

/Денисова Д.А./

«27» февраля 2024 г

МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ
«ЧИСЛЕННЫЕ МЕТОДЫ»

Направление подготовки
«Информационные системы и технологии»

Направленность
«Интеллектуальные информационные системы и технологии»

ОБРАЗОВАТЕЛЬНАЯ ПРОГРАММА ВЫСШЕГО ОБРАЗОВАНИЯ - ПРОГРАММА
БАКАЛАВРИАТА

Форма обучения

Очная, заочная

Москва, 2024 г.

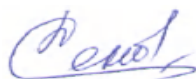
Методические материалы по дисциплине (модулю) «Численные методы» разработаны на основании федерального государственного образовательного стандарта высшего образования – *бакалавриата* по направлению подготовки *09.03.02 Информационные системы и технологии*, утвержденного приказом Министерства образования и науки Российской Федерации от 19.09.2017 № 926, учебного плана по основной профессиональной образовательной программе высшего образования – программе бакалавриата по направлению подготовки *09.03.02 Информационные системы и технологии* (далее – «ОПОП»).

Методические материалы по дисциплине (модулю) разработаны канд. физ-мат. наук, доцентом Н.П. Третьяковым.

Методические материалы по дисциплине (модулю) разработаны на заседании кафедры комплекса естественно-научных дисциплин.

Протокол № 9 от «27» февраля 2024 года

Заведующий кафедрой
канд. техн. наук, доцент



Д.А. Денисова

(подпись)

СОДЕРЖАНИЕ

1. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ К ЛЕКЦИОННЫМ И ПРАКТИЧЕСКИМ ЗАНЯТИЯМ.....	4
1.1. Методические материалы к проведению лекционных занятий по дисциплине (модулю)	4
1.2. Методические материалы по подготовке к практическим занятиям по дисциплине (модулю)	6
2. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ) И САМОСТОЯТЕЛЬНОЙ РАБОТЕ.....	10
3. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНИВАНИЯ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ И (ИЛИ) ОПЫТА ДЕЯТЕЛЬНОСТИ, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ.....	20
3.1. Организационные основы применения балльно-рейтинговой системы оценки успеваемости обучающихся по дисциплине (модулю)	20
3.2. Проведение текущего контроля успеваемости обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося.....	20
3.3. Проведение промежуточной аттестации обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося.....	21
ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ	23

1. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ К ЛЕКЦИОННЫМ И ПРАКТИЧЕСКИМ ЗАНЯТИЯМ

1.1. Методические материалы к проведению лекционных занятий по дисциплине (модулю)

Лекция - один из методов обучения, одна из основных системообразующих форм организации учебного процесса в вузе. Лекционное занятие представляет собой систематическое, последовательное, монологическое изложение педагогическим работником учебного материала, как правило, теоретического характера. Такое занятие представляет собой элемент технологии представления учебного материала путем логически стройного, систематически последовательного и ясного изложения.

Цель лекции – организация целенаправленной познавательной деятельности обучающихся по овладению программным материалом дисциплины (модуля). Чтение курса лекций позволяет дать связанное, последовательное изложение материала в соответствии с новейшими данными науки, сообщить слушателям основное содержание предмета в целостном, систематизированном виде. В ряде случаев лекция выполняет функцию основного источника информации: при отсутствии учебников и учебных пособий, чаще по новым курсам; в случае, когда новые научные данные по той или иной теме не нашли отражения в учебниках; отдельные разделы и темы очень сложны для самостоятельного изучения. В таких случаях только лектор может методически помочь обучающимся в освоении сложного материала.

Возможные формы проведения лекций:

- Вводная лекция – один из наиболее важных и трудных видов лекции при чтении систематических курсов. От успеха этой лекции во многом зависит успех усвоения всего курса. Она может содержать: определение дисциплины (модуля); краткую историческую справку о дисциплине (модуле); цели и задачи дисциплины (модуля), ее роль в общей системе обучения и связь со смежными дисциплинами (модулями); основные проблемы (понятия и определения) данной науки; основную и дополнительную учебную литературу; особенности самостоятельной работы обучающихся над дисциплиной (модулем) и формы участия в научно-исследовательской работе; отчетность по курсу.

- Информационная лекция ориентирована на изложение и объяснение обучающимся научной информации, подлежащей осмыслению и запоминанию. Это самый традиционный тип лекций в практике высшей школы.

- Заключительная лекция предназначена для обобщения полученных знаний и раскрытия перспектив дальнейшего развития данной науки.

- Обзорная лекция – это систематизация научных знаний на высоком уровне, допускающая большое число ассоциативных связей в процессе осмысления информации, излагаемой при раскрытии внутрипредметной и межпредметной связей, исключая детализацию и конкретизацию. Как правило, стержень излагаемых теоретических положений составляет научно-понятийная и концептуальная основа всего курса или крупных его разделов.

- Лекция-беседа - непосредственный контакт педагогического работника с аудиторией - диалог. По ходу лекции педагогический работник задает вопросы для выяснения мнений и уровня осведомленности обучающихся по рассматриваемой проблеме.

- Лекция-дискуссия - свободный обмен мнениями в ходе изложения лекционного материала. Педагогический работник активизирует участие в обсуждении отдельными вопросами, сопоставляет между собой различные мнения и тем самым развивает дискуссию, стремясь направить ее в нужное русло.

- Лекция с применением обратной связи включает в себе то, что в начале и конце каждого раздела лекции задаются вопросы. Первый - для того, чтобы узнать, насколько обучающиеся ориентируются в излагаемом материале, вопрос в конце раздела предназначен для выяснения степени усвоения только что изложенного материала. При

неудовлетворительных результатах контрольного опроса педагогический работник возвращается к уже прочитанному разделу, изменив при этом методику подачи материала.

- Проблемная лекция опирается на логику последовательно моделируемых проблемных ситуаций путем постановки проблемных вопросов или предъявления проблемных задач. Проблемный вопрос - это диалектическое противоречие, требующее для своего решения размышления, сравнения, поиска, приобретения и применения новых знаний. Проблемная задача содержит дополнительную вводную информацию и при необходимости некоторые ориентиры поиска ее решения.

- Программированная лекция - консультация – педагогический работник сам составляет и предлагает обучающимся вопросы. На подготовленные вопросы педагогический работник сначала просит ответить обучающихся, а затем проводит анализ и обсуждение неправильных ответов. В лекциях можно использовать наглядные материалы, а также подготовить презентацию. Что касается презентации, то в качестве визуальной поддержки ее можно органично интегрировать во все вышеупомянутые лекции. В то же время лекцию-презентацию возможно выделить и в качестве самостоятельной формы. Лекция-презентация должна отражать суть основных и (или) проблемных вопросов лекции, на которые особо следует обратить внимание обучающихся. В условиях применения активного метода проведения занятий презентация представляется весьма удачным способом донесения информации до слушателей. Единственное, на что следует обратить внимание при подготовке слайдов, - это их оформление и текст. Слайд не должен быть перегружен картинками и лишней информацией, которая будет отвлекать от основного аспекта того или иного вопроса лекции. Во время лекции можно задавать вопросы аудитории в отношении того или иного слайда, тем самым еще больше вовлекая обучающихся в проблематику.

Краткое содержание лекционных занятий

Наименование разделов и тем	Содержание учебного материала
РАЗДЕЛ 1. Основные понятия и методы вычислительной математики	
Тема 1.1. Предмет вычислительной математики и ее место в практике	Предмет, структура и задачи курса, его связь с другими дисциплинами. Этапы развития и классификация методов. Понятие алгоритма. Универсальные численные методы и программы и он-лайн средства специального назначения.
Тема 1.2. Действия над приближенными величинами	Основные источники и классификация погрешностей. Абсолютная и относительная погрешности. Особенности машинной арифметики. Абсолютные погрешности суммы и разности. Относительные погрешности произведения и частного. Устойчивость численных методов к накоплению погрешностей округления.
РАЗДЕЛ 2. Основные алгоритмы вычислительной математики	
Тема 2.1. Пределы последовательностей и степенные ряды	Определение предела последовательности и методы их вычисления.
Тема 2.2. Решение нелинейных уравнений и систем уравнений	Отделение корней алгебраических и трансцендентных уравнений. Уточнение корня алгебраического уравнения методом половинного деления. Метод простой итерации. Метод Ньютона, метод Хорда. Различные методы решения систем нелинейных уравнений: модификации метода Ньютона, гибридные методы.
Тема 2.3. Аппроксимация функций	Интерполяция. Интерполяционный полином в форме

	Лагранжа. Интерполяционный полином в форме Ньютона. Сплайн-интерполяция. Сглаживание. Метод наименьших квадратов.
Тема 2.4. Численное интегрирование	Метод прямоугольников. Метод трапеций. Метод парабол (Симпсона)
Тема 2.5. Поиск собственных значений и векторов	Определение собственных значений и векторов. Уравнение на собственные значения. Методы вычислений. Примеры применения.
Тема 2.6. Решение обыкновенных дифференциальных уравнений	Решение задачи Коши. Метод Эйлера. Модифицированный метод Эйлера. Метод Рунге-Кутты.
Тема 2.7. Методы оптимизации и примеры	Общая постановка задачи линейного программирования. Целевая функция. Методы вычислений. Примеры.

1.2. Методические материалы по подготовке к практическим занятиям по дисциплине (модулю)

Практические (семинарские) занятия - одна из форм учебного занятия, направленная на развитие самостоятельности обучающихся и приобретение умений и навыков. Данные учебные занятия углубляют, расширяют, детализируют полученные ранее знания. Практическое занятие предполагает выполнение обучающимися по заданию и под руководством преподавателей одной или нескольких практических работ.

Цель практических занятий и семинаров состоит в развитии познавательных способностей, самостоятельности мышления и творческой активности обучающихся; углублении, расширении, детализировании знаний, полученных на лекции в обобщенной форме, и содействии выработке навыков профессиональной деятельности. В отдельных случаях на практических занятиях и семинарах руководителем занятия сообщаются дополнительные знания.

Для достижения поставленных целей и решения требуемого перечня задач практические занятия и семинары проводятся традиционными технологиями или с использованием активных и интерактивных образовательных технологий.

Возможные формы проведения практических (семинарских) занятий:

- Деловая игра - это метод группового обучения совместной деятельности в процессе решения общих задач в условиях максимально возможного приближения к реальным проблемным ситуациям. Имитационные игры - на занятиях имитируется деятельность какой-либо организации, предприятия или его подразделения. Имитироваться могут события, конкретная деятельность людей (деловое совещание, обсуждение плана) и обстановка, условия, в которых происходит событие или осуществляется деятельность (кабинет начальника цеха, зал заседаний). Исполнение ролей (ролевые игры) - в этих играх отрабатывается тактика поведения, действий, выполнение функций и обязанностей конкретного лица. Для проведения игр с исполнением роли разрабатывается модель-пьеса ситуации, между студентами распределяются роли с «обязательным содержанием», характеризующиеся различными интересами; в процессе их взаимодействия должно быть найдено компромиссное решение. «Деловой театр» (метод инсценировки) - в нем разыгрывается какая-либо ситуация, поведение человека в этой обстановке, обучающийся должен вжиться в образ определенного лица, понять его действия, оценить обстановку и найти правильную линию поведения. Основная задача метода инсценировки - научить ориентироваться в различных обстоятельствах, давать объективную оценку своему поведению, учитывать возможности других людей, влиять на их интересы, потребности и деятельность, не прибегая к формальным атрибутам власти, к приказу.

- Игровое проектирование - является практическим занятием или циклом занятий, суть которых состоит в разработке инженерного, конструкторского, технологического и других

видов проектов в игровых условиях, максимально воссоздающих реальность. Этот метод отличается высокой степенью сочетания индивидуальной и совместной работы обучающихся.

- Познавательные-дидактические игры не относятся к деловым играм. Они предполагают лишь включение изучаемого материала в необычный игровой контекст и иногда содержат лишь элементы ролевых игр. Такие игры могут проводиться в виде копирования научных, культурных, социальных явлений (конкурс знатоков, «Поле чудес», КВН и т.д.) и в виде предметно-содержательных моделей, (например, игры-путешествия, когда надо разработать рациональный маршрут, пользуясь различными картами).

- Анализ конкретных ситуаций. Конкретная ситуация – это любое событие, которое содержит в себе противоречие или вступает в противоречие с окружающей средой. Ситуации могут нести в себе как позитивный, так и отрицательный опыт. Все ситуации делятся на простые, критические и экстремальные.

- Кейс-метод (от английского case – случай, ситуация) – усовершенствованный метод анализа конкретных ситуаций, метод активного проблемно-ситуационного анализа, основанный на обучении путем решения конкретных задач – ситуаций (решение кейсов). Непосредственная цель метода case-study – обучающиеся должны проанализировать ситуацию, разобраться в сути проблем, предложить возможные решения и выбрать лучшее из них. Кейсы делятся на практические (отражающие реальные жизненные ситуации), обучающие (искусственно созданные, содержащие значительные элементы условности при отражении в нем жизни) и исследовательские (ориентированные на проведение исследовательской деятельности посредством применения метода моделирования). Метод конкретных ситуаций (метод case-study) относится к неигровым имитационным активным методам обучения.

- Тренинг (англ. training от train — обучать, воспитывать) – метод активного обучения, направленный на развитие знаний, умений и навыков и социальных установок. Тренинг – форма интерактивного обучения, целью которого является развитие компетентности межличностного и профессионального поведения в общении. Достоинство тренинга заключается в том, что он обеспечивает активное вовлечение всех участников в процесс обучения. Можно выделить основные типы тренингов по критерию направленности воздействия и изменений – навыков, психотерапевтический, социально-психологический, бизнес-тренинг.

- Метод Сократа (Майевтика) – метод вопросов, предполагающих критическое отношение к догматическим утверждениям, называется еще как метод «сократовской иронии». Это умение извлекать скрытое в человеке знание с помощью искусных наводящих вопросов, подразумевающего короткий, простой и заранее предсказуемый ответ.

- Интерактивная лекция – выступление ведущего обучающего перед большой аудиторией с применением следующих активных форм обучения: дискуссия, беседа, демонстрация слайдов или учебных фильмов, мозговой штурм.

- Групповая, научная дискуссия, диспут. Дискуссия — это целенаправленное обсуждение конкретного вопроса, сопровождающееся обменом мнениями, идеями между двумя и более лицами. Задача дискуссии - обнаружить различия в понимании вопроса и в споре установить истину. Дискуссии могут быть свободными и управляемыми. К технике управляемой дискуссии относятся: четкое определение цели, прогнозирование реакции оппонентов, планирование своего поведения, ограничение времени на выступления и их заданная очередность. Разновидностью свободной дискуссии является форум, где каждому желающему дается неограниченное время на выступление, при условии, что его выступление вызывает интерес аудитории. Каждый конкретный форум имеет свою тематику — достаточно широкую, чтобы в её пределах можно было вести многоплановое обсуждение.

- Дебаты – это четко структурированный и специально организованный публичный обмен мыслями между двумя сторонами по актуальным темам. Это разновидность публичной дискуссии участников дебатов, направляющая на переубеждение в своей правоте третьей стороны, а не друг друга. Поэтому вербальные и невербальные средства, которые

используются участниками дебатов, имеют целью получения определённого результата — сформировать у слушателей положительное впечатление от собственной позиции.

- Метод работы в малых группах. Групповое обсуждение кого-либо вопроса направлено на достижение лучшего взаимопонимания и нахождения истины. Групповое обсуждение способствует лучшему усвоению изучаемого материала. Оптимальное количество участников - 5-7 человек. Перед обучающимися ставится проблема, выделяется определенное время, в течение которого они должны подготовить аргументированный обдуманный ответ. Педагогический работник может устанавливать правила проведения группового обсуждения — задавать определенные рамки обсуждения, ввести алгоритм выработки общего мнения, назначить лидера и др.

- Круглый стол - общество, собрание в рамках более крупного мероприятия (съезда, симпозиума, конференции). Мероприятие, как правило, на которое приглашаются эксперты и специалисты из разных сфер деятельности для обсуждения актуальных вопросов. Данная модель обсуждения, основываясь на соглашениях, в качестве итогов даёт результаты, которые, в свою очередь, являются новыми соглашениями.

- Коллоквиум - (лат. colloquium — разговор, беседа) - одна из форм учебных занятий в системе образования, имеющая целью выяснение и повышение знаний обучающихся. На коллоквиумах обсуждаются: отдельные части, разделы, темы, вопросы изучаемого курса (обычно не включаемые в тематику семинарских и других практических учебных занятий), рефераты, проекты и др. работы обучающихся. Это научные собрания, на которых заслушиваются и обсуждаются доклады. Коллоквиум — это и форма контроля, массового опроса, позволяющая преподавателю в сравнительно небольшой срок выяснить уровень знаний студентов по данной теме дисциплины. Коллоквиум проходит обычно в форме дискуссии, в ходе которой обучающимся предоставляется возможность высказать свою точку зрения на рассматриваемую проблему, учиться обосновывать и защищать ее. Аргументируя и отстаивая свое мнение, обучающийся в то же время демонстрирует, насколько глубоко и осознанно он усвоил изученный материал.

- Метод «мозговой штурм» (мозговой штурм, мозговая атака, англ. brainstorming) — оперативный метод решения проблемы на основе стимулирования творческой активности, при котором участникам обсуждения предлагают высказывать как можно большее количество вариантов решения, в том числе самых фантастичных. Затем из общего числа высказанных идей отбирают наиболее удачные, которые могут быть использованы на практике. Является методом экспертного оценивания.

- Метод проектов - это способ достижения дидактической цели через детальную разработку проблемы (технологию), которая должна завершиться вполне реальным, осязаемым практическим результатом, оформленным тем или иным образом; это совокупность приёмов, действий обучающихся в их определённой последовательности для достижения поставленной задачи — решения проблемы, лично значимой для учащихся и оформленной в виде некоего конечного продукта. Основное предназначение метода проектов состоит в предоставлении учащимся возможности самостоятельного приобретения знаний в процессе решения практических задач или проблем, требующего интеграции знаний из различных предметных областей.

- Брифинг - (англ. briefing от англ. brief — короткий, недолгий) — краткая пресс-конференция, посвященная одному вопросу. Основное отличие: отсутствует презентационная часть. То есть практически сразу идут ответы на вопросы журналистов.

- Метод портфолио (итал. portfolio — 'портфель, англ. - папка для документов) - современная образовательная технология, в основе которой используется метод аутентичного оценивания результатов образовательной и профессиональной деятельности. Портфолио как подборка сертифицированных достижений, наиболее значимых работ и отзывов на них.

Вопросы для самоподготовки к практическим (семинарским) занятиям по разделам (темам) дисциплины (модуля)

Задания для самостоятельной работы к Разделу 1

Вопросы для самостоятельной работы к Разделу 1

1. Основные источники и классификация погрешностей.
2. Абсолютная и относительная погрешности.
3. Особенности машинной арифметики.
4. Абсолютные погрешности суммы и разности.
5. Относительные погрешности произведения и частного.
6. Формула полной погрешности.
7. Устойчивость численных методов к накоплению погрешностей округления.

Литература для самостоятельного изучения к Разделу 1.

Основная литература

1. Зенков, А. В. Численные методы : учебное пособие для вузов / А. В. Зенков. — Москва : Издательство Юрайт, 2020. — 122 с. — (Высшее образование). — ISBN 978-5-534-10893-4. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/452695> (дата обращения: 31.12.2022).
2. Зализняк, В. Е. Численные методы. Основы научных вычислений : учебник и практикум для вузов / В. Е. Зализняк. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2020. — 356 с. — (Высшее образование). — ISBN 978-5-534-02714-3. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/449891> (дата обращения: 31.12.2020). А. П. Архитектура ЭВМ : учебное пособие для вузов / А. П. Толстобров. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2023. — 154 с. — (Высшее образование). — ISBN 978-5-534-12377-7. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/518719>.

Дополнительная литература

1. Численные методы : учебник и практикум для академического бакалавриата / У. Г. Пирумов [и др.] ; под редакцией У. Г. Пирумова. — 5-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2019. — 421 с. — (Бакалавр. Академический курс). — ISBN 978-5-534-03141-6. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/431961> (дата обращения: 31.12.2020).
2. Сухарев, А. Г. Численные методы оптимизации : учебник и практикум для академического бакалавриата / А. Г. Сухарев, А. В. Тимохов, В. В. Федоров. — 3-е изд., испр. и доп. — Москва : Издательство Юрайт, 2019. — 367 с. — (Бакалавр. Академический курс). — ISBN 978-5-534-04449-2. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/427001> (дата обращения: 31.12.2022).

Задания для самостоятельной работы к Разделу 2

Вопросы для самостоятельной работы к Разделу 2

1. Метод Гаусса решения систем линейных алгебраических уравнений.
2. Применение LU - разложения матрицы для решения систем линейных алгебраических уравнений, вычисления определителя и обратной матрицы.
3. Разложение симметричных матриц, метод квадратного корня (метод Холецкого).
4. Метод прогонки решения систем уравнений с трёхдиагональными матрицами.
5. Нормы векторов и матриц.

6. Обусловленность систем линейных алгебраических уравнений.
7. Корректные и некорректные задачи.
8. Двухслойные итерационные методы решения систем линейных алгебраических уравнений.
9. Методы Якоби и Зейделя. Метод простой итерации.
10. Исследование сходимости и оценка скорости сходимости итерационных методов.
11. Метод Гаусса.
12. Метод Крамера.
13. Вычисление определителей и обращение матриц Методы Якоби и Зейделя.
14. Метод простой итерации.
15. Метод простой итерации, метод Ньютона.
16. Решение систем уравнений с помощью инструментальных средств
17. Исследование сходимости и оценка скорости сходимости итерационных методов.

Литература для самостоятельного изучения к Разделу 2.

Основная литература

1. Зенков, А. В. Численные методы : учебное пособие для вузов / А. В. Зенков. — Москва : Издательство Юрайт, 2020. — 122 с. — (Высшее образование). — ISBN 978-5-534-10893-4. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/452695> (дата обращения: 31.12.2022).
2. Зализняк, В. Е. Численные методы. Основы научных вычислений : учебник и практикум для вузов / В. Е. Зализняк. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2020. — 356 с. — (Высшее образование). — ISBN 978-5-534-02714-3. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/449891> (дата обращения: 31.12.2020).
- А. П. Архитектура ЭВМ : учебное пособие для вузов / А. П. Толстобров. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2023. — 154 с. — (Высшее образование). — ISBN 978-5-534-12377-7. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/518719>.

Дополнительная литература

1. Численные методы : учебник и практикум для академического бакалавриата / У. Г. Пирумов [и др.] ; под редакцией У. Г. Пирумова. — 5-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2019. — 421 с. — (Бакалавр. Академический курс). — ISBN 978-5-534-03141-6. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/431961> (дата обращения: 31.12.2020).
- Сухарев, А. Г. Численные методы оптимизации : учебник и практикум для академического бакалавриата / А. Г. Сухарев, А. В. Тимохов, В. В. Федоров. — 3-е изд., испр. и доп. — Москва : Издательство Юрайт, 2019. — 367 с. — (Бакалавр. Академический курс). — ISBN 978-5-534-04449-2. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/427001> (дата обращения: 31.12.2022).

2. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ) И САМОСТОЯТЕЛЬНОЙ РАБОТЕ

Освоение обучающимся дисциплины (модуля) «Технологии возможностей и безбарьерной среды» предполагает изучение материалов дисциплины на аудиторных занятиях и в ходе самостоятельной работы. Аудиторные занятия проходят в форме лекций, семинаров, практических и лабораторных занятий. Самостоятельная работа включает разнообразный комплекс видов и форм работы обучающихся.

Для успешного освоения дисциплины (модуля) и достижения поставленных целей необходимо внимательно ознакомиться с рабочей программы дисциплины (модуля), доступной в электронной информационно-образовательной среде РГСУ.

Следует обратить внимание на списки основной и дополнительной литературы, на предлагаемые преподавателем ресурсы информационно-телекоммуникационной сети Интернет. Эта информация необходима для самостоятельной работы обучающегося.

При подготовке к аудиторным занятиям необходимо помнить особенности каждой формы его проведения.

Подготовка к учебному занятию лекционного типа заключается в следующем.

С целью обеспечения успешного обучения обучающийся должен готовиться к лекции, поскольку она является важнейшей формой организации учебного процесса, поскольку:

- знакомит с новым учебным материалом;
- разъясняет учебные элементы, трудные для понимания;
- систематизирует учебный материал;
- ориентирует в учебном процессе.

С этой целью:

- внимательно прочитайте материал предыдущей лекции;
- ознакомьтесь с учебным материалом по учебнику и учебным пособиям с темой прочитанной лекции;
- внесите дополнения к полученным ранее знаниям по теме лекции на полях лекционной тетради;
- запишите возможные вопросы, которые вы зададите лектору на лекции по материалу изученной лекции;
- постарайтесь уяснить место изучаемой темы в своей подготовке;
- узнайте тему предстоящей лекции (по тематическому плану, по информации лектора) и запишите информацию, которой вы владеете по данному вопросу.

Подготовка к занятию семинарского типа.

При подготовке и работе во время проведения занятий семинарского типа следует обратить внимание на следующие моменты: на процесс предварительной подготовки, на работу во время занятия, обработку полученных результатов, исправление полученных замечаний.

Предварительная подготовка к учебному занятию семинарского типа заключается в изучении теоретического материала в отведенное для самостоятельной работы время, ознакомление с инструктивными материалами с целью осознания задач практического занятия, техники безопасности при работе с приборами, веществами.

Работа во время проведения учебного занятия семинарского типа включает:

- консультирование студентов преподавателями и вспомогательным персоналом с целью предоставления исчерпывающей информации, необходимой для самостоятельного выполнения предложенных преподавателем задач.
- самостоятельное выполнение заданий согласно обозначенной рабочей программой дисциплины (модуля) тематики.

Самостоятельная работа.

Самостоятельная работа - планируемая учебная, учебно-исследовательская, научно-исследовательская работа студентов, выполняемая во внеаудиторное (аудиторное) время по заданию и при методическом руководстве преподавателя, но без его непосредственного участия (при частичном непосредственном участии преподавателя, оставляющем ведущую роль за работой студентов).

Для более углубленного изучения темы задания для самостоятельной работы рекомендуется выполнять параллельно с изучением данной темы. При выполнении заданий по возможности используйте наглядное представление материала.

Самостоятельная работа студентов в ВУЗе является важным видом учебной и научной деятельности студента. Самостоятельная работа студентов играет значительную роль в рейтинговой технологии обучения.

К современному специалисту в области медицины общество предъявляет достаточно широкий перечень требований, среди которых немаловажное значение имеет наличие у выпускников определенных навыков (компетенций) и умения самостоятельно добывать знания из различных источников, систематизировать полученную информацию, давать оценку конкретной профессиональной ситуации. Формирование такого умения происходит в течение всего периода обучения.

В процессе самостоятельной работы студент приобретает необходимые для будущей специальности компетенции, навыки самоорганизации, самоконтроля, самоуправления, саморефлексии и становится активным самостоятельным субъектом учебной деятельности.

Самостоятельная работа студентов должна оказывать важное влияние на формирование личности будущего специалиста, его компетентность. Каждый студент самостоятельно определяет режим своей работы и меру труда, затрачиваемого на овладение учебным содержанием по каждой дисциплине. Он выполняет внеаудиторную работу по личному индивидуальному плану, в зависимости от его подготовки, времени и других условий.

Виды самостоятельной работы.

Работа с литературой.

При работе с книгой необходимо подобрать литературу, научиться правильно ее читать, вести записи. Для подбора литературы в библиотеке используются алфавитный и систематический каталоги. Важно помнить, что рациональные навыки работы с книгой - это всегда большая экономия времени и сил. Правильный подбор учебников рекомендуется преподавателем, читающим лекционный курс. Необходимая литература может быть также указана в методических разработках по данному курсу. Изучая материал по учебнику, следует переходить к следующему вопросу только после правильного уяснения предыдущего, описывая на бумаге все выкладки и вычисления (в том числе те, которые в учебнике опущены или на лекции даны для самостоятельного вывода). При изучении любой дисциплины большую и важную роль играет самостоятельная индивидуальная работа. Особое внимание следует обратить на определение основных понятий курса. Студент должен подробно разбирать примеры, которые поясняют такие определения, и уметь строить аналогичные примеры самостоятельно. Нужно добиваться точного представления о том, что изучаешь. Полезно составлять опорные конспекты. При изучении материала по учебнику полезно в тетради (на специально отведенных полях) дополнять конспект лекций. Там же следует отмечать вопросы, выделенные студентом для консультации с преподавателем. Выводы, полученные в результате изучения, рекомендуется в конспекте выделять, чтобы они при перечитывании записей лучше запоминались. Опыт показывает, что многим студентам помогает составление листа опорных сигналов, содержащего важнейшие и наиболее часто употребляемые формулы и понятия. Такой лист помогает запомнить формулы, основные положения лекции, а также может служить постоянным справочником для студента. Различают два вида чтения: первичное и вторичное. Первичное - это внимательное, неторопливое чтение, при котором можно остановиться на трудных местах.

