МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ТЕХНОЛОГИЙ И УПРАВЛЕНИЯ ИМЕНИ К.Г. РАЗУМОВСКОГО (ПЕРВЫЙ КАЗАЧИЙ УНИВЕРСИТЕТ)»
Смоленский казачий институт промышленных технологий и бизнеса (филиал)

УТВЕРЖДАЮ
Председатель ИЦК
_____Максаков С. А.
«10» апреля 2026 г.

Рабочая программа дисциплины
МДК.01.02 Базы данных

Закреплена за **ПЦК общепрофессиональных дисциплин; профессиональных модулей специальности 10.02.05 Обеспечение информационной безопасности автоматизированных систем**

Специальность: **10.02.05 ОБЕСПЕЧЕНИЕ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ АВТОМАТИЗИРОВАННЫХ СИСТЕМ**

Направленность

на базе основного общего образования

Квалификация **техник по защите информации**

Форма обучения **очная**

Общая трудоемкость **140 часов**

Часов по учебному плану 140

в том числе:

контактная работа 128

самостоятельная работа 12

Виды контроля в семестрах:

Контрольная работа - 4,5

Распределение часов дисциплины по семестрам

Вид учебной работы	Трудоемкость, часов					
	семестр 4		**семестр 5**		Итого	
	УП	РП	УП	РП	УП	РП
Лекции	44	44	14	14	58	58
Лабораторные	20	20	28	28	48	48
Практические	22	22			22	22
В т.ч. в форме практ. подгот.	0	0	0	0	0	0
Контактная работа	86	86	42	42	128	128
Сам. работа	4	4	8	8	12	12
Часы на контроль						

ДОКУМЕНТ ПОДПИСАН ЭЛЕКТРОННОЙ ПОДПИСЬЮ
Сертификат: 00F5 665F CCDB 5046 05DD 6E08 EBCB D00A DC
Действителен: с 21 октября 2025 г. по 14 января 2027 г.
Владелец: Коржикова Надежда Сергеевна
Должность: Директор

Москва, 2026

Программу составил:
Преподаватель Максаков Сергей Анатольевич _____

Рабочая программа дисциплины
Базы данных

Разработана в соответствии с:
Федеральный государственный образовательный стандарт среднего профессионального образования по специальности 10.02.05 ОБЕСПЕЧЕНИЕ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ АВТОМАТИЗИРОВАННЫХ СИСТЕМ (приказ Минобрнауки России от 09.12.2016 г. № 1553)
06.033. Профессиональный стандарт "СПЕЦИАЛИСТ ПО ЗАЩИТЕ ИНФОРМАЦИИ В АВТОМАТИЗИРОВАННЫХ СИСТЕМАХ", утверждённый приказом Министерства труда и социальной защиты Российско Федерации от 14 сентября 2022 г. N 525н (зарегистрирован Министерством юстиции Российской Федерации 14 октября 2022 г., регистрационный N 70543)

Разработана на основании учебного плана, утвержденного Учёным советом ФГБОУ ВО "МГУТУ им. К.Г. Разумовского (ПКУ)" от 26.03.2026 протокол №9.

СОДЕРЖАНИЕ

1. МЕСТО ДИСЦИПЛИНЫ (УЧЕБНОГО ПРЕДМЕТА) В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

2. ЦЕЛИ, ЗАДАЧИ И ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОСВОЕНИЯ ДИСЦИПЛИНЫ (УЧЕБНОГО ПРЕДМЕТА)

3. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ (УЧЕБНОГО ПРЕДМЕТА)

4. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДИСЦИПЛИНЫ (УЧЕБНОГО ПРЕДМЕТА)

5. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (УЧЕБНОГО ПРЕДМЕТА)

6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (УЧЕБНОГО ПРЕДМЕТА)

7. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ (УЧЕБНОГО ПРЕДМЕТА)

8. ОРГАНИЗАЦИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ДЛЯ ЛИЦ С ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ ЗДОРОВЬЯ И ИНВАЛИДОВ

1. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ	
Цикл (раздел) ОП:	П

Дисциплина Базы данных является частью неопознанный объект и обязательна для изучения.

Требования к предварительной подготовке обучающегося:	
1	Информатика
2	Операционные системы
Дисциплины (модули) и практики, для которых освоение данной дисциплины необходимо как предшествующее:	
1	Производственная практика (эксплуатация автоматизированных (информационных) систем в защищенном исполнении)
2	Учебная практика (эксплуатация автоматизированных (информационных) систем в защищенном исполнении)
3	Экзамен по модулю (эксплуатация автоматизированных (информационных) систем в защищенном исполнении)

2. ЦЕЛИ, ЗАДАЧИ И ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОСВОЕНИЯ ДИСЦИПЛИНЫ (УЧЕБНОГО ПРЕДМЕТА)

Цели освоения: Формирование у студентов теоретических знаний и практических навыков проектирования, администрирования и защиты баз данных (БД) как ключевого компонента автоматизированных систем, обеспечивающего целостность, доступность и конфиденциальность информации
Задачи: 1. Изучение моделей структур данных;
2. Знакомство со способами классификации СУБД в зависимости от реализуемых моделей данных и способов их использования;
3. Изучение способов хранения данных на физическом уровне, типов и способов организации файловых систем;
4. Изучение реляционной модели данных и СУБД, реализующих эту модель, языка запросов SQL;
5. Понимание проблем и основных способов их решения при коллективном доступе к данным;
6. Изучение возможностей СУБД, поддерживающих различные модели организации данных, преимуществ и недостатков этих СУБД при реализации различных структур данных
Изучение дисциплины направлено на формирование у обучающихся следующих компетенций:

ПК 1.1.: Производить установку и настройку компонентов автоматизированных (информационных) систем в защищенном исполнении в соответствии с требованиями эксплуатационной документации
ПК 1.1.1: Знать состав и принципы работы автоматизированных систем, операционных систем и сред; принципы разработки алгоритмов программ, основных приемов программирования; модели баз данных; принципы построения, физические основы работы периферийных устройств
ПК 1.1.2: Уметь осуществлять комплектование, конфигурирование, настройку автоматизированных систем в защищенном исполнении и компонент систем защиты информации автоматизированных систем
ПК 1.1.3: Иметь практический опыт установки и настройки компонентов систем защиты информации автоматизированных (информационных) систем

2.3	**Иметь практический опыт:**
2.3.1	ПК 1.1: Иметь навыки установки и настройки компонентов автоматизированных (информационных) систем в защищенном исполнении в соответствии с требованиями эксплуатационной документации.
2.3.2	ПК 1.2: Иметь навыки администрирования программных и программно-аппаратных компонентов автоматизированной (информационной) системы в защищенном исполнении.
2.3.3	ПК 1.3: Иметь навыки обеспечения бесперебойной работы автоматизированных (информационных) систем в защищенном исполнении в соответствии с требованиями эксплуатационной документации.
2.3.4	ПК 1.4: Иметь навыки и опыт деятельности по проверке технического состояния, техническому обслуживанию, текущему ремонту, устранению отказов и восстановлению работоспособности автоматизированных (информационных) систем в защищенном исполнении.

3. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Код занятия	Наименование разделов, тем и содержание занятий /вид занятия/	Семестр / Курс	Часов/ в том числе	Компетен-ции	Виды контроля
	1. Основы теории баз данных				
1. 1	Тема 1.1. Введение в базы данных Основные понятия баз данных. Компоненты системы баз данных. Централизованное управление базами данными, основные требования. Архитектура систем базы данных Знать: основные понятия баз данных (базы данных, системы управления базами данных, данные, метаданные, модель данных); компоненты системы баз данных; принципы	4	4/0	ПК 1.1.1	Устный опрос

	централизации; требования к централизованным БД; архитектуру систем баз данных /Лек/				
1. 2	Практическая работа №1. Основные понятия и архитектуры систем баз данных. Уметь: приводить примеры данных и метаданных из повседневной жизни; отличать модель данных (реляционную, иерархическую, сетевую) по описанию структуры; перечислять компоненты любой информационной системы с хранением данных; рисовать простую блок-схему клиент-серверной архитектуры. Владеть: терминологией БД. /Пр/	4	2/0	ПК 1.1.2,ПК 1.1.3	Отчет по практической работе
1. 3	Тема 1.2. Модели данных Понятие модели данных. Иерархическая модель данных. Сетевая модель данных. Объектно-ориентированная модель данных. Реляционная модель данных. Двенадцать правил Кодда для определения концепции реляционной модели. Постреляционная модель данных Знать: понятие модели данных; виды моделей данных; достоинства и недостатки моделей данных; основные понятия реляционной модели данных /Лек/	4	4/0	ПК 1.1.1	Устный опрос
1. 4	Самостоятельная работа по теме «Модели данных» /СР/	4	2/0	ПК 1.1.1,ПК 1.1.2,ПК 1.1.3	Вопросы для самоподготовки
1. 5	Практическая работа №2. Сравнительный анализ видов моделей данных Уметь: различать основные модели данных; выбирать модель данных под задачу Владеть: терминологическим аппаратом моделей данных (корень, уровень, вершина, дуга, кортеж, атрибут, класс, объект, наследование), инструментарием сравнительного анализа для заполнения таблиц и формулирования выводов о применимости моделей. /Пр/	4	2/0	ПК 1.1.2,ПК 1.1.3	Отчет по практической работе
1. 6	Тема 1.3. Основы реляционной алгебры Основы реляционной алгебры. Традиционные операции над отношениями. Специальные операции над отношениями. Операции над отношениями дополненные Дейтом. Знать: понятие реляционной алгебры; традиционные операции над отношениями /Лек/	4	4/0	ПК 1.1.1	Устный опрос
1. 7	Практическая работа №3. Основы реляционной алгебры: традиционные и специальные операции над отношениями Уметь: выполнять традиционные (объединение, пересечение, разность, декартово произведение) и специальные (выборка, проекция, соединение, деление) операции реляционной алгебры над таблицами Владеть: навыками записи последовательности операций для формулирования запросов, различать совместимость отношений по типу и смыслу, применять дополненные Дейтом операции (полусоединение, антиполусоединение, внешние соединения) и анализировать результат каждой операции в виде результирующей таблицы	4	4/0	ПК 1.1.2,ПК 1.1.3	Отчет по практической работе