Задача вторичного чтения - полное усвоение смысла целого (по счету это чтение может быть и не вторым, а третьим или четвертым). Самостоятельная работа с учебниками и книгами (а также самостоятельное теоретическое исследование проблем, обозначенных преподавателем на лекциях) – это важнейшее условие формирования у себя научного способа познания.

Чтение научного текста является частью познавательной деятельности. Ее цель – извлечение из текста необходимой информации. От того на сколько осознанна читающим собственная внутренняя установка при обращении к печатному слову (найти нужные сведения, усвоить информацию полностью или частично, критически проанализировать материал и т.п.) во многом зависит эффективность осуществляемого действия.

Методические рекомендации по составлению конспекта:

1. Внимательно прочитайте текст. Уточните в справочной литературе непонятные слова. При записи не забудьте вынести справочные данные на поля конспекта;
2. Выделите главное, составьте план;
3. Кратко сформулируйте основные положения текста, отметьте аргументацию автора;
4. Законспектируйте материал, четко следуя пунктам плана. При конспектировании старайтесь выразить мысль своими словами. Записи следует вести четко, ясно.

5. Грамотно записывайте цитаты. Цитируя, учитывайте лаконичность, значимость мысли. В тексте конспекта желательно приводить не только тезисные положения, но и их доказательства. При оформлении конспекта необходимо стремиться к емкости каждого предложения. Мысли автора книги следует излагать кратко, заботясь о стиле и выразительности написанного. Число дополнительных элементов конспекта должно быть логически обоснованным, записи должны распределяться в определенной последовательности, отвечающей логической структуре произведения. Для уточнения и дополнения необходимо оставлять поля. Овладение навыками конспектирования требует от студента целеустремленности, повседневной самостоятельной работы.

Методические материалы по самостоятельному решению задач

При самостоятельном решении задач нужно обосновывать каждый этап решения, исходя из теоретических положений курса. Если студент видит несколько путей решения проблемы (задачи), то нужно сравнить их и выбрать самый рациональный. Полезно до начала вычислений составить краткий план решения проблемы (задачи). Решение проблемных задач или примеров следует излагать подробно, вычисления располагать в строгом порядке, отделяя вспомогательные вычисления от основных. Решения при необходимости нужно сопровождать комментариями, схемами, чертежами и рисунками. Следует помнить, что решение каждой учебной задачи должно доводиться до окончательного логического ответа, которого требует условие, и по возможности с выводом. Полученный ответ следует проверить способами, вытекающими из существа данной задачи. Полезно также (если возможно) решать несколькими способами и сравнить полученные результаты. Решение задач данного типа нужно продолжать до приобретения твердых навыков в их решении.

Методические материалы к выполнению реферата

Реферат (от лат. *refere* – сообщать) – краткое изложение в письменном виде или в форме публикации доклада, содержания научного труда (трудов), литературы по теме. Работа над рефератом условно разделяется на выбор темы, подбор литературы, подготовку и защиту плана; написание теоретической части и всего текста с указанием библиографических данных используемых источников, подготовку доклада, выступление с ним. Тематика рефератов полностью связана с основными вопросами изучаемого курса.

Список литературы к темам не дается, и обучающиеся самостоятельно ведут библиографический поиск, причем им не рекомендуется ограничиваться университетской библиотекой.

Важно учитывать, что написание реферата требует от обучающихся определенных усилий. Особое внимание следует уделить подбору литературы, методике ее изучения с целью отбора и обработки собранного материала, обоснованию актуальности темы и теоретического уровня обоснованности используемых в качестве примеров фактов какой-либо деятельности.

Выбрав тему реферата, начав работу над литературой, необходимо составить план. Изучая литературу, продолжается обдумывание темы, осмысливание прочитанного, делаются выписки, сопоставляются точки зрения разных авторов и т.д. Реферативная работа сводится к тому, чтобы в ней выделились две взаимосвязанные стороны: во-первых, ее следует рассматривать как учебное задание, которое должен выполнить обучаемый, а во-вторых, как форму научной работы, творческого воображения при выполнении учебного задания.

Наличие плана реферата позволяет контролировать ход работы, избежать формального переписывания текстов из первоисточников.

Оформление реферата включает титульный лист, оглавление и краткий список использованной литературы. Список использованной литературы размещается на последней странице рукописи или печатной форме реферата. Реферат выполняется в письменной или печатной форме на белых листах формата А4 (210 x 297 мм). Шрифт Times New Roman, кегель 14, через 1,5 интервала при соблюдении следующих размеров текста: верхнее поле – 25 мм, нижнее – 20 мм, левое – 30 мм, правое – 15 мм. Нумерация страниц производится сверху листа, по центру. Титульный лист нумерации не подлежит.

Рефераты должны быть написаны простым, ясным языком, без претензий на наукообразность. Следует избегать сложных грамматических оборотов, непривычных терминов и символов. Если же такие термины и символы все-таки приводятся, то необходимо разъяснять их значение при первом упоминании в тексте реферата.

Объем реферата предполагает тщательный отбор информации, необходимой для краткого изложения вопроса. Важнейший этап – редактирование готового текста реферата и подготовка к обсуждению. Обсуждение требует хорошей ориентации в материале темы, умения выделить главное, поставить дискуссионный вопрос, привлечь внимание слушателей к интересной литературе, логично и убедительно изложить свои мысли.

Рефераты обязательно подлежат защите. Процедура защиты начинается с определения оппонентов защищающего свою работу. Они стремятся дать основательный анализ работы обучающимся, обращают внимание на положительные моменты и недостатки реферата, дают общую оценку содержанию, форме преподнесения материала, характеру использованной литературы. Иногда они дополняют тот или иной раздел реферата. Последнее особенно ценно, ибо говорит о глубоком знании обучающимся-оппонентом изучаемой проблемы.

Обсуждение не ограничивается заслушиванием оппонентов. Другие обучающиеся имеют право уточнить или опровергнуть какое-либо утверждение. Преподаватель предлагает любому обучающемуся задать вопрос по существу доклада или попытаться подвести итог обсуждению.

Алгоритм работы над рефератом

1. Выбор темы

Тема должна быть сформулирована грамотно (с литературной точки зрения);

В названии реферата следует поставить четкие рамки рассмотрения темы;

Желательно избегать слишком длинных названий;

Следует по возможности воздерживаться от использования в названии спорных с научной точки зрения терминов, излишней наукообразности, а также чрезмерного упрощения формулировок.

2. Реферат следует составлять из пяти основных частей: введения; основной части; заключения; списка литературы; приложений.

3. Основные требования к введению:

Во введении не следует концентрироваться на содержании; введение должно включать краткое обоснование актуальности темы реферата, где требуется показать, почему данный вопрос может представлять научный интерес и есть ли связь представляемого материала с современностью. Таким образом, тема реферата должна быть актуальна либо с научной точки зрения, либо с современных позиций.

Очень важно выделить цель, а также задачи, которые требуется решить для выполнения цели.

Введение должно содержать краткий обзор изученной литературы, в котором указывается взятый из того или иного источника материал, кратко анализируются изученные источники, показываются их сильные и слабые стороны;

Объем введения составляет две страницы текста.

4. Требования к основной части реферата:

Основная часть содержит материал, отобранный для рассмотрения проблемы;

Также основная часть должна включать в себя собственно мнение обучающихся и сформулированные самостоятельные выводы, опирающиеся на приведенные факты;

Материал, представленный в основной части, должен быть логически изложен и распределен по параграфам, имеющим свои названия;

В изложении основной части необходимо использовать сноски (в первую очередь, когда приводятся цифры и чьи-то цитаты);

Основная часть должна содержать иллюстративный материал (графики, таблицы и т. д.);

Объем основной части составляет около 10 страниц.

5. Требования к заключению:

В заключении формулируются выводы по параграфам, обращается внимание на выдвинутые во введении задачи и цели;

Заключение должно быть четким, кратким, вытекающим из содержания основной части.

6. Требования к оформлению списка литературы (по ГОСТу):

Необходимо соблюдать правильность последовательности записи источников: сначала следует писать фамилию, а после инициалы; название работы не ставится в кавычки; после названия сокращенно пишется место издания; затем идет год издания; наконец, называется процитированная страница.

Критерии оценки реферата

Обучающийся, защищающий реферат, должен рассказать о его актуальности, поставленных целях и задачах, изученной литературе, структуре основной части, сделанных в ходе работы выводах.

По окончанию выступления ему может быть задано несколько вопросов по представленной проблеме.

Оценка складывается из соблюдения требований к реферату, грамотного раскрытия темы, умения четко рассказывать о представленном реферате, способности понять суть задаваемых по работе вопросов и найти точные ответы на них.

Методические материалы к выполнению эссе

Эссе – литературное произведение небольшого объема, обычно прозаическое, свободной композиции, передающее индивидуальные впечатления, суждения, соображения автора о той или иной проблеме, теме, о том или ином событии или явлении. Это вид самостоятельной исследовательской работы обучающихся, с целью углубления и закрепления теоретических знаний и освоения практических навыков. Цель эссе состоит в развитии самостоятельного творческого мышления и письменного изложения собственных мыслей. При написании эссе обучающийся должен представить развернутый письменный ответ на теоретический или практический актуальный вопрос, объявленный преподавателем в аудитории непосредственно перед ее написанием. В процессе написания эссе разрешается пользоваться нормативно-правовыми актами, конспектом лекций (в печатном виде). Использование интернет-ресурсов не допускается. Темы эссе преподаватель предлагает из числа тех, которые обучающиеся уже рассматривали на лекциях или семинарских занятиях, исходя из содержания заданий в составе оценочных средств. По решению преподавателя, в качестве темы эссе может быть выбрана одна или несколько тем, которые могут быть распределены между обучающимися по желанию.

Требования к выполнению эссе:

1. Проводится письменно.

2. Эссе выполняется на компьютере (гарнитура Times New Roman, шрифт 14) через 1,5 интервала с полями: верхнее, нижнее – 2; правое – 3; левое – 1,5. Отступ первой строки абзаца – 1,25. Сноски – постраничные. Таблицы и рисунки встраиваются в текст работы. При этом обязательный заголовок таблицы надо размещать над табличным полем, а рисунки сопровождать подрисуночными подписями. При включении в эссе нескольких таблиц и/или рисунков их нумерация обязательна. Обязательна и нумерация страниц. Их целесообразно проставлять внизу страницы – по середине или в правом углу. Номер страницы не ставится на титульном листе, но в общее число страниц он включается. Объем эссе, без учета

приложений, не должен превышать 5 страниц. Значительное превышение установленного объема является недостатком работы и указывает на то, что обучающийся не сумел отобрать и переработать необходимый материал.

3. Работа должна содержать собственные умозаключения по сути поставленной проблемы, включать самостоятельно проведенный анализ по сути этой проблемы, выводы, обобщающие авторскую позицию по поставленной проблеме.

Критерии оценки эссе:

«Отлично» – исключительные знания материала, абсолютное понимание сути, безукоризненное знание основных понятий и положений, логически и лексически грамотно изложенный, содержательный, аргументированный, конкретный и исчерпывающий ответ.

«Хорошо» – глубокие знания материала, правильное понимание сути, знание основных понятий и положений, содержательный, полный и конкретный ответ.

«Удовлетворительно» – твердые, но недостаточно полные знания, верное понимание сути, в целом правильный ответ.

«Неудовлетворительно» – непонимание сущности задания, грубые ошибки в ответе.

Методические материалы по выполнению тестирования.

Тестовые задания содержат вопросы и 3-4 варианта ответа по базовым положениям изучаемой темы/раздела, составлены с расчетом на знания, полученные обучающимся в процессе изучения темы/раздела.

Тестовые задания выполняются в письменной форме и сдаются преподавателю, ведущему дисциплину (модуль). На выполнение тестовых заданий обучающимся отводится 45 минут.

При обработке результатов оценочной процедуры используются: критерии оценки по содержанию и качеству полученных ответов, ключи, оценочные листы.

Критерии оценки теста:

«Зачтено» - если обучающийся правильно выполнил не менее 2/3 всей работы или допустил не более одной грубой ошибки и двух недочетов, не более одной грубой и одной негрубой ошибки, не более трех негрубых ошибок, одной негрубой ошибки и трех недочетов, при наличии четырех-пяти недочетов.

«Не зачтено» - если число ошибок и недочетов превысило норму для оценки 3 или правильно выполнено менее 2/3 всей работы.

Методические материалы по выполнению доклада.

Рекомендуется следующая структура доклада:

1. титульный лист, содержание доклада;
2. краткое изложение;
3. цели и задачи;
4. изложение характера исследований и рассмотренных проблем, гипотезы, спорные вопросы;
5. источники информации, методы сбора и анализа данных, степень их полноты и достоверности;
6. анализ и толкование полученных в работе результатов;
7. выводы и оценки;
8. библиография и приложения.

Время выступления докладчика не должно превышать 10 минут.

Основные требования к оформлению доклада:

- титульный лист должен включать название доклада, наименование предметной (цикловой) комиссии, фамилию обучающегося;
- все использованные литературные источники сопровождаются библиографическим описанием;
- приводимая цитата из источника берется в кавычки (оформляются сноски);
- единицы измерения должны применяться в соответствии с действующими стандартами;

- все названия литературных источников следует приводить в соответствии с новейшими изданиями;
- рекомендуется включение таблиц, графиков, схем, если они отражают основное содержание или улучшают ее наглядность;
- названия фирм, учреждений, организаций и предприятий должны именоваться так, как они указываются в источнике;

Критерии оценки доклада

При выполнении доклада обучающийся должен продемонстрировать умение кратко излагать прочитанный материал, а также умение обобщать и анализировать материал по теме доклада.

Презентация

Методические материалы к презентациям

1. Объём презентации 10 -20 слайдов.
2. На титульном слайде должно быть отражено:
 - наименование факультета;
 - тема презентации;
 - фамилия, имя, отчество, направление подготовки/ специальность, направленность (профиль)/ специализация, форма обучения, номер группы автора презентации;
 - фамилия, имя, отчество, степень, звание, должность руководитель работы;
 - год выполнения работы.
3. В презентации должны быть отражены обоснование актуальности представляемого материала, цели и задачи работы.
4. Содержание презентации должно включать наиболее значимый материал доклада, а также, при необходимости, таблицы, диаграммы, рисунки, фотографии, карты, видео – вставки, звуковое сопровождение.
5. Заключительный слайд должен содержать информацию об источниках информации для презентации.

Критерии оценки презентации

1. Объём презентации 10 -20 слайдов.
2. Правильность оформления титульного слайда.
3. Актуальность отобранного материала, обоснованность формулировки цели и задач работы.
4. Наглядность и логичность презентации, обоснованность использования таблиц, диаграмм, рисунков, фотографий, карт, видео – вставок, звукового сопровождения; правильный выбор шрифтов, фона, других элементов дизайна слайда.
5. Объём и качество источников информации (не менее 2-х интернет – источников и не менее 2-х литературных источников).

Методические материалы по подготовке к опросу

Самостоятельная работа обучающихся включает подготовку к опросу на практических занятиях. Для этого обучающийся изучает лекции, основную и дополнительную литературу, публикации, информацию из Интернет-ресурсов.

Тема и вопросы к практическим занятиям, вопросы для самоконтроля содержатся в рабочей программе и доводятся до студентов заранее.

Для подготовки к опросу обучающемуся необходимо ознакомиться с материалом, посвященным теме практического занятия, в учебнике или другой рекомендованной литературе, конспекте лекции, обратить внимание на усвоение основных понятий дисциплины, выявить неясные вопросы и подобрать дополнительную литературу для их освещения.

Критерии оценки опроса

«Отлично»:

- дан полный, развернутый ответ на поставленный вопрос;

- в ответе прослеживается четкая структура, логическая последовательность, отражающая сущность раскрываемых понятий, теорий, явлений;
- знание по предмету демонстрируются на фоне понимания его в системе данной науки и междисциплинарных связей;

- свободное владение терминологией;

- ответы на дополнительные вопросы четкие, краткие;

«Хорошо»:

- дан полный, развернутый ответ на поставленный вопрос, показано умение выделять существенные и несущественные признаки, причинно-следственные связи;

- ответ недостаточно логичен с единичными ошибками в частностях, исправленные студентом с помощью преподавателя;

- единичные ошибки в терминологии;

- ответы на дополнительные вопросы правильные, недостаточно полные и четкие.

«Удовлетворительно»:

- ответ не полный, с ошибками в деталях, умение раскрыть значение обобщённых знаний не показано, речевое оформление требует поправок, коррекции;

- логика и последовательность изложения имеют нарушения, студент не способен самостоятельно выделить существенные и несущественные признаки и причинно-следственные связи;

- ошибки в раскрываемых понятиях, терминах;

- студент не ориентируется в теме, допускает серьезные ошибки;

- студент не может ответить на большую часть дополнительных вопросов.

«Неудовлетворительно»:

- ответ представляет собой разрозненные знания с существенными ошибками по вопросу;

- присутствуют фрагментарность, нелогичность изложения, студент не осознает связь обсуждаемого вопроса с другими объектами дисциплины, речь неграмотная;

- незнание терминологии;

- ответы на дополнительные вопросы неправильные.

Методические материалы по выполнению практического задания

При выполнении практического задания обучающийся придерживается следующего алгоритма:

1. Записать дату, тему и цель задания;

2. Ознакомиться с правилами и условия выполнения практического задания;

3. Повторить теоретические задания, необходимые для рациональной работы и других практических действий, используя конспекты лекций и рекомендованную литературу, представленной в программе;

4. Выполнить работу по предложенному алгоритму действий;

5. Обобщить результаты работы, сформулировать выводы / дать ответы на контрольные вопросы;

Работа должна быть выполнена грамотно, с соблюдением культуры изложения. При использовании данных из учебных, методических пособий и другой литературы, периодических изданий, Интернет-источников должны иметься ссылки на вышеперечисленные.

Критерии оценки практического задания:

«Отлично» – правильный ответ, дается четкое обоснование принятому решению; рассуждения четкие последовательные логические; используются ссылки на полученные при изучении дисциплины знания; правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Хорошо» – правильный ответ, дается обоснование принятому решению; но с не существенными ошибками, в рассуждениях отсутствует логическая последовательность; используются ссылки на полученные при изучении дисциплины знания, правильно

используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Удовлетворительно» – правильный ответ, допускаются грубые ошибки в обосновании принятого решения; рассуждения не последовательные сумбурные; используются ссылки на полученные при изучении дисциплины знания; используются формулы, процедуры, понятия, имеющие прямое значение для подтверждения принятого решения, однако, при обращении к ним допускаются серьезные ошибки, студент не может правильно ими воспользоваться.

«Неудовлетворительно, не зачтено» – ответ неверный, отсутствует обоснование принятому решению; студент демонстрирует полное непонимание сути вопроса.

Для оценки решения ситуационной задачи (аналитического задания):

Оценка «отлично» выставляется, если задача решена грамотно, ответы на вопросы сформулированы четко. Эталонный ответ полностью соответствует решению студента, которое хорошо обосновано теоретически.

Оценка «хорошо» выставляется, если задача решена, ответы на вопросы сформулированы не достаточно четко. Решение студента в целом соответствует эталонному ответу, но не достаточно хорошо обосновано теоретически.

Оценка «удовлетворительно» выставляется, если задача решена не полностью, ответы не содержат всех необходимых обоснований решения.

Оценка «неудовлетворительно» выставляется, если задача не решена или имеет грубые теоретические ошибки в ответе на поставленные вопросы.

Методические материалы по выполнению лабораторного задания

При выполнении лабораторного задания обучающийся придерживается следующего алгоритма:

1. Записать дату, тему и цель задания;
2. Ознакомиться с правилами и условия выполнения задания;
3. Повторить теоретические задания, необходимые для рациональной работы и других практических действий, используя конспекты лекций и рекомендованную литературу, представленной в программе;
4. Выполнить работу по предложенному алгоритму действий;
5. Обобщить результаты работы, сформулировать выводы / дать ответы на контрольные вопросы;

Работа должна быть выполнена грамотно, с соблюдением культуры изложения. При использовании данных из учебных, методических пособий и другой литературы, периодических изданий, Интернет-источников должны иметься ссылки на вышеперечисленные.

Критерии оценки лабораторного задания:

«Отлично» – правильный ответ, дается четкое обоснование принятому решению; рассуждения четкие последовательные логические; используются ссылки на полученные при изучении дисциплины знания; правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Хорошо» – правильный ответ, дается обоснование принятому решению; но с не существенными ошибками, в рассуждениях отсутствует логическая последовательность; используются ссылки на полученные при изучении дисциплины знания, правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Удовлетворительно» – правильный ответ, допускаются грубые ошибки в обосновании принятого решения; рассуждения не последовательные сумбурные; используются ссылки на полученные при изучении дисциплины знания; используются формулы, процедуры, понятия, имеющие прямое значение для подтверждения принятого решения, однако, при обращении к ним допускаются серьезные ошибки, студент не может правильно ими воспользоваться.

«Неудовлетворительно, не зачтено» – ответ неверный, отсутствует обоснование принятому решению; студент демонстрирует полное непонимание сути вопроса.

Методические указания для подготовки к промежуточной аттестации.

Изучение учебных дисциплин (модулей) завершается зачетом/зачетом с оценкой или экзаменом. Подготовка к промежуточной аттестации способствует закреплению, углублению

и обобщению знаний, получаемых, в процессе обучения, а также применению их к решению практических задач. Готовясь к экзамену, студент ликвидирует имеющиеся пробелы в знаниях, углубляет, систематизирует и упорядочивает свои знания. На зачете или экзамене студент демонстрирует то, что он освоил в процессе обучения по дисциплине (модулю).

Вначале следует просмотреть весь материал по дисциплине (модулю), отметить для себя трудные вопросы. Обязательно в них разобраться. В заключение еще раз целесообразно повторить основные положения, используя при этом листы опорных сигналов. Систематическая подготовка к занятиям в течение семестра позволит использовать время промежуточной аттестации для систематизации знаний.

3. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНИВАНИЯ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ И (ИЛИ) ОПЫТА ДЕЯТЕЛЬНОСТИ, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ

3.1. Организационные основы применения балльно-рейтинговой системы оценки успеваемости обучающихся по дисциплине (модулю)

Оценка качества освоения обучающимися дисциплины (модуля) реализуется в формате балльно-рейтинговой системы оценки успеваемости обучающихся (БРСО).

БРСО в ходе текущего контроля успеваемости и промежуточной аттестации осуществляется по 100-балльной шкале.

Академический рейтинг обучающегося по дисциплине (модулю) складывается из результатов:

- текущего контроля успеваемости (максимальный текущий рейтинг обучающегося 80 рейтинговых баллов);
- промежуточной аттестации (максимальный рубежный рейтинг обучающегося 20 рейтинговых баллов).

Условия оценки освоения обучающимся дисциплины (модуля) в формате БРСО доводятся преподавателем до сведения обучающихся на первом учебном занятии, а также размещены в свободном доступе в электронной информационно-образовательной среде Университета.

3.2. Проведение текущего контроля успеваемости обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося

В течение учебного семестра до промежуточной аттестации на основании утвержденной рабочей программы дисциплины (модуля) формируется текущий рейтинг обучающегося. Текущий рейтинг обучающегося складывается как сумма рейтинговых баллов, полученных им в течение учебного семестра по всем видам учебных занятий по учебной дисциплине.

В процессе текущего контроля оцениваются следующие действия обучающегося, направленные на освоение компетенций в рамках изучения учебной дисциплины:

- академическая активность (посещаемость учебных занятий, самостоятельное изучение содержания учебной дисциплины в электронной информационно-образовательной среде, соблюдение сроков сдачи практических заданий и текущих контрольных мероприятий и др.);
- выполнение и сдача текущих и итогового практических заданий (эссе, рефераты, творческие задания, кейс-задания, лабораторные работы, расчетные задания и др., активное участие в групповых интерактивных занятиях (дискуссии, WiKi-проекты и др.), защита проектов и др.);
- прохождение рубежей текущего контроля, включая соблюдение графика их прохождения в электронной информационно-образовательной среде.

Для планирования расчета текущего рейтинга обучающегося используются следующие пропорции:

Вид учебного действия	Максимальная рейтинговая оценка, баллов
академическая активность	10
практические задания	40
<i>из них: текущие практические задания</i>	20
<i>итоговое практическое задание</i>	20
рубежи текущего контроля	30
ИТОГО:	80

В течение учебного семестра по дисциплине (модулю) обучающимся должен быть накоплен текущий рейтинг не менее 52 рейтинговых баллов (65% от максимального значения текущего рейтинга).

Необходимыми условиями допуска обучающегося к промежуточной аттестации по дисциплине являются положительное прохождение обучающимся не менее 65% рубежей текущего контроля с накоплением не менее 65% максимального рейтингового балла за каждый рубеж текущего контроля и положительное выполнение итогового практического задания с накоплением не менее 65% максимального рейтингового балла, установленного за итоговое практическое задание.

Невыполнение вышеуказанных условий является текущей академической задолженностью, которая должна быть ликвидирована обучающимся до контрольного мероприятия промежуточной аттестации.

Сведения о наличии у обучающихся текущей академической задолженности, сроках и порядке добора рейтинговых баллов для её ликвидации доводятся до обучающихся педагогическим работником.

В случае неликвидации текущей академической задолженности, педагогический работник обязан во время контрольного мероприятия промежуточной аттестации поставить обучающемуся 0 рейтинговых баллов. В этом случае ликвидация текущей академической задолженности возможна в периоды проведения повторной промежуточной аттестации.

3.3. Проведение промежуточной аттестации обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося

Промежуточная аттестация по дисциплине (модулю) проводится в соответствии с Положением о промежуточной аттестации обучающихся по основным профессиональным образовательным программам высшего образования – программ специалитета в Российском государственном социальном университете и Положением о балльно-рейтинговой системе оценки успеваемости обучающихся по основным профессиональным образовательным программам высшего образования - программам бакалавриата, программам специалитета, программам магистратуры в Российском государственном социальном университете.

На промежуточную аттестацию отводится 20 рейтинговых баллов.

Ответы обучающегося на контрольном мероприятии промежуточной аттестации оцениваются педагогическим работником по 20 - балльной шкале, а итоговая оценка по дисциплине (модулю) выставляется по системе зачтено/не зачтено.

Критерии выставления оценки определяются Положением о балльно-рейтинговой системе оценки успеваемости обучающихся по основным профессиональным образовательным программам высшего образования - программам бакалавриата, программам специалитета, программам магистратуры в Российском государственном социальном университете.

В процессе определения рубежного рейтинга обучающегося используется следующая шкала:

Рубежный рейтинг	Критерии оценки освоения обучающимся учебной дисциплины в ходе контрольных мероприятий промежуточной аттестации
19-20 рейтинговых баллов	обучающийся глубоко и прочно усвоил программный материал, исчерпывающе, последовательно, грамотно и логически стройно его излагает, тесно увязывает с задачами и будущей деятельностью, не затрудняется с ответом при видоизменении задания, свободно справляется с задачами и практическими заданиями, правильно обосновывает принятые решения, умеет самостоятельно обобщать и излагать материал, не допуская
16-18 рейтинговых баллов	обучающийся твердо знает программный материал, грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос, может правильно применять теоретические положения и владеет необходимыми умениями и навыками при выполнении практических
13-15 рейтинговых баллов	обучающийся освоил основной материал, но не знает отдельных деталей, допускает неточности, недостаточно правильные формулировки, нарушает последовательность в изложении программного материала и испытывает затруднения в выполнении практических заданий
1-12 рейтинговых баллов	обучающийся не знает значительной части программного материала, допускает существенные ошибки, с большими затруднениями выполняет практические задания
0 рейтинговых баллов	не аттестован

Если результат контроля успеваемости в рамках проведения контрольных мероприятий промежуточной аттестации (рубежный рейтинг обучающегося) неудовлетворительный (получено менее 13 рейтинговых баллов), то промежуточная аттестация по учебной дисциплине (модулю) невозможна даже при наличии высокого текущего рейтинга, полученного по итогам текущего контроля по учебной дисциплине (модулю).

ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

№ п/п	Содержание изменения	Реквизиты документа об утверждении изменения	Дата введения изменения
1.			
2.			
3.			
4.			



**Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Российский государственный социальный университет»**

УТВЕРЖДАЮ
Заведующий кафедрой комплекса естественно-
научных дисциплин

/Денисова Д.А./

«27» февраля 2024 г.

**МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ
«ДИСКРЕТНАЯ МАТЕМАТИКА»**

Направление подготовки
«Информационные системы и технологии»

Направленность
«Интеллектуальные информационные системы и технологии»

**ОБРАЗОВАТЕЛЬНАЯ ПРОГРАММА ВЫСШЕГО ОБРАЗОВАНИЯ - ПРОГРАММА
БАКАЛАВРИАТА**

Форма обучения

Очная, заочная

Москва, 2024 г.

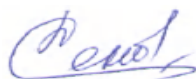
Методические материалы по дисциплине (модулю) «Дискретная математика» разработаны на основании федерального государственного образовательного стандарта высшего образования – *бакалавриата* по направлению подготовки *09.03.02 Информационные системы и технологии*, утвержденного приказом Министерства образования и науки Российской Федерации от 19.09.2017 № 926, учебного плана по основной профессиональной образовательной программе высшего образования – программе бакалавриата по направлению подготовки *09.03.02 Информационные системы и технологии* (далее – «ОПОП»).

Методические материалы по дисциплине (модулю) разработаны канд. пед. наук, доцентом Пивневой С.В.

Методические материалы по дисциплине (модулю) разработаны на заседании кафедры комплекса естественно-научных дисциплин.

Протокол № 9 от «27» февраля 2024 года

Заведующий кафедрой
канд. техн. наук, доцент



Д.А. Денисова

(подпись)

СОДЕРЖАНИЕ

1. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ К ЛЕКЦИОННЫМ И ПРАКТИЧЕСКИМ ЗАНЯТИЯМ.....	4
1.1. Методические материалы к проведению лекционных занятий по дисциплине (модулю)	4
1.2. Методические материалы по подготовке к практическим занятиям по дисциплине (модулю)	6
2. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ) И САМОСТОЯТЕЛЬНОЙ РАБОТЕ.....	10
3. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНИВАНИЯ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ И (ИЛИ) ОПЫТА ДЕЯТЕЛЬНОСТИ, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ.....	20
3.1. Организационные основы применения балльно-рейтинговой системы оценки успеваемости обучающихся по дисциплине (модулю)	20
3.2. Проведение текущего контроля успеваемости обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося.....	20
3.3. Проведение промежуточной аттестации обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося.....	21
ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ	23

1. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ К ЛЕКЦИОННЫМ И ПРАКТИЧЕСКИМ ЗАНЯТИЯМ

1.1. Методические материалы к проведению лекционных занятий по дисциплине (модулю)

Лекция - один из методов обучения, одна из основных системообразующих форм организации учебного процесса в вузе. Лекционное занятие представляет собой систематическое, последовательное, монологическое изложение педагогическим работником учебного материала, как правило, теоретического характера. Такое занятие представляет собой элемент технологии представления учебного материала путем логически стройного, систематически последовательного и ясного изложения.

Цель лекции – организация целенаправленной познавательной деятельности обучающихся по овладению программным материалом дисциплины (модуля). Чтение курса лекций позволяет дать связанное, последовательное изложение материала в соответствии с новейшими данными науки, сообщить слушателям основное содержание предмета в целостном, систематизированном виде. В ряде случаев лекция выполняет функцию основного источника информации: при отсутствии учебников и учебных пособий, чаще по новым курсам; в случае, когда новые научные данные по той или иной теме не нашли отражения в учебниках; отдельные разделы и темы очень сложны для самостоятельного изучения. В таких случаях только лектор может методически помочь обучающимся в освоении сложного материала.

Возможные формы проведения лекций:

- Вводная лекция – один из наиболее важных и трудных видов лекции при чтении систематических курсов. От успеха этой лекции во многом зависит успех усвоения всего курса. Она может содержать: определение дисциплины (модуля); краткую историческую справку о дисциплине (модуле); цели и задачи дисциплины (модуля), ее роль в общей системе обучения и связь со смежными дисциплинами (модулями); основные проблемы (понятия и определения) данной науки; основную и дополнительную учебную литературу; особенности самостоятельной работы обучающихся над дисциплиной (модулем) и формы участия в научно-исследовательской работе; отчетность по курсу.

- Информационная лекция ориентирована на изложение и объяснение обучающимся научной информации, подлежащей осмыслению и запоминанию. Это самый традиционный тип лекций в практике высшей школы.

- Заключительная лекция предназначена для обобщения полученных знаний и раскрытия перспектив дальнейшего развития данной науки.

- Обзорная лекция – это систематизация научных знаний на высоком уровне, допускающая большое число ассоциативных связей в процессе осмысления информации, излагаемой при раскрытии внутрипредметной и межпредметной связей, исключая детализацию и конкретизацию. Как правило, стержень излагаемых теоретических положений составляет научно-понятийная и концептуальная основа всего курса или крупных его разделов.

- Лекция-беседа - непосредственный контакт педагогического работника с аудиторией - диалог. По ходу лекции педагогический работник задает вопросы для выяснения мнений и уровня осведомленности обучающихся по рассматриваемой проблеме.

- Лекция-дискуссия - свободный обмен мнениями в ходе изложения лекционного материала. Педагогический работник активизирует участие в обсуждении отдельными вопросами, сопоставляет между собой различные мнения и тем самым развивает дискуссию, стремясь направить ее в нужное русло.

- Лекция с применением обратной связи включает в себе то, что в начале и конце каждого раздела лекции задаются вопросы. Первый - для того, чтобы узнать, насколько обучающиеся ориентируются в излагаемом материале, вопрос в конце раздела предназначен для выяснения степени усвоения только что изложенного материала. При

неудовлетворительных результатах контрольного опроса педагогический работник возвращается к уже прочитанному разделу, изменив при этом методику подачи материала.

- Проблемная лекция опирается на логику последовательно моделируемых проблемных ситуаций путем постановки проблемных вопросов или предъявления проблемных задач. Проблемный вопрос - это диалектическое противоречие, требующее для своего решения размышления, сравнения, поиска, приобретения и применения новых знаний. Проблемная задача содержит дополнительную вводную информацию и при необходимости некоторые ориентиры поиска ее решения.

- Программированная лекция - консультация – педагогический работник сам составляет и предлагает обучающимся вопросы. На подготовленные вопросы педагогический работник сначала просит ответить обучающихся, а затем проводит анализ и обсуждение неправильных ответов. В лекциях можно использовать наглядные материалы, а также подготовить презентацию. Что касается презентации, то в качестве визуальной поддержки ее можно органично интегрировать во все вышеупомянутые лекции. В то же время лекцию-презентацию возможно выделить и в качестве самостоятельной формы. Лекция-презентация должна отражать суть основных и (или) проблемных вопросов лекции, на которые особо следует обратить внимание обучающихся. В условиях применения активного метода проведения занятий презентация представляется весьма удачным способом донесения информации до слушателей. Единственное, на что следует обратить внимание при подготовке слайдов, - это их оформление и текст. Слайд не должен быть перегружен картинками и лишней информацией, которая будет отвлекать от основного аспекта того или иного вопроса лекции. Во время лекции можно задавать вопросы аудитории в отношении того или иного слайда, тем самым еще больше вовлекая обучающихся в проблематику.

Краткое содержание лекционных занятий

Наименование разделов	Содержание учебного материала
Раздел 1. Комбинаторика	
Тема 1.1. Комбинаторные формулы для сочетаний, размещений, перестановок	Элементы комбинаторики. Основные принципы комбинаторики. Перестановки, размещения, сочетания. Свойства сочетаний.
Тема 1.2. Комбинаторные формулы для сочетаний, размещений, перестановок с повторениями	Перестановки с повторениями, размещения с повторениями, сочетания с повторениями.
Раздел 2. Функции алгебры логики	
Тема 2.1. Логические операции	Конъюнкция. Дизъюнкция. Отрицание. Импликация. Эквивалентность.
Тема 2.2. Булевы функции	Булева функция. Фиктивные и существенные переменные.
Раздел 3. Элементы теории графов	
Тема 3.1. Основы теории графов	Возникновение дискретной математики. Задачи на графах. Понятие графа, методы представления графа. Примеры. Свойства матриц графа. Изоморфизм графов. Мультиграф, орграф, взвешенный граф. Степень вершины. Свойства степеней вершин графа. Полный граф. Дополнение к графу. Цепь и путь в графе. Примеры. Связность графа. Циклы графа. Деревья. Свойства деревьев. Остовное дерево графа. Конъюнкция. Дизъюнкция. Отрицание. Импликация. Эквивалентность.
Тема 3.2. Эйлеровы и Гамильтоновы графы	Задача о Кенигсбергских мостах. Эйлеров цикл. Уникурсальная линия. Теорема Эйлера. Алгоритм

	построения Эйлера цикла. Гамильтонов путь и цикл. Задача Гамильтона. Признаки существования Гамильтонова пути и цикла в графе. Задача коммивояжера. Путь и цепь в орграфе.
Раздел 4. Элементы теории автоматов и вычислимых функций	
Тема 4.1. Понятие конечного автомата и способы его задания	Понятие автомата, способы задания автоматов. Эквивалентность автоматов. Автоматы Мура и их связь с автоматами Мили. Минимизация конечных автоматов.
Тема 4.2. Машины Тьюринга	Понятие машины Тьюринга. Тезис Тьюринга.

1.2. Методические материалы по подготовке к практическим занятиям по дисциплине (модулю)

Практические (семинарские) занятия - одна из форм учебного занятия, направленная на развитие самостоятельности обучающихся и приобретение умений и навыков. Данные учебные занятия углубляют, расширяют, детализируют полученные ранее знания. Практическое занятие предполагает выполнение обучающимися по заданию и под руководством преподавателей одной или нескольких практических работ.

Цель практических занятий и семинаров состоит в развитии познавательных способностей, самостоятельности мышления и творческой активности обучающихся; углублении, расширении, детализировании знаний, полученных на лекции в обобщенной форме, и содействии выработке навыков профессиональной деятельности. В отдельных случаях на практических занятиях и семинарах руководителем занятия сообщаются дополнительные знания.

Для достижения поставленных целей и решения требуемого перечня задач практические занятия и семинары проводятся традиционными технологиями или с использованием активных и интерактивных образовательных технологий.

Возможные формы проведения практических (семинарских) занятий:

- Деловая игра - это метод группового обучения совместной деятельности в процессе решения общих задач в условиях максимально возможного приближения к реальным проблемным ситуациям. Имитационные игры - на занятиях имитируется деятельность какой-либо организации, предприятия или его подразделения. Имитироваться могут события, конкретная деятельность людей (деловое совещание, обсуждение плана) и обстановка, условия, в которых происходит событие или осуществляется деятельность (кабинет начальника цеха, зал заседаний). Исполнение ролей (ролевые игры) - в этих играх отрабатывается тактика поведения, действий, выполнение функций и обязанностей конкретного лица. Для проведения игр с исполнением роли разрабатывается модель-пьеса ситуации, между студентами распределяются роли с «обязательным содержанием», характеризующиеся различными интересами; в процессе их взаимодействия должно быть найдено компромиссное решение. «Деловой театр» (метод инсценировки) - в нем разыгрывается какая-либо ситуация, поведение человека в этой обстановке, обучающийся должен вжиться в образ определенного лица, понять его действия, оценить обстановку и найти правильную линию поведения. Основная задача метода инсценировки - научить ориентироваться в различных обстоятельствах, давать объективную оценку своему поведению, учитывать возможности других людей, влиять на их интересы, потребности и деятельность, не прибегая к формальным атрибутам власти, к приказу.

- Игровое проектирование - является практическим занятием или циклом занятий, суть которых состоит в разработке инженерного, конструкторского, технологического и других видов проектов в игровых условиях, максимально воссоздающих реальность. Этот метод отличается высокой степенью сочетания индивидуальной и совместной работы обучающихся.

- Познавательно-дидактические игры не относятся к деловым играм. Они предполагают лишь включение изучаемого материала в необычный игровой контекст и иногда содержат лишь элементы ролевых игр. Такие игры могут проводиться в виде копирования научных, культурных, социальных явлений (конкурс знатоков, «Поле чудес», КВН и т.д.) и в виде предметно-содержательных моделей, (например, игры-путешествия, когда надо разработать рациональный маршрут, пользуясь различными картами).

- Анализ конкретных ситуаций. Конкретная ситуация – это любое событие, которое содержит в себе противоречие или вступает в противоречие с окружающей средой. Ситуации могут нести в себе как позитивный, так и отрицательный опыт. Все ситуации делятся на простые, критические и экстремальные.

- Кейс-метод (от английского case – случай, ситуация) – усовершенствованный метод анализа конкретных ситуаций, метод активного проблемно-ситуационного анализа, основанный на обучении путем решения конкретных задач – ситуаций (решение кейсов). Непосредственная цель метода case-study – обучающиеся должны проанализировать ситуацию, разобраться в сути проблем, предложить возможные решения и выбрать лучшее из них. Кейсы делятся на практические (отражающие реальные жизненные ситуации), обучающие (искусственно созданные, содержащие значительные элементы условности при отражении в нем жизни) и исследовательские (ориентированные на проведение исследовательской деятельности посредством применения метода моделирования). Метод конкретных ситуаций (метод case-study) относится к неигровым имитационным активным методам обучения.

- Тренинг (англ. training от train — обучать, воспитывать) – метод активного обучения, направленный на развитие знаний, умений и навыков и социальных установок. Тренинг – форма интерактивного обучения, целью которого является развитие компетентности межличностного и профессионального поведения в общении. Достоинство тренинга заключается в том, что он обеспечивает активное вовлечение всех участников в процесс обучения. Можно выделить основные типы тренингов по критерию направленности воздействия и изменений – навыковый, психотерапевтический, социально-психологический, бизнес-тренинг.

- Метод Сократа (Майевтика) – метод вопросов, предполагающих критическое отношение к догматическим утверждениям, называется еще как метод «сократовской иронии». Это умение извлекать скрытое в человеке знание с помощью искусных наводящих вопросов, подразумевающего короткий, простой и заранее предсказуемый ответ.

- Интерактивная лекция – выступление ведущего обучающего перед большой аудиторией с применением следующих активных форм обучения: дискуссия, беседа, демонстрация слайдов или учебных фильмов, мозговой штурм.

- Групповая, научная дискуссия, диспут. Дискуссия — это целенаправленное обсуждение конкретного вопроса, сопровождающееся обменом мнениями, идеями между двумя и более лицами. Задача дискуссии – обнаружить различия в понимании вопроса и в споре установить истину. Дискуссии могут быть свободными и управляемыми. К технике управляемой дискуссии относятся: четкое определение цели, прогнозирование реакции оппонентов, планирование своего поведения, ограничение времени на выступления и их заданная очередность. Разновидностью свободной дискуссии является форум, где каждому желающему дается неограниченное время на выступление, при условии, что его выступление вызывает интерес аудитории. Каждый конкретный форум имеет свою тематику — достаточно широкую, чтобы в её пределах можно было вести многоплановое обсуждение.

- Дебаты – это четко структурированный и специально организованный публичный обмен мыслями между двумя сторонами по актуальным темам. Это разновидность публичной дискуссии участников дебатов, направляющая на переубеждение в своей правоте третьей стороны, а не друг друга. Поэтому вербальные и невербальные средства, которые используются участниками дебатов, имеют целью получения определённого результата — сформировать у слушателей положительное впечатление от собственной позиции.

- Метод работы в малых группах. Групповое обсуждение кого-либо вопроса направлено на достижение лучшего взаимопонимания и нахождения истины. Групповое обсуждение способствует лучшему усвоению изучаемого материала. Оптимальное количество участников - 5-7 человек. Перед обучающимися ставится проблема, выделяется определенное время, в течение которого они должны подготовить аргументированный обдуманный ответ. Педагогический работник может устанавливать правила проведения группового обсуждения – задавать определенные рамки обсуждения, ввести алгоритм выработки общего мнения, назначить лидера и др.

- Круглый стол - общество, собрание в рамках более крупного мероприятия (съезда, симпозиума, конференции). Мероприятие, как правило, на которое приглашаются эксперты и специалисты из разных сфер деятельности для обсуждения актуальных вопросов. Данная модель обсуждения, основываясь на соглашениях, в качестве итогов даёт результаты, которые, в свою очередь, являются новыми соглашениями.

- Коллоквиум - (лат. colloquium — разговор, беседа) - одна из форм учебных занятий в системе образования, имеющая целью выяснение и повышение знаний обучающихся. На коллоквиумах обсуждаются: отдельные части, разделы, темы, вопросы изучаемого курса (обычно не включаемые в тематику семинарских и других практических учебных занятий), рефераты, проекты и др. работы обучающихся. Это научные собрания, на которых заслушиваются и обсуждаются доклады. Коллоквиум – это и форма контроля, массового опроса, позволяющая преподавателю в сравнительно небольшой срок выяснить уровень знаний студентов по данной теме дисциплины. Коллоквиум проходит обычно в форме дискуссии, в ходе которой обучающимся предоставляется возможность высказать свою точку зрения на рассматриваемую проблему, учиться обосновывать и защищать ее. Аргументируя и отстаивая свое мнение, обучающийся в то же время демонстрирует, насколько глубоко и осознанно он усвоил изученный материал.

- Метод «мозговой штурм» (мозговой штурм, мозговая атака, англ. brainstorming) — оперативный метод решения проблемы на основе стимулирования творческой активности, при котором участникам обсуждения предлагают высказывать как можно большее количество вариантов решения, в том числе самых фантастичных. Затем из общего числа высказанных идей отбирают наиболее удачные, которые могут быть использованы на практике. Является методом экспертного оценивания.

- Метод проектов - это способ достижения дидактической цели через детальную разработку проблемы (технологию), которая должна завершиться вполне реальным, осязаемым практическим результатом, оформленным тем или иным образом; это совокупность приёмов, действий обучающихся в их определённой последовательности для достижения поставленной задачи – решения проблемы, лично значимой для учащихся и оформленной в виде некоего конечного продукта. Основное предназначение метода проектов состоит в предоставлении учащимся возможности самостоятельного приобретения знаний в процессе решения практических задач или проблем, требующего интеграции знаний из различных предметных областей.

- Брифинг - (англ. briefing от англ. brief – короткий, недолгий) – краткая пресс-конференция, посвященная одному вопросу. Основное отличие: отсутствует презентационная часть. То есть практически сразу идут ответы на вопросы журналистов.

- Метод портфолио (итал. portfolio — 'портфель, англ. - папка для документов) - современная образовательная технология, в основе которой используется метод аутентичного оценивания результатов образовательной и профессиональной деятельности. Портфолио как подборка сертифицированных достижений, наиболее значимых работ и отзывов на них.

Вопросы для самоподготовки к практическим (семинарским) занятиям по разделам (темам) дисциплины (модуля)

Задания для самостоятельной работы к Разделу 1

Литература для самостоятельного изучения к Разделу 1.

1. Дискретная математика : учебное пособие для вузов / Д. С. Ананичев [и др.] ; под научной редакцией А. Н. Сесекина. — Москва : Издательство Юрайт, 2022. — 108 с. — (Высшее образование). — ISBN 978-5-534-08214-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/492307> (дата обращения: 03.03.2023).
2. Гашков, С. Б. Дискретная математика : учебник и практикум для вузов / С. Б. Гашков, А. Б. Фролов. — 3-е изд., испр. и доп. — Москва : Издательство Юрайт, 2023. — 483 с. — (Высшее образование). — ISBN 978-5-534-11613-7. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511483> (дата обращения: 03.03.2023).
3. Таранников, Ю. В. Дискретная математика. Задачник : учебное пособие для вузов / Ю. В. Таранников. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-534-01180-7. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511496> (дата обращения: 03.03.2023).
4. Палий, И. А. Дискретная математика и математическая логика : учебное пособие для вузов / И. А. Палий. — 3-е изд., испр. и доп. — Москва : Издательство Юрайт, 2023. — 370 с. — (Высшее образование). — ISBN 978-5-534-12446-0. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/515003> (дата обращения: 03.03.2023).

Задания для самостоятельной работы к Разделу 2

Литература для самостоятельного изучения к Разделу 2.

1. Дискретная математика : учебное пособие для вузов / Д. С. Ананичев [и др.] ; под научной редакцией А. Н. Сесекина. — Москва : Издательство Юрайт, 2022. — 108 с. — (Высшее образование). — ISBN 978-5-534-08214-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/492307> (дата обращения: 03.03.2023).
2. Гашков, С. Б. Дискретная математика : учебник и практикум для вузов / С. Б. Гашков, А. Б. Фролов. — 3-е изд., испр. и доп. — Москва : Издательство Юрайт, 2023. — 483 с. — (Высшее образование). — ISBN 978-5-534-11613-7. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511483> (дата обращения: 03.03.2023).
3. Таранников, Ю. В. Дискретная математика. Задачник : учебное пособие для вузов / Ю. В. Таранников. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-534-01180-7. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511496> (дата обращения: 03.03.2023).
4. Палий, И. А. Дискретная математика и математическая логика : учебное пособие для вузов / И. А. Палий. — 3-е изд., испр. и доп. — Москва : Издательство Юрайт, 2023. — 370 с. — (Высшее образование). — ISBN 978-5-534-12446-0. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/515003> (дата обращения: 03.03.2023).

Задания для самостоятельной работы к Разделу 3

Литература для самостоятельного изучения к Разделу 3.

1. Дискретная математика : учебное пособие для вузов / Д. С. Ананичев [и др.] ; под научной редакцией А. Н. Сесекина. — Москва : Издательство Юрайт, 2022. — 108 с. —

(Высшее образование). — ISBN 978-5-534-08214-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/492307> (дата обращения: 03.03.2023).

2. Гашков, С. Б. Дискретная математика : учебник и практикум для вузов / С. Б. Гашков, А. Б. Фролов. — 3-е изд., испр. и доп. — Москва : Издательство Юрайт, 2023. — 483 с. — (Высшее образование). — ISBN 978-5-534-11613-7. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511483> (дата обращения: 03.03.2023).

3. Таранников, Ю. В. Дискретная математика. Задачник : учебное пособие для вузов / Ю. В. Таранников. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-534-01180-7. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511496> (дата обращения: 03.03.2023).

4. Палий, И. А. Дискретная математика и математическая логика : учебное пособие для вузов / И. А. Палий. — 3-е изд., испр. и доп. — Москва : Издательство Юрайт, 2023. — 370 с. — (Высшее образование). — ISBN 978-5-534-12446-0. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/515003> (дата обращения: 03.03.2023).

Задания для самостоятельной работы к Разделу 4

Литература для самостоятельного изучения к Разделу 4.

1. Дискретная математика : учебное пособие для вузов / Д. С. Ананичев [и др.] ; под научной редакцией А. Н. Сесекина. — Москва : Издательство Юрайт, 2022. — 108 с. — (Высшее образование). — ISBN 978-5-534-08214-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/492307> (дата обращения: 03.03.2023).

2. Гашков, С. Б. Дискретная математика : учебник и практикум для вузов / С. Б. Гашков, А. Б. Фролов. — 3-е изд., испр. и доп. — Москва : Издательство Юрайт, 2023. — 483 с. — (Высшее образование). — ISBN 978-5-534-11613-7. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511483> (дата обращения: 03.03.2023).

3. Таранников, Ю. В. Дискретная математика. Задачник : учебное пособие для вузов / Ю. В. Таранников. — Москва : Издательство Юрайт, 2023. — 385 с. — (Высшее образование). — ISBN 978-5-534-01180-7. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511496> (дата обращения: 03.03.2023).

4. Палий, И. А. Дискретная математика и математическая логика : учебное пособие для вузов / И. А. Палий. — 3-е изд., испр. и доп. — Москва : Издательство Юрайт, 2023. — 370 с. — (Высшее образование). — ISBN 978-5-534-12446-0. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/515003> (дата обращения: 03.03.2023).

2. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ) И САМОСТОЯТЕЛЬНОЙ РАБОТЕ

Освоение обучающимся дисциплины (модуля) «Технологии возможностей и безбарьерной среды» предполагает изучение материалов дисциплины на аудиторных занятиях и в ходе самостоятельной работы. Аудиторные занятия проходят в форме лекций, семинаров, практических и лабораторных занятий. Самостоятельная работа включает разнообразный комплекс видов и форм работы обучающихся.

Для успешного освоения дисциплины (модуля) и достижения поставленных целей необходимо внимательно ознакомиться с рабочей программы дисциплины (модуля), доступной в электронной информационно-образовательной среде РГСУ.

Следует обратить внимание на списки основной и дополнительной литературы, на предлагаемые преподавателем ресурсы информационно-телекоммуникационной сети Интернет. Эта информация необходима для самостоятельной работы обучающегося.

При подготовке к аудиторным занятиям необходимо помнить особенности каждой формы его проведения.

Подготовка к учебному занятию лекционного типа заключается в следующем.

С целью обеспечения успешного обучения обучающийся должен готовиться к лекции, поскольку она является важнейшей формой организации учебного процесса, поскольку:

- знакомит с новым учебным материалом;
- разъясняет учебные элементы, трудные для понимания;
- систематизирует учебный материал;
- ориентирует в учебном процессе.

С этой целью:

- внимательно прочитайте материал предыдущей лекции;
- ознакомьтесь с учебным материалом по учебнику и учебным пособиям с темой прочитанной лекции;
- внесите дополнения к полученным ранее знаниям по теме лекции на полях лекционной тетради;
- запишите возможные вопросы, которые вы зададите лектору на лекции по материалу изученной лекции;
- постарайтесь уяснить место изучаемой темы в своей подготовке;
- узнайте тему предстоящей лекции (по тематическому плану, по информации лектора) и запишите информацию, которой вы владеете по данному вопросу.

Подготовка к занятию семинарского типа.

При подготовке и работе во время проведения занятий семинарского типа следует обратить внимание на следующие моменты: на процесс предварительной подготовки, на работу во время занятия, обработку полученных результатов, исправление полученных замечаний.

Предварительная подготовка к учебному занятию семинарского типа заключается в изучении теоретического материала в отведенное для самостоятельной работы время, ознакомление с инструктивными материалами с целью осознания задач практического занятия, техники безопасности при работе с приборами, веществами.

Работа во время проведения учебного занятия семинарского типа включает:

- консультирование студентов преподавателями и вспомогательным персоналом с целью предоставления исчерпывающей информации, необходимой для самостоятельного выполнения предложенных преподавателем задач.
- самостоятельное выполнение заданий согласно обозначенной рабочей программой дисциплины (модуля) тематики.

Самостоятельная работа.

Самостоятельная работа - планируемая учебная, учебно-исследовательская, научно-исследовательская работа студентов, выполняемая во внеаудиторное (аудиторное) время по заданию и при методическом руководстве преподавателя, но без его непосредственного участия (при частичном непосредственном участии преподавателя, оставляющем ведущую роль за работой студентов).

Для более углубленного изучения темы задания для самостоятельной работы рекомендуется выполнять параллельно с изучением данной темы. При выполнении заданий по возможности используйте наглядное представление материала.

Самостоятельная работа студентов в ВУЗе является важным видом учебной и научной деятельности студента. Самостоятельная работа студентов играет значительную роль в рейтинговой технологии обучения.

К современному специалисту в области медицины общество предъявляет достаточно широкий перечень требований, среди которых немаловажное значение имеет наличие у выпускников определенных навыков (компетенций) и умения самостоятельно добывать знания из различных источников, систематизировать полученную информацию, давать оценку конкретной профессиональной ситуации. Формирование такого умения происходит в течение всего периода обучения.

В процессе самостоятельной работы студент приобретает необходимые для будущей специальности компетенции, навыки самоорганизации, самоконтроля, самоуправления, саморефлексии и становится активным самостоятельным субъектом учебной деятельности.

Самостоятельная работа студентов должна оказывать важное влияние на формирование личности будущего специалиста, его компетентность. Каждый студент самостоятельно определяет режим своей работы и меру труда, затрачиваемого на овладение учебным содержанием по каждой дисциплине. Он выполняет внеаудиторную работу по личному индивидуальному плану, в зависимости от его подготовки, времени и других условий.

Виды самостоятельной работы.

Работа с литературой.

При работе с книгой необходимо подобрать литературу, научиться правильно ее читать, вести записи. Для подбора литературы в библиотеке используются алфавитный и систематический каталоги. Важно помнить, что рациональные навыки работы с книгой - это всегда большая экономия времени и сил. Правильный подбор учебников рекомендуется преподавателем, читающим лекционный курс. Необходимая литература может быть также указана в методических разработках по данному курсу. Изучая материал по учебнику, следует переходить к следующему вопросу только после правильного уяснения предыдущего, описывая на бумаге все выкладки и вычисления (в том числе те, которые в учебнике опущены или на лекции даны для самостоятельного вывода). При изучении любой дисциплины большую и важную роль играет самостоятельная индивидуальная работа. Особое внимание следует обратить на определение основных понятий курса. Студент должен подробно разбирать примеры, которые поясняют такие определения, и уметь строить аналогичные примеры самостоятельно. Нужно добиваться точного представления о том, что изучаешь. Полезно составлять опорные конспекты. При изучении материала по учебнику полезно в тетради (на специально отведенных полях) дополнять конспект лекций. Там же следует отмечать вопросы, выделенные студентом для консультации с преподавателем. Выводы, полученные в результате изучения, рекомендуется в конспекте выделять, чтобы они при перечитывании записей лучше запоминались. Опыт показывает, что многим студентам помогает составление листа опорных сигналов, содержащего важнейшие и наиболее часто употребляемые формулы и понятия. Такой лист помогает запомнить формулы, основные положения лекции, а также может служить постоянным справочником для студента. Различают два вида чтения: первичное и вторичное. Первичное - это внимательное, неторопливое чтение, при котором можно остановиться на трудных местах.

Задача вторичного чтения - полное усвоение смысла целого (по счету это чтение может быть и не вторым, а третьим или четвертым). Самостоятельная работа с учебниками и книгами (а также самостоятельное теоретическое исследование проблем, обозначенных преподавателем на лекциях) – это важнейшее условие формирования у себя научного способа познания.

Чтение научного текста является частью познавательной деятельности. Ее цель – извлечение из текста необходимой информации. От того на сколько осознанна читающим собственная внутренняя установка при обращении к печатному слову (найти нужные сведения, усвоить информацию полностью или частично, критически проанализировать материал и т.п.) во многом зависит эффективность осуществляемого действия.

Методические рекомендации по составлению конспекта:

1. Внимательно прочитайте текст. Уточните в справочной литературе непонятные слова. При записи не забудьте вынести справочные данные на поля конспекта;
2. Выделите главное, составьте план;
3. Кратко сформулируйте основные положения текста, отметьте аргументацию автора;
4. Законспектируйте материал, четко следуя пунктам плана. При конспектировании старайтесь выразить мысль своими словами. Записи следует вести четко, ясно.

5. Грамотно записывайте цитаты. Цитируя, учитывайте лаконичность, значимость мысли. В тексте конспекта желательно приводить не только тезисные положения, но и их доказательства. При оформлении конспекта необходимо стремиться к емкости каждого предложения. Мысли автора книги следует излагать кратко, заботясь о стиле и выразительности написанного. Число дополнительных элементов конспекта должно быть логически обоснованным, записи должны распределяться в определенной последовательности, отвечающей логической структуре произведения. Для уточнения и дополнения необходимо оставлять поля. Овладение навыками конспектирования требует от студента целеустремленности, повседневной самостоятельной работы.

Методические материалы по самостоятельному решению задач

При самостоятельном решении задач нужно обосновывать каждый этап решения, исходя из теоретических положений курса. Если студент видит несколько путей решения проблемы (задачи), то нужно сравнить их и выбрать самый рациональный. Полезно до начала вычислений составить краткий план решения проблемы (задачи). Решение проблемных задач или примеров следует излагать подробно, вычисления располагать в строгом порядке, отделяя вспомогательные вычисления от основных. Решения при необходимости нужно сопровождать комментариями, схемами, чертежами и рисунками. Следует помнить, что решение каждой учебной задачи должно доводиться до окончательного логического ответа, которого требует условие, и по возможности с выводом. Полученный ответ следует проверить способами, вытекающими из существа данной задачи. Полезно также (если возможно) решать несколькими способами и сравнить полученные результаты. Решение задач данного типа нужно продолжать до приобретения твердых навыков в их решении.

Методические материалы к выполнению реферата

Реферат (от лат. *refere* – сообщать) – краткое изложение в письменном виде или в форме публикации доклада, содержания научного труда (трудов), литературы по теме. Работа над рефератом условно разделяется на выбор темы, подбор литературы, подготовку и защиту плана; написание теоретической части и всего текста с указанием библиографических данных используемых источников, подготовку доклада, выступление с ним. Тематика рефератов полностью связана с основными вопросами изучаемого курса.

Список литературы к темам не дается, и обучающиеся самостоятельно ведут библиографический поиск, причем им не рекомендуется ограничиваться университетской библиотекой.

Важно учитывать, что написание реферата требует от обучающихся определенных усилий. Особое внимание следует уделить подбору литературы, методике ее изучения с целью отбора и обработки собранного материала, обоснованию актуальности темы и теоретического уровня обоснованности используемых в качестве примеров фактов какой-либо деятельности.

Выбрав тему реферата, начав работу над литературой, необходимо составить план. Изучая литературу, продолжается обдумывание темы, осмысливание прочитанного, делаются выписки, сопоставляются точки зрения разных авторов и т.д. Реферативная работа сводится к тому, чтобы в ней выделились две взаимосвязанные стороны: во-первых, ее следует рассматривать как учебное задание, которое должен выполнить обучаемый, а во-вторых, как форму научной работы, творческого воображения при выполнении учебного задания.

Наличие плана реферата позволяет контролировать ход работы, избежать формального переписывания текстов из первоисточников.

Оформление реферата включает титульный лист, оглавление и краткий список использованной литературы. Список использованной литературы размещается на последней странице рукописи или печатной форме реферата. Реферат выполняется в письменной или печатной форме на белых листах формата А4 (210 х 297 мм). Шрифт Times New Roman, кегель 14, через 1,5 интервала при соблюдении следующих размеров текста: верхнее поле – 25 мм, нижнее – 20 мм, левое – 30 мм, правое – 15 мм. Нумерация страниц производится вверху листа, по центру. Титульный лист нумерации не подлежит.

Рефераты должны быть написаны простым, ясным языком, без претензий на наукообразность. Следует избегать сложных грамматических оборотов, непривычных терминов и символов. Если же такие термины и символы все-таки приводятся, то необходимо разъяснять их значение при первом упоминании в тексте реферата.

Объем реферата предполагает тщательный отбор информации, необходимой для краткого изложения вопроса. Важнейший этап – редактирование готового текста реферата и подготовка к обсуждению. Обсуждение требует хорошей ориентации в материале темы, умения выделить главное, поставить дискуссионный вопрос, привлечь внимание слушателей к интересной литературе, логично и убедительно изложить свои мысли.

Рефераты обязательно подлежат защите. Процедура защиты начинается с определения оппонентов защищающего свою работу. Они стремятся дать основательный анализ работы обучающимся, обращают внимание на положительные моменты и недостатки реферата, дают общую оценку содержанию, форме преподнесения материала, характеру использованной литературы. Иногда они дополняют тот или иной раздел реферата. Последнее особенно ценно, ибо говорит о глубоком знании обучающимся-оппонентом изучаемой проблемы.

Обсуждение не ограничивается заслушиванием оппонентов. Другие обучающиеся имеют право уточнить или опровергнуть какое-либо утверждение. Преподаватель предлагает любому обучающемуся задать вопрос по существу доклада или попытаться подвести итог обсуждению.

Алгоритм работы над рефератом

1. Выбор темы

Тема должна быть сформулирована грамотно (с литературной точки зрения);

В названии реферата следует поставить четкие рамки рассмотрения темы;

Желательно избегать слишком длинных названий;

Следует по возможности воздерживаться от использования в названии спорных с научной точки зрения терминов, излишней наукообразности, а также чрезмерного упрощения формулировок.

2. Реферат следует составлять из пяти основных частей: введения; основной части; заключения; списка литературы; приложений.

3. Основные требования к введению:

Во введении не следует концентрироваться на содержании; введение должно включать краткое обоснование актуальности темы реферата, где требуется показать, почему данный вопрос может представлять научный интерес и есть ли связь представляемого материала с современностью. Таким образом, тема реферата должна быть актуальна либо с научной точки зрения, либо с современных позиций.

Очень важно выделить цель, а также задачи, которые требуется решить для выполнения цели.

Введение должно содержать краткий обзор изученной литературы, в котором указывается взятый из того или иного источника материал, кратко анализируются изученные источники, показываются их сильные и слабые стороны;

Объем введения составляет две страницы текста.

4. Требования к основной части реферата:

Основная часть содержит материал, отобранный для рассмотрения проблемы;

Также основная часть должна включать в себя собственно мнение обучающихся и сформулированные самостоятельные выводы, опирающиеся на приведенные факты;

Материал, представленный в основной части, должен быть логически изложен и распределен по параграфам, имеющим свои названия;

В изложении основной части необходимо использовать сноски (в первую очередь, когда приводятся цифры и чьи-то цитаты);

Основная часть должна содержать иллюстративный материал (графики, таблицы и т. д.);

Объем основной части составляет около 10 страниц.

5. Требования к заключению:

В заключении формулируются выводы по параграфам, обращается внимание на выдвинутые во введении задачи и цели;

Заключение должно быть четким, кратким, вытекающим из содержания основной части.

6. Требования к оформлению списка литературы (по ГОСТу):

Необходимо соблюдать правильность последовательности записи источников: сначала следует писать фамилию, а после инициалы; название работы не ставится в кавычки; после названия сокращенно пишется место издания; затем идет год издания; наконец, называется процитированная страница.

Критерии оценки реферата

Обучающийся, защищающий реферат, должен рассказать о его актуальности, поставленных целях и задачах, изученной литературе, структуре основной части, сделанных в ходе работы выводах.

По окончанию выступления ему может быть задано несколько вопросов по представленной проблеме.

Оценка складывается из соблюдения требований к реферату, грамотного раскрытия темы, умения четко рассказывать о представленном реферате, способности понять суть задаваемых по работе вопросов и найти точные ответы на них.

Методические материалы к выполнению эссе

Эссе – литературное произведение небольшого объема, обычно прозаическое, свободной композиции, передающее индивидуальные впечатления, суждения, соображения автора о той или иной проблеме, теме, о том или ином событии или явлении. Это вид самостоятельной исследовательской работы обучающихся, с целью углубления и закрепления теоретических знаний и освоения практических навыков. Цель эссе состоит в развитии самостоятельного творческого мышления и письменного изложения собственных мыслей. При написании эссе обучающийся должен представить развернутый письменный ответ на теоретический или практический актуальный вопрос, объявленный преподавателем в аудитории непосредственно перед ее написанием. В процессе написания эссе разрешается пользоваться нормативно-правовыми актами, конспектом лекций (в печатном виде). Использование интернет-ресурсов не допускается. Темы эссе преподаватель предлагает из числа тех, которые обучающиеся уже рассматривали на лекциях или семинарских занятиях, исходя из содержания заданий в составе оценочных средств. По решению преподавателя, в качестве темы эссе может быть выбрана одна или несколько тем, которые могут быть распределены между обучающимися по желанию.

Требования к выполнению эссе:

1. Проводится письменно.

2. Эссе выполняется на компьютере (гарнитура Times New Roman, шрифт 14) через 1,5 интервала с полями: верхнее, нижнее – 2; правое – 3; левое – 1,5. Отступ первой строки абзаца – 1,25. Сноски – постраничные. Таблицы и рисунки встраиваются в текст работы. При этом обязательный заголовок таблицы надо размещать над табличным полем, а рисунки сопровождать подрисуночными подписями. При включении в эссе нескольких таблиц и/или рисунков их нумерация обязательна. Обязательна и нумерация страниц. Их целесообразно проставлять внизу страницы – по середине или в правом углу. Номер страницы не ставится на титульном листе, но в общее число страниц он включается. Объем эссе, без учета

приложений, не должен превышать 5 страниц. Значительное превышение установленного объема является недостатком работы и указывает на то, что обучающийся не сумел отобрать и переработать необходимый материал.

3. Работа должна содержать собственные умозаключения по сути поставленной проблемы, включать самостоятельно проведенный анализ по сути этой проблемы, выводы, обобщающие авторскую позицию по поставленной проблеме.

Критерии оценки эссе:

«Отлично» – исключительные знания материала, абсолютное понимание сути, безукоризненное знание основных понятий и положений, логически и лексически грамотно изложенный, содержательный, аргументированный, конкретный и исчерпывающий ответ.

«Хорошо» – глубокие знания материала, правильное понимание сути, знание основных понятий и положений, содержательный, полный и конкретный ответ.

«Удовлетворительно» – твердые, но недостаточно полные знания, верное понимание сути, в целом правильный ответ.

«Неудовлетворительно» – непонимание сущности задания, грубые ошибки в ответе.

Методические материалы по выполнению тестирования.

Тестовые задания содержат вопросы и 3-4 варианта ответа по базовым положениям изучаемой темы/раздела, составлены с расчетом на знания, полученные обучающимся в процессе изучения темы/раздела.

Тестовые задания выполняются в письменной форме и сдаются преподавателю, ведущему дисциплину (модуль). На выполнение тестовых заданий обучающимся отводится 45 минут.

При обработке результатов оценочной процедуры используются: критерии оценки по содержанию и качеству полученных ответов, ключи, оценочные листы.

Критерии оценки теста:

«Зачтено» - если обучающийся правильно выполнил не менее 2/3 всей работы или допустил не более одной грубой ошибки и двух недочетов, не более одной грубой и одной негрубой ошибки, не более трех негрубых ошибок, одной негрубой ошибки и трех недочетов, при наличии четырех-пяти недочетов.

«Не зачтено» - если число ошибок и недочетов превысило норму для оценки 3 или правильно выполнено менее 2/3 всей работы.

Методические материалы по выполнению доклада.

Рекомендуется следующая структура доклада:

1. титульный лист, содержание доклада;
2. краткое изложение;
3. цели и задачи;
4. изложение характера исследований и рассмотренных проблем, гипотезы, спорные вопросы;
5. источники информации, методы сбора и анализа данных, степень их полноты и достоверности;
6. анализ и толкование полученных в работе результатов;
7. выводы и оценки;
8. библиография и приложения.

Время выступления докладчика не должно превышать 10 минут.

Основные требования к оформлению доклада:

- титульный лист должен включать название доклада, наименование предметной (цикловой) комиссии, фамилию обучающегося;
- все использованные литературные источники сопровождаются библиографическим описанием;
- приводимая цитата из источника берется в кавычки (оформляются сноски);
- единицы измерения должны применяться в соответствии с действующими стандартами;

- все названия литературных источников следует приводить в соответствии с новейшими изданиями;
- рекомендуется включение таблиц, графиков, схем, если они отражают основное содержание или улучшают ее наглядность;
- названия фирм, учреждений, организаций и предприятий должны именоваться так, как они указываются в источнике;

Критерии оценки доклада

При выполнении доклада обучающийся должен продемонстрировать умение кратко излагать прочитанный материал, а также умение обобщать и анализировать материал по теме доклада.

Презентация

Методические материалы к презентациям

1. Объём презентации 10 -20 слайдов.
2. На титульном слайде должно быть отражено:
 - наименование факультета;
 - тема презентации;
 - фамилия, имя, отчество, направление подготовки/ специальность, направленность (профиль)/ специализация, форма обучения, номер группы автора презентации;
 - фамилия, имя, отчество, степень, звание, должность руководитель работы;
 - год выполнения работы.
3. В презентации должны быть отражены обоснование актуальности представляемого материала, цели и задачи работы.
4. Содержание презентации должно включать наиболее значимый материал доклада, а также, при необходимости, таблицы, диаграммы, рисунки, фотографии, карты, видео – вставки, звуковое сопровождение.
5. Заключительный слайд должен содержать информацию об источниках информации для презентации.

Критерии оценки презентации

1. Объём презентации 10 -20 слайдов.
2. Правильность оформления титульного слайда.
3. Актуальность отобранного материала, обоснованность формулировки цели и задач работы.
4. Наглядность и логичность презентации, обоснованность использования таблиц, диаграмм, рисунков, фотографий, карт, видео – вставок, звукового сопровождения; правильный выбор шрифтов, фона, других элементов дизайна слайда.
5. Объём и качество источников информации (не менее 2-х интернет – источников и не менее 2-х литературных источников).

Методические материалы по подготовке к опросу

Самостоятельная работа обучающихся включает подготовку к опросу на практических занятиях. Для этого обучающийся изучает лекции, основную и дополнительную литературу, публикации, информацию из Интернет-ресурсов.

Тема и вопросы к практическим занятиям, вопросы для самоконтроля содержатся в рабочей программе и доводятся до студентов заранее.

Для подготовки к опросу обучающемуся необходимо ознакомиться с материалом, посвященным теме практического занятия, в учебнике или другой рекомендованной литературе, конспекте лекции, обратить внимание на усвоение основных понятий дисциплины, выявить неясные вопросы и подобрать дополнительную литературу для их освещения.

Критерии оценки опроса

«Отлично»:

- дан полный, развернутый ответ на поставленный вопрос;

- в ответе прослеживается четкая структура, логическая последовательность, отражающая сущность раскрываемых понятий, теорий, явлений;
- знание по предмету демонстрируются на фоне понимания его в системе данной науки и междисциплинарных связей;

- свободное владение терминологией;

- ответы на дополнительные вопросы четкие, краткие;

«Хорошо»:

- дан полный, развернутый ответ на поставленный вопрос, показано умение выделять существенные и несущественные признаки, причинно-следственные связи;

- ответ недостаточно логичен с единичными ошибками в частностях, исправленные студентом с помощью преподавателя;

- единичные ошибки в терминологии;

- ответы на дополнительные вопросы правильные, недостаточно полные и четкие.

«Удовлетворительно»:

- ответ не полный, с ошибками в деталях, умение раскрыть значение обобщённых знаний не показано, речевое оформление требует поправок, коррекции;

- логика и последовательность изложения имеют нарушения, студент не способен самостоятельно выделить существенные и несущественные признаки и причинно-следственные связи;

- ошибки в раскрываемых понятиях, терминах;

- студент не ориентируется в теме, допускает серьезные ошибки;

- студент не может ответить на большую часть дополнительных вопросов.

«Неудовлетворительно»:

- ответ представляет собой разрозненные знания с существенными ошибками по вопросу;

- присутствуют фрагментарность, нелогичность изложения, студент не осознает связь обсуждаемого вопроса с другими объектами дисциплины, речь неграмотная;

- незнание терминологии;

- ответы на дополнительные вопросы неправильные.

Методические материалы по выполнению практического задания

При выполнении практического задания обучающийся придерживается следующего алгоритма:

1. Записать дату, тему и цель задания;

2. Ознакомиться с правилами и условия выполнения практического задания;

3. Повторить теоретические задания, необходимые для рациональной работы и других практических действий, используя конспекты лекций и рекомендованную литературу, представленной в программе;

4. Выполнить работу по предложенному алгоритму действий;

5. Обобщить результаты работы, сформулировать выводы / дать ответы на контрольные вопросы;

Работа должна быть выполнена грамотно, с соблюдением культуры изложения. При использовании данных из учебных, методических пособий и другой литературы, периодических изданий, Интернет-источников должны иметься ссылки на вышеперечисленные.

Критерии оценки практического задания:

«Отлично» – правильный ответ, дается четкое обоснование принятому решению; рассуждения четкие последовательные логические; используются ссылки на полученные при изучении дисциплины знания; правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Хорошо» – правильный ответ, дается обоснование принятому решению; но с не существенными ошибками, в рассуждениях отсутствует логическая последовательность; используются ссылки на полученные при изучении дисциплины знания, правильно

используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Удовлетворительно» – правильный ответ, допускаются грубые ошибки в обосновании принятого решения; рассуждения не последовательные сумбурные; используются ссылки на полученные при изучении дисциплины знания; используются формулы, процедуры, понятия, имеющие прямое значение для подтверждения принятого решения, однако, при обращении к ним допускаются серьезные ошибки, студент не может правильно ими воспользоваться.

«Неудовлетворительно, не зачтено» – ответ неверный, отсутствует обоснование принятому решению; студент демонстрирует полное непонимание сути вопроса.

Для оценки решения ситуационной задачи (аналитического задания):

Оценка «отлично» выставляется, если задача решена грамотно, ответы на вопросы сформулированы четко. Эталонный ответ полностью соответствует решению студента, которое хорошо обосновано теоретически.

Оценка «хорошо» выставляется, если задача решена, ответы на вопросы сформулированы не достаточно четко. Решение студента в целом соответствует эталонному ответу, но не достаточно хорошо обосновано теоретически.

Оценка «удовлетворительно» выставляется, если задача решена не полностью, ответы не содержат всех необходимых обоснований решения.

Оценка «неудовлетворительно» выставляется, если задача не решена или имеет грубые теоретические ошибки в ответе на поставленные вопросы.

Методические материалы по выполнению лабораторного задания

При выполнении лабораторного задания обучающийся придерживается следующего алгоритма:

1. Записать дату, тему и цель задания;
2. Ознакомиться с правилами и условия выполнения задания;
3. Повторить теоретические задания, необходимые для рациональной работы и других практических действий, используя конспекты лекций и рекомендованную литературу, представленной в программе;
4. Выполнить работу по предложенному алгоритму действий;
5. Обобщить результаты работы, сформулировать выводы / дать ответы на контрольные вопросы;

Работа должна быть выполнена грамотно, с соблюдением культуры изложения. При использовании данных из учебных, методических пособий и другой литературы, периодических изданий, Интернет-источников должны иметься ссылки на вышеперечисленные.

Критерии оценки лабораторного задания:

«Отлично» – правильный ответ, дается четкое обоснование принятому решению; рассуждения четкие последовательные логические; используются ссылки на полученные при изучении дисциплины знания; правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Хорошо» – правильный ответ, дается обоснование принятому решению; но с не существенными ошибками, в рассуждениях отсутствует логическая последовательность; используются ссылки на полученные при изучении дисциплины знания, правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Удовлетворительно» – правильный ответ, допускаются грубые ошибки в обосновании принятого решения; рассуждения не последовательные сумбурные; используются ссылки на полученные при изучении дисциплины знания; используются формулы, процедуры, понятия, имеющие прямое значение для подтверждения принятого решения, однако, при обращении к ним допускаются серьезные ошибки, студент не может правильно ими воспользоваться.

«Неудовлетворительно, не зачтено» – ответ неверный, отсутствует обоснование принятому решению; студент демонстрирует полное непонимание сути вопроса.

Методические указания для подготовки к промежуточной аттестации.

Изучение учебных дисциплин (модулей) завершается зачетом/зачетом с оценкой или экзаменом. Подготовка к промежуточной аттестации способствует закреплению, углублению

и обобщению знаний, получаемых, в процессе обучения, а также применению их к решению практических задач. Готовясь к экзамену, студент ликвидирует имеющиеся пробелы в знаниях, углубляет, систематизирует и упорядочивает свои знания. На зачете или экзамене студент демонстрирует то, что он освоил в процессе обучения по дисциплине (модулю).

Вначале следует просмотреть весь материал по дисциплине (модулю), отметить для себя трудные вопросы. Обязательно в них разобраться. В заключение еще раз целесообразно повторить основные положения, используя при этом листы опорных сигналов. Систематическая подготовка к занятиям в течение семестра позволит использовать время промежуточной аттестации для систематизации знаний.

3. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНИВАНИЯ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ И (ИЛИ) ОПЫТА ДЕЯТЕЛЬНОСТИ, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ

3.1. Организационные основы применения балльно-рейтинговой системы оценки успеваемости обучающихся по дисциплине (модулю)

Оценка качества освоения обучающимися дисциплины (модуля) реализуется в формате балльно-рейтинговой системы оценки успеваемости обучающихся (БРСО).

БРСО в ходе текущего контроля успеваемости и промежуточной аттестации осуществляется по 100-балльной шкале.

Академический рейтинг обучающегося по дисциплине (модулю) складывается из результатов:

- текущего контроля успеваемости (максимальный текущий рейтинг обучающегося 80 рейтинговых баллов);
- промежуточной аттестации (максимальный рубежный рейтинг обучающегося 20 рейтинговых баллов).

Условия оценки освоения обучающимся дисциплины (модуля) в формате БРСО доводятся преподавателем до сведения обучающихся на первом учебном занятии, а также размещены в свободном доступе в электронной информационно-образовательной среде Университета.

3.2. Проведение текущего контроля успеваемости обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося

В течение учебного семестра до промежуточной аттестации на основании утвержденной рабочей программы дисциплины (модуля) формируется текущий рейтинг обучающегося. Текущий рейтинг обучающегося складывается как сумма рейтинговых баллов, полученных им в течение учебного семестра по всем видам учебных занятий по учебной дисциплине.

В процессе текущего контроля оцениваются следующие действия обучающегося, направленные на освоение компетенций в рамках изучения учебной дисциплины:

- академическая активность (посещаемость учебных занятий, самостоятельное изучение содержания учебной дисциплины в электронной информационно-образовательной среде, соблюдение сроков сдачи практических заданий и текущих контрольных мероприятий и др.);
- выполнение и сдача текущих и итогового практических заданий (эссе, рефераты, творческие задания, кейс-задания, лабораторные работы, расчетные задания и др., активное участие в групповых интерактивных занятиях (дискуссии, Wiki-проекты и др.), защита проектов и др.);
- прохождение рубежей текущего контроля, включая соблюдение графика их прохождения в электронной информационно-образовательной среде.

Для планирования расчета текущего рейтинга обучающегося используются следующие пропорции:

Вид учебного действия	Максимальная рейтинговая оценка, баллов
академическая активность	10
практические задания	40
<i>из них: текущие практические задания</i>	20
<i>итоговое практическое задание</i>	20
рубежи текущего контроля	30
ИТОГО:	80

В течение учебного семестра по дисциплине (модулю) обучающимся должен быть накоплен текущий рейтинг не менее 52 рейтинговых баллов (65% от максимального значения текущего рейтинга).

Необходимыми условиями допуска обучающегося к промежуточной аттестации по дисциплине являются положительное прохождение обучающимся не менее 65% рубежей текущего контроля с накоплением не менее 65% максимального рейтингового балла за каждый рубеж текущего контроля и положительное выполнение итогового практического задания с накоплением не менее 65% максимального рейтингового балла, установленного за итоговое практическое задание.

Невыполнение вышеуказанных условий является текущей академической задолженностью, которая должна быть ликвидирована обучающимся до контрольного мероприятия промежуточной аттестации.

Сведения о наличии у обучающихся текущей академической задолженности, сроках и порядке добора рейтинговых баллов для её ликвидации доводятся до обучающихся педагогическим работником.

В случае неликвидации текущей академической задолженности, педагогический работник обязан во время контрольного мероприятия промежуточной аттестации поставить обучающемуся 0 рейтинговых баллов. В этом случае ликвидация текущей академической задолженности возможна в периоды проведения повторной промежуточной аттестации.

3.3. Проведение промежуточной аттестации обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося

Промежуточная аттестация по дисциплине (модулю) проводится в соответствии с Положением о промежуточной аттестации обучающихся по основным профессиональным образовательным программам высшего образования – программ специалитета в Российском государственном социальном университете и Положением о балльно-рейтинговой системе оценки успеваемости обучающихся по основным профессиональным образовательным программам высшего образования - программам бакалавриата, программам специалитета, программам магистратуры в Российском государственном социальном университете.

На промежуточную аттестацию отводится 20 рейтинговых баллов.

Ответы обучающегося на контрольном мероприятии промежуточной аттестации оцениваются педагогическим работником по 20 - балльной шкале, а итоговая оценка по дисциплине (модулю) выставляется по пятибалльной системе.

Критерии выставления оценки определяются Положением о балльно-рейтинговой системе оценки успеваемости обучающихся по основным профессиональным образовательным программам высшего образования - программам бакалавриата, программам специалитета, программам магистратуры в Российском государственном социальном университете.

В процессе определения рубежного рейтинга обучающегося используется следующая шкала:

Рубежный рейтинг	Критерии оценки освоения обучающимся учебной дисциплины в ходе контрольных мероприятий промежуточной аттестации
19-20 рейтинговых баллов	обучающийся глубоко и прочно усвоил программный материал, исчерпывающе, последовательно, грамотно и логически стройно его излагает, тесно увязывает с задачами и будущей деятельностью, не затрудняется с ответом при видоизменении задания, свободно справляется с задачами и практическими заданиями, правильно обосновывает принятые решения, умеет самостоятельно обобщать и излагать материал, не допуская
16-18 рейтинговых баллов	обучающийся твердо знает программный материал, грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос, может правильно применять теоретические положения и владеет необходимыми умениями и навыками при выполнении практических
13-15 рейтинговых баллов	обучающийся освоил основной материал, но не знает отдельных деталей, допускает неточности, недостаточно правильные формулировки, нарушает последовательность в изложении программного материала и испытывает затруднения в выполнении практических заданий
1-12 рейтинговых баллов	обучающийся не знает значительной части программного материала, допускает существенные ошибки, с большими затруднениями выполняет практические задания
0 рейтинговых баллов	не аттестован

Если результат контроля успеваемости в рамках проведения контрольных мероприятий промежуточной аттестации (рубежный рейтинг обучающегося) неудовлетворительный (получено менее 13 рейтинговых баллов), то промежуточная аттестация по учебной дисциплине (модулю) невозможна даже при наличии высокого текущего рейтинга, полученного по итогам текущего контроля по учебной дисциплине (модулю).

ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

№ п/п	Содержание изменения	Реквизиты документа об утверждении изменения	Дата введения изменения
1.			
2.			
3.			
4.			



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Российский государственный социальный университет»

УТВЕРЖДАЮ

декан факультета политических и социальных

технологий  /Пивнева С.В./

«28» февраля 2024 года

МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ
ИНЖЕНЕРНАЯ ГЕОМЕТРИЯ И КОМПЬЮТЕРНАЯ ГРАФИКА

Направление подготовки
«Информационные системы и технологии»

Направленность
«Интеллектуальные информационные системы и технологии»

ОБРАЗОВАТЕЛЬНАЯ ПРОГРАММА ВЫСШЕГО ОБРАЗОВАНИЯ - ПРОГРАММА
БАКАЛАВРИАТА

Форма обучения

Очная, заочная

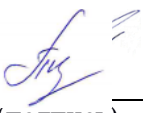
Москва, 2024 г.

Методические материалы по дисциплине (модулю) «Инженерная геометрия и компьютерная графика» разработаны на основании федерального государственного образовательного стандарта высшего образования – *бакалавриата* по направлению подготовки *09.03.02 Информационные системы и технологии*, утвержденного приказом Министерства образования и науки Российской Федерации от 19.09.2017 № 926, учебного плана по основной профессиональной образовательной программе высшего образования – программе бакалавриата по направлению подготовки *09.03.02 Информационные системы и технологии* (далее – «ОПОП»).

Методические материалы по дисциплине (модулю) разработаны рабочей группой в составе: канд. техн. наук А.О. Блинов, канд. тех. наук, доцент С.М. Бобровский.

Методические материалы по дисциплине (модулю) обсуждены и утверждены на заседании кафедры информационных технологий, искусственного интеллекта и общественно-социальных технологий цифрового общества факультета социальных и политических технологий (Протокол утверждения РПД и ММ № 11 от 28 февраля 2024 г.)

Заведующий кафедрой
канд. пед. наук, доцент



(подпись) С.В. Пивнева

СОДЕРЖАНИЕ

1. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ К ЛЕКЦИОННЫМ, ЛАБОРАТОРНЫМ ЗАНЯТИЯМ.....	4
1.1. Методические материалы к проведению лекционных занятий по дисциплине (модулю)	4
1.2. Методические материалы по подготовке к практическим занятиям по дисциплине (модулю)	5
1.4. Методические материалы по подготовке к лабораторным занятиям по дисциплине (модулю)	13
2. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ) И САМОСТОЯТЕЛЬНОЙ РАБОТЕ	13
3. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНИВАНИЯ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ И (ИЛИ) ОПЫТА ДЕЯТЕЛЬНОСТИ, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ.....	23
3.1. Организационные основы применения балльно-рейтинговой системы оценки успеваемости обучающихся по дисциплине (модулю)	23
3.2. Проведение текущего контроля успеваемости обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося.....	23
3.3. Проведение промежуточной аттестации обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося	24
ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ	26

1. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ К ЛЕКЦИОННЫМ, ЛАБОРАТОРНЫМ ЗАНЯТИЯМ

1.1. Методические материалы к проведению лекционных занятий по дисциплине (модулю)

Лекция - один из методов обучения, одна из основных системообразующих форм организации учебного процесса в вузе. Лекционное занятие представляет собой систематическое, последовательное, монологическое изложение педагогическим работником учебного материала, как правило, теоретического характера. Такое занятие представляет собой элемент технологии представления учебного материала путем логически стройного, систематически последовательного и ясного изложения.

Цель лекции – организация целенаправленной познавательной деятельности обучающихся по овладению программным материалом дисциплины (модуля). Чтение курса лекций позволяет дать связанное, последовательное изложение материала в соответствии с новейшими данными науки, сообщить слушателям основное содержание предмета в целостном, систематизированном виде. В ряде случаев лекция выполняет функцию основного источника информации: при отсутствии учебников и учебных пособий, чаще по новым курсам; в случае, когда новые научные данные по той или иной теме не нашли отражения в учебниках; отдельные разделы и темы очень сложны для самостоятельного изучения. В таких случаях только лектор может методически помочь обучающимся в освоении сложного материала.

Возможные формы проведения лекций:

- Вводная лекция – один из наиболее важных и трудных видов лекции при чтении систематических курсов. От успеха этой лекции во многом зависит успех усвоения всего курса. Она может содержать: определение дисциплины (модуля); краткую историческую справку о дисциплине (модуле); цели и задачи дисциплины (модуля), ее роль в общей системе обучения и связь со смежными дисциплинами (модулями); основные проблемы (понятия и определения) данной науки; основную и дополнительную учебную литературу; особенности самостоятельной работы обучающихся над дисциплиной (модулем) и формы участия в научно-исследовательской работе; отчетность по курсу.

- Информационная лекция ориентирована на изложение и объяснение обучающимся научной информации, подлежащей осмыслению и запоминанию. Это самый традиционный тип лекций в практике высшей школы.

- Заключительная лекция предназначена для обобщения полученных знаний и раскрытия перспектив дальнейшего развития данной науки.

- Обзорная лекция – это систематизация научных знаний на высоком уровне, допускающая большое число ассоциативных связей в процессе осмысления информации, излагаемой при раскрытии внутрипредметной и межпредметной связей, исключая детализацию и конкретизацию. Как правило, стержень излагаемых теоретических положений составляет научно-понятийная и концептуальная основа всего курса или крупных его разделов.

- Лекция-беседа - непосредственный контакт педагогического работника с аудиторией - диалог. По ходу лекции педагогический работник задает вопросы для выяснения мнений и уровня осведомленности обучающихся по рассматриваемой проблеме.

- Лекция-дискуссия - свободный обмен мнениями в ходе изложения лекционного материала. Педагогический работник активизирует участие в обсуждении отдельными вопросами, сопоставляет между собой различные мнения и тем самым развивает дискуссию, стремясь направить ее в нужное русло.

- Лекция с применением обратной связи включает в себе то, что в начале и конце каждого раздела лекции задаются вопросы. Первый - для того, чтобы узнать, насколько обучающиеся ориентируются в излагаемом материале, вопрос в конце раздела предназначен для выяснения степени усвоения только что изложенного материала. При

неудовлетворительных результатах контрольного опроса педагогический работник возвращается к уже прочитанному разделу, изменив при этом методику подачи материала.

- Проблемная лекция опирается на логику последовательно моделируемых проблемных ситуаций путем постановки проблемных вопросов или предъявления проблемных задач. Проблемный вопрос - это диалектическое противоречие, требующее для своего решения размышления, сравнения, поиска, приобретения и применения новых знаний. Проблемная задача содержит дополнительную вводную информацию и при необходимости некоторые ориентиры поиска ее решения.

- Программированная лекция - консультация – педагогический работник сам составляет и предлагает обучающимся вопросы. На подготовленные вопросы педагогический работник сначала просит ответить обучающихся, а затем проводит анализ и обсуждение неправильных ответов. В лекциях можно использовать наглядные материалы, а также подготовить презентацию. Что касается презентации, то в качестве визуальной поддержки ее можно органично интегрировать во все вышеупомянутые лекции. В то же время лекцию-презентацию возможно выделить и в качестве самостоятельной формы. Лекция-презентация должна отражать суть основных и (или) проблемных вопросов лекции, на которые особо следует обратить внимание обучающихся. В условиях применения активного метода проведения занятий презентация представляется весьма удачным способом донесения информации до слушателей. Единственное, на что следует обратить внимание при подготовке слайдов, - это их оформление и текст. Слайд не должен быть перегружен картинками и лишней информацией, которая будет отвлекать от основного аспекта того или иного вопроса лекции. Во время лекции можно задавать вопросы аудитории в отношении того или иного слайда, тем самым еще больше вовлекая обучающихся в проблематику.

Краткое содержание лекционных занятий

Наименование разделов и тем	Содержание учебного материала
Раздел 1. Введение в компьютерную графику	Области применения компьютерной графики. Классификация, обзор и тенденции построения современных графических систем. Основные принципы и функциональные возможности современных графических систем. Стандарты в области разработки графических систем
Раздел 2. Средства и технологии 2D и 3D моделирования. Компьютерная геометрия и моделирование	Технические средства компьютерной графики. 2D и 3D моделирование, способы и форматы создания, хранения, ввода и вывода графической информации. Системы координат, типы преобразований графической информации. Виды геометрических моделей их свойства, параметризация моделей; геометрические операции над моделями. Алгоритмы визуализации: отсечения, развертки, удаления невидимых линий и поверхностей, закраски. Роль и место геометрических моделей в процессе автоматизированного проектирования; - классификацию, основные свойства, способы создания и описания геометрических моделей; - сущность и методы твердотельного моделирования; - методы поверхностного моделирования; - основные компоненты, классы и стандарты графических систем; - системы подготовки и выпуска конструкторско-технологической документации.
Раздел 3. Элементы инженерной графики	Конструкторская документация, система стандартов ЕСКД. Оформление чертежей. Резьбовые изделия. Разъёмные и неразъёмные соединения. Составление чертежа детали. Детализование. Способы преобразования комплексного чертежа. Чертёж сборочной единицы. Чертежи зданий.

1.2. Методические материалы по подготовке к практическим занятиям по дисциплине (модулю)

Практические (семинарские) занятия - одна из форм учебного занятия, направленная на развитие самостоятельности обучающихся и приобретение умений и навыков. Данные учебные занятия углубляют, расширяют, детализируют полученные ранее знания.

Практическое занятие предполагает выполнение обучающимися по заданию и под руководством преподавателей одной или нескольких практических работ.

Цель практических занятий и семинаров состоит в развитии познавательных способностей, самостоятельности мышления и творческой активности обучающихся; углублении, расширении, детализировании знаний, полученных на лекции в обобщенной форме, и содействии выработке навыков профессиональной деятельности. В отдельных случаях на практических занятиях и семинарах руководителем занятия сообщаются дополнительные знания.

Для достижения поставленных целей и решения требуемого перечня задач практические занятия и семинары проводятся традиционными технологиями или с использованием активных и интерактивных образовательных технологий.

Возможные формы проведения практических (семинарских) занятий:

- Деловая игра - это метод группового обучения совместной деятельности в процессе решения общих задач в условиях максимально возможного приближения к реальным проблемным ситуациям. Имитационные игры - на занятиях имитируется деятельность какой-либо организации, предприятия или его подразделения. Имитироваться могут события, конкретная деятельность людей (деловое совещание, обсуждение плана) и обстановка, условия, в которых происходит событие или осуществляется деятельность (кабинет начальника цеха, зал заседаний). Исполнение ролей (ролевые игры) - в этих играх отрабатывается тактика поведения, действий, выполнение функций и обязанностей конкретного лица. Для проведения игр с исполнением роли разрабатывается модель-пьеса ситуации, между студентами распределяются роли с «обязательным содержанием», характеризующиеся различными интересами; в процессе их взаимодействия должно быть найдено компромиссное решение. «Деловой театр» (метод инсценировки) - в нем разыгрывается какая-либо ситуация, поведение человека в этой обстановке, обучающийся должен вжиться в образ определенного лица, понять его действия, оценить обстановку и найти правильную линию поведения. Основная задача метода инсценировки - научить ориентироваться в различных обстоятельствах, давать объективную оценку своему поведению, учитывать возможности других людей, влиять на их интересы, потребности и деятельность, не прибегая к формальным атрибутам власти, к приказу.

- Игровое проектирование - является практическим занятием или циклом занятий, суть которых состоит в разработке инженерного, конструкторского, технологического и других видов проектов в игровых условиях, максимально воссоздающих реальность. Этот метод отличается высокой степенью сочетания индивидуальной и совместной работы обучающихся.

- Познавательно-дидактические игры не относятся к деловым играм. Они предполагают лишь включение изучаемого материала в необычный игровой контекст и иногда содержат лишь элементы ролевых игр. Такие игры могут проводиться в виде копирования научных, культурных, социальных явлений (конкурс знатоков, «Поле чудес», КВН и т.д.) и в виде предметно-содержательных моделей, (например, игры-путешествия, когда надо разработать рациональный маршрут, пользуясь различными картами).

- Анализ конкретных ситуаций. Конкретная ситуация – это любое событие, которое содержит в себе противоречие или вступает в противоречие с окружающей средой. Ситуации могут нести в себе как позитивный, так и отрицательный опыт. Все ситуации делятся на простые, критические и экстремальные.

- Кейс-метод (от английского case – случай, ситуация) – усовершенствованный метод анализа конкретных ситуаций, метод активного проблемно-ситуационного анализа, основанный на обучении путем решения конкретных задач – ситуаций (решение кейсов). Непосредственная цель метода case-study - обучающиеся должны проанализировать ситуацию, разобраться в сути проблем, предложить возможные решения и выбрать лучшее из них. Кейсы делятся на практические (отражающие реальные жизненные ситуации), обучающие (искусственно созданные, содержащие значительные элементы условности при отражении в

нем жизни) и исследовательские (ориентированные на проведение исследовательской деятельности посредством применения метода моделирования). Метод конкретных ситуаций (метод case-study) относится к неигровым имитационным активным методам обучения.

- Тренинг (англ. training от train — обучать, воспитывать) – метод активного обучения, направленный на развитие знаний, умений и навыков и социальных установок. Тренинг – форма интерактивного обучения, целью которого является развитие компетентности межличностного и профессионального поведения в общении. Достоинство тренинга заключается в том, что он обеспечивает активное вовлечение всех участников в процесс обучения. Можно выделить основные типы тренингов по критерию направленности воздействия и изменений – навыковый, психотерапевтический, социально-психологический, бизнес-тренинг.

- Метод Сократа (Майевтика) – метод вопросов, предполагающих критическое отношение к догматическим утверждениям, называется еще как метод «сократовской иронии». Это умение извлекать скрытое в человеке знание с помощью искусных наводящих вопросов, подразумевающего короткий, простой и заранее предсказуемый ответ.

- Интерактивная лекция – выступление ведущего обучающего перед большой аудиторией с применением следующих активных форм обучения: дискуссия, беседа, демонстрация слайдов или учебных фильмов, мозговой штурм.

- Групповая, научная дискуссия, диспут. Дискуссия — это целенаправленное обсуждение конкретного вопроса, сопровождающееся обменом мнениями, идеями между двумя и более лицами. Задача дискуссии - обнаружить различия в понимании вопроса и в споре установить истину. Дискуссии могут быть свободными и управляемыми. К технике управляемой дискуссии относятся: четкое определение цели, прогнозирование реакции оппонентов, планирование своего поведения, ограничение времени на выступления и их заданная очередность. Разновидностью свободной дискуссии является форум, где каждому желающему дается неограниченное время на выступление, при условии, что его выступление вызывает интерес аудитории. Каждый конкретный форум имеет свою тематику — достаточно широкую, чтобы в её пределах можно было вести многоплановое обсуждение.

- Дебаты – это чётко структурированный и специально организованный публичный обмен мыслями между двумя сторонами по актуальным темам. Это разновидность публичной дискуссии участников дебатов, направляющая на переубеждение в своей правоте третьей стороны, а не друг друга. Поэтому вербальные и невербальные средства, которые используются участниками дебатов, имеют целью получения определённого результата — сформировать у слушателей положительное впечатление от собственной позиции.

- Метод работы в малых группах. Групповое обсуждение кого-либо вопроса направлено на достижение лучшего взаимопонимания и нахождения истины. Групповое обсуждение способствует лучшему усвоению изучаемого материала. Оптимальное количество участников - 5-7 человек. Перед обучающимися ставится проблема, выделяется определенное время, в течение которого они должны подготовить аргументированный обдуманный ответ. Педагогический работник может устанавливать правила проведения группового обсуждения – задавать определенные рамки обсуждения, ввести алгоритм выработки общего мнения, назначить лидера и др.

- Круглый стол - общество, собрание в рамках более крупного мероприятия (съезда, симпозиума, конференции). Мероприятие, как правило, на которое приглашаются эксперты и специалисты из разных сфер деятельности для обсуждения актуальных вопросов. Данная модель обсуждения, основываясь на соглашениях, в качестве итогов даёт результаты, которые, в свою очередь, являются новыми соглашениями.

- Коллоквиум - (лат. colloquium — разговор, беседа) - одна из форм учебных занятий в системе образования, имеющая целью выяснение и повышение знаний обучающихся. На коллоквиумах обсуждаются: отдельные части, разделы, темы, вопросы изучаемого курса (обычно не включаемые в тематику семинарских и других практических учебных занятий), рефераты, проекты и др. работы обучающихся. Это научные собрания, на которых

заслушиваются и обсуждаются доклады. Коллоквиум – это и форма контроля, массового опроса, позволяющая преподавателю в сравнительно небольшой срок выяснить уровень знаний студентов по данной теме дисциплины. Коллоквиум проходит обычно в форме дискуссии, в ходе которой обучающимся предоставляется возможность высказать свою точку зрения на рассматриваемую проблему, учиться обосновывать и защищать ее. Аргументируя и отстаивая свое мнение, обучающийся в то же время демонстрирует, насколько глубоко и осознанно он усвоил изученный материал.

- Метод «мозговой штурм» (мозговой штурм, мозговая атака, англ. brainstorming) — оперативный метод решения проблемы на основе стимулирования творческой активности, при котором участникам обсуждения предлагают высказывать как можно большее количество вариантов решения, в том числе самых фантастичных. Затем из общего числа высказанных идей отбирают наиболее удачные, которые могут быть использованы на практике. Является методом экспертного оценивания.

- Метод проектов - это способ достижения дидактической цели через детальную разработку проблемы (технологию), которая должна завершиться вполне реальным, осязаемым практическим результатом, оформленным тем или иным образом; это совокупность приёмов, действий обучающихся в их определённой последовательности для достижения поставленной задачи – решения проблемы, лично значимой для учащихся и оформленной в виде некоего конечного продукта. Основное предназначение метода проектов состоит в предоставлении учащимся возможности самостоятельного приобретения знаний в процессе решения практических задач или проблем, требующего интеграции знаний из различных предметных областей.

- Брифинг - (англ. briefing от англ. brief – короткий, недолгий) – краткая пресс-конференция, посвященная одному вопросу. Основное отличие: отсутствует презентационная часть. То есть практически сразу идут ответы на вопросы журналистов.

- Метод портфолио (итал. portfolio — 'портфель, англ. - папка для документов) - современная образовательная технология, в основе которой используется метод аутентичного оценивания результатов образовательной и профессиональной деятельности. Портфолио как подборка сертифицированных достижений, наиболее значимых работ и отзывов на них.

Вопросы для самоподготовки к практическим (семинарским) занятиям по разделам (темам) дисциплины (модуля)

Задания для самостоятельной работы к Разделу 1

Вопросы для самостоятельной работы к Разделу 1

1. Основные задачи компьютерной графики.
2. Виды графических систем. Основные достоинства и недостатки.
3. Графические системы с векторным сканированием.
4. Растровые графические системы. Основные характеристики растра.
5. Растровые графические системы. Построчная и чересстрочная развертки растра.
6. Форматы графических файлов.
7. Векторные форматы графических файлов. Основные достоинства и недостатки.
8. Растровые форматы графических файлов. Основные достоинства и недостатки.
9. Методы сжатия растровых файлов.
10. Векторные и растровые прикладные графические редакторы. Области применения.
11. Аддитивная цветовая модель RGB.
12. Субтрактивная цветовая модель CMY, CMYK.
13. Цветовая модель HSB.
14. Базовые растровые алгоритмы. Основные решаемые задачи. Понятие связности.
15. Растровое представление отрезка. Алгоритм Брезенхэма.

16. Устранение ступенчатого эффекта в растровых изображениях.
17. Заполнение области (закрашивание).
18. Закрашивание многоугольников, заданных своими вершинами.
19. Отсечение многоугольников относительно видимого окна.

Литература для самостоятельного изучения к Разделу 1

Основная литература

1. Боресков, А. В. Основы компьютерной графики : учебник и практикум для вузов / А. В. Боресков, Е. В. Шикин. — Москва : Издательство Юрайт, 2023. — 219 с. — (Высшее образование). — ISBN 978-5-534-13196-3. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511419> (дата обращения: 08.03.2023)..
2. Инженерная 3D-компьютерная графика в 2 т. Том 1 : учебник и практикум для вузов / А. Л. Хейфец, А. Н. Логиновский, И. В. Буторина, В. Н. Васильева ; под редакцией А. Л. Хейфеца. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 328 с. — (Высшее образование). — ISBN 978-5-534-02957-4. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/513027> (дата обращения: 08.03.2023).
3. Инженерная 3D-компьютерная графика в 2 т. Том 2 : учебник и практикум для вузов / А. Л. Хейфец, А. Н. Логиновский, И. В. Буторина, В. Н. Васильева ; под редакцией А. Л. Хейфеца. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 279 с. — (Высшее образование). — ISBN 978-5-534-02959-8. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/513028> (дата обращения: 08.03.2023).5.1.2.

Дополнительная литература

1. Кордонская, И. Б. Инженерная и компьютерная графика : учебник / И. Б. Кордонская, Е. А. Богданова. — Самара : ПГУТИ, 2020. — 264 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/255455> (дата обращения: 08.03.2023). — Режим доступа: для авториз. пользователей.
2. Инженерная и компьютерная графика : учебник и практикум для вузов / Р. Р. Анамова [и др.] ; под общей редакцией Р. Р. Анамовой, С. А. Леоновой, Н. В. Пшеничной. — Москва : Издательство Юрайт, 2023. — 246 с. — (Высшее образование). — ISBN 978-5-9916-8262-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/512176> (дата обращения: 08.03.2023).

Задания для самостоятельной работы к Разделу 2

Вопросы для самостоятельной работы к Разделу 2

1. Аффинные преобразования на плоскости. Основные частные случаи. Применение однородных координат для матричной формы записи уравнений аффинных преобразований.
2. Аффинные преобразования в пространстве. Основные частные случаи. Композиция преобразований.
3. Проецирование. Виды плоских геометрических проекций.
4. Виды параллельных проекций. Искажения объекта при параллельном проецировании.
5. Ортографическая проекция.
6. Аксонометрические проекции.
7. Косоугольные проекции.

8. Перспективные (центральные) проекции.
9. Системы координат в компьютерной графике. Переход от мировых к экранным координатам.
10. Основные геометрические модели трехмерных объектов.
11. Каркасная и граневая геометрические модели трехмерных объектов. Достоинства и недостатки, область применения.
12. Граневая геометрическая модель трехмерных объектов. Полигональная сетка, параметрические бикубические куски.
13. Объемно-параметрическая геометрическая модель трехмерных объектов.
14. Кинематическая геометрическая модель трехмерных объектов.
15. Способы визуализации трехмерных изображений.
16. Способы задания полигональной сетки. Основные достоинства и недостатки.
17. Основные способы математического описания кривых и поверхностей. Достоинства параметрического способа описания.
18. Форма Эрмита для задания параметрической кубической кривой. Основные достоинства и недостатки. Условия непрерывности.
19. Форма Безье для задания параметрической кубической кривой. Область применения. Условия непрерывности.
20. Форма В-сплайнов для задания параметрической кубической кривой. Область применения.
21. Форма Эрмита для задания параметрической бикубической поверхности. Условия непрерывности.
22. Форма Безье для задания параметрической бикубической поверхности. Условия непрерывности.
23. Форма В-сплайнов для задания параметрической бикубической поверхности. Область применения.
24. Аффинные преобразования параметрических кривых и поверхностей.
25. Удаление скрытых ребер и поверхностей. Сортировка граней по глубине.
26. Удаление скрытых ребер и поверхностей. Метод плавающего горизонта.
27. Удаление скрытых ребер и поверхностей. Метод z-буфера.
28. Удаление скрытых ребер и поверхностей. Методы оптимизации. Метод порталов.
29. Модели отражения и преломления цвета. Определение цвета закрашивания.
30. Методы закрашивания поверхностей: плоское закрашивание, закрашивание методами Гуро, Фонга. Тени.
31. Методы закрашивания поверхностей: трассировка лучей, метод анализа излучательности.
32. Детализация поверхностей цветом и фактурой.
33. Текстуры.
34. Моделирование микрорельефа поверхности.
35. Применение эффекта «затуманивания» для передачи глубины пространства.
36. Построение трехмерных сцен. Граф сцены. Форматы файлов трехмерной графики.
37. Понятие интерфейса прикладного программирования (API).

Литература для самостоятельного изучения к Разделу 2

Основная литература

1. Боресков, А. В. Основы компьютерной графики : учебник и практикум для вузов / А. В. Боресков, Е. В. Шикин. — Москва : Издательство Юрайт, 2023. — 219 с. — (Высшее образование). — ISBN 978-5-534-13196-3. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511419> (дата обращения: 08.03.2023)..

2. Инженерная 3D-компьютерная графика в 2 т. Том 1 : учебник и практикум для вузов / А. Л. Хейфец, А. Н. Логиновский, И. В. Буторина, В. Н. Васильева ; под редакцией А. Л. Хейфеца. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 328 с. — (Высшее образование). — ISBN 978-5-534-02957-4. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/513027> (дата обращения: 08.03.2023).

Дополнительная литература

1. Кордонская, И. Б. Инженерная и компьютерная графика : учебник / И. Б. Кордонская, Е. А. Богданова. — Самара : ПГУТИ, 2020. — 264 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/255455> (дата обращения: 08.03.2023). — Режим доступа: для авториз. пользователей.
2. Инженерная и компьютерная графика : учебник и практикум для вузов / Р. Р. Анамова [и др.] ; под общей редакцией Р. Р. Анамовой, С. А. Леоновой, Н. В. Пшеничновой. — Москва : Издательство Юрайт, 2023. — 246 с. — (Высшее образование). — ISBN 978-5-9916-8262-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/512176> (дата обращения: 08.03.2023).

Здания для самостоятельной работы к Разделу 3

Вопросы для самостоятельной работы к Разделу 3

1. Форматы основные и дополнительные. Масштабы чертежа. Линии чертежа.
2. Шрифты чертёжные (классификация, основные параметры, правила выполнения надписей)
3. Правила нанесения размеров на чертеже.
4. Прямоугольный метод проецирования. Проецирование на три плоскости проекций.
5. Основные виды. Виды местные и дополнительные (понятие, назначение, обозначение на чертеже).
6. Разрезы простые (понятие, назначение, классификация, обозначение на чертежах, правила выполнения).
7. Разрезы сложные (понятие, назначение, классификация, обозначение на чертеже, правила выполнения).
8. Сечения (понятие, назначение, классификация, обозначение на чертеже, правила выполнения).
9. Правила соединения части вида и части разреза (для симметричных и несимметричных деталей).
10. Резьба. Типы резьбы, обозначение на чертеже. Правила изображения детали с наружной резьбой.
11. Резьба. Типы резьбы, обозначение на чертеже. Правила изображения детали с внутренней резьбой.
12. Рабочий чертёж детали (понятие, назначение, правила выполнения).
13. Шероховатость поверхности. Допуски и посадки. Обозначение на чертеже.
14. Сборочный чертёж (понятие, назначение, содержание, правила выполнения).
15. Спецификация (назначение, содержание, правила выполнения).
16. Сборочный чертёж, упрощения при выполнении сборочного чертежа.
17. Строительный чертёж. Правила выполнения (масштаб, линии чертежа, изображения на чертеже).

18. Нанесение размеров на строительном чертеже (на плане, фасаде, разрезе здания). Координационные оси.
19. Конструктивные элементы здания фундамент, стены, перекрытия, цоколь, кровля (понятие, обозначение на чертеже).
20. Схемы (понятие, классификация, правила выполнения схем). Перечень элементов.
21. Правила привязки технологического оборудования на чертеже.
22. Разъёмные соединения (резьбовые, шпоночные, шлицевые). Назначение, правила выполнения на чертеже.
23. Разрезы простые (понятие, назначение, классификация, обозначение на чертежах, правила выполнения).
24. План и разрез здания, правила выполнения и оформления.