	/Пр/				
1. 8	Тема 1.4. Системы управления базами данных (СУБД) Базовые понятия СУБД. Этапы развития СУБД и ведущие производители. Функции, реализуемые в СУБД. Структура СУБД. Интерфейс СУБД. Языковые средства СУБД. Классификация СУБД. Знакомство с СУБД Ms Access Знать: базовые понятия СУБД; этапы развития СУБД; функции СУБД; структуру СУБД; языковые средства СУБД; классификацию СУБД /Лек/	4	4/0	ПК 1.1.1	Устный опрос
1. 9	Тема 1.5. Целостность данных как ключевое понятие баз данных Понятие целостности и непротиворечивости данных. Классификация ограничений целостности. Механизм каскадных изменений. Примеры нарушения целостности и непротиворечивости данных Знать: понятие целостности и непротиворечивости данных; классификация ограничений целостности; механизм каскадных изменений; /Лек/	4	4/0	ПК 1.1.1	Устный опрос
1. 10	Лабораторная работа №1. Создание простейших баз данных с помощью СУБД Ms Access Уметь: создавать и настраивать базу данных; работать с таблицами Ms Access; создавать формы и кнопочное меню Владеть: СУБД Ms Access /Лаб/	4	8/0	ПК 1.1.2,ПК 1.1.3	Отчет по лабораторной работе
	2. Проектирование баз данных				
2. 1	Тема 2.1. Этапы проектирования баз данных. Системный анализ предметной области Основные этапы проектирования баз данных. Анализ предметной области Знать: этапы проектирования баз данных; понятие сущности и связи между ними /Лек/	4	4/0	ПК 1.1.1	Устный опрос
2. 2	Тема 2.2. Инфологическое проектирование баз данных. ER-диаграммы Инфологическое проектирование БД, общие сведения. Требования и компоненты концептуальной модели. ER-диаграмма в нотации Чена. Этапы построения ER-диаграммы. ER-диаграммы в нотациях Баркера, IDEF1X. Переход от ER-диаграмм к реляционным отношениям Знать: понятие инфологического проектирования; понятие и назначение ER-диаграммы; этапы построения ER-диаграммы /Лек/	4	4/0	ПК 1.1.1	Устный опрос
2. 3	Тема 2.3. Даталогическое проектирование Выбор СУБД. Логическое (даталогическое) проектирование, общие сведения. Понятие нормализации. Аномалии вставки, удаления и обновления. Функциональная и многозначная зависимости. Приведение таблицы к нормальным формам Знать: критерии выбора СУБД; понятие логического проектирования; понятие нормализации; понятие функциональной и многозначной зависимости; нормальные формы /Лек/	4	4/0	ПК 1.1.1	Устный опрос
2. 4	Практическая работа №4. Построение ER-диаграмм в различных нотациях Уметь: выполнять инфологическое проектирование предметной области (выделять	4	6/0	ПК 1.1.2,ПК 1.1.3	Отчет по практической работе

	сущности, атрибуты, ключи и связи), строить ER-диаграммы в нотациях Чена, Баркера и IDEF1X (с использованием правильных графических обозначений каждого типа связи и обязательности), перечислять этапы построения ER-диаграммы в правильной последовательности Владеть: программными средствами для построения ER-диаграмм /Пр/				
2. 5	Тема 2.4. Средства автоматизации проектирования CASE-средства, CASE-система и CASE-технология. Классификация CASE-средств. Метод функционального моделирования SADT. Основные элементы IDEF0-диаграмм. DFD-диаграммы. Введение в UML. Виды UML-диаграмм. Диаграммы вариантов использования Знать: понятие CASE-средства; понятие CASE-технологии; классификацию CASE-средств; сущность SADT; элементы IDEF)-модели; сущность и назначение DFD-диаграммы; назначение UML-диаграмм и их виды /Лек/	4	4/0	ПК 1.1.1	Устный опрос
2. 6	Практическая работа №5. IDEF0-диаграммы Уметь: строить функциональные модели в нотации IDEF0 (выделять входы, выходы, управления и механизмы, выполнять декомпозицию, использовать основные типы связей между блоками и туннелирование стрелок), отличать IDEF0 от DFD по назначению и составу элементов, перечислять основные виды UML-диаграмм с их кратким назначением Владеть: инструментами и методологиями моделирования при построении IDEF0-диаграмм /Пр/	4	8/0	ПК 1.1.2,ПК 1.1.3	Отчет по практической работе
2. 7	Лабораторная работа №2. Проектирование реляционной базы данных: от концептуальной модели до физической реализации в СУБД (на примере Microsoft Access) Уметь: применять методологию проектирования реляционных баз данных (выделять сущности, определять ключи, нормализовать структуры до 3НФ, устанавливать связи между таблицами) для создания в СУБД MS Access многотабличных баз данных Владеть: навыками разработки пользовательских форм для ввода данных и построения итоговых отчётов /Лаб/	4	8/0	ПК 1.1.2,ПК 1.1.3	Отчет по лабораторной работе
	3. Организация баз данных				
3. 1	Тема 3.1. Создание базы данных. Манипулирование данными Создание базы данных. Создание и изменение структуры таблиц. Наполнение таблицы данными. Работа с базой данных: восстановление и сжатие. Открытие и модификация данных. Навигация по набору данных Знать: способы восстановления и сжатия данных; способы навигации по набору данных /Лек/	4	4/0	ПК 1.1.1	Устный опрос
3. 2	Тема 3.2. Индексы. Связи между таблицами. Объединение таблиц Последовательный поиск данных. Сортировка и фильтрация данных. Индексирование таблиц. Рабочие области и псевдонимы. Связь таблиц Знать: понятие индекса и его назначение; понятие сортировки и фильтрации данных; различие	4	4/0	ПК 1.1.1	Устный опрос

	между сортировкой и фильтрацией; назначение псевдонима /Лек/				
3. 3	Лабораторная работа №3. Индексы, связи между таблицами и объединение таблиц Уметь: сортировать и фильтровать данные в БД Владеть: программными средствами по сортировке и фильтрации данных в БД /Лаб/	4	4/0	ПК 1.1.2,ПК 1.1.3	Отчет по лабораторной работе
3. 4	Контрольная работа ПК 1.1.: Производить установку и настройку компонентов автоматизированных (информационных) систем в защищенном исполнении в соответствии с требованиями эксплуатационной документации ПК 1.1.1: Знать состав и принципы работы автоматизированных систем, операционных систем и сред; принципы разработки алгоритмов программ, основных приемов программирования; модели баз данных; принципы построения, физические основы работы периферийных устройств ПК 1.1.2: Уметь осуществлять комплектование, конфигурирование, настройку автоматизированных систем в защищенном исполнении и компонент систем защиты информации автоматизированных систем ПК 1.1.3: Иметь практический опыт установки и настройки компонентов систем защиты информации автоматизированных (информационных) систем /К/	4	2/0	ПК 1.1.1,ПК 1.1.2,ПК 1.1.3	Вопросы для контрольной работы
	4. Управление базой данных с помощью SQL				
4. 1	Тема 4.1. Структурированный язык запросов SQL Общая характеристика языка структурированных запросов SQL. Структуры и типы данных. Стандарты языка SQL. Команды определения данных и манипулирования данными Знать: /Лек/	5	2/0	ПК 1.1.1	Устный опрос
4. 2	Тема 4.2. Операторы и функции языка SQL Структура команды Select. Условие Where. Групповые функции. Функции даты и времени. Символьные функции Знать: назначение и структуру команды Select; назначение и структуру условия Where; виды групповых функции; функции даты и времени; символьные функции /Лек/	5	2/0	ПК 1.1.1	Устный опрос
4. 3	Лабораторная работа №4. Структурированный язык запросов SQL Уметь: писать базовые SQL-запросы; применять условия и сортировку; использовать встроенные функции; работать с соединениями таблиц; обрабатывать NULL-значения Владеть: программными средствами по работе с операторами и функциями языка SQL /Лаб/	5	8/0	ПК 1.1.2,ПК 1.1.3	Отчет по лабораторной работе
	5. Системы управления распределенными базами данных				
5. 1	Тема 5.1. Архитектуры распределенных баз данных Архитектуры клиент-сервер. Достоинства и недостатки моделей архитектуры клиент-сервер. Проектирование базы данных под конкретную архитектуру: клиент-сервер, распределенные базы данных, параллельная обработка данных.	5	2/0	ПК 1.1.1	Устный опрос

	Знать: определение клиент-серверной архитектуры; модели клиент-серверного взаимодействия; достоинства и недостатки клиент -серверной архитектуры; параллельную обработку данных /Лек/				
5. 2	Лабораторная работа №5. Архитектуры распределенных баз данных Уметь: различать виды клиент-серверных архитектур; проектировать оптимальную структуру БД для заданной архитектуры; формулировать достоинства и недостатки различных архитектурных решений; разрабатывать SQL-запросы для работы с данными в разных архитектурах; оценивать производительность и масштабируемость предлагаемых решений Владеть: методологией проектирования реляционных баз данных; навыками работы с системами управления базами данных; техниками оптимизации запросов и структуры БД /Лаб/	5	6/0	ПК 1.1.2,ПК 1.1.3	Отчет по лабораторной работе
5. 3	Тема 5.2. Клиентская часть распределенной базы данных Планирование и развёртывание СУБД для работы с клиентскими приложениями. Планирование приложений. Организация интерфейса с пользователем. Знакомство с мастерами и конструкторами при проектировании форм и отчетов. Типы меню. Работа с меню: создание, модификация. Использование объектно-ориентированных языков программирования для создания клиентской части базы данных. Технологии доступа. Оптимизация производительности работы СУБД Знать: основные компоненты серверной части; механизмы распределения данных; способы планирования приложения; понятие интерфейса с пользователем при разработке БД; понятие меню; способы оптимизации производительности работы СУБД /Лек/	5	2/0	ПК 1.1.1	Устный опрос
5. 4	Лабораторная работа №6. Разработка клиентской части для работы с распределенной базой данных Уметь: разрабатывать клиентские интерфейсы для работы с распределенными БД; оптимизировать запросы с учетом особенностей распределенных систем; обрабатывать возможные ошибки и задержки при работе с распределенными данными; реализовывать кэширование данных на клиентской стороне Владеть: языками программирования для клиентской разработки; принципами проектирования пользовательских интерфейсов; методами отладки сетевых приложений /Лаб/	5	6/0	ПК 1.1.2,ПК 1.1.3	Отчет по лабораторной работе
	6. Администрирование и безопасность баз данных				
6. 1	Тема 6.1. Обеспечение целостности, достоверности и непротиворечивости данных Угрозы целостности СУБД. Основные виды и причины возникновения угроз целостности. Способы противодействия. Правила, ограничения. Понятие хранимой процедуры. Достоинства и недостатки использования хранимых процедур. Понятие триггера. Язык хранимых процедур и триггеров. Каскадные	5	4/0	ПК 1.1.1	Устный опрос

	воздействия. Управление транзакциями и кэширование памяти Знать: виды угроз СУБД; причины возникновения угроз целостности; способы противодействия угрозам целостности; достоинства и недостатки использования хранимых процедур; понятие триггера /Лек/				
6. 2	Лабораторная работа №7. Обеспечение целостности, достоверности и непротиворечивости данных Уметь: определять и реализовывать ограничения целостности данных; проектировать и использовать триггеры для поддержания непротиворечивости; реализовывать транзакционные механизмы обработки данных; разрабатывать процедуры проверки достоверности информации Владеть: языком SQL (DDL, DML, TCL); технологиями реализации ограничений в различных СУБД; методами проектирования транзакционных систем; инструментами мониторинга целостности данных; принципами работы распределенных транзакций /Лаб/	5	4/2	ПК 1.1.2,ПК 1.1.3	Отчет по лабораторной работе
6. 3	Тема 6.2. Механизмы защиты информации в СУБД. Копирование и перенос данных. Восстановление данных Средства идентификации и аутентификации. Общие сведения. Организация взаимодействия СУБД и базовой ОС. Средства управления доступом. Основные понятия: субъекты и объекты, группы пользователей, привилегии, роли и представления. Языковые средства разграничения доступа. Виды привилегий: привилегии безопасности и доступа. Концепция и реализация механизма ролей. Соотношение прав доступа, определяемых ОС и СУБД. Создание резервных копий всей базы данных, журнала транзакций, а также одного или нескольких файлов или файловых групп. Параллелизм операций модификации данных и копирования. Типы резервного копирования. Управление резервными копиями. Автоматизация процессов копирования. Восстановление данных /Лек/	5	2/0	ПК 1.1.1	Устный опрос
6. 4	Лабораторная работа №8.Реализация механизмов защиты информации в СУБД Уметь: настраивать систему аутентификации и авторизации в СУБД; реализовывать механизмы шифрования данных; настраивать аудит безопасности Владеть: технологиями шифрования данных в СУБД; инструментами аудита безопасности /Лаб/	5	4/2	ПК 1.1.2,ПК 1.1.3	Отчет по лабораторной работе
6. 5	Самостоятельная работа по теме «Идентификация и аутентификация при разработке баз данных» /СР/	5	2/0	ПК 1.1.1,ПК 1.1.2,ПК 1.1.3	Вопросы для самоподготовки
6. 6	Самостоятельная работа по теме «Резервное копирование и восстановление баз данных» /СР/	5	4/0	ПК 1.1.1,ПК 1.1.2,ПК 1.1.3	Вопросы для самоподготовки
6. 7	Контрольная работа	5	2/0	ПК	Вопросы для