Литература для самостоятельного изучения к Разделу 3

Основная литература

1. Инженерная 3D-компьютерная графика в 2 т. Том 1 : учебник и практикум для вузов / А. Л. Хейфец, А. Н. Логиновский, И. В. Буторина, В. Н. Васильева ; под редакцией А. Л. Хейфеца. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 328 с. — (Высшее образование). — ISBN 978-5-534-02957-4. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/513027> (дата обращения: 08.03.2023).
2. Инженерная 3D-компьютерная графика в 2 т. Том 2 : учебник и практикум для вузов / А. Л. Хейфец, А. Н. Логиновский, И. В. Буторина, В. Н. Васильева ; под редакцией А. Л. Хейфеца. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 279 с. — (Высшее образование). — ISBN 978-5-534-02959-8. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/513028> (дата обращения: 08.03.2023).5.1.2.
3. Чекмарев, А. А. Инженерная графика : учебник для вузов / А. А. Чекмарев. — 13-е изд., испр. и доп. — Москва : Издательство Юрайт, 2023. — 355 с. — (Высшее образование). — ISBN 978-5-534-12795-9. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511257> (дата обращения: 08.03.2023)

Дополнительная литература

1. Кордонская, И. Б. Инженерная и компьютерная графика : учебник / И. Б. Кордонская, Е. А. Богданова. — Самара : ПГУТИ, 2020. — 264 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/255455> (дата обращения: 08.03.2023). — Режим доступа: для авториз. пользователей.
2. Инженерная и компьютерная графика : учебник и практикум для вузов / Р. Р. Анамова [и др.] ; под общей редакцией Р. Р. Анамовой, С. А. Леоновой, Н. В. Пшеничной. — Москва : Издательство Юрайт, 2023. — 246 с. — (Высшее образование). — ISBN 978-5-9916-8262-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/512176> (дата обращения: 08.03.2023).
3. Колошкина, И. Е. Инженерная графика. CAD : учебник и практикум для вузов / И. Е. Колошкина, В. А. Селезнев. — Москва : Издательство Юрайт, 2023. — 220 с. — (Высшее образование). — ISBN 978-5-534-10412-7. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/517264> (дата обращения: 08.03.2023).

4. Хейфец, А. Л. Компьютерная графика для строителей : учебник для вузов / А. Л. Хейфец, В. Н. Васильева, И. В. Буторина ; под редакцией А. Л. Хейфеца. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2022. — 258 с. — (Высшее образование). — ISBN 978-5-534-10969-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/490414> (дата обращения: 12.04.2022).

1.4. Методические материалы по подготовке к лабораторным занятиям по дисциплине (модулю)

Лабораторные занятия - одна из форм учебного занятия, направленная на развитие самостоятельности обучающихся и приобретение умений и навыков. Данные учебные занятия углубляют, расширяют, детализируют полученные ранее знания. Лабораторное занятие предполагает выполнение обучающимися по заданию и под руководством преподавателей одной или нескольких работ.

Цель лабораторных занятий состоит в развитии познавательных способностей, самостоятельности мышления и творческой активности обучающихся; углублении, расширении, детализировании знаний, полученных на лекции в обобщенной форме, и содействии выработке навыков профессиональной деятельности. В отдельных случаях на лабораторных занятиях руководителем занятия сообщаются дополнительные знания.

Для достижения поставленных целей и решения требуемого перечня задач лабораторные занятия проводятся традиционными технологиями или с использованием активных и интерактивных образовательных технологий.

При подготовке и работе во время проведения лабораторных занятий следует обратить внимание на следующие моменты: на процесс предварительной подготовки, на работу во время занятия, обработку полученных результатов, исправление полученных замечаний.

Предварительная подготовка к лабораторному занятию заключается в изучении теоретического материала в отведенное для самостоятельной работы время, ознакомление с инструктивными материалами с целью осознания задач занятия, техники безопасности при работе с приборами, веществами.

Работа во время проведения занятия включает:

- консультирование студентов преподавателями и вспомогательным персоналом с целью предоставления исчерпывающей информации, необходимой для самостоятельного выполнения предложенных преподавателем задач.
- самостоятельное выполнение заданий согласно обозначенной рабочей программой дисциплины (модуля) тематики.

2. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ) И САМОСТОЯТЕЛЬНОЙ РАБОТЕ

Освоение обучающимся дисциплины (модуля) «Инженерная геометрия и компьютерная графика (модуль)» предполагает изучение материалов дисциплины на аудиторных занятиях и в ходе самостоятельной работы. Аудиторные занятия проходят в форме лекций, семинаров, практических и лабораторных занятий. Самостоятельная работа включает разнообразный комплекс видов и форм работы обучающихся.

Для успешного освоения дисциплины (модуля) и достижения поставленных целей необходимо внимательно ознакомиться с рабочей программы дисциплины (модуля), доступной в электронной информационно-образовательной среде РГСУ.

Следует обратить внимание на списки основной и дополнительной литературы, на предлагаемые преподавателем ресурсы информационно-телекоммуникационной сети Интернет. Эта информация необходима для самостоятельной работы обучающегося.

При подготовке к аудиторным занятиям необходимо помнить особенности каждой формы его проведения.

Подготовка к учебному занятию лекционного типа заключается в следующем.

С целью обеспечения успешного обучения обучающийся должен готовиться к лекции, поскольку она является важнейшей формой организации учебного процесса, поскольку:

- знакомит с новым учебным материалом;
- разъясняет учебные элементы, трудные для понимания;
- систематизирует учебный материал;
- ориентирует в учебном процессе.

С этой целью:

- внимательно прочитайте материал предыдущей лекции;
- ознакомьтесь с учебным материалом по учебнику и учебным пособиям с темой прочитанной лекции;
- внесите дополнения к полученным ранее знаниям по теме лекции на полях лекционной тетради;
- запишите возможные вопросы, которые вы зададите лектору на лекции по материалу изученной лекции;
- постарайтесь уяснить место изучаемой темы в своей подготовке;
- узнайте тему предстоящей лекции (по тематическому плану, по информации лектора) и запишите информацию, которой вы владеете по данному вопросу.

Подготовка к занятию семинарского типа.

При подготовке и работе во время проведения занятий семинарского типа следует обратить внимание на следующие моменты: на процесс предварительной подготовки, на работу во время занятия, обработку полученных результатов, исправление полученных замечаний.

Предварительная подготовка к учебному занятию семинарского типа заключается в изучении теоретического материала в отведенное для самостоятельной работы время, ознакомление с инструктивными материалами с целью осознания задач практического занятия, техники безопасности при работе с приборами, веществами.

Работа во время проведения учебного занятия семинарского типа включает:

- консультирование студентов преподавателями и вспомогательным персоналом с целью предоставления исчерпывающей информации, необходимой для самостоятельного выполнения предложенных преподавателем задач.
- самостоятельное выполнение заданий согласно обозначенной рабочей программой дисциплины (модуля) тематики.

Самостоятельная работа.

Самостоятельная работа - планируемая учебная, учебно-исследовательская, научно-исследовательская работа студентов, выполняемая во внеаудиторное (аудиторное) время по заданию и при методическом руководстве преподавателя, но без его непосредственного участия (при частичном непосредственном участии преподавателя, оставляющем ведущую роль за работой студентов).

Для более углубленного изучения темы задания для самостоятельной работы рекомендуется выполнять параллельно с изучением данной темы. При выполнении заданий по возможности используйте наглядное представление материала.

Самостоятельная работа студентов в ВУЗе является важным видом учебной и научной деятельности студента. Самостоятельная работа студентов играет значительную роль в рейтинговой технологии обучения.

К современному специалисту в области медицины общество предъявляет достаточно широкий перечень требований, среди которых немаловажное значение имеет наличие у выпускников определенных навыков (компетенций) и умения самостоятельно добывать знания из различных источников, систематизировать полученную информацию, давать оценку

конкретной профессиональной ситуации. Формирование такого умения происходит в течение всего периода обучения.

В процессе самостоятельной работы студент приобретает необходимые для будущей специальности компетенции, навыки самоорганизации, самоконтроля, самоуправления, саморефлексии и становится активным самостоятельным субъектом учебной деятельности.

Самостоятельная работа студентов должна оказывать важное влияние на формирование личности будущего специалиста, его компетентность. Каждый студент самостоятельно определяет режим своей работы и меру труда, затрачиваемого на овладение учебным содержанием по каждой дисциплине. Он выполняет внеаудиторную работу по личному индивидуальному плану, в зависимости от его подготовки, времени и других условий.

Виды самостоятельной работы.

Работа с литературой.

При работе с книгой необходимо подобрать литературу, научиться правильно ее читать, вести записи. Для подбора литературы в библиотеке используются алфавитный и систематический каталоги. Важно помнить, что рациональные навыки работы с книгой - это всегда большая экономия времени и сил. Правильный подбор учебников рекомендуется преподавателем, читающим лекционный курс. Необходимая литература может быть также указана в методических разработках по данному курсу. Изучая материал по учебнику, следует переходить к следующему вопросу только после правильного уяснения предыдущего, описывая на бумаге все выкладки и вычисления (в том числе те, которые в учебнике опущены или на лекции даны для самостоятельного вывода). При изучении любой дисциплины большую и важную роль играет самостоятельная индивидуальная работа. Особое внимание следует обратить на определение основных понятий курса. Студент должен подробно разбирать примеры, которые поясняют такие определения, и уметь строить аналогичные примеры самостоятельно. Нужно добиваться точного представления о том, что изучаешь. Полезно составлять опорные конспекты. При изучении материала по учебнику полезно в тетради (на специально отведенных полях) дополнять конспект лекций. Там же следует отмечать вопросы, выделенные студентом для консультации с преподавателем. Выводы, полученные в результате изучения, рекомендуется в конспекте выделять, чтобы они при перечитывании записей лучше запоминались. Опыт показывает, что многим студентам помогает составление листа опорных сигналов, содержащего важнейшие и наиболее часто употребляемые формулы и понятия. Такой лист помогает запомнить формулы, основные положения лекции, а также может служить постоянным справочником для студента. Различают два вида чтения: первичное и вторичное. Первичное - это внимательное, неторопливое чтение, при котором можно остановиться на трудных местах.

Задача вторичного чтения - полное усвоение смысла целого (по счету это чтение может быть и не вторым, а третьим или четвертым). Самостоятельная работа с учебниками и книгами (а также самостоятельное теоретическое исследование проблем, обозначенных преподавателем на лекциях) – это важнейшее условие формирования у себя научного способа познания.

Чтение научного текста является частью познавательной деятельности. Ее цель – извлечение из текста необходимой информации. От того, насколько осознанно читающим собственная внутренняя установка при обращении к печатному слову (найти нужные сведения, усвоить информацию полностью или частично, критически проанализировать материал и т.п.) во многом зависит эффективность осуществляемого действия.

Методические рекомендации по составлению конспекта:

1. Внимательно прочитайте текст. Уточните в справочной литературе непонятные слова. При записи не забудьте вынести справочные данные на поля конспекта;
2. Выделите главное, составьте план;
3. Кратко сформулируйте основные положения текста, отметьте аргументацию автора;
4. Законспектируйте материал, четко следуя пунктам плана. При конспектировании старайтесь выразить мысль своими словами. Записи следует вести четко, ясно.

5. Грамотно записывайте цитаты. Цитируя, учитывайте лаконичность, значимость мысли. В тексте конспекта желательно приводить не только тезисные положения, но и их доказательства. При оформлении конспекта необходимо стремиться к емкости каждого предложения. Мысли автора книги следует излагать кратко, заботясь о стиле и выразительности написанного. Число дополнительных элементов конспекта должно быть логически обоснованным, записи должны распределяться в определенной последовательности, отвечающей логической структуре произведения. Для уточнения и дополнения необходимо оставлять поля. Овладение навыками конспектирования требует от студента целеустремленности, повседневной самостоятельной работы.

Методические материалы по самостоятельному решению задач

При самостоятельном решении задач нужно обосновывать каждый этап решения, исходя из теоретических положений курса. Если студент видит несколько путей решения проблемы (задачи), то нужно сравнить их и выбрать самый рациональный. Полезно до начала вычислений составить краткий план решения проблемы (задачи). Решение проблемных задач или примеров следует излагать подробно, вычисления располагать в строгом порядке, отделяя вспомогательные вычисления от основных. Решения при необходимости нужно сопровождать комментариями, схемами, чертежами и рисунками. Следует помнить, что решение каждой учебной задачи должно доводиться до окончательного логического ответа, которого требует условие, и по возможности с выводом. Полученный ответ следует проверить способами, вытекающими из существа данной задачи. Полезно также (если возможно) решать несколькими способами и сравнить полученные результаты. Решение задач данного типа нужно продолжать до приобретения твердых навыков в их решении.

Методические материалы к выполнению реферата

Реферат (от лат. *refere* – сообщать) – краткое изложение в письменном виде или в форме публикации доклада, содержания научного труда (трудов), литературы по теме. Работа над рефератом условно разделяется на выбор темы, подбор литературы, подготовку и защиту плана; написание теоретической части и всего текста с указанием библиографических данных используемых источников, подготовку доклада, выступление с ним. Тематика рефератов полностью связана с основными вопросами изучаемого курса.

Список литературы к темам не дается, и обучающиеся самостоятельно ведут библиографический поиск, причем им не рекомендуется ограничиваться университетской библиотекой.

Важно учитывать, что написание реферата требует от обучающихся определенных усилий. Особое внимание следует уделить подбору литературы, методике ее изучения с целью отбора и обработки собранного материала, обоснованию актуальности темы и теоретического уровня обоснованности используемых в качестве примеров фактов какой-либо деятельности.

Выбрав тему реферата, начав работу над литературой, необходимо составить план. Изучая литературу, продолжается обдумывание темы, осмысливание прочитанного, делаются выписки, сопоставляются точки зрения разных авторов и т.д. Реферативная работа сводится к тому, чтобы в ней выделились две взаимосвязанные стороны: во-первых, ее следует рассматривать как учебное задание, которое должен выполнить обучаемый, а во-вторых, как форму научной работы, творческого воображения при выполнении учебного задания.

Наличие плана реферата позволяет контролировать ход работы, избежать формального переписывания текстов из первоисточников.

Оформление реферата включает титульный лист, оглавление и краткий список использованной литературы. Список использованной литературы размещается на последней странице рукописи или печатной форме реферата. Реферат выполняется в письменной или печатной форме на белых листах формата А4 (210 x 297 мм). Шрифт Times New Roman, кегель 14, через 1,5 интервала при соблюдении следующих размеров текста: верхнее поле – 25 мм, нижнее – 20 мм, левое – 30 мм, правое – 15 мм. Нумерация страниц производится сверху листа, по центру. Титульный лист нумерации не подлежит.

Рефераты должны быть написаны простым, ясным языком, без претензий на наукообразность. Следует избегать сложных грамматических оборотов, непривычных терминов и символов. Если же такие термины и символы все-таки приводятся, то необходимо разъяснять их значение при первом упоминании в тексте реферата.

Объем реферата предполагает тщательный отбор информации, необходимой для краткого изложения вопроса. Важнейший этап – редактирование готового текста реферата и подготовка к обсуждению. Обсуждение требует хорошей ориентации в материале темы, умения выделить главное, поставить дискуссионный вопрос, привлечь внимание слушателей к интересной литературе, логично и убедительно изложить свои мысли.

Рефераты обязательно подлежат защите. Процедура защиты начинается с определения оппонентов защищающего свою работу. Они стремятся дать основательный анализ работы обучающимся, обращают внимание на положительные моменты и недостатки реферата, дают общую оценку содержанию, форме преподнесения материала, характеру использованной литературы. Иногда они дополняют тот или иной раздел реферата. Последнее особенно ценно, ибо говорит о глубоком знании обучающимся-оппонентом изучаемой проблемы.

Обсуждение не ограничивается заслушиванием оппонентов. Другие обучающиеся имеют право уточнить или опровергнуть какое-либо утверждение. Преподаватель предлагает любому обучающемуся задать вопрос по существу доклада или попытаться подвести итог обсуждению.

Алгоритм работы над рефератом

1. Выбор темы

Тема должна быть сформулирована грамотно (с литературной точки зрения);

В названии реферата следует поставить четкие рамки рассмотрения темы;

Желательно избегать слишком длинных названий;

Следует по возможности воздерживаться от использования в названии спорных с научной точки зрения терминов, излишней наукообразности, а также чрезмерного упрощения формулировок.

2. Реферат следует составлять из пяти основных частей: введения; основной части; заключения; списка литературы; приложений.

3. Основные требования к введению:

Во введении не следует концентрироваться на содержании; введение должно включать краткое обоснование актуальности темы реферата, где требуется показать, почему данный вопрос может представлять научный интерес и есть ли связь представляемого материала с современностью. Таким образом, тема реферата должна быть актуальна либо с научной точки зрения, либо с современных позиций.

Очень важно выделить цель, а также задачи, которые требуется решить для выполнения цели.

Введение должно содержать краткий обзор изученной литературы, в котором указывается взятый из того или иного источника материал, кратко анализируются изученные источники, показываются их сильные и слабые стороны;

Объем введения составляет две страницы текста.

4. Требования к основной части реферата:

Основная часть содержит материал, отобранный для рассмотрения проблемы;

Также основная часть должна включать в себя собственно мнение обучающихся и сформулированные самостоятельные выводы, опирающиеся на приведенные факты;

Материал, представленный в основной части, должен быть логически изложен и распределен по параграфам, имеющим свои названия;

В изложении основной части необходимо использовать сноски (в первую очередь, когда приводятся цифры и чьи-то цитаты);

Основная часть должна содержать иллюстративный материал (графики, таблицы и т. д.);

Объем основной части составляет около 10 страниц.

5. Требования к заключению:

В заключении формулируются выводы по параграфам, обращается внимание на выдвинутые во введении задачи и цели;

Заключение должно быть четким, кратким, вытекающим из содержания основной части.

6. Требования к оформлению списка литературы (по ГОСТу):

Необходимо соблюдать правильность последовательности записи источников: сначала следует писать фамилию, а после инициалы; название работы не ставится в кавычки; после названия сокращенно пишется место издания; затем идет год издания; наконец, называется процитированная страница.

Критерии оценки реферата

Обучающийся, защищающий реферат, должен рассказать о его актуальности, поставленных целях и задачах, изученной литературе, структуре основной части, сделанных в ходе работы выводах.

По окончанию выступления ему может быть задано несколько вопросов по представленной проблеме.

Оценка складывается из соблюдения требований к реферату, грамотного раскрытия темы, умения четко рассказывать о представленном реферате, способности понять суть задаваемых по работе вопросов и найти точные ответы на них.

Методические материалы к выполнению эссе

Эссе – литературное произведение небольшого объема, обычно прозаическое, свободной композиции, передающее индивидуальные впечатления, суждения, соображения автора о той или иной проблеме, теме, о том или ином событии или явлении. Это вид самостоятельной исследовательской работы обучающихся, с целью углубления и закрепления теоретических знаний и освоения практических навыков. Цель эссе состоит в развитии самостоятельного творческого мышления и письменного изложения собственных мыслей. При написании эссе обучающийся должен представить развернутый письменный ответ на теоретический или практический актуальный вопрос, объявленный преподавателем в аудитории непосредственно перед ее написанием. В процессе написания эссе разрешается пользоваться нормативно-правовыми актами, конспектом лекций (в печатном виде). Использование интернет-ресурсов не допускается. Темы эссе преподаватель предлагает из числа тех, которые обучающиеся уже рассматривали на лекциях или семинарских занятиях, исходя из содержания заданий в составе оценочных средств. По решению преподавателя, в качестве темы эссе может быть выбрана одна или несколько тем, которые могут быть распределены между обучающимися по желанию.

Требования к выполнению эссе:

1. Проводится письменно.

2. Эссе выполняется на компьютере (гарнитура Times New Roman, шрифт 14) через 1,5 интервала с полями: верхнее, нижнее – 2; правое – 3; левое – 1,5. Отступ первой строки абзаца – 1,25. Сноски – постраничные. Таблицы и рисунки встраиваются в текст работы. При этом обязательный заголовок таблицы надо размещать над табличным полем, а рисунки сопровождать подрисуночными подписями. При включении в эссе нескольких таблиц и/или рисунков их нумерация обязательна. Обязательна и нумерация страниц. Их целесообразно проставлять внизу страницы – по середине или в правом углу. Номер страницы не ставится на титульном листе, но в общее число страниц он включается. Объем эссе, без учета приложений, не должен превышать 5 страниц. Значительное превышение установленного объема является недостатком работы и указывает на то, что обучающийся не сумел отобрать и переработать необходимый материал.

3. Работа должна содержать собственные умозаключения по сути поставленной проблемы, включать самостоятельно проведенный анализ по сути этой проблемы, выводы, обобщающие авторскую позицию по поставленной проблеме.

Критерии оценки эссе:

«Отлично» – исключительные знания материала, абсолютное понимание сути, безукоризненное знание основных понятий и положений, логически и лексически грамотно изложенный, содержательный, аргументированный, конкретный и исчерпывающий ответ.

«Хорошо» – глубокие знания материала, правильное понимание сути, знание основных понятий и положений, содержательный, полный и конкретный ответ.

«Удовлетворительно» – твердые, но недостаточно полные знания, верное понимание сути, в целом правильный ответ.

«Неудовлетворительно» – непонимание сущности задания, грубые ошибки в ответе.

Методические материалы по выполнению тестирования.

Тестовые задания содержат вопросы и 3-4 варианта ответа по базовым положениям изучаемой темы/раздела, составлены с расчетом на знания, полученные обучающимся в процессе изучения темы/раздела.

Тестовые задания выполняются в письменной форме и сдаются преподавателю, ведущему дисциплину (модуль). На выполнение тестовых заданий обучающимся отводится 45 минут.

При обработке результатов оценочной процедуры используются: критерии оценки по содержанию и качеству полученных ответов, ключи, оценочные листы.

Критерии оценки теста:

«Зачтено» - если обучающийся правильно выполнил не менее 2/3 всей работы или допустил не более одной грубой ошибки и двух недочетов, не более одной грубой и одной негрубой ошибки, не более трех негрубых ошибок, одной негрубой ошибки и трех недочетов, при наличии четырех-пяти недочетов.

«Не зачтено» - если число ошибок и недочетов превысило норму для оценки 3 или правильно выполнено менее 2/3 всей работы.

Методические материалы по выполнению доклада.

Рекомендуется следующая структура доклада:

1. титульный лист, содержание доклада;
2. краткое изложение;
3. цели и задачи;
4. изложение характера исследований и рассмотренных проблем, гипотезы, спорные вопросы;
5. источники информации, методы сбора и анализа данных, степень их полноты и достоверности;
6. анализ и толкование полученных в работе результатов;
7. выводы и оценки;
8. библиография и приложения.

Время выступления докладчика не должно превышать 10 минут.

Основные требования к оформлению доклада:

- титульный лист должен включать название доклада, наименование предметной (цикловой) комиссии, фамилию обучающегося;
- все использованные литературные источники сопровождаются библиографическим описанием;
- приводимая цитата из источника берется в кавычки (оформляются сноски);
- единицы измерения должны применяться в соответствии с действующими стандартами;
- все названия литературных источников следует приводить в соответствии с новейшими изданиями;
- рекомендуется включение таблиц, графиков, схем, если они отражают основное содержание или улучшают ее наглядность;
- названия фирм, учреждений, организаций и предприятий должны именоваться так, как они указываются в источнике;

Критерии оценки доклада

При выполнении доклада обучающийся должен продемонстрировать умение кратко излагать прочитанный материал, а также умение обобщать и анализировать материал по теме доклада.

Презентация

Методические материалы к презентациям

1. Объём презентации 10 -20 слайдов.
2. На титульном слайде должно быть отражено:
 - наименование факультета;
 - тема презентации;
 - фамилия, имя, отчество, направление подготовки/ специальность, направленность (профиль)/ специализация, форма обучения, номер группы автора презентации;
 - фамилия, имя, отчество, степень, звание, должность руководитель работы;
 - год выполнения работы.
3. В презентации должны быть отражены обоснование актуальности представляемого материала, цели и задачи работы.
4. Содержание презентации должно включать наиболее значимый материал доклада, а также, при необходимости, таблицы, диаграммы, рисунки, фотографии, карты, видео – вставки, звуковое сопровождение.
5. Заключительный слайд должен содержать информацию об источниках информации для презентации.

Критерии оценки презентации

1. Объём презентации 10 -20 слайдов.
2. Правильность оформления титульного слайда.
3. Актуальность отобранного материала, обоснованность формулировки цели и задач работы.
4. Наглядность и логичность презентации, обоснованность использования таблиц, диаграмм, рисунков, фотографий, карт, видео – вставок, звукового сопровождения; правильный выбор шрифтов, фона, других элементов дизайна слайда.
5. Объём и качество источников информации (не менее 2-х интернет – источников и не менее 2-х литературных источников).

Методические материалы по подготовке к опросу

Самостоятельная работа обучающихся включает подготовку к опросу на практических занятиях. Для этого обучающийся изучает лекции, основную и дополнительную литературу, публикации, информацию из Интернет-ресурсов.

Тема и вопросы к практическим занятиям, вопросы для самоконтроля содержатся в рабочей программе и доводятся до студентов заранее.

Для подготовки к опросу обучающемуся необходимо ознакомиться с материалом, посвященным теме практического занятия, в учебнике или другой рекомендованной литературе, конспекте лекции, обратить внимание на усвоение основных понятий дисциплины, выявить неясные вопросы и подобрать дополнительную литературу для их освещения.

Критерии оценки опроса

- «Отлично»:
- дан полный, развернутый ответ на поставленный вопрос;
 - в ответе прослеживается четкая структура, логическая последовательность, отражающая сущность раскрываемых понятий, теорий, явлений;
 - знание по предмету демонстрируется на фоне понимания его в системе данной науки и междисциплинарных связей;
 - свободное владение терминологией;
 - ответы на дополнительные вопросы четкие, краткие;
- «Хорошо»:

– дан полный, развернутый ответ на поставленный вопрос, показано умение выделять существенные и несущественные признаки, причинно-следственные связи;

– ответ недостаточно логичен с единичными ошибками в частностях, исправленные студентом с помощью преподавателя;

– единичные ошибки в терминологии;

– ответы на дополнительные вопросы правильные, недостаточно полные и четкие.

«Удовлетворительно»:

– ответ не полный, с ошибками в деталях, умение раскрыть значение обобщённых знаний не показано, речевое оформление требует поправок, коррекции;

– логика и последовательность изложения имеют нарушения, студент не способен самостоятельно выделить существенные и несущественные признаки и причинно-следственные связи;

– ошибки в раскрываемых понятиях, терминах;

– студент не ориентируется в теме, допускает серьезные ошибки;

– студент не может ответить на большую часть дополнительных вопросов.

«Неудовлетворительно»:

– ответ представляет собой разрозненные знания с существенными ошибками по вопросу;

– присутствуют фрагментарность, нелогичность изложения, студент не осознает связь обсуждаемого вопроса с другими объектами дисциплины, речь неграмотная;

– незнание терминологии;

– ответы на дополнительные вопросы неправильные.

Методические материалы по выполнению практического задания

При выполнении практического задания обучающийся придерживается следующего алгоритма:

1. Записать дату, тему и цель задания;

2. Ознакомиться с правилами и условия выполнения практического задания;

3. Повторить теоретические задания, необходимые для рациональной работы и других практических действий, используя конспекты лекций и рекомендованную литературу, представленной в программе;

4. Выполнить работу по предложенному алгоритму действий;

5. Обобщить результаты работы, сформулировать выводы / дать ответы на контрольные вопросы;

Работа должна быть выполнена грамотно, с соблюдением культуры изложения. При использовании данных из учебных, методических пособий и другой литературы, периодических изданий, Интернет-источников должны иметься ссылки на вышеперечисленные.

Критерии оценки практического задания:

«Отлично» – правильный ответ, дается четкое обоснование принятому решению; рассуждения четкие последовательные логические; используются ссылки на полученные при изучении дисциплины знания; правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Хорошо» – правильный ответ, дается обоснование принятому решению; но с не существенными ошибками, в рассуждениях отсутствует логическая последовательность; используются ссылки на полученные при изучении дисциплины знания, правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Удовлетворительно» – правильный ответ, допускаются грубые ошибки в обосновании принятого решения; рассуждения не последовательные сумбурные; используются ссылки на полученные при изучении дисциплины знания; используются формулы, процедуры, понятия, имеющие прямое значение для подтверждения принятого решения, однако, при обращении к ним допускаются серьезные ошибки, студент не может правильно ими воспользоваться.

«Неудовлетворительно, не зачтено» – ответ неверный, отсутствует обоснование принятому решению; студент демонстрирует полное непонимание сути вопроса.

Для оценки решения ситуационной задачи (аналитического задания):

Оценка «отлично» выставляется, если задача решена грамотно, ответы на вопросы сформулированы четко. Эталонный ответ полностью соответствует решению студента, которое хорошо обосновано теоретически.

Оценка «хорошо» выставляется, если задача решена, ответы на вопросы сформулированы не достаточно четко. Решение студента в целом соответствует эталонному ответу, но не достаточно хорошо обосновано теоретически.

Оценка «удовлетворительно» выставляется, если задача решена не полностью, ответы не содержат всех необходимых обоснований решения.

Оценка «неудовлетворительно» выставляется, если задача не решена или имеет грубые теоретические ошибки в ответе на поставленные вопросы.

Методические материалы по выполнению лабораторного задания

При выполнении лабораторного задания обучающийся придерживается следующего алгоритма:

1. Записать дату, тему и цель задания;
2. Ознакомиться с правилами и условия выполнения задания;
3. Повторить теоретические задания, необходимые для рациональной работы и других практических действий, используя конспекты лекций и рекомендованную литературу, представленной в программе;
4. Выполнить работу по предложенному алгоритму действий;
5. Обобщить результаты работы, сформулировать выводы / дать ответы на контрольные вопросы;

Работа должна быть выполнена грамотно, с соблюдением культуры изложения. При использовании данных из учебных, методических пособий и другой литературы, периодических изданий, Интернет-источников должны иметься ссылки на вышеперечисленные.

Критерии оценки лабораторного задания:

«Отлично» – правильный ответ, дается четкое обоснование принятому решению; рассуждения четкие последовательные логические; используются ссылки на полученные при изучении дисциплины знания; правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Хорошо» – правильный ответ, дается обоснование принятому решению; но с не существенными ошибками, в рассуждениях отсутствует логическая последовательность; используются ссылки на полученные при изучении дисциплины знания, правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Удовлетворительно» – правильный ответ, допускаются грубые ошибки в обосновании принятого решения; рассуждения не последовательные сумбурные; используются ссылки на полученные при изучении дисциплины знания; используются формулы, процедуры, понятия, имеющие прямое значение для подтверждения принятого решения, однако, при обращении к ним допускаются серьезные ошибки, студент не может правильно ими воспользоваться.

«Неудовлетворительно, не зачтено» – ответ неверный, отсутствует обоснование принятому решению; студент демонстрирует полное непонимание сути вопроса.

Методические указания для подготовки к промежуточной аттестации.

Изучение учебных дисциплин (модулей) завершается зачетом/зачетом с оценкой или экзаменом. Подготовка к промежуточной аттестации способствует закреплению, углублению и обобщению знаний, получаемых, в процессе обучения, а также применению их к решению практических задач. Готовясь к экзамену, студент ликвидирует имеющиеся пробелы в знаниях, углубляет, систематизирует и упорядочивает свои знания. На зачете или экзамене студент демонстрирует то, что он освоил в процессе обучения по дисциплине (модулю).

Вначале следует просмотреть весь материал по дисциплине (модулю), отметить для себя трудные вопросы. Обязательно в них разобраться. В заключение еще раз целесообразно повторить основные положения, используя при этом листы опорных сигналов.

Систематическая подготовка к занятиям в течение семестра позволит использовать время промежуточной аттестации для систематизации знаний.

3. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНИВАНИЯ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ И (ИЛИ) ОПЫТА ДЕЯТЕЛЬНОСТИ, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ

3.1. Организационные основы применения балльно-рейтинговой системы оценки успеваемости обучающихся по дисциплине (модулю)

Оценка качества освоения обучающимися дисциплины (модуля) реализуется в формате балльно-рейтинговой системы оценки успеваемости обучающихся (БРСО).

БРСО в ходе текущего контроля успеваемости и промежуточной аттестации осуществляется по 100-балльной шкале.

Академический рейтинг обучающегося по дисциплине (модулю) складывается из результатов:

- текущего контроля успеваемости (максимальный текущий рейтинг обучающегося 80 рейтинговых баллов);
- промежуточной аттестации (максимальный рубежный рейтинг обучающегося 20 рейтинговых баллов).

Условия оценки освоения обучающимся дисциплины (модуля) в формате БРСО доводятся преподавателем до сведения обучающихся на первом учебном занятии, а также размещены в свободном доступе в электронной информационно-образовательной среде Университета.

3.2. Проведение текущего контроля успеваемости обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося

В течение учебного семестра до промежуточной аттестации на основании утвержденной рабочей программы дисциплины (модуля) формируется текущий рейтинг обучающегося. Текущий рейтинг обучающегося складывается как сумма рейтинговых баллов, полученных им в течение учебного семестра по всем видам учебных занятий по учебной дисциплине.

В процессе текущего контроля оцениваются следующие действия обучающегося, направленные на освоение компетенций в рамках изучения учебной дисциплины:

- академическая активность (посещаемость учебных занятий, самостоятельное изучение содержания учебной дисциплины в электронной информационно-образовательной среде, соблюдение сроков сдачи практических заданий и текущих контрольных мероприятий и др.);
- выполнение и сдача текущих и итогового практических заданий (эссе, рефераты, творческие задания, кейс-задания, лабораторные работы, расчетные задания и др., активное участие в групповых интерактивных занятиях (дискуссии, WiKi-проекты и др.), защита проектов и др.);
- прохождение рубежей текущего контроля, включая соблюдение графика их прохождения в электронной информационно-образовательной среде.

Для планирования расчета текущего рейтинга обучающегося используются следующие пропорции:

Вид учебного действия	Максимальная рейтинговая оценка, баллов
академическая активность	10
практические задания	40
из них: текущие практические задания	20

<i>итоговое практическое задание</i>	<i>20</i>
рубежи текущего контроля	<i>30</i>
<i>ИТОГО:</i>	<i>80</i>

В течение учебного семестра по дисциплине (модулю) обучающимся должен быть накоплен текущий рейтинг не менее 52 рейтинговых баллов (65% от максимального значения текущего рейтинга).

Необходимыми условиями допуска обучающегося к промежуточной аттестации по дисциплине являются положительное прохождение обучающимся не менее 65% рубежей текущего контроля с накоплением не менее 65% максимального рейтингового балла за каждый рубеж текущего контроля и положительное выполнение итогового практического задания с накоплением не менее 65% максимального рейтингового балла, установленного за итоговое практическое задание.

Невыполнение вышеуказанных условий является текущей академической задолженностью, которая должна быть ликвидирована обучающимся до контрольного мероприятия промежуточной аттестации.

Сведения о наличии у обучающихся текущей академической задолженности, сроках и порядке добора рейтинговых баллов для её ликвидации доводятся до обучающихся педагогическим работником.

В случае неликвидации текущей академической задолженности, педагогический работник обязан во время контрольного мероприятия промежуточной аттестации поставить обучающемуся 0 рейтинговых баллов. В этом случае ликвидация текущей академической задолженности возможна в периоды проведения повторной промежуточной аттестации.

3.3. Проведение промежуточной аттестации обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося

Промежуточная аттестация по дисциплине (модулю) проводится в соответствии с Положением о промежуточной аттестации обучающихся по основным профессиональным образовательным программам высшего образования – программ специалитета в Российском государственном социальном университете и Положением о балльно-рейтинговой системе оценки успеваемости обучающихся по основным профессиональным образовательным программам высшего образования - программам бакалавриата, программам специалитета, программам магистратуры в Российском государственном социальном университете.

На промежуточную аттестацию отводится 20 рейтинговых баллов.

Ответы обучающегося на контрольном мероприятии промежуточной аттестации оцениваются педагогическим работником по 20 - балльной шкале, а итоговая оценка по дисциплине (модулю) выставляется по пятибалльной системе для экзамена.

Критерии выставления оценки определяются Положением о балльно-рейтинговой системе оценки успеваемости обучающихся по основным профессиональным образовательным программам высшего образования - программам бакалавриата, программам специалитета, программам магистратуры в Российском государственном социальном университете.

В процессе определения рубежного рейтинга обучающегося используется следующая шкала:

Рубежный рейтинг	Критерии оценки освоения обучающимся учебной дисциплины в ходе контрольных мероприятий промежуточной аттестации

19-20 рейтинговых баллов	обучающийся глубоко и прочно усвоил программный материал, исчерпывающе, последовательно, грамотно и логически стройно его излагает, тесно увязывает с задачами и будущей деятельностью, не затрудняется с ответом при видоизменении задания, свободно справляется с задачами и практическими заданиями, правильно обосновывает принятые решения, умеет самостоятельно обобщать и излагать материал, не допуская
16-18 рейтинговых баллов	обучающийся твердо знает программный материал, грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос, может правильно применять теоретические положения и владеет необходимыми умениями и навыками при выполнении практических
13-15 рейтинговых баллов	обучающийся освоил основной материал, но не знает отдельных деталей, допускает неточности, недостаточно правильные формулировки, нарушает последовательность в изложении программного материала и испытывает затруднения в выполнении практических заданий
1-12 рейтинговых баллов	обучающийся не знает значительной части программного материала, допускает существенные ошибки, с большими затруднениями выполняет практические задания
0 рейтинговых баллов	не аттестован

Если результат контроля успеваемости в рамках проведения контрольных мероприятий промежуточной аттестации (рубежный рейтинг обучающегося) неудовлетворительный (получено менее 13 рейтинговых баллов), то промежуточная аттестация по учебной дисциплине (модулю) невозможна даже при наличии высокого текущего рейтинга, полученного по итогам текущего контроля по учебной дисциплине (модулю).

ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

№ п/п	Содержание изменения	Реквизиты документа об утверждении изменения	Дата введения изменения
1.			
2.			
3.			
4.			



**Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Российский государственный социальный университет»**

УТВЕРЖДАЮ

Декан факультета политических и социальных
технологий

/Пивнева С.В./

28 февраля 2024 г.

**МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ
ТЕХНОЛОГИИ РАЗРАБОТКИ ВИРТУАЛЬНОЙ И ДОПОЛНЕННОЙ РЕАЛЬНОСТИ**

**Направление подготовки
«Информационные системы и технологии»**

**Направленность
«Интеллектуальные информационные системы и технологии»**

**ОБРАЗОВАТЕЛЬНАЯ ПРОГРАММА ВЫСШЕГО ОБРАЗОВАНИЯ - ПРОГРАММА
БАКАЛАВРИАТА**

Форма обучения

Очная, заочная

Москва, 2024 г.

Методические материалы по дисциплине (модулю) «Технологии разработки виртуальной и дополненной реальности» разработаны на основании федерального государственного образовательного стандарта высшего образования – *бакалавриата* по направлению подготовки *09.03.02 Информационные системы и технологии*, утвержденного приказом Министерства образования и науки Российской Федерации от 19.09.2017 № 926, учебного плана по основной профессиональной образовательной программе высшего образования – программе бакалавриата по направлению подготовки *09.03.02 Информационные системы и технологии* (далее – «ОПОП»).

Методические материалы по дисциплине (модулю) «Технологии разработки виртуальной и дополненной реальности» разработана рабочей группой в составе: канд. пед. наук, доцент Мнацаканян О.Л.

Методические материалы по дисциплине (модулю) обсуждены и утверждены на заседании кафедры информационных технологий, искусственного интеллекта и общественно-социальных технологий цифрового общества факультета социальных и политических технологий.

Протокол № 11 от «28» февраля 2024 г.

Заведующий кафедрой
канд. пед. наук, доцент

_____  _____ С.В. Пивнева
(подпись)

СОДЕРЖАНИЕ

1. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ К ЛЕКЦИОННЫМ, ЛАБОРАТОРНЫМ ЗАНЯТИЯМ	4
1.1. Методические материалы к проведению лекционных занятий по дисциплине (модулю)	4
1.2. Методические материалы по подготовке к лабораторным занятиям по дисциплине (модулю)	7
2. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ) И САМОСТОЯТЕЛЬНОЙ РАБОТЕ	7
3. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНИВАНИЯ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ И (ИЛИ) ОПЫТА ДЕЯТЕЛЬНОСТИ, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ	16
3.1. Организационные основы применения балльно-рейтинговой системы оценки успеваемости обучающихся по дисциплине (модулю)	17
3.2. Проведение текущего контроля успеваемости обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося	17
3.3. Проведение промежуточной аттестации обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося	18
ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ	20

1. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ К ЛЕКЦИОННЫМ, ЛАБОРАТОРНЫМ ЗАНЯТИЯМ

1.1. Методические материалы к проведению лекционных занятий по дисциплине (модулю)

Лекция – один из методов обучения, одна из основных системообразующих форм организации учебного процесса в вузе. Лекционное занятие представляет собой систематическое, последовательное, монологическое изложение педагогическим работником учебного материала, как правило, теоретического характера. Такое занятие представляет собой элемент технологии представления учебного материала путем логически стройного, систематически последовательного и ясного изложения.

Цель лекции – организация целенаправленной познавательной деятельности обучающихся по овладению программным материалом дисциплины (модуля). Чтение курса лекций позволяет дать связанное, последовательное изложение материала в соответствии с новейшими данными науки, сообщить слушателям основное содержание предмета в целостном, систематизированном виде. В ряде случаев лекция выполняет функцию основного источника информации: при отсутствии учебников и учебных пособий, чаще по новым курсам; в случае, когда новые научные данные по той или иной теме не нашли отражения в учебниках; отдельные разделы и темы очень сложны для самостоятельного изучения. В таких случаях только лектор может методически помочь обучающимся в освоении сложного материала.

Возможные формы проведения лекций:

- Вводная лекция – один из наиболее важных и трудных видов лекции при чтении систематических курсов. От успеха этой лекции во многом зависит успех усвоения всего курса. Она может содержать: определение дисциплины (модуля); краткую историческую справку о дисциплине (модуле); цели и задачи дисциплины (модуля), ее роль в общей системе обучения и связь со смежными дисциплинами (модулями); основные проблемы (понятия и определения) данной науки; основную и дополнительную учебную литературу; особенности самостоятельной работы обучающихся над дисциплиной (модулем) и формы участия в научно-исследовательской работе; отчетность по курсу.

- Информационная лекция ориентирована на изложение и объяснение обучающимся научной информации, подлежащей осмыслению и запоминанию. Это самый традиционный тип лекций в практике высшей школы.

- Заключительная лекция предназначена для обобщения полученных знаний и раскрытия перспектив дальнейшего развития данной науки.

- Обзорная лекция – это систематизация научных знаний на высоком уровне, допускающая большое число ассоциативных связей в процессе осмысления информации, излагаемой при раскрытии внутрипредметной и межпредметной связей, исключая детализацию и конкретизацию. Как правило, стержень излагаемых теоретических положений составляет научно-понятийная и концептуальная основа всего курса или крупных его разделов.

- Лекция-беседа – непосредственный контакт педагогического работника с аудиторией - диалог. По ходу лекции педагогический работник задает вопросы для выяснения мнений и уровня осведомленности обучающихся по рассматриваемой проблеме.

- Лекция-дискуссия – свободный обмен мнениями в ходе изложения лекционного материала. Педагогический работник активизирует участие в обсуждении отдельными вопросами, сопоставляет между собой различные мнения и тем самым развивает дискуссию, стремясь направить ее в нужное русло.

- Лекция с применением обратной связи включает в себе то, что в начале и конце каждого раздела лекции задаются вопросы. Первый – для того, чтобы узнать, насколько обучающиеся ориентируются в излагаемом материале, вопрос в конце раздела предназначен для выяснения степени усвоения только что изложенного материала. При

неудовлетворительных результатах контрольного опроса педагогический работник возвращается к уже прочитанному разделу, изменив при этом методику подачи материала.

- Проблемная лекция опирается на логику последовательно моделируемых проблемных ситуаций путем постановки проблемных вопросов или предъявления проблемных задач. Проблемный вопрос – это диалектическое противоречие, требующее для своего решения размышления, сравнения, поиска, приобретения и применения новых знаний. Проблемная задача содержит дополнительную вводную информацию и при необходимости некоторые ориентиры поиска ее решения.

- Программированная лекция-консультация – педагогический работник сам составляет и предлагает обучающимся вопросы. На подготовленные вопросы педагогический работник сначала просит ответить обучающихся, а затем проводит анализ и обсуждение неправильных ответов. В лекциях можно использовать наглядные материалы, а также подготовить презентацию. Что касается презентации, то в качестве визуальной поддержки ее можно органично интегрировать во все вышеупомянутые лекции. В то же время лекцию-презентацию возможно выделить и в качестве самостоятельной формы. Лекция-презентация должна отражать суть основных и (или) проблемных вопросов лекции, на которые особо следует обратить внимание обучающихся. В условиях применения активного метода проведения занятий презентация представляется весьма удачным способом донесения информации до слушателей. Единственное, на что следует обратить внимание при подготовке слайдов, - это их оформление и текст. Слайд не должен быть перегружен картинками и лишней информацией, которая будет отвлекать от основного аспекта того или иного вопроса лекции. Во время лекции можно задавать вопросы аудитории в отношении того или иного слайда, тем самым еще больше вовлекая обучающихся в проблематику.

Краткое содержание лекционных занятий

Наименование разделов и тем	Содержание учебного материала
РАЗДЕЛ 1. Основы технологий виртуальной и дополненной реальности	
Тема 1.1. Базовые понятия и определения технологий виртуальной и дополненной реальности	Введение в технологии виртуальной, дополненной и смешанной реальностей. Технология дополненной реальности. Технология виртуальной реальности. Основные различия между меточной и точечной технологии дополненной реальности. Достоинства и недостатки различных типов технологий дополненной реальности. История развития виртуальной и дополненной реальностей.
Тема 1.2. Сферы применения и использования технологий виртуальной и дополненной реальности	VR и AR в образовании. Эксплуатация и обучение персонала. Тренажеры и экзамены в VR. Выставочная деятельность, торговля, реклама и маркетинг. Виртуальные путешествия, фото и видео 360 (панорамное).
РАЗДЕЛ 2. Устройства визуализации и взаимодействия для иммерсивных сред	
Тема 2.1. Классификация устройств визуализации и взаимодействия для иммерсивных сред	Иммерсивные технологии. Использование иммерсивных технологий. Принцип работы носимых устройств виртуальной реальности. VR шлемы, очки дополненной реальности, панели и мониторы для отображения виртуальных объектов. Проблемы формирования изображения в системах виртуальной, дополненной и смешанной реальностей.
Тема 2.2. Устройства визуализации виртуальных	Компонентная база очков виртуальной реальности. Принцип работы оптической части индикатора на лобовом

объектов	стекле. “Bird bath” архитектура оптической системы дополненной реальности. Оптическая система наשלменного индикатора. Оптические системы дополненной реальности, построенные на принципе передачи изображения по свето- и волноводным каналам. Принцип работы устройства смешанной реальности.
Тема 2.3. Устройства взаимодействия с виртуальными объектами в иммерсивных средах	Устройства взаимодействия с виртуальными объектами в иммерсивных средах: системы трекинга головы, глаз, движений тела; перчатки, 3D контроллеры, устройства с обратной связью, платформы, датчики. Дисторсия и исправление дисторсии в окулярах систем виртуальной реальности. Принцип работы системы отслеживания положения зрачка. Эффект аккомодации человеческого зрения. Эффект рассогласования аккомодации и вергенции человеческого зрения. Принципиальная схема работы проекционного устройства дополненной реальности.
РАЗДЕЛ 3. Разработка приложений дополненной реальности	
Тема 3.1. Распознавание образов	Распознавание образов. Методы распознавания образов. Типы задач распознавания образов. Ограничения технологии дополненной реальности. Обзор средств разработки приложений дополненной реальности.
Тема 3.2. Технологии дополненной реальности	Структура приложения дополненной реальности. Модуль отслеживания камеры. Модуль хранения объектов. Модуль визуализации. Модуль пользовательского интерфейса. Алгоритм работы приложения дополненной реальности.
Тема 3.3. Маркерные технологии дополненной реальности	Технология трекинга. Маркерная технология распознавания. Безмаркерная технология распознавания. Контраст особых точек. Уникальность особых точек. Распределение особых точек. Масштаб особых точек. Физический размер метки.
РАЗДЕЛ 4. Разработка приложений виртуальной реальности	
Тема 4.1. Основы работы с SDK Unity 3D	Базовые основы формирования стереоизображений. Основы работы с SDK Unity 3D. Особенности взаимодействия с пользователем в виртуальной реальности. Технологии создания стереоизображений. Создание анаглифа. Создание стереограммы.
Тема 4.2. Создание VR-приложения с использованием SDK Unity	Принципы создание VR с применением SDK Unity. Создание VR-приложения с использованием SDK Unity. Создание VR-приложения с использованием SDK Unity и библиотеки ALPS-VR. Создание VR-приложения с использованием SDK Unity и библиотеки Fibrum SDK. ArtoolKit.
Тема 4.3. Программное обеспечение функционирования аппаратной составляющей взаимодействия с объектами виртуальной реальности	Перечень изучаемых элементов содержания объектов виртуальной реальности. Проектирование макета пользовательского интерфейса. Основные компоненты приложения дополненной реальности. Предназначение AR Camera. Понятие тестирования программного обеспечения. Основные компоненты содержит в себе набор метрик.

1.2. Методические материалы по подготовке к лабораторным занятиям по дисциплине (модулю)

Лабораторные занятия – одна из форм учебного занятия, направленная на развитие самостоятельности обучающихся и приобретение умений и навыков. Данные учебные занятия углубляют, расширяют, детализируют полученные ранее знания. Лабораторное занятие предполагает выполнение обучающимися по заданию и под руководством преподавателей одной или нескольких работ.

Цель лабораторных занятий состоит в развитии познавательных способностей, самостоятельности мышления и творческой активности обучающихся; углублении, расширении, детализировании знаний, полученных на лекции в обобщенной форме, и содействии выработке навыков профессиональной деятельности. В отдельных случаях на лабораторных занятиях руководителем занятия сообщаются дополнительные знания.

Для достижения поставленных целей и решения требуемого перечня задач лабораторные занятия проводятся традиционными технологиями или с использованием активных и интерактивных образовательных технологий.

При подготовке и работе во время проведения лабораторных занятий следует обратить внимание на следующие моменты: на процесс предварительной подготовки, на работу во время занятия, обработку полученных результатов, исправление полученных замечаний.

Предварительная подготовка к лабораторному занятию заключается в изучении теоретического материала в отведенное для самостоятельной работы время, ознакомление с инструктивными материалами с целью осознания задач занятия, техники безопасности при работе с приборами, веществами.

Работа во время проведения занятия включает:

- консультирование студентов преподавателями и вспомогательным персоналом с целью предоставления исчерпывающей информации, необходимой для самостоятельного выполнения предложенных преподавателем задач.
- самостоятельное выполнение заданий согласно обозначенной рабочей программой дисциплины (модуля) тематики.

2. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ) И САМОСТОЯТЕЛЬНОЙ РАБОТЕ

Освоение обучающимся дисциплины (модуля) «Технологии разработки виртуальной и дополненной реальности» предполагает изучение материалов дисциплины на аудиторных занятиях и в ходе самостоятельной работы. Аудиторные занятия проходят в форме лекций, семинаров, практических и лабораторных занятий. Самостоятельная работа включает разнообразный комплекс видов и форм работы обучающихся.

Для успешного освоения дисциплины (модуля) и достижения поставленных целей необходимо внимательно ознакомиться с рабочей программой дисциплины (модуля), доступной в электронной информационно-образовательной среде РГСУ.

Следует обратить внимание на списки основной и дополнительной литературы, на предлагаемые преподавателем ресурсы информационно-телекоммуникационной сети Интернет. Эта информация необходима для самостоятельной работы обучающегося.

При подготовке к аудиторным занятиям необходимо помнить особенности каждой формы его проведения.

Подготовка к учебному занятию лекционного типа заключается в следующем.

С целью обеспечения успешного обучения обучающийся должен готовиться к лекции, поскольку она является важнейшей формой организации учебного процесса, поскольку:

- знакомит с новым учебным материалом;
- разъясняет учебные элементы, трудные для понимания;
- систематизирует учебный материал;

- ориентирует в учебном процессе.
- С этой целью:
- внимательно прочитайте материал предыдущей лекции;
 - ознакомьтесь с учебным материалом по учебнику и учебным пособиям с темой прочитанной лекции;
 - внесите дополнения к полученным ранее знаниям по теме лекции на полях лекционной тетради;
 - запишите возможные вопросы, которые вы зададите лектору на лекции по материалу изученной лекции;
 - постарайтесь уяснить место изучаемой темы в своей подготовке;
 - узнайте тему предстоящей лекции (по тематическому плану, по информации лектора) и запишите информацию, которой вы владеете по данному вопросу.

Подготовка к лабораторному занятию.

При подготовке и работе во время проведения лабораторных занятий следует обратить внимание на следующие моменты: на процесс предварительной подготовки, на работу во время занятия, обработку полученных результатов, исправление полученных замечаний.

Предварительная подготовка к лабораторному занятию заключается в изучении теоретического материала в отведенное для самостоятельной работы время, ознакомление с инструктивными материалами с целью осознания задач занятия, техники безопасности при работе с приборами, веществами.

Работа во время проведения занятия включает:

- консультирование студентов преподавателями и вспомогательным персоналом с целью предоставления исчерпывающей информации, необходимой для самостоятельного выполнения предложенных преподавателем задач.
- самостоятельное выполнение заданий согласно обозначенной рабочей программой дисциплины (модуля) тематики.

Самостоятельная работа.

Самостоятельная работа - планируемая учебная, учебно-исследовательская, научно-исследовательская работа студентов, выполняемая во внеаудиторное (аудиторное) время по заданию и при методическом руководстве преподавателя, но без его непосредственного участия (при частичном непосредственном участии преподавателя, оставляющем ведущую роль за работой студентов).

Для более углубленного изучения темы задания для самостоятельной работы рекомендуется выполнять параллельно с изучением данной темы. При выполнении заданий по возможности используйте наглядное представление материала.

Самостоятельная работа студентов в ВУЗе является важным видом учебной и научной деятельности студента. Самостоятельная работа студентов играет значительную роль в рейтинговой технологии обучения.

К современному специалисту в области медицины общество предъявляет достаточно широкий перечень требований, среди которых немаловажное значение имеет наличие у выпускников определенных навыков (компетенций) и умения самостоятельно добывать знания из различных источников, систематизировать полученную информацию, давать оценку конкретной профессиональной ситуации. Формирование такого умения происходит в течение всего периода обучения.

В процессе самостоятельной работы студент приобретает необходимые для будущей специальности компетенции, навыки самоорганизации, самоконтроля, самоуправления, саморефлексии и становится активным самостоятельным субъектом учебной деятельности.

Самостоятельная работа студентов должна оказывать важное влияние на формирование личности будущего специалиста, его компетентность. Каждый студент самостоятельно определяет режим своей работы и меру труда, затрачиваемого на овладение учебным содержанием по каждой дисциплине. Он выполняет внеаудиторную работу по личному индивидуальному плану, в зависимости от его подготовки, времени и других условий.

Виды самостоятельной работы.

Работа с литературой.

При работе с книгой необходимо подобрать литературу, научиться правильно ее читать, вести записи. Для подбора литературы в библиотеке используются алфавитный и систематический каталоги. Важно помнить, что рациональные навыки работы с книгой - это всегда большая экономия времени и сил. Правильный подбор учебников рекомендуется преподавателем, читающим лекционный курс. Необходимая литература может быть также указана в методических разработках по данному курсу. Изучая материал по учебнику, следует переходить к следующему вопросу только после правильного уяснения предыдущего, описывая на бумаге все выкладки и вычисления (в том числе те, которые в учебнике опущены или на лекции даны для самостоятельного вывода). При изучении любой дисциплины большую и важную роль играет самостоятельная индивидуальная работа. Особое внимание следует обратить на определение основных понятий курса. Студент должен подробно разбирать примеры, которые поясняют такие определения, и уметь строить аналогичные примеры самостоятельно. Нужно добиваться точного представления о том, что изучаешь. Полезно составлять опорные конспекты. При изучении материала по учебнику полезно в тетради (на специально отведенных полях) дополнять конспект лекций. Там же следует отмечать вопросы, выделенные студентом для консультации с преподавателем. Выводы, полученные в результате изучения, рекомендуется в конспекте выделять, чтобы они при перечитывании записей лучше запоминались. Опыт показывает, что многим студентам помогает составление листа опорных сигналов, содержащего важнейшие и наиболее часто употребляемые формулы и понятия. Такой лист помогает запомнить формулы, основные положения лекции, а также может служить постоянным справочником для студента. Различают два вида чтения: первичное и вторичное. Первичное - это внимательное, неторопливое чтение, при котором можно остановиться на трудных местах.