	ПК 1.1.: Производить установку и настройку компонентов автоматизированных (информационных) систем в защищенном исполнении в соответствии с требованиями эксплуатационной документации ПК 1.1.1: Знать состав и принципы работы автоматизированных систем, операционных систем и сред; принципы разработки алгоритмов программ, основных приемов программирования; модели баз данных; принципы построения, физические основы работы периферийных устройств ПК 1.1.2: Уметь осуществлять комплектование, конфигурирование, настройку автоматизированных систем в защищенном исполнении и компонент систем защиты информации автоматизированных систем ПК 1.1.3: Иметь практический опыт установки и настройки компонентов систем защиты информации автоматизированных (информационных) систем /К/			1.1.1,ПК 1.1.2,ПК 1.1.3	контрольной работы

Список выбранных образовательный технологий:

Название	Описание
Проблемно-развивающая технология	Основанная на создании научной проблемной ситуации, при решении которой учащиеся получают новые учебные знания, овладевают умениями и навыками практической деятельности
Информационные технологии	Личностно ориентированная технология, способ организации самостоятельной деятельности учащихся, направленный на решение задачи учебного проекта
Кейс-технология	Технология включает в себя: индивидуальную самостоятельную работу обучаемых с материалами кейса (идентификация проблемы, формулирование ключевых альтернатив, предложение решения или рекомендуемого действия); работу в малых группах по согласованию видения ключевой проблемы и ее решений; презентацию и экспертизу результатов малых групп на общей дискуссии (в
Компьютерная технология обучения	Основана на использовании информационных технологий в учебном процессе. Реализация данной технологии осуществляется посредством компьютера и иных мультимедийных средств. Использование компьютерных технологий делает учебный процесс не только современным и познавательным, но интересным для обучающихся

4. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДИСЦИПЛИНЫ

Оценочные средства текущего контроля успеваемости:

ВОПРОСЫ ДЛЯ УСТНОГО ОПРОСА

РАЗДЕЛ 1. ОСНОВЫ ТЕОРИИ БАЗ ДАННЫХ

Тема 1.1. Введение в базы данных
1. Что понимается под термином «база данных» с точки зрения организации и хранения взаимосвязанных данных?
2. Какие основные компоненты входят в состав любой системы баз данных?
3. Чем отличаются данные (сырые факты) от метаданных в контексте баз данных?
4. Каковы главные преимущества централизованного управления данными по сравнению с файловыми системами?
5. Какие основные требования предъявляются к централизованным системам управления базами данных?
6. Что означает понятие «независимость данных» и на какие два уровня она делится в архитектуре СУБД?
7. Какие три уровня описывает стандартная архитектура «база данных – СУБД – пользователь»?
8. В чем заключается основное различие между логической и физической независимостью данных?
9. Какие функции выполняет централизованная СУБД при обеспечении многопользовательского доступа?
10. Что такое схема базы данных и как она связана с экземпляром (состоянием) базы данных?
11. Какие требования к целостности и согласованности данных предъявляются к централизованным базам данных?
12. Как в архитектуре систем баз данных реализуется защита данных от несанкционированного доступа?
13. Какие роли пользователей (администратор БД, прикладной программист, конечный пользователь) участвуют во взаимодействии с системой базы данных?
14. Что понимается под централизованным управлением базами данных в отличие от децентрализованного подхода?
15. Какие основные требования к восстановлению данных после сбоев должна обеспечивать система баз данных?

Тема 1.2. Модели данных
1. Что понимается под термином «модель данных» в контексте проектирования баз данных?

2. Каковы основные структурные элементы и принципы организации иерархической модели данных?
3. В чем заключается главное ограничение иерархической модели, которое привело к появлению сетевой модели?
4. Как в сетевой модели данных организуются связи между записями в отличие от иерархической?
5. Какие основные понятия лежат в основе объектно-ориентированной модели данных?
6. Чем отличаются понятия «класс» и «объект» в объектно-ориентированной модели баз данных?
7. Что представляет собой реляционная модель данных и каковы её три основные составляющие (структурная, манипуляционная, целостная)?
8. Какие основные типы данных и структуры (отношение, кортеж, атрибут, домен) используются в реляционной модели?
9. В чем суть двенадцати правил Кодда и для какой цели они были сформулированы?
10. Какое из двенадцати правил Кодда считается главным (нулевым или правилом 0) и что оно требует от системы?
11. Что означает правило «гарантированного доступа» в правилах Кодда?
12. Какое правило Кодда требует поддержки высокоуровневых языков манипулирования данными, например SQL?
13. Чем постреляционная модель данных отличается от классической реляционной модели?
14. Какие возможности для работы со сложными структурами данных (например, вложенными отношениями) предоставляет постреляционная модель?
15. Какие недостатки реляционной модели пытаются устранить с помощью объектно-реляционных и постреляционных моделей данных?

Тема 1.3. Основы реляционной алгебры
1. Что понимается под замкнутостью реляционной алгебры и почему это свойство считается фундаментальным?
2. Какие операции над отношениями относятся к категории традиционных операций теории множеств?
3. Как операция объединения двух отношений отличается от операции пересечения с точки зрения результирующих кортежей?
4. При каких условиях (условиях совместимости по типу) возможно применение традиционных операций объединения, пересечения и разности к двум отношениям?
5. В чем суть декартова произведения двух отношений и как оно влияет на степень и мощность результирующего отношения?
6. Какие операции над отношениями относятся к специальным реляционным операциям?
7. Как работает операция ограничения (выборки, SELECT или σ) и какой результат она возвращает?
8. Что делает операция проекции (π) и чем она отличается от операции ограничения?
9. Как выполняется операция соединения (в частности, эквисоединение и естественное соединение) двух отношений?
10. В чем заключается принципиальное различие между операциями декартова произведения и естественного соединения?
11. Что представляет собой операция деления отношений и для решения каких типов запросов она предназначена?
12. Какие операции, дополненные Крисом Дейтом, расширяют классический набор операций реляционной алгебры?
13. В чем суть операции полуприсоединения и для каких целей она была предложена Дейтом?
14. Что означает операция «расширения» в дополнении Дейта и как она позволяет создавать вычисляемые атрибуты?
15. Какая операция из дополнений Дейта позволяет переименовывать атрибуты в отношении, и зачем это может потребоваться?

Тема 1.4. Системы управления базами данных (СУБД)
1. Что понимается под системой управления базы данных и какова ее основная роль по отношению к базе данных?
2. Какие основные этапы прошло развитие систем управления базами данных с момента появления первых иерархических и сетевых СУБД до современных решений?
3. Какие ведущие мировые и отечественные производители СУБД доминируют на современном рынке?
4. Какие основные функции обязана реализовывать любая полноценная система управления базами данных?
5. В чем заключается функция журнализации изменений в СУБД и для чего используется протокол Write Ahead Log (WAL)?
6. Из каких основных структурных компонентов (подсистем и ядра) состоит современная СУБД?
7. Какие виды интерфейсов (формы, командный интерфейс и другие) предоставляются пользователю для взаимодействия с СУБД?
8. Какие языковые средства входят в состав любой СУБД и для чего служит каждый из них?
9. Как классифицируются СУБД по модели данных, по сфере применения и по архитектуре построения?
10. Чем отличаются настольные СУБД от корпоративных систем с точки зрения функциональности и режима доступа?
11. В чем заключаются ключевые различия между файл-серверной и клиент-серверной архитектурами СУБД?
12. Что представляет собой Microsoft Access как система управления базами данных, и какие возможности она предоставляет конечному пользователю?
13. Какими способами можно создать новую базу данных в MS Access, и какие объекты обязательно входят в ее состав?
14. Какую роль выполняет ядро СУБД в общей структуре системы управления базами данных?
15. Какие требования к языковым средствам и уровню представления данных предъявляются в правилах Кодда к реляционным СУБД?

Тема 1.5. Целостность данных как ключевое понятие баз данных
1. Что понимается под целостностью и непротиворечивостью данных в контексте баз данных?
2. Чем отличается понятие целостности данных от понятия безопасности данных в системах управления базами данных?
3. На какие основные классы принято делить ограничения целостности в реляционных базах данных?

4. Какие ограничения целостности относятся к категории внутриотношенческих (ограничений на атрибут и кортеж)?
5. В чем суть ограничения «домен» и какие проверки целостности оно накладывает на значения атрибута?
6. Что такое первичный ключ и какое ограничение целостности (уникальности и непустоты) он обеспечивает в отношении?
7. Какие ограничения относятся к межотношенческим (ссылкам) и какое понятие лежит в их основе?
8. Что представляет собой внешний ключ и как он обеспечивает ссылочную целостность между двумя отношениями?
9. Как работает механизм каскадных изменений при обновлении или удалении записи, на которую ссылаются внешние ключи?
10. Какие варианты поведения (каскадное удаление, ограничение, обнуление) поддерживает механизм каскадных изменений и чем они отличаются?
11. Каковы примеры нарушения целостности данных на уровне домена (недопустимого значения атрибута)?
12. Какие ситуации в базе данных иллюстрируют нарушение ссылочной целостности (сиротские записи)?
13. Чем отличается нарушение целостности от нарушения непротиворечивости данных, и может ли база данных быть целостной, но противоречивой?
14. В чем заключается проблема аномалий изменения и как она связана с нарушением целостности?
15. Какие декларативные механизмы (ограничения CHECK, UNIQUE, NOT NULL, FOREIGN KEY) в стандарте SQL служат для поддержания целостности данных?

РАЗДЕЛ 2. ПРОЕКТИРОВАНИЕ БАЗ ДАННЫХ

Тема 2.1. Этапы проектирования баз данных. Системный анализ предметной области
1. Какие основные этапы включает в себя процесс проектирования базы данных от постановки задачи до физической реализации?
2. В чем заключается цель и содержание этапа системного анализа предметной области при проектировании базы данных?
3. Какие методы сбора информации о предметной области используются на начальном этапе проектирования?
4. Что представляет собой концептуальная модель данных и на каком этапе проектирования она создается?
5. Чем отличается логическое проектирование базы данных от физического проектирования?
6. Какие типы сущностей (сильные, слабые, ассоциативные) выделяются при анализе предметной области?
7. Как на этапе системного анализа выявляются и документируются все виды связей между сущностями предметной области?
8. Какие степени (мощности) связей (один-к-одному, один-ко-многим, многие-ко-многим) могут существовать между сущностями?
9. Что такое атрибуты сущностей и как они классифицируются (идентифицирующие, описательные, составные, многозначные)?
10. Какие ограничения предметной области должны быть выявлены и задокументированы для последующего их отражения в структуре базы данных?
11. Что представляет собой спецификация требований к базе данных как результат системного анализа?
12. Как на этапе анализа предметной области определяются предполагаемые запросы и операции над данными?
13. Какие проблемы могут возникнуть при недостаточном или некорректном системном анализе предметной области?
14. Какие нотации и средства моделирования (например, нотация «сущность-связь») используются для формализации результатов анализа предметной области?
15. В чем заключается роль системного аналитика и почему этот этап считается критически важным для успеха всего проекта разработки базы данных?

Тема 2.2. Инфологическое проектирование баз данных. ER-диаграммы
1. Что понимается под инфологическим (концептуальным) проектированием баз данных и каков его конечный результат?
2. В чем заключается основная цель инфологического моделирования и почему оно не зависит от конкретной СУБД?
3. Какие основные элементы (сущность, атрибут, связь) используются в нотации «сущность-связь» (ER-нотации)?
4. Как на ER-диаграмме графически обозначаются сущности, атрибуты и типы связей между ними?
5. Какие типы сущностей (сильные и слабые) существуют в ER-модели и чем они отличаются друг от друга?
6. Как на ER-диаграмме изображаются простые и составные атрибуты, а также однозначные и многозначные атрибуты?
7. Какие степени связи (один-к-одному, один-ко-многим, многие-ко-многим) могут быть представлены в ER-диаграммах?
8. Как на ER-диаграмме отображаются классы принадлежности сущностей к связи (обязательная и необязательная связь)?
9. Что такое роль сущности в связи и как она указывается на ER-диаграмме для разрешения неоднозначности?
10. Какие нотации (нотация Питера Чена, нотация Мартина, нотация IDEF1X, «воронья лапка») наиболее распространены для построения ER-диаграмм?
11. Как на ER-диаграмме моделируются рекурсивные связи, когда сущность связана сама с собой?
12. Какие ограничения целостности предметной области могут быть отражены непосредственно на ER-диаграмме?
13. Каков процесс преобразования ER-диаграммы с типом связи «многие-ко-многим» в реляционную модель данных?
14. Чем отличается идентифицирующая связь от неидентифицирующей связи при моделировании зависимых (слабых) сущностей?
15. Какие инструментальные средства (CASE-средства) используются для автоматизированного построения ER-диаграмм и генерации схем баз данных?

Тема 2.3. Даталогическое проектирование
1. Что понимается под даталогическим проектированием баз данных и какова его основная цель?
2. Чем отличается даталогическое проектирование от инфологического и физического этапов проектирования?
3. Какой формальный аппарат (нормализация или правила преобразования ER-модели) используется при переходе от инфологической модели к даталогической?
4. Каковы правила преобразования сущностей и их атрибутов из ER-диаграммы в реляционную схему данных?
5. Как на этапе даталогического проектирования преобразуются связи типа «один-к-одному» между сущностями в структуру реляционных таблиц?
6. Каким образом связь типа «один-ко-многим» отображается в схему реляционной базы данных при даталогическом проектировании?
7. Как преобразуется связь типа «многие-ко-многим» из инфологической модели в даталогическую схему?
8. Каким способом на этапе даталогического проектирования разрешаются многозначные атрибуты сущностей из ER-модели?
9. Как на уровне даталогического проектирования фиксируются первичные и внешние ключи для обеспечения ссылочной целостности?
10. Что происходит с иерархией сущностей (обобщение/специализация) при переходе к даталогической модели?
11. Как на этапе даталогического проектирования документируются типы данных, ограничения и значения по умолчанию для каждого атрибута?
12. Каким образом в даталогической модели отражаются домены, определенные на инфологическом этапе?
13. Что представляет собой даталогическая схема базы данных в терминах конкретной модели данных (например, реляционной)?
14. Какие ограничения целостности, определенные на инфологическом уровне, переносятся в даталогическую модель без изменений?
15. Каков результат даталогического проектирования, передаваемый на следующий этап физического проектирования базы данных?

Тема 2.4. Средства автоматизации проектирования
1. Что понимается под CASE-средствами, CASE-системой и CASE-технологией в контексте автоматизации проектирования баз данных?
2. По каким основным признакам (классификациям) принято разделять CASE-средства в зависимости от этапа жизненного цикла и функциональности?
3. В чем суть метода функционального моделирования SADT и для решения каких задач он применяется?
4. Какие основные элементы входят в состав IDEF0-диаграммы и что обозначает каждый из них?
5. Что означают четыре типа стрелок (вход, управление, выход, механизм) на IDEF0-диаграмме?
6. Как принцип декомпозиции реализуется в методологии IDEF0 для построения иерархии диаграмм?
7. Что представляют собой DFD-диаграммы (диаграммы потоков данных) и чем они отличаются от IDEF0-диаграмм?
8. Какие основные элементы (внешняя сущность, процесс, хранилище данных, поток данных) используются в DFD-диаграммах?
9. Что такое UML и какие задачи проектирования информационных систем решаются с помощью этого языка визуального моделирования?
10. Какие виды UML-диаграмм относятся к структурным (статическим), а какие – к поведенческим (динамическим)?
11. Что представляет собой диаграмма вариантов использования (use case diagram) и какие элементы в нее входят?
12. Как на диаграмме вариантов использования отображаются отношения между актерами и прецедентами (вариантами использования)?
13. Какие типы отношений (ассоциация, включение, расширение, обобщение) существуют между вариантами использования в UML?
14. На каких этапах проектирования баз данных и информационных систем применяются рассмотренные CASE-средства и нотации?
15. Какие современные CASE-средства (например, Erwin, PowerDesigner, Draw.io, PlantUML) поддерживают создание IDEF0-, DFD- и UML-диаграмм?

РАЗДЕЛ 3. ОРГАНИЗАЦИЯ БАЗ ДАННЫХ

Тема 3.1. Создание базы данных. Манипулирование данными
1. Какими способами (через графический интерфейс или SQL-запрос) можно создать новую базу данных в системах управления базами данных?
2. Какие параметры (имя файла, размер, способ роста, кодировка) необходимо определить при физическом создании базы данных?
3. Как создается структура таблицы и какие обязательные элементы (имя таблицы, имена и типы столбцов) должны быть определены?
4. Какие команды языка определения данных DDL (CREATE, ALTER, DROP) используются для создания и изменения структуры таблиц?
5. Как с помощью команды ALTER TABLE можно добавить новый столбец, изменить тип данных существующего столбца или удалить столбец?
6. Какими способами (прямой ввод, импорт из внешних источников, выполнение INSERT-запросов) производится наполнение таблицы данными?
7. Что понимается под восстановлением базы данных и в каких ситуациях (сбой системы, ошибочное удаление данных) эта операция необходима?

8. Для чего выполняется сжатие базы данных и чем оно отличается от восстановления и от дефрагментации?
9. Какие операции относятся к открытию и модификации данных в контексте манипулирования содержимым базы данных?
10. Как осуществляется навигация по набору данных (перемещение по записям, переход к первой, последней, следующей, предыдущей записи) в интерактивном режиме?
11. Какие существуют способы фильтрации и поиска данных при навигации по большому набору записей?
12. Как реализована пакетная модификация данных (обновление, вставка, удаление нескольких записей одновременно) с помощью языка манипулирования данными DML?
13. Что такое транзакция и как она связана с безопасной модификацией данных (фиксация изменений COMMIT и откат ROLLBACK)?
14. Какие ограничения целостности (первичный ключ, уникальность, внешний ключ, проверочные ограничения) проверяются при вставке и модификации данных?
15. Как осуществляется восстановление базы данных из резервной копии по сравнению с восстановлением до согласованного состояния с помощью журнала транзакций?

Тема 3.2. Индексы. Связи между таблицами. Объединение таблиц
1. Что такое последовательный (полный) поиск данных в таблице и в чем заключаются его основные недостатки при большом объеме данных?
2. Как выполняются сортировка и фильтрация данных в реляционной базе данных без использования индексов?
3. Что представляет собой индекс в базе данных и какую структуру данных (чаще всего B-дерево) он использует для ускорения поиска?
4. Чем отличается кластерный индекс от некластерного с точки зрения физического хранения данных?
5. Каковы преимущества и недостатки создания индексов с точки зрения скорости операций чтения и модификации данных?
6. Что понимается под рабочей областью в контексте работы с базами данных и каково ее назначение?
7. Для чего используются псевдонимы (алиасы) таблиц и как они упрощают написание сложных запросов?
8. Как устанавливается связь между таблицами через внешний ключ и какую роль при этом играют первичные ключи?
9. Какие типы связей (один-к-одному, один-ко-многим, многие-ко-многим) можно реализовать между таблицами в реляционной базе данных?
10. Что такое объединение таблиц (JOIN) и какие основные типы объединений (INNER, LEFT, RIGHT, FULL) существуют?
11. Как работает внутреннее объединение (INNER JOIN) и какие строки попадают в результирующий набор данных?
12. Чем отличается левое внешнее объединение (LEFT OUTER JOIN) от правого внешнего объединения (RIGHT OUTER JOIN)?
13. В каких случаях применяется полное внешнее объединение (FULL OUTER JOIN) и что оно возвращает?
14. Как связаны между собой индексы и производительность операций объединения таблиц (JOIN)?
15. Каким образом при объединении нескольких таблиц обрабатываются неоднозначные имена столбцов с помощью псевдонимов?

РАЗДЕЛ 4. УПРАВЛЕНИЕ БАЗОЙ ДАННЫХ С ПОМОЩЬЮ SQL

Тема 4.1. Структурированный язык запросов SQL
1. Что понимается под структурированным языком запросов SQL и для каких целей он был разработан?
2. На какие три основные подмножества команд делится язык SQL в зависимости от выполняемых функций?
3. Какие команды относятся к языку определения данных (DDL) и для чего предназначена каждая из них?
4. Какие команды входят в состав языка манипулирования данными (DML) и какие операции они выполняют?
5. Что представляет собой язык управления данными (DCL) и какие команды (GRANT, REVOKE) к нему относятся?
6. Как с помощью команды CREATE DATABASE создать новую базу данных и какие параметры можно при этом указать?
7. Каков синтаксис команды CREATE TABLE и какие элементы (имя таблицы, список столбцов с типами данных, ограничения) обязательно должны присутствовать?
8. Как с помощью команды ALTER TABLE можно изменить структуру существующей таблицы (добавить, удалить или изменить столбец)?
9. Что делает команда DROP TABLE и какие последствия влечет за собой ее выполнение?
10. Каков базовый синтаксис команды SELECT для извлечения данных из таблицы и какие предложения являются обязательными?
11. Как с помощью предложения WHERE в команде SELECT выполняется фильтрация строк по заданным условиям?
12. Для чего служит команда INSERT INTO и какие существуют формы ее использования (вставка одной записи, вставка результата подзапроса)?
13. Как с помощью команды UPDATE выполняется модификация существующих данных в таблице и что произойдет при отсутствии предложения WHERE?
14. Каков синтаксис команды DELETE FROM и чем отличается удаление всех записей этой командой от выполнения TRUNCATE TABLE?
15. Что такое представление в SQL и как оно связано с безопасностью и упрощением доступа к данным?

Тема 4.2. Операторы и функции языка SQL
1. Какие арифметические операторы (+, -, *, /) поддерживаются в SQL и как они применяются в запросах?
2. Какие операторы сравнения (=, <>, >, <, >=, <=) используются в SQL для построения условий фильтрации?

3. Для чего служат логические операторы AND, OR и NOT и как они влияют на комбинирование условий в предложении WHERE?
4. Что делает оператор BETWEEN и как с его помощью проверить принадлежность значения заданному диапазону?
5. Как работает оператор IN и чем его использование отличается от объединения нескольких условий через OR?
6. Для каких целей предназначен оператор LIKE и какие символы-шаблоны (%) и (_) используются при поиске по шаблону?
7. Что делает оператор IS NULL и почему для проверки на пустые значения нельзя использовать обычное сравнение (= NULL)?
8. Какие агрегатные функции (COUNT, SUM, AVG, MAX, MIN) поддерживаются в SQL и для решения каких задач применяется каждая из них?
9. Как работает предложение GROUP BY и для чего оно используется совместно с агрегатными функциями?
10. Чем отличается предложение WHERE от предложения HAVING при фильтрации данных в запросах с группировкой?
11. Какие строковые функции (CONCAT, UPPER, LOWER, LENGTH, SUBSTRING, TRIM) наиболее часто используются в SQL-запросах?
12. Какие функции для работы с датой и временем (CURRENT_DATE, EXTRACT, DATEADD, DATEDIFF) поддерживаются в стандарте SQL?
13. Какие математические функции (ROUND, CEIL, FLOOR, ABS, POWER, SQRT) могут применяться при вычислениях в запросах?
14. Какие функции преобразования типов данных (CAST, CONVERT) используются для явного приведения одного типа к другому?
15. Что представляют собой оконные функции (ROW_NUMBER, RANK, LAG, LEAD) и чем их применение отличается от использования агрегатных функций с GROUP BY?

РАЗДЕЛ 5. СИСТЕМЫ УПРАВЛЕНИЯ РАСПРЕДЕЛЕННЫМИ БАЗАМИ ДАННЫХ

Тема 5.1. Архитектуры распределенных баз данных
1. Что понимается под распределенной базой данных (РБД) и чем она отличается от централизованной базы данных?
2. Каковы основные принципы прозрачности (локации, фрагментации, репликации) в распределенных базах данных?
3. Какие основные архитектуры распределенных баз данных существуют (однородные и неоднородные, автономные и неавтономные)?
4. Что представляет собой архитектура «клиент-сервер» в контексте распределенных баз данных и как в ней распределяются функции?
5. Как организована многоуровневая архитектура (трехуровневая и более) распределенной базы данных?
6. Что такое фрагментация данных в распределенной базе данных и какие виды фрагментации (горизонтальная, вертикальная, смешанная) существуют?
7. Как репликация данных повышает доступность и производительность распределенной базы данных?
8. Какие проблемы синхронизации данных возникают при использовании репликации и как они решаются?
9. Что представляет собой распределенный процессор транзакций и как он обеспечивает согласованность данных в РБД?
10. Как работает протокол двухфазной фиксации (2PC) при выполнении распределенных транзакций?
11. Какие требования предъявляются к сетевому взаимодействию узлов в архитектуре распределенной базы данных?
12. Что такое глобальный каталог (словарь) данных в распределенной базе данных и какую информацию он хранит?
13. Какие стратегии размещения данных (централизованная, секционированная, полностью реплицированная, гибридная) используются в архитектурах РБД?
14. Как архитектура распределенной базы данных обеспечивает отказоустойчивость при выходе из строя одного или нескольких узлов?
15. Какие известные реализации распределенных баз данных (например, Oracle RAC, Google Spanner, Apache Cassandra) существуют на современном рынке?

Тема 5.2. Клиентская часть распределенной базы данных
1. Что понимается под клиентской частью распределенной базы данных и какие компоненты в нее входят?
2. Какие функции выполняет клиентское приложение при работе с распределенной базой данных?
3. Чем отличается «толстый» клиент от «тонкого» клиента в архитектуре распределенной базы данных?
4. Как клиентская часть обрабатывает запросы пользователя и преобразует их в команды для серверной части РБД?
5. Что представляет собой локальный кэш данных на стороне клиента и для каких целей он используется?
6. Как на клиентской стороне реализуется временное хранение данных при отсутствии постоянного соединения с сервером (автономный режим)?
7. Какие механизмы синхронизации данных между клиентской частью и сервером применяются при восстановлении соединения?
8. Как в клиентской части распределенной базы данных реализуется управление транзакциями, затрагивающими несколько узлов?
9. Какие требования предъявляются к интерфейсу пользователя клиентской части для удобной работы с распределенными данными?
10. Как клиентская часть определяет, на каком узле распределенной базы данных находятся запрашиваемые данные?
11. Какие протоколы и форматы обмена данными (например, ODBC, JDBC, OLE DB, REST API) используются клиентской частью для взаимодействия с сервером?
12. Как на клиентской стороне обрабатываются ошибки и тайм-ауты при недоступности одного из узлов распределенной базы данных?

13. Что такое «ленивая» загрузка данных в клиентском приложении и для чего она применяется?
14. Какие механизмы безопасности и аутентификации реализуются на клиентской стороне для доступа к распределенной базе данных?
15. Как клиентская часть распределенной базы данных поддерживает работу с фрагментированными и реплицированными данными без участия пользователя?

РАЗДЕЛ 6. АДМИНИСТРИРОВАНИЕ И БЕЗОПАСНОСТЬ БАЗ ДАННЫХ

Тема 6.1. Обеспечение целостности, достоверности и непротиворечивости данных
1. В чем разница между понятиями целостности, достоверности и непротиворечивости данных в контексте баз данных?
2. Какие типы ограничений целостности (декларативные и процедурные) могут быть реализованы в современных СУБД?
3. Как обеспечивается целостность по сущности с помощью первичных ключей и ограничения NOT NULL?
4. Что такое ссылочная целостность и какие правила (RESTRICT, CASCADE, SET NULL) определяют поведение при обновлении или удалении связанных записей?
5. Как ограничение UNIQUE отличается от ограничения PRIMARY KEY с точки зрения обеспечения целостности?
6. Что представляют собой проверочные ограничения (CHECK) и какие условия целостности они могут контролировать?
7. Каким образом доменная целостность реализуется через типы данных и пользовательские домены в СУБД?
8. Что такое транзакция и как свойства ACID (атомарность, согласованность, изоляция, долговечность) обеспечивают целостность данных?
9. Какие уровни изоляции транзакций (READ UNCOMMITTED, READ COMMITTED, REPEATABLE READ, SERIALIZABLE) существуют и как они влияют на непротиворечивость?
10. Что такое аномалия параллельного доступа (грязное чтение, неповторяемое чтение, фантомное чтение) и как уровни изоляции ее предотвращают?
11. Какие механизмы блокировок (разделяемые, исключительные, гранулярность блокировок) используются для поддержания целостности при параллельной работе пользователей?
12. Как оптимистическая и пессимистическая блокировки отличаются по подходу к обеспечению непротиворечивости данных?
13. Что такое журнал транзакций (журнал упорядоченных изменений) и как он обеспечивает возможность восстановления целостности после сбоев?
14. Каким образом триггеры (хранимые процедуры, автоматически выполняемые при событиях) могут поддерживать сложные бизнес-правила целостности?
15. Что понимается под достоверностью данных и как она обеспечивается через контроль ввода, валидацию и аудит изменений?

Тема 6.2. Механизмы защиты информации в СУБД. Копирование и перенос данных. Восстановление данных
1. Какие основные угрозы безопасности данных существуют в системах управления базами данных?
2. Как разграничение доступа на уровне пользователей реализуется через механизмы аутентификации и авторизации в СУБД?
3. Что представляют собой привилегии (права доступа) и как команды GRANT и REVOKE управляют ими?
4. Какие уровни привилегий (системные, на объекты базы данных, на столбцы) могут быть назначены пользователям в СУБД?
5. Что такое роли в СУБД и как они упрощают администрирование прав доступа для групп пользователей?
6. Как механизмы шифрования данных (прозрачное шифрование, шифрование на уровне столбцов, SSL/TLS для передачи) защищают информацию в СУБД?
7. Что представляет собой аудит (журналирование действий) в СУБД и для выявления каких нарушений он используется?
8. Какие существуют стратегии резервного копирования (полное, дифференциальное, инкрементное) и чем они отличаются друг от друга?
9. Как выполняется копирование данных между различными СУБД и какие инструменты (экспорт/импорт, средства миграции) для этого применяются?
10. Что такое мгновенные снимки (snapshot) и как они используются для создания согласованных копий базы данных?
11. Каким образом производится восстановление базы данных до состояния на определенный момент времени (point-in-time recovery)?
12. Что представляет собой восстановление из резервной копии и чем оно отличается от восстановления с применением журнала транзакций?
13. Какие действия необходимо выполнить для переноса базы данных с одного сервера на другой с минимальным временем простоя?
14. Как механизмы защиты от несанкционированного доступа различаются для настольных (например, MS Access) и серверных (например, MS SQL Server, Oracle) СУБД?
15. Что понимается под неотказуемостью (non-repudiation) в контексте защиты данных и как она обеспечивается в системах баз данных?

ВОПРОСЫ ДЛЯ САМОПОДГОТОВКИ

Самостоятельная работа по теме «Модели данных»
1. Какие четыре основные модели данных (не считая постреляционной) традиционно выделяются в теории баз данных

и каковы их отличительные признаки?
2. В чем заключается основное ограничение иерархической модели данных, которое делает ее неудобной для моделирования сложных связей «многие-ко-многим»?
3. Как сетевая модель данных решает проблему ограничений иерархической модели и какие дополнительные возможности она предоставляет?
4. Какие основные принципы (инкапсуляция, наследование, полиморфизм) объектно-ориентированной модели данных заимствованы из объектно-ориентированного программирования?
5. Что представляет собой отношение (таблица) в реляционной модели данных и какие свойства (отсутствие дубликатов строк, порядок строк не важен, атомарность значений) ему присущи?
6. Какое из двенадцати правил Кодда гарантирует, что все данные в реляционной СУБД должны быть доступны только через табличную структуру?
7. Чем отличается постреляционная модель от классической реляционной модели и для работы с какими типами данных она лучше подходит?
8. Как в постреляционной модели (например, в системе UniData) реализуются многозначные атрибуты без необходимости создания дополнительных таблиц?
9. Для какой предметной области объектно-ориентированная модель данных имеет преимущества перед реляционной, и какие проблемы присущи объектно-ориентированным СУБД?
10. Почему реляционная модель данных стала доминирующей в коммерческих системах управления базами данных, несмотря на преимущества других моделей?

Самостоятельная работа по теме «Идентификация и аутентификация при разработке баз данных»
1. В чем разница между понятиями идентификации и аутентификации в контексте безопасности баз данных?
2. Какие типы аутентификации (на основе пароля, встроенная, сертификатная, биометрическая) могут использоваться в системах управления базами данных?
3. Как встроенная аутентификация (Windows-аутентификация) отличается от серверной аутентификации (SQL Server Authentication) в продуктах Microsoft?
4. Что такое двухфакторная аутентификация и почему она повышает безопасность доступа к базам данных?
5. Каким образом идентификатор пользователя (логин) связывается с правами доступа к объектам базы данных через механизм сопоставления с пользователем БД?
6. Что представляют собой политики паролей (сложность, срок действия, блокировка после неудачных попыток) и как они реализуются в СУБД?
7. Как хранятся пароли пользователей в современных СУБД и почему их хранение в открытом виде является грубым нарушением безопасности?
8. Что такое аутентификация по сертификату и в каких сценариях (высокозащищенные системы, распределенные приложения) она применяется?
9. Как при разработке базы данных следует разделять пользователей-администраторов и обычных пользователей с точки зрения их идентификаторов?
10. Какие уязвимости могут возникнуть при передаче идентификационных данных по сети и как защищенные протоколы (SSL/TLS, Kerberos) решают эту проблему?

Самостоятельная работа по теме «Резервное копирование и восстановление баз данных»
1. Какие основные цели преследует резервное копирование баз данных и чем оно отличается от простого копирования файлов?
2. Чем отличаются полное, дифференциальное и инкрементное резервное копирование по объему сохраняемых данных и времени восстановления?
3. В чем суть резервного копирования только журнала транзакций и для восстановления до какого состояния оно применяется?
4. Что такое момент восстановления (point-in-time recovery) и какие типы резервных копий необходимы для его выполнения?
5. Какие факторы (размер базы данных, допустимое время простоя, частота изменений) влияют на выбор стратегии резервного копирования?
6. Как отличается процесс восстановления базы данных из полной резервной копии по сравнению с восстановлением из последовательности полной, дифференциальной и инкрементной копий?
7. Что такое «холодное», «горячее» и «теплое» резервное копирование и в каких ситуациях применяется каждый из этих режимов?
8. Какую роль играет журнал транзакций при восстановлении базы данных после сбоя оборудования или программной ошибки?
9. Какие инструменты и средства (встроенные средства СУБД, сторонние решения, облачные сервисы) используются для автоматизации резервного копирования?
10. Как часто следует выполнять тестовое восстановление из резервных копий и почему это критически важно для надежности системы резервного копирования?

Задания для практических и лабораторных работ см. в приложение А.

Оценочные средства промежуточной аттестации:

КОНТРОЛЬНАЯ РАБОТА (4 СЕМЕСТР)

ПК 1.1. Производить установку и настройку компонентов автоматизированных (информационных) систем в

защищенном исполнении в соответствии с требованиями эксплуатационной документации

1. Что такое база данных?
2. Что такое СУБД?
3. Перечислите три уровня архитектуры систем баз данных.
4. Что такое модель данных?
5. Назовите четыре основные модели данных.
6. В чем суть реляционной модели данных?
7. Что такое первичный ключ?
8. Что такое внешний ключ?
9. Назовите три традиционные операции реляционной алгебры.
10. Назовите три специальные операции реляционной алгебры.
11. Что такое целостность данных?
12. Что такое ссылочная целостность?
13. Перечислите основные этапы проектирования баз данных.
14. Что такое ER-диаграмма?
15. Назовите три типа связей между сущностями в ER-модели.
16. Как преобразуется связь «многие-ко-многим» в реляционной модели?
17. Что такое нормализация базы данных?
18. Что такое независимость данных?
19. Назовите одно из двенадцати правил Кодда.
20. Что такое CASE-средства?

Тестовые задания контрольной работы (5 семестр) см. в Приложении Б

Итоговый тест

ПК 1.1. Производить установку и настройку компонентов автоматизированных (информационных) систем в защищенном исполнении в соответствии с требованиями эксплуатационной документации

1. Какие из перечисленных утверждений правильно описывают базу данных по сравнению с файловой системой?
А) В базе данных минимизируется избыточность данных
Б) База данных не поддерживает совместный доступ пользователей
В) В базе данных обеспечивается централизованное управление и целостность
Г) Данные в базе данных всегда хранятся в одном единственном файле
2. Какие модели данных относятся к классическим моделям?
А) Реляционная модель
Б) Иерархическая модель
В) Графовая модель
Г) Сетевая модель
3. Что из перечисленного является обязательным свойством первичного ключа в реляционной таблице?
А) Значения могут повторяться
Б) Значения не могут быть пустыми (NULL)
В) Значения должны быть уникальными
Г) Значения должны быть текстовыми
4. Какие операции относятся к специальным операциям реляционной алгебры?
А) Объединение
Б) Проекция
В) Ограничение (выборка)
Г) Декартово произведение
5. Какие из перечисленных функций обязательно реализует современная система управления базами данных (СУБД)?
А) Управление транзакциями и восстановление после сбоев
Б) Создание графических изображений
В) Обеспечение многопользовательского доступа с разграничением прав
Г) Компиляция программ на языке C++
6. Какие утверждения о ссылочной целостности являются верными?
А) Ссылочная целостность обеспечивается механизмом внешних ключей
Б) При нарушении ссылочной целостности СУБД всегда автоматически удаляет все данные
В) Каскадное удаление означает, что при удалении родительской записи удаляются и связанные дочерние записи
Г) Внешний ключ обязательно должен иметь то же имя, что и первичный ключ
7. Какие из перечисленных этапов входят в процесс проектирования базы данных?
А) Инфологическое (концептуальное) проектирование
Б) Программирование пользовательского интерфейса
В) Даталогическое (логическое) проектирование
Г) Системный анализ предметной области
8. Какие типы связей между сущностями могут быть представлены в ER-диаграмме?
А) Один-к-одному (1:1)
Б) Один-ко-многим (1:N)

В) Многие-ко-многим (M:N)
Г) Один-к-нулю (1:0)
9. Какие операторы относятся к языку определения данных (DDL) в SQL?
А) SELECT
Б) CREATE TABLE
В) INSERT INTO
Г) ALTER TABLE
10. Какие команды SQL используются для управления транзакциями?
А) BEGIN TRANSACTION
Б) GRANT
В) COMMIT
Г) ROLLBACK
11. Какие утверждения о распределенных базах данных являются верными?
А) В распределенной базе данные физически находятся на нескольких узлах
Б) Распределенная база данных всегда медленнее централизованной
В) Репликация данных повышает доступность и отказоустойчивость
Г) Протокол двухфазной фиксации обеспечивает согласованность распределенных транзакций
12. Какие компоненты входят в архитектуру «клиент-сервер» применительно к базам данных?
А) Клиентское приложение
Б) Принтер
В) Сервер баз данных
Г) Сетевое соединение
13. Какие задачи относятся к функциям администратора баз данных (DBA)?
А) Разработка дизайна веб-сайта
Б) Настройка прав доступа пользователей
В) Организация резервного копирования и восстановления
Г) Мониторинг производительности СУБД
14. Какие команды SQL используются для управления правами доступа?
А) GRANT
Б) UPDATE
В) REVOKE
Г) DELETE
15. Какие из перечисленных свойств описывают транзакцию (свойства ACID)?
А) Атомарность (все изменения выполняются как единое целое или не выполняются вовсе)
Б) Анонимность (пользователь транзакции не известен системе)
В) Изоляция (параллельные транзакции не влияют друг на друга)
Г) Долговечность (результаты фиксированной транзакции сохраняются после сбоя)

Темы индивидуальных проектов, курсовых работ (проектов), индивидуальных заданий на практику:

Учебным планом не предусмотрено

Описание критериев оценивания успеваемости

НЕДОСТАТОЧНЫЙ УРОВЕНЬ Обучающийся демонстрирует: - существенные пробелы в знаниях учебного материала; - принципиальные ошибки при ответе на основные вопросы билета, отсутствует знание и понимание основных понятий и категорий; - непонимание сущности дополнительных вопросов в рамках заданий билета; - отсутствие умения выполнять практические задания, предусмотренные программой дисциплины; - отсутствие готовности (способности) к дискуссии и низкая степень контактности.	ПОРОГОВЫЙ УРОВЕНЬ Обучающийся демонстрирует: - знания теоретического материала; - неполные ответы на основные вопросы, ошибки в ответе, недостаточное понимание сущности излагаемых вопросов; - неуверенные и неточные ответы на дополнительные вопросы; - недостаточное владение литературой, рекомендованной программой дисциплины; - умение без грубых ошибок решать практические задания, которые следует выполнить.	ПРОДВИНУТЫЙ УРОВЕНЬ Обучающийся демонстрирует: - знание и понимание основных вопросов контролируемого объема программного материала; - твердые знания теоретического материала; - способность устанавливать и объяснять связь практики и теории, выявлять противоречия, проблемы и тенденции развития; - правильные и конкретные, без грубых ошибок ответы на поставленные вопросы; - умение решать практические задания, которые следует выполнить; - владение основной литературой, рекомендованной программой дисциплины; - наличие собственной обоснованной позиции по обсуждаемым вопросам; - незначительные оговорки и неточности в раскрытии отдельных положений вопросов билета, присутствует неуверенность в ответах на дополнительные вопросы.	ВЫСОКИЙ УРОВЕНЬ Обучающийся демонстрирует: - глубокие, всесторонние и аргументированные знания программного материала; - полное понимание сущности и взаимосвязи рассматриваемых процессов и явлений, точное знание основных понятий в рамках обсуждаемых заданий; - способность устанавливать и объяснять связь практики и теории; - логически последовательные, содержательные, конкретные и исчерпывающие ответы на все задания билета, а также дополнительные вопросы экзаменатора; - умение решать практические задания; - свободное использование в ответах на вопросы материалов рекомендованной основной и дополнительной литературы.
Оценка «незачет», «неудовлетворительно»	Оценка «зачтено/удовлетворительно» ,«удовлетворительно»	Оценка «зачтено/хорошо», «хорошо»	Оценка «зачтено/отлично», «отлично»

ПК 1.1.: Производить установку и настройку компонентов автоматизированных (информационных) систем в защищенном исполнении в соответствии с требованиями эксплуатационной документации

Недостаточный уровень:
Знает отдельные термины и названия, относящиеся к составу автоматизированных систем, операционных систем и сред; имеет фрагментарные представления о принципах разработки алгоритмов и основных приемах программирования; различает по названиям некоторые модели баз данных (например, реляционную); может перечислить несколько периферийных устройств, но не объясняет принципы их построения и физические основы работы. Воспроизведение материала возможно только с опорой на образец или подсказки преподавателя.

Умеет под руководством преподавателя или по образцу подбирать отдельные компоненты для простейшей автоматизированной системы; выполнять элементарные действия по конфигурированию (например, установить пароль) или настройке одного компонента защиты информации при наличии подробной инструкции. Самостоятельно осуществлять комплектование, конфигурирование и настройку защищенных автоматизированных систем не способен, действия носят фрагментарный и подражательный характер.

Имеет практический опыт установки отдельных компонентов систем защиты информации (СЗИ) в автоматизированной системе по готовой инструкции с использованием установочных пакетов и настройки простейших параметров (изменение пароля, выбор базового профиля).

Пороговый уровень:
Знает состав и типовые принципы работы автоматизированных и операционных систем в стандартных условиях; основные принципы разработки алгоритмов (линейные, разветвляющиеся, циклические) и базовые приемы программирования; классификацию моделей баз данных (иерархическую, сетевую, реляционную) и их краткую характеристику; принципы построения (назначение, способы подключения) и физические основы работы (например, преобразование сигналов, электрические и оптические принципы) основных периферийных устройств (принтеры, сканеры, мониторы, клавиатура, мышь). Знания носят репродуктивный характер и воспроизводятся по знакомому алгоритму.

Умеет осуществлять комплектование типовой автоматизированной системы в защищенном исполнении по заданной спецификации (подбирать аппаратное и программное обеспечение с учетом базовых требований к защите информации); выполнять стандартное конфигурирование (настройка сетевых параметров, прав доступа,

Умеет осуществлять комплектование типовой автоматизированной системы в защищенном исполнении по заданной спецификации (подбирать аппаратное и программное обеспечение с учетом базовых требований к защите информации); выполнять стандартное конфигурирование (настройка сетевых параметров, прав доступа, политик паролей) и базовую настройку компонентов системы защиты информации (например, антивирусных средств, встроенного межсетевого экрана) по готовому алгоритму или инструкции. Умеет действовать в типовых, заранее известных условиях.

Имеет практический опыт самостоятельной установки и настройки типовых компонентов систем защиты информации (средств аутентификации, контроля доступа, антивирусной защиты, межсетевых экранов) в автоматизированной системе в соответствии с эксплуатационной документацией и типовыми политиками безопасности.

Продвинутый уровень:
Знает в систематизированном виде состав, функции и принципы взаимодействия компонентов автоматизированных систем, операционных систем и сред (включая сетевые взаимодействия, файловые системы, управление процессами); свободно оперирует принципами разработки эффективных алгоритмов и современными приемами программирования (структурное, модульное, объектно-ориентированное программирование); развернуто характеризует все основные модели баз данных с примерами их применения, преимуществами и недостатками; объясняет принципы построения и физические основы работы широкого круга периферийных устройств (включая лазерные и струйные принтеры, ЖК-мониторы, твердотельные накопители, оптические сканеры) с опорой на физические эффекты. Знания позволяют самостоятельно решать типовые и измененные профессиональные задачи.

Умеет самостоятельно осуществлять комплектование автоматизированной системы в защищенном исполнении с учетом нестандартных требований заказчика, особенностей среды функционирования и угроз информационной безопасности; выполнять полное конфигурирование и настройку автоматизированной системы и всех основных компонентов системы защиты информации (межсетевые экраны, системы обнаружения вторжений, средства криптографической защиты, контроля доступа) в измененных условиях; умеет выбирать оптимальные параметры настройки, анализировать совместимость компонентов и корректность выполнения функций защиты. Действия носят осознанный и продуктивный характер.

Имеет практический опыт установки, настройки и интеграции компонентов систем защиты информации в гетерогенной автоматизированной системе, включая настройку взаимодействия компонентов СЗИ между собой и с прикладным ПО, разрешение конфликтов, корректировку исключений и верификацию применения политик ИБ.

Высокий уровень:
Знает на углубленном уровне состав, принципы работы и взаимосвязи автоматизированных систем, операционных систем и сред, включая нестандартные и специализированные решения (реального времени, распределенные, встраиваемые системы); владеет принципами разработки сложных, оптимизированных по времени и памяти алгоритмов, продвинутыми приемами программирования (рекурсия, динамическое программирование, параллельные и асинхронные вычисления); глубоко понимает все модели баз данных (включая объектно-реляционные, колоночные, документо-ориентированные NoSQL), умеет анализировать их применимость для нестандартных задач; развернуто объясняет физические основы работы периферийных устройств на уровне физических законов (электростатика, оптика, магнетизм, пьезоэффект), принципы их построения, диагностики неисправностей и эксплуатационные ограничения. Знания готовы к творческой адаптации, инновационной деятельности и передаче другим специалистам.

Умеет на высоком профессиональном уровне осуществлять комплектование, конфигурирование и настройку сложных, распределенных автоматизированных систем в защищенном исполнении, включая нестандартные и уникальные конфигурации; умеет разрабатывать и реализовывать нештатные схемы настройки компонентов системы защиты информации для сложных сред (виртуализация, облачные платформы, гетерогенные сети); умеет оптимизировать параметры конфигурирования по критериям безопасности, производительности и удобства администрирования; способен самостоятельно находить и устранять конфликты и ошибки настройки, адаптировать решения под новые, непредвиденные условия эксплуатации, а также передавать опыт и методики настройки другим специалистам.

Имеет практический опыт установки, настройки, оптимизации и адаптации компонентов систем защиты информации в сложных распределённых автоматизированных системах, включая нештатные сценарии развёртывания, автоматизацию процесса настройки (скрипты, групповые политики), экспорт/импорт конфигураций, аудит корректности установки и настройки, а также разработку рекомендаций по модификации конфигураций компонентов СЗИ.

Перечень практических навыков, приобретаемых при изучении дисциплины:

ПК 1.1: Иметь навыки установки и настройки компонентов автоматизированных (информационных) систем в защищенном исполнении в соответствии с требованиями эксплуатационной документации.
ПК 1.2: Иметь навыки администрирования программных и программно-аппаратных компонентов автоматизированной (информационной) системы в защищенном исполнении.
ПК 1.3: Иметь навыки обеспечения бесперебойной работы автоматизированных (информационных) систем в защищенном исполнении в соответствии с требованиями эксплуатационной документации.
ПК 1.4: Иметь навыки и опыт деятельности по проверке технического состояния, техническому обслуживанию, текущему ремонту, устранению отказов и восстановлению работоспособности автоматизированных (информационных) систем в защищенном исполнении.

Методы оценки	Критерии оценки
Экспертное наблюдение за решением обучающимися проблемно-ситуационных задач в рамках практической подготовки	Оценка «отлично» – задание выполнено полностью, в рамках регламента, установленного на публичную презентацию, студент(ы) приводит (подготовили) полную четкую аргументацию выбранного решения на основе качественно сделанного анализа задания. Демонстрируются хорошие теоретические знания, имеется собственная обоснованная точка зрения на проблему(ы) и причины ее (их) возникновения. В случае ряда выявленных проблем четко определяет их иерархию. При устной презентации уверенно и быстро отвечает на заданные вопросы, выступление сопровождается приемами визуализации. В случае письменного отчета-презентации по выполнению кейс-задания сделан структурированный и детализированный анализ, представлены возможные варианты решения (3-5), четко и аргументировано обоснован окончательный выбор одного из альтернативных решений. Оценка «хорошо» – задание выполнено полностью, но в рамках установленного на выступление регламента, студент(ы) не приводит (не подготовили) полную четкую аргументацию выбранного решения. Имеет место излишнее теоретизирование, или наоборот, теоретическое обоснование ограничено, имеется собственная точка зрения на проблемы, но не все причины ее возникновения установлены. При устной презентации на дополнительные вопросы выступающий отвечает с некоторым затруднением, подготовленная устная презентации выполненного задания не очень структурирована. При письменном отчете-презентации по выполнению задания сделан неполный анализ, без учета ряда фактов, выявлены не все возможные проблемы, для решения могла быть выбрана второстепенная, а не главная проблема, количество представленных возможных вариантов решения – 2-3, затруднена четкая аргументация окончательного выбора одного из альтернативных решений. Оценка «удовлетворительно» – задание выполнено более чем на 2/3, но в рамках установленного на выступление регламента, студент(ы) расплывчато раскрывает решение, не может четко аргументировать сделанный выбор, показывает явный недостаток теоретических знаний. Выводы слабые, свидетельствуют о недостаточном анализе фактов, в основе решения может иметь место интерпретация фактов или предположения, собственная точка зрения на причины возникновения проблемы не обоснована или отсутствует. При устной презентации на вопросы отвечает с трудом или не отвечает совсем. Подготовленная презентация выполненного задания не структурирована. В случае письменной презентации по выполнению кейсзадания не сделан детальный анализ, далеко не все факты учтены, для решения выбрана второстепенная, а не главная проблема, количество представленных возможных вариантов решения – 1-2, отсутствует четкая аргументация окончательного выбора решения. Оценка «неудовлетворительно» – задание не выполнено, или выполнено менее чем на треть. Отсутствует детализация при анализе задания, изложение устное или письменное не структурировано. Если решение и обозначено в выступлении или отчете-презентации, то оно не является решением проблемы, которая заложена в задании.

5. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

5.1. Рекомендуемая литература

5.1.1. Основная литература

Л.1.1	Волк В. К. Базы данных. Проектирование, программирование, управление и администрирование [Электронный ресурс]:учебник для вузов. - Санкт-Петербург: Лань, 2023. - 244 с. – Режим доступа: https://e.lanbook.com/book/346439
Л.1.2	Мамедли Р. Э. Базы данных. Лабораторный практикум [Электронный ресурс]:учебное пособие для вузов. - Санкт-Петербург: Лань, 2023. - 152 с. – Режим доступа: https://e.lanbook.com/book/319400
Л.1.3	Волк В. К. Базы данных. Проектирование, программирование, управление и администрирование [Электронный ресурс]:учебник для спо. - Санкт-Петербург: Лань, 2022. - 340 с. – Режим доступа: https://e.lanbook.com/book/198584
Л.1.4	Кумскова И.А. Базы данных [Электронный ресурс]:Учебник. - Москва: КноРус, 2022. - 400 с. – Режим доступа: https://book.ru/book/943244
Л.1.5	Ратушняк Г.Я., Золкин А.Л., Никитин А.Л. Базы данных [Электронный ресурс]:Учебное пособие. - Москва: Русайнс, 2022. - 127 с. – Режим доступа: https://book.ru/book/947106
Л.1.6	Ратушняк Г.Я., Золкин А.Л., Никитин А.Л. Базы данных [Электронный ресурс]:Учебное пособие. - Москва: Русайнс, 2024. - 127 с. – Режим доступа: https://book.ru/book/951769
Л.1.7	Кумскова И.А. Базы данных [Электронный ресурс]:Учебник. - Москва: КноРус, 2024. - 400 с. – Режим доступа: https://book.ru/book/952917
Л.1.8	Волк В. К. Базы данных. Проектирование, программирование, управление и администрирование [Электронный ресурс]:учебник для спо. - Санкт-Петербург: Лань, 2024. - 340 с. – Режим доступа: https://e.lanbook.com/book/382310

Л.1.9	Васильева М. А., Филипченко К. М., Балакина Е. П. Информационное обеспечение систем управления. Проектирование базы данных с заданиями [Электронный ресурс]:учебник для вузов. - Санкт-Петербург: Лань, 2023. - 200 с. – Режим доступа: https://e.lanbook.com/book/339740
Л.1.10	Кондрашов Ю.Н. Язык SQL. Сборник ситуационных задач по дисциплине «Базы данных» [Электронный ресурс]:Учебно-практическое пособие. - Москва: Русайнс, 2023. - 125 с. – Режим доступа: https://book.ru/book/947081
Л.1.11	Кондрашов Ю.Н. Язык SQL. Сборник ситуационных задач по дисциплине «Базы данных» [Электронный ресурс]:Учебно-практическое пособие. - Москва: Русайнс, 2026. - 125 с. – Режим доступа: https://book.ru/book/959301
Л.1.12	Ратушняк Г.Я., Золкин А.Л., Никитин А.Л. Базы данных [Электронный ресурс]:Учебное пособие. - Москва: Русайнс, 2026. - 127 с. – Режим доступа: https://book.ru/book/960549
Л.1.13	Мамедли Р. Э. Базы данных. Лабораторный практикум [Электронный ресурс]:учебное пособие для спо. - Санкт-Петербург: Лань, 2026. - 152 с. – Режим доступа: https://e.lanbook.com/book/508780

5.2. Перечень программного обеспечения

Kaspersky Endpoint Security для бизнеса – Расширенный Russian Edition

5.3. Перечень информационных справочных систем

ПЛАТФОРМА ОНЛАЙН-ОБРАЗОВАНИЯ «РАЗУМ»
Научная электронная библиотека "eLIBRARY.RU"

6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Адрес: 215119, Смоленская область, Вяземский район, город Вязьма, ул. Ленина, д. 77: аудитория 18-1302 - Лаборатория
Учебная аудитория для проведения занятий лекционного типа; занятий семинарского типа, лабораторных и практических занятий; для курсового проектирования (выполнения курсовых работ); для проведения групповых и индивидуальных консультаций; для текущего контроля и промежуточной аттестации : Рабочее место преподавателя; маркерная доска; 10 рабочих мест обучающихся, оснащенных ПЭВМ с подключением к сети интернет и обеспечением доступа в электронную информационно-образовательную среду;

Адрес: 215119, Смоленская область, Вяземский район, город Вязьма, ул. Ленина, д. 77: аудитория 18-1310 - Лаборатория
Учебная аудитория для проведения занятий лекционного типа; занятий семинарского типа, лабораторных и практических занятий; для курсового проектирования (выполнения курсовых работ); для проведения групповых и индивидуальных консультаций; для текущего контроля и промежуточной аттестации : Рабочие места обучающихся; Рабочее место преподавателя; Классная доска; 15 рабочих мест обучающихся оснащенные ПЭВМ с подключением к сети интернет и обеспечением доступа в электронную информационно-образовательную среду; Ноутбук переносной; Переносной проектор; Переносной экран; Учебно-наглядные пособия

7. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ

Методические рекомендации по организации самостоятельной работы

СРС – планируемая учебная, выполняемая во внеаудиторное (аудиторное) время по заданию и при методическом руководстве преподавателя, но без его непосредственного участия (возможно частичное непосредственное участие преподавателя при сохранении ведущей роли студентов). Целью СРС является овладение фундаментальными знаниями, профессиональными умениями и навыками по профилю будущей специальности, опытом творческой, исследовательской деятельности, развитие самостоятельности, ответственности и организованности, творческого подхода к решению проблем учебного и профессионального уровней. Задачи СРС: систематизация и закрепление полученных теоретических знаний и практических умений студентов; углубление и расширение теоретической подготовки; формирование умений использовать нормативную, правовую, справочную документацию и специальную литературу; развитие познавательных способностей и активности студентов: творческой инициативы, самостоятельности, ответственности и организованности; формирование самостоятельности мышления, способностей к саморазвитию, самосовершенствованию и самореализации; развитие исследовательских умений; использование материала, собранного и полученного в ходе самостоятельных занятий на практических занятиях, при написании курсовых и выпускной квалификационной работ, для эффективной подготовки к текущей и промежуточной аттестации. Функции СРС: развивающая (повышение культуры умственного труда, приобщение к 10 творческим видам деятельности, обогащение интеллектуальных способностей студентов); информационно-обучающая (учебная деятельность студентов на аудиторных занятиях, неподкрепленная самостоятельной работой, становится мало результативной); ориентирующая и стимулирующая (процессу обучения придается ускорение и мотивация); воспитательная (формируются и развиваются профессиональные качества специалиста и гражданина); исследовательская (новый уровень профессионально-творческого мышления).

Самостоятельная работа студентов является обязательным компонентом учебного процесса для каждого студента и определяется учебным планом. Самостоятельная работа определяется как индивидуальная или коллективная учебная деятельность, осуществляемая без непосредственного руководства педагога, но по его заданиям и под его контролем. Самостоятельная работа – это познавательная учебная деятельность, когда последовательность мышления студента, его умственных и практических операций и действий зависит и определяется самим студентом.

Самостоятельная работа студентов способствует развитию самостоятельности, ответственности и организованности, творческого подхода к решению проблем учебного и профессионального уровня, что в итоге приводит к развитию навыка самостоятельного планирования и реализации деятельности. Целью самостоятельной работы студентов является овладение необходимыми компетенциями по своему направлению подготовки, опытом творческой и исследовательской деятельности. На основании компетентностного подхода к реализации профессиональных образовательных программ, видами заданий для самостоятельной работы являются:

- для овладения знаниями: чтение текста (учебника, первоисточника, дополнительной литературы), составление плана

текста, графическое изображение структуры текста, конспектирование текста, выписки из текста, работа со словарями и справочниками, ознакомление с нормативными документами, учебно-исследовательская работа, использование аудио- и видеозаписей, компьютерной техники и информационно-телекоммуникационной сети Интернет и др.
- для закрепления и систематизации знаний: работа с конспектом лекции, обработка текста (учебника, первоисточника, дополнительной литературы, аудио и видеозаписей), повторная работа над учебным материалом, составление плана, составление таблиц для систематизации учебного материала, ответ на контрольные вопросы, заполнение рабочей тетради, аналитическая обработка текста (аннотирование, рецензирование, реферирование, конспект-анализ и др.), завершение аудиторных практических работ и оформление отчётов по ним, подготовка мультимедиа сообщений/докладов к выступлению на семинаре (конференции), материалов-презентаций, подготовка реферата, составление библиографии, тематических кроссвордов, тестирование и др.
- для формирования умений: решение задач и упражнений по образцу, решение вариативных задач, выполнение чертежей, схем, выполнение расчетов (графических работ), решение ситуационных (профессиональных) задач, подготовка к деловым играм, проектирование и моделирование разных видов и компонентов профессиональной деятельности, рефлексивный анализ профессиональных умений с использованием аудио- и видеотехники и др.
Самостоятельная работа может осуществляться индивидуально или группами студентов в зависимости от цели, объема, конкретной тематики самостоятельной работы, уровня сложности, уровня умений студентов.

Методические рекомендации по освоению дисциплины (учебного предмета)

Методические рекомендации по подготовке к лекционным занятиям

Просмотрите конспект сразу после занятий. Пометьте материал конспекта лекций, который вызывает затруднения для понимания. Попытайтесь найти ответы на затруднительные вопросы, используя предлагаемую литературу. Если самостоятельно не удалось разобраться в материале, сформулируйте вопросы и обратитесь на текущей консультации или на ближайшей лекции за помощью к преподавателю. Каждую неделю рекомендуется отводить время для повторения пройденного материала, проверяя свои знания, умения и навыки по контрольным вопросам.
Работа с рекомендованной литературой:
При работе с основной и дополнительной литературой целесообразно придерживаться такой последовательности. Сначала прочитать весь заданный текст в быстром темпе. Цель такого чтения заключается в том, чтобы создать общее представление об изучаемом материале, понять общий смысл прочитанного. Затем прочитать вторично, более медленно, чтобы в ходе чтения понять и запомнить смысл каждой фразы, каждого положения и вопроса в целом. Чтение приносит пользу и становится продуктивным, когда сопровождается записями. Это может быть составление плана прочитанного текста, тезисы или выписки, конспектирование и др. Выбор вида записи зависит от характера изучаемого материала и целей работы с ним. Если содержание материала несложное, легко усваиваемое, можно ограничиться составлением плана. Если материал содержит новую и трудно усваиваемую информацию, целесообразно его законспектировать. План – это схема прочитанного материала, перечень вопросов, отражающих структуру и последовательность материала. Конспект – это систематизированное, логичное изложение материала источника. Различаются четыре типа конспектов:
- план-конспект – это развернутый детализированный план, в котором по наиболее сложным вопросам даются подробные пояснения,
- текстуальный конспект – это воспроизведение наиболее важных положений и фактов источника,
- свободный конспект – это четко и кратко изложенные основные положения в результате глубокого изучения материала, могут присутствовать выписки, цитаты, тезисы; часть материала может быть представлена планом,
- тематический конспект – составляется на основе изучения ряда источников и дает ответ по изучаемому вопросу. В процессе изучения материала источника и составления конспекта нужно обязательно применять различные выделения, подзаголовки, создавая блочную структуру конспекта. Это делает конспект легко воспринимаемым и удобным для работы.

Методические рекомендации по подготовке к практическим занятиям

Практические занятия представляют особую форму сочетания теории и практики. Их назначение – углубление проработки теоретического материала предмета путем регулярной и планомерной самостоятельной работы студентов на протяжении всего курса. Процесс подготовки к практическим занятиям включает изучение нормативных документов, обязательной и дополнительной литературы по рассматриваемому вопросу. Непосредственное проведение практического занятия предполагает, например:
- индивидуальные выступления студентов с сообщениями по какому-либо вопросу изучаемой темы;
- фронтальное обсуждение рассматриваемой проблемы, обобщения и выводы;
- решение задач и упражнений по образцу;
- решение вариантных задач и упражнений;
- решение ситуационных производственных (профессиональных) задач;
- проектирование и моделирование разных видов и компонентов профессиональной деятельности.
- выполнение контрольных работ;
- работу с тестами.
При подготовке к практическим занятиям студентам рекомендуется: внимательно ознакомиться с тематикой практического занятия; прочесть конспект лекции по теме, изучить рекомендованную литературу;
составить краткий план ответа на каждый вопрос практического занятия; проверить свои знания, отвечая на вопросы для самопроверки; если встретятся незнакомые термины, обязательно обратиться к словарю и зафиксировать их в тетради.
Все письменные задания выполнять в рабочей тетради. Практические занятия развивают у студентов навыки самостоятельной работы по решению конкретных задач.

Методические рекомендации по подготовке к лабораторным работам

Лабораторные работы представляют одну из форм освоения теоретического материала с одновременным формированием практических навыков в изучаемой дисциплине. Их назначение – углубление проработки теоретического материала, формирование практических навыков путем регулярной и планомерной самостоятельной работы студентов на протяжении всего курса. Процесс подготовки к лабораторным работам включает изучение нормативных документов, обязательной и дополнительной литературы по рассматриваемому вопросу. Непосредственное проведение лабораторной работы предполагает:

- изучение теоретического материала по теме лабораторной работы (по вопросам изучаемой темы);
- выполнение необходимых расчетов и экспериментов;
- оформление отчета с заполнением необходимых таблиц, построением графиков, подготовкой выводов по проделанным экспериментам и теоретическим расчетам;
- по каждой лабораторной работе проводится контроль: проверяется содержание отчета, проверяется усвоение теоретического материала. Контроль усвоения теоретического материала является индивидуальным.

Методические указания по выполнению отчёта к лабораторным работам

Основным требованием по выполнению лабораторных и практических работ является полное исчерпывающее описание всей проделанной работы, позволяющее судить о полученных результатах, степени выполнения и профессиональной подготовки студентов.
Методические указания обеспечивают комплексный подход в учебной работе студентов, единство и преемственность требований к оформлению результатов работы на разных этапах обучения. С единых позиций приведены основные требования по структуре, оформлению и содержанию отчета по лабораторным и практическим работам.
Структура отчёта:
- цель работы;
- краткие теоретические сведения;
- ход выполнения работы;
- выводы.

Дополнительными элементы:
- приложения;
- библиографический список.

Требования к содержанию отчёта:
1. Титульный лист
В верхнем поле листа указывают полное наименование учебного заведения.
В среднем поле указывается вид работы, в данном случае лабораторная или практическая работа с указанием курса, по которому она выполнена, и ниже ее название. Название работы приводится без слова тема и в кавычки не заключается