Задача вторичного чтения - полное усвоение смысла целого (по счету это чтение может быть и не вторым, а третьим или четвертым). Самостоятельная работа с учебниками и книгами (а также самостоятельное теоретическое исследование проблем, обозначенных преподавателем на лекциях) – это важнейшее условие формирования у себя научного способа познания.

Чтение научного текста является частью познавательной деятельности. Ее цель – извлечение из текста необходимой информации. От того на сколько осознанно читающим собственная внутренняя установка при обращении к печатному слову (найти нужные сведения, усвоить информацию полностью или частично, критически проанализировать материал и т.п.) во многом зависит эффективность осуществляемого действия.

Методические рекомендации по составлению конспекта:

1. Внимательно прочитайте текст. Уточните в справочной литературе непонятные слова. При записи не забудьте вынести справочные данные на поля конспекта;
2. Выделите главное, составьте план;
3. Кратко сформулируйте основные положения текста, отметьте аргументацию автора;
4. Законспектируйте материал, четко следуя пунктам плана. При конспектировании старайтесь выразить мысль своими словами. Записи следует вести четко, ясно.

5. Грамотно записывайте цитаты. Цитируя, учитывайте лаконичность, значимость мысли. В тексте конспекта желательно приводить не только тезисные положения, но и их доказательства. При оформлении конспекта необходимо стремиться к емкости каждого предложения. Мысли автора книги следует излагать кратко, заботясь о стиле и выразительности написанного. Число дополнительных элементов конспекта должно быть логически обоснованным, записи должны распределяться в определенной последовательности, отвечающей логической структуре произведения. Для уточнения и дополнения необходимо оставлять поля. Овладение навыками конспектирования требует от студента целеустремленности, повседневной самостоятельной работы.

Методические материалы по самостоятельному решению задач

При самостоятельном решении задач нужно обосновывать каждый этап решения, исходя из теоретических положений курса. Если студент видит несколько путей решения проблемы (задачи), то нужно сравнить их и выбрать самый рациональный. Полезно до начала вычислений составить краткий план решения проблемы (задачи). Решение проблемных задач или примеров следует излагать подробно, вычисления располагать в строгом порядке, отделяя вспомогательные вычисления от основных. Решения при необходимости нужно сопровождать комментариями, схемами, чертежами и рисунками. Следует помнить, что решение каждой учебной задачи должно доводиться до окончательного логического ответа, которого требует условие, и по возможности с выводом. Полученный ответ следует проверить способами, вытекающими из существа данной задачи. Полезно также (если возможно) решать несколькими способами и сравнить полученные результаты. Решение задач данного типа нужно продолжать до приобретения твердых навыков в их решении.

Методические материалы к выполнению реферата

Реферат (от лат. *refere* – сообщать) – краткое изложение в письменном виде или в форме публикации доклада, содержания научного труда (трудов), литературы по теме. Работа над рефератом условно разделяется на выбор темы, подбор литературы, подготовку и защиту плана; написание теоретической части и всего текста с указанием библиографических данных используемых источников, подготовку доклада, выступление с ним. Тематика рефератов полностью связана с основными вопросами изучаемого курса.

Список литературы к темам не дается, и обучающиеся самостоятельно ведут библиографический поиск, причем им не рекомендуется ограничиваться университетской библиотекой.

Важно учитывать, что написание реферата требует от обучающихся определенных усилий. Особое внимание следует уделить подбору литературы, методике ее изучения с целью отбора и обработки собранного материала, обоснованию актуальности темы и теоретического уровня обоснованности используемых в качестве примеров фактов какой-либо деятельности.

Выбрав тему реферата, начав работу над литературой, необходимо составить план. Изучая литературу, продолжается обдумывание темы, осмысливание прочитанного, делаются выписки, сопоставляются точки зрения разных авторов и т.д. Реферативная работа сводится к тому, чтобы в ней выделились две взаимосвязанные стороны: во-первых, ее следует рассматривать как учебное задание, которое должен выполнить обучаемый, а во-вторых, как форму научной работы, творческого воображения при выполнении учебного задания.

Наличие плана реферата позволяет контролировать ход работы, избежать формального переписывания текстов из первоисточников.

Оформление реферата включает титульный лист, оглавление и краткий список использованной литературы. Список использованной литературы размещается на последней странице рукописи или печатной форме реферата. Реферат выполняется в письменной или печатной форме на белых листах формата А4 (210 x 297 мм). Шрифт Times New Roman, кегель 14, через 1,5 интервала при соблюдении следующих размеров текста: верхнее поле – 25 мм, нижнее – 20 мм, левое – 30 мм, правое – 15 мм. Нумерация страниц производится сверху листа, по центру. Титульный лист нумерации не подлежит.

Рефераты должны быть написаны простым, ясным языком, без претензий на наукообразность. Следует избегать сложных грамматических оборотов, непривычных терминов и символов. Если же такие термины и символы все-таки приводятся, то необходимо разъяснять их значение при первом упоминании в тексте реферата.

Объем реферата предполагает тщательный отбор информации, необходимой для краткого изложения вопроса. Важнейший этап – редактирование готового текста реферата и подготовка к обсуждению. Обсуждение требует хорошей ориентации в материале темы, умения выделить главное, поставить дискуссионный вопрос, привлечь внимание слушателей к интересной литературе, логично и убедительно изложить свои мысли.

Рефераты обязательно подлежат защите. Процедура защиты начинается с определения оппонентов защищающего свою работу. Они стремятся дать основательный анализ работы

обучающимся, обращают внимание на положительные моменты и недостатки реферата, дают общую оценку содержанию, форме преподнесения материала, характеру использованной литературы. Иногда они дополняют тот или иной раздел реферата. Последнее особенно ценно, ибо говорит о глубоком знании обучающимся-оппонентом изучаемой проблемы.

Обсуждение не ограничивается выслушиванием оппонентов. Другие обучающиеся имеют право уточнить или опровергнуть какое-либо утверждение. Преподаватель предлагает любому обучающемуся задать вопрос по существу доклада или попытаться подвести итог обсуждению.

Алгоритм работы над рефератом

1. Выбор темы

Тема должна быть сформулирована грамотно (с литературной точки зрения);

В названии реферата следует поставить четкие рамки рассмотрения темы;

Желательно избегать слишком длинных названий;

Следует по возможности воздерживаться от использования в названии спорных с научной точки зрения терминов, излишней наукообразности, а также чрезмерного упрощения формулировок.

2. Реферат следует составлять из пяти основных частей: введения; основной части; заключения; списка литературы; приложений.

3. Основные требования к введению:

Во введении не следует концентрироваться на содержании; введение должно включать краткое обоснование актуальности темы реферата, где требуется показать, почему данный вопрос может представлять научный интерес и есть ли связь представляемого материала с современностью. Таким образом, тема реферата должна быть актуальна либо с научной точки зрения, либо с современных позиций.

Очень важно выделить цель, а также задачи, которые требуется решить для выполнения цели.

Введение должно содержать краткий обзор изученной литературы, в котором указывается взятый из того или иного источника материал, кратко анализируются изученные источники, показываются их сильные и слабые стороны;

Объем введения составляет две страницы текста.

4. Требования к основной части реферата:

Основная часть содержит материал, отобранный для рассмотрения проблемы;

Также основная часть должна включать в себя собственно мнение обучающихся и сформулированные самостоятельные выводы, опирающиеся на приведенные факты;

Материал, представленный в основной части, должен быть логически изложен и распределен по параграфам, имеющим свои названия;

В изложении основной части необходимо использовать сноски (в первую очередь, когда приводятся цифры и чьи-то цитаты);

Основная часть должна содержать иллюстративный материал (графики, таблицы и т. д.);

Объем основной части составляет около 10 страниц.

5. Требования к заключению:

В заключении формулируются выводы по параграфам, обращается внимание на выдвинутые во введении задачи и цели;

Заключение должно быть четким, кратким, вытекающим из содержания основной части.

6. Требования к оформлению списка литературы (по ГОСТу):

Необходимо соблюдать правильность последовательности записи источников: сначала следует писать фамилию, а после инициалы; название работы не ставится в кавычки; после названия сокращенно пишется место издания; затем идет год издания; наконец, называется процитированная страница.

Критерии оценки реферата

Обучающийся, защищающий реферат, должен рассказать о его актуальности, поставленных целях и задачах, изученной литературе, структуре основной части, сделанных в ходе работы выводах.

По окончании выступления ему может быть задано несколько вопросов по представленной проблеме.

Оценка складывается из соблюдения требований к реферату, грамотного раскрытия темы, умения четко рассказывать о представленном реферате, способности понять суть задаваемых по работе вопросов и найти точные ответы на них.

Методические материалы к выполнению эссе

Эссе – литературное произведение небольшого объема, обычно прозаическое, свободной композиции, передающее индивидуальные впечатления, суждения, соображения автора о той или иной проблеме, теме, о том или ином событии или явлении. Это вид самостоятельной исследовательской работы обучающихся, с целью углубления и закрепления теоретических знаний и освоения практических навыков. Цель эссе состоит в развитии самостоятельного творческого мышления и письменного изложения собственных мыслей. При написании эссе обучающийся должен представить развернутый письменный ответ на теоретический или практический актуальный вопрос, объявленный преподавателем в аудитории непосредственно перед ее написанием. В процессе написания эссе разрешается пользоваться нормативно-правовыми актами, конспектом лекций (в печатном виде). Использование интернет-ресурсов не допускается. Темы эссе преподаватель предлагает из числа тех, которые обучающиеся уже рассматривали на лекциях или семинарских занятиях, исходя из содержания заданий в составе оценочных средств. По решению преподавателя, в качестве темы эссе может быть выбрана одна или несколько тем, которые могут быть распределены между обучающимися по желанию.

Требования к выполнению эссе:

1. Проводится письменно.
2. Эссе выполняется на компьютере (гарнитура Times New Roman, шрифт 14) через 1,5 интервала с полями: верхнее, нижнее – 2; правое – 3; левое – 1,5. Отступ первой строки абзаца – 1,25. Сноски – постраничные. Таблицы и рисунки встраиваются в текст работы. При этом обязательный заголовок таблицы надо размещать над табличным полем, а рисунки сопровождать подрисовочными подписями. При включении в эссе нескольких таблиц и/или рисунков их нумерация обязательна. Обязательна и нумерация страниц. Их целесообразно проставлять внизу страницы – по середине или в правом углу. Номер страницы не ставится на титульном листе, но в общее число страниц он включается. Объем эссе, без учета приложений, не должен превышать 5 страниц. Значительное превышение установленного объема является недостатком работы и указывает на то, что обучающийся не сумел отобрать и переработать необходимый материал.

3. Работа должна содержать собственные умозаключения по сути поставленной проблемы, включать самостоятельно проведенный анализ по сути этой проблемы, выводы, обобщающие авторскую позицию по поставленной проблеме.

Критерии оценки эссе:

«Отлично» – исключительные знания материала, абсолютное понимание сути, безукоризненное знание основных понятий и положений, логически и лексически грамотно изложенный, содержательный, аргументированный, конкретный и исчерпывающий ответ.

«Хорошо» – глубокие знания материала, правильное понимание сути, знание основных понятий и положений, содержательный, полный и конкретный ответ.

«Удовлетворительно» – твердые, но недостаточно полные знания, верное понимание сути, в целом правильный ответ.

«Неудовлетворительно» – непонимание сущности задания, грубые ошибки в ответе.

Методические материалы по выполнению тестирования.

Тестовые задания содержат вопросы и 3-4 варианта ответа по базовым положениям изучаемой темы/раздела, составлены с расчетом на знания, полученные обучающимся в процессе изучения темы/раздела.

Тестовые задания выполняются в письменной форме и сдаются преподавателю, ведущему дисциплину (модуль). На выполнение тестовых заданий обучающимся отводится 45 минут.

При обработке результатов оценочной процедуры используются: критерии оценки по содержанию и качеству полученных ответов, ключи, оценочные листы.

Критерии оценки теста:

«Зачтено» - если обучающийся правильно выполнил не менее 2/3 всей работы или допустил не более одной грубой ошибки и двух недочетов, не более одной грубой и одной негрубой ошибки, не более трех негрубых ошибок, одной негрубой ошибки и трех недочетов, при наличии четырех-пяти недочетов.

«Не зачтено» - если число ошибок и недочетов превысило норму для оценки 3 или правильно выполнено менее 2/3 всей работы.

Методические материалы по выполнению доклада.

Рекомендуется следующая структура доклада:

1. титульный лист, содержание доклада;
2. краткое изложение;
3. цели и задачи;
4. изложение характера исследований и рассмотренных проблем, гипотезы, спорные вопросы;
5. источники информации, методы сбора и анализа данных, степень их полноты и достоверности;
6. анализ и толкование полученных в работе результатов;
7. выводы и оценки;
8. библиография и приложения.

Время выступления докладчика не должно превышать 10 минут.

Основные требования к оформлению доклада:

- титульный лист должен включать название доклада, наименование предметной (цикловой) комиссии, фамилию обучающегося;
- все использованные литературные источники сопровождаются библиографическим описанием;
- приводимая цитата из источника берется в кавычки (оформляются сноски);
- единицы измерения должны применяться в соответствии с действующими стандартами;
- все названия литературных источников следует приводить в соответствии с новейшими изданиями;
- рекомендуется включение таблиц, графиков, схем, если они отражают основное содержание или улучшают ее наглядность;
- названия фирм, учреждений, организаций и предприятий должны именоваться так, как они указываются в источнике;

Критерии оценки доклада

При выполнении доклада обучающийся должен продемонстрировать умение кратко излагать прочитанный материал, а также умение обобщать и анализировать материал по теме доклада.

Презентация

Методические материалы к презентациям

1. Объем презентации 10 -20 слайдов.
2. На титульном слайде должно быть отражено:
 - наименование факультета;
 - тема презентации;

- фамилия, имя, отчество, направление подготовки/ специальность, направленность (профиль)/ специализация, форма обучения, номер группы автора презентации;
- фамилия, имя, отчество, степень, звание, должность руководитель работы;
- год выполнения работы.

3. В презентации должны быть отражены обоснование актуальности представляемого материала, цели и задачи работы.

4. Содержание презентации должно включать наиболее значимый материал доклада, а также, при необходимости, таблицы, диаграммы, рисунки, фотографии, карты, видео – вставки, звуковое сопровождение.

5. Заключительный слайд должен содержать информацию об источниках информации для презентации.

Критерии оценки презентации

1. Объём презентации 10 -20 слайдов.
2. Правильность оформления титульного слайда.
3. Актуальность отобранного материала, обоснованность формулировки цели и задач работы.

4. Наглядность и логичность презентации, обоснованность использования таблиц, диаграмм, рисунков, фотографий, карт, видео – вставок, звукового сопровождения; правильный выбор шрифтов, фона, других элементов дизайна слайда.

5. Объём и качество источников информации (не менее 2-х интернет – источников и не менее 2-х литературных источников).

Методические материалы по подготовке к опросу

Самостоятельная работа обучающихся включает подготовку к опросу на практических занятиях. Для этого обучающийся изучает лекции, основную и дополнительную литературу, публикации, информацию из Интернет-ресурсов.

Тема и вопросы к практическим занятиям, вопросы для самоконтроля содержатся в рабочей программе и доводятся до студентов заранее.

Для подготовки к опросу обучающемуся необходимо ознакомиться с материалом, посвященным теме практического занятия, в учебнике или другой рекомендованной литературе, конспекте лекции, обратить внимание на усвоение основных понятий дисциплины, выявить неясные вопросы и подобрать дополнительную литературу для их освещения.

Критерии оценки опроса

«Отлично»:

- дан полный, развернутый ответ на поставленный вопрос;
- в ответе прослеживается четкая структура, логическая последовательность, отражающая сущность раскрываемых понятий, теорий, явлений;
- знание по предмету демонстрируется на фоне понимания его в системе данной науки и междисциплинарных связей;
- свободное владение терминологией;
- ответы на дополнительные вопросы четкие, краткие;

«Хорошо»:

- дан полный, развернутый ответ на поставленный вопрос, показано умение выделять существенные и несущественные признаки, причинно-следственные связи;
- ответ недостаточно логичен с единичными ошибками в частностях, исправленные студентом с помощью преподавателя;
- единичные ошибки в терминологии;
- ответы на дополнительные вопросы правильные, недостаточно полные и четкие.

«Удовлетворительно»:

- ответ не полный, с ошибками в деталях, умение раскрыть значение обобщённых знаний не показано, речевое оформление требует поправок, коррекции;

– логика и последовательность изложения имеют нарушения, студент не способен самостоятельно выделить существенные и несущественные признаки и причинно-следственные связи;

- ошибки в раскрываемых понятиях, терминах;
- студент не ориентируется в теме, допускает серьезные ошибки;
- студент не может ответить на большую часть дополнительных вопросов.

«Неудовлетворительно»:

– ответ представляет собой разрозненные знания с существенными ошибками по вопросу;

– присутствуют фрагментарность, нелогичность изложения, студент не осознает связь обсуждаемого вопроса с другими объектами дисциплины, речь неграмотная;

- незнание терминологии;
- ответы на дополнительные вопросы неправильные.

Методические материалы по выполнению практического задания

При выполнении практического задания обучающийся придерживается следующего алгоритма:

1. Записать дату, тему и цель задания;
2. Ознакомиться с правилами и условия выполнения практического задания;
3. Повторить теоретические задания, необходимые для рациональной работы и других практических действий, используя конспекты лекций и рекомендованную литературу, представленной в программе;
4. Выполнить работу по предложенному алгоритму действий;
5. Обобщить результаты работы, сформулировать выводы / дать ответы на контрольные вопросы;

Работа должна быть выполнена грамотно, с соблюдением культуры изложения. При использовании данных из учебных, методических пособий и другой литературы, периодических изданий, Интернет-источников должны иметься ссылки на вышеперечисленные.

Критерии оценки практического задания:

«Отлично» – правильный ответ, дается четкое обоснование принятому решению; рассуждения четкие последовательные логические; используются ссылки на полученные при изучении дисциплины знания; правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Хорошо» – правильный ответ, дается обоснование принятому решению; но с не существенными ошибками, в рассуждениях отсутствует логическая последовательность; используются ссылки на полученные при изучении дисциплины знания, правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Удовлетворительно» – правильный ответ, допускаются грубые ошибки в обосновании принятого решения; рассуждения не последовательные сумбурные; используются ссылки на полученные при изучении дисциплины знания; используются формулы, процедуры, понятия, имеющие прямое значение для подтверждения принятого решения, однако, при обращении к ним допускаются серьезные ошибки, студент не может правильно ими воспользоваться.

«Неудовлетворительно, не зачтено» – ответ неверный, отсутствует обоснование принятому решению; студент демонстрирует полное непонимание сути вопроса.

Для оценки решения ситуационной задачи (аналитического задания):

Оценка «отлично» выставляется, если задача решена грамотно, ответы на вопросы сформулированы четко. Эталонный ответ полностью соответствует решению студента, которое хорошо обосновано теоретически.

Оценка «хорошо» выставляется, если задача решена, ответы на вопросы сформулированы не достаточно четко. Решение студента в целом соответствует эталонному ответу, но не достаточно хорошо обосновано теоретически.

Оценка «удовлетворительно» выставляется, если задача решена не полностью, ответы не содержат всех необходимых обоснований решения.

Оценка «неудовлетворительно» выставляется, если задача не решена или имеет грубые теоретические ошибки в ответе на поставленные вопросы.

Методические материалы по выполнению лабораторного задания

При выполнении лабораторного задания обучающийся придерживается следующего алгоритма:

1. Записать дату, тему и цель задания;
2. Ознакомиться с правилами и условия выполнения задания;
3. Повторить теоретические задания, необходимые для рациональной работы и других практических действий, используя конспекты лекций и рекомендованную литературу, представленной в программе;
4. Выполнить работу по предложенному алгоритму действий;
5. Обобщить результаты работы, сформулировать выводы / дать ответы на контрольные вопросы;

Работа должна быть выполнена грамотно, с соблюдением культуры изложения. При использовании данных из учебных, методических пособий и другой литературы, периодических изданий, Интернет-источников должны иметься ссылки на вышеперечисленные.

Критерии оценки лабораторного задания:

«Отлично» – правильный ответ, дается четкое обоснование принятому решению; рассуждения четкие последовательные логические; используются ссылки на полученные при изучении дисциплины знания; правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Хорошо» – правильный ответ, дается обоснование принятому решению; но с не существенными ошибками, в рассуждениях отсутствует логическая последовательность; используются ссылки на полученные при изучении дисциплины знания, правильно используются формулы, понятия, процедуры, имеющие прямое отношение к задаче для подтверждения принятого решения.

«Удовлетворительно» – правильный ответ, допускаются грубые ошибки в обосновании принятого решения; рассуждения не последовательные сумбурные; используются ссылки на полученные при изучении дисциплины знания; используются формулы, процедуры, понятия, имеющие прямое значение для подтверждения принятого решения, однако, при обращении к ним допускаются серьезные ошибки, студент не может правильно ими воспользоваться.

«Неудовлетворительно, не зачтено» – ответ неверный, отсутствует обоснование принятому решению; студент демонстрирует полное непонимание сути вопроса.

Методические указания для подготовки к промежуточной аттестации.

Изучение учебных дисциплин (модулей) завершается зачетом/зачетом с оценкой или экзаменом. Подготовка к промежуточной аттестации способствует закреплению, углублению и обобщению знаний, получаемых, в процессе обучения, а также применению их к решению практических задач. Готовясь к экзамену, студент ликвидирует имеющиеся пробелы в знаниях, углубляет, систематизирует и упорядочивает свои знания. На зачете или экзамене студент демонстрирует то, что он освоил в процессе обучения по дисциплине (модулю).

Вначале следует просмотреть весь материал по дисциплине (модулю), отметить для себя трудные вопросы. Обязательно в них разобраться. В заключение еще раз целесообразно повторить основные положения, используя при этом листы опорных сигналов. Систематическая подготовка к занятиям в течение семестра позволит использовать время промежуточной аттестации для систематизации знаний.

3. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНИВАНИЯ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ И (ИЛИ) ОПЫТА ДЕЯТЕЛЬНОСТИ, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ

3.1. Организационные основы применения балльно-рейтинговой системы оценки успеваемости обучающихся по дисциплине (модулю)

Оценка качества освоения обучающимися дисциплины (модуля) реализуется в формате балльно-рейтинговой системы оценки успеваемости обучающихся (БРСО).

БРСО в ходе текущего контроля успеваемости и промежуточной аттестации осуществляется по 100-балльной шкале.

Академический рейтинг обучающегося по дисциплине (модулю) складывается из результатов:

- текущего контроля успеваемости (максимальный текущий рейтинг обучающегося 80 рейтинговых баллов);
- промежуточной аттестации (максимальный рубежный рейтинг обучающегося 20 рейтинговых баллов).

Условия оценки освоения обучающимся дисциплины (модуля) в формате БРСО доводятся преподавателем до сведения обучающихся на первом учебном занятии, а также размещены в свободном доступе в электронной информационно-образовательной среде Университета.

3.2. Проведение текущего контроля успеваемости обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося

В течение учебного семестра до промежуточной аттестации на основании утвержденной рабочей программы дисциплины (модуля) формируется текущий рейтинг обучающегося. Текущий рейтинг обучающегося складывается как сумма рейтинговых баллов, полученных им в течение учебного семестра по всем видам учебных занятий по учебной дисциплине.

В процессе текущего контроля оцениваются следующие действия обучающегося, направленные на освоение компетенций в рамках изучения учебной дисциплины:

- академическая активность (посещаемость учебных занятий, самостоятельное изучение содержания учебной дисциплины в электронной информационно-образовательной среде, соблюдение сроков сдачи практических заданий и текущих контрольных мероприятий и др.);
- выполнение и сдача текущих и итогового практических заданий (эссе, рефераты, творческие задания, кейс-задания, лабораторные работы, расчетные задания и др., активное участие в групповых интерактивных занятиях (дискуссии, Wiki-проекты и др.), защита проектов и др.);
- прохождение рубежей текущего контроля, включая соблюдение графика их прохождения в электронной информационно-образовательной среде.

Для планирования расчета текущего рейтинга обучающегося используются следующие пропорции:

Вид учебного действия	Максимальная рейтинговая оценка, баллов
академическая активность	10
практические задания	40
<i>из них: текущие практические задания</i>	20
<i>итоговое практическое задание</i>	20
рубежи текущего контроля	30
ИТОГО:	80

В течение учебного семестра по дисциплине (модулю) обучающимся должен быть накоплен текущий рейтинг не менее 52 рейтинговых баллов (65% от максимального значения текущего рейтинга).

Необходимыми условиями допуска обучающегося к промежуточной аттестации по дисциплине являются положительное прохождение обучающимся не менее 65% рубежей текущего контроля с накоплением не менее 65% максимального рейтингового балла за каждый рубеж текущего контроля и положительное выполнение итогового практического задания с накоплением не менее 65% максимального рейтингового балла, установленного за итоговое практическое задание.

Невыполнение вышеуказанных условий является текущей академической задолженностью, которая должна быть ликвидирована обучающимся до контрольного мероприятия промежуточной аттестации.

Сведения о наличии у обучающихся текущей академической задолженности, сроках и порядке добора рейтинговых баллов для её ликвидации доводятся до обучающихся педагогическим работником.

В случае неликвидации текущей академической задолженности, педагогический работник обязан во время контрольного мероприятия промежуточной аттестации поставить обучающемуся 0 рейтинговых баллов. В этом случае ликвидация текущей академической задолженности возможна в периоды проведения повторной промежуточной аттестации.

3.3. Проведение промежуточной аттестации обучающихся по дисциплине (модулю) в соответствии с балльно-рейтинговой системой оценки успеваемости обучающегося

Промежуточная аттестация по дисциплине (модулю) проводится в соответствии с Положением о промежуточной аттестации обучающихся по основным профессиональным образовательным программам высшего образования – программ специалитета в Российском государственном социальном университете и Положением о балльно-рейтинговой системе оценки успеваемости обучающихся по основным профессиональным образовательным программам высшего образования - программам бакалавриата, программам специалитета, программам магистратуры в Российском государственном социальном университете.

На промежуточную аттестацию отводится 20 рейтинговых баллов.

Ответы обучающегося на контрольном мероприятии промежуточной аттестации оцениваются педагогическим работником по 20 - балльной шкале, а итоговая оценка по дисциплине (модулю) выставляется по пятибалльной системе для экзамена.

Критерии выставления оценки определяются Положением о балльно-рейтинговой системе оценки успеваемости обучающихся по основным профессиональным образовательным программам высшего образования - программам бакалавриата, программам специалитета, программам магистратуры в Российском государственном социальном университете.

В процессе определения рубежного рейтинга обучающегося используется следующая шкала:

Рубежный рейтинг	Критерии оценки освоения обучающимся учебной дисциплины в ходе контрольных мероприятий промежуточной аттестации
19-20 рейтинговых баллов	обучающийся глубоко и прочно усвоил программный материал, исчерпывающе, последовательно, грамотно и логически стройно его излагает, тесно увязывает с задачами и будущей деятельностью, не затрудняется с ответом при видоизменении задания, свободно справляется с задачами и практическими заданиями, правильно обосновывает принятые решения, умеет самостоятельно обобщать и излагать материал, не допуская
16-18 рейтинговых баллов	обучающийся твердо знает программный материал, грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос, может правильно применять теоретические положения и владеет необходимыми умениями и навыками при выполнении практических

13-15 рейтинговых баллов	обучающийся освоил основной материал, но не знает отдельных деталей, допускает неточности, недостаточно правильные формулировки, нарушает последовательность в изложении программного материала и испытывает затруднения в выполнении практических заданий
1-12 рейтинговых баллов	обучающийся не знает значительной части программного материала, допускает существенные ошибки, с большими затруднениями выполняет практические задания
0 рейтинговых баллов	не аттестован

Если результат контроля успеваемости в рамках проведения контрольных мероприятий промежуточной аттестации (рубежный рейтинг обучающегося) неудовлетворительный (получено менее 13 рейтинговых баллов), то промежуточная аттестация по учебной дисциплине (модулю) невозможна даже при наличии высокого текущего рейтинга, полученного по итогам текущего контроля по учебной дисциплине (модулю).

ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

№ п/п	Содержание изменения	Реквизиты документа об утверждении изменения	Дата введения изменения
1.			
2.			
3.			
4.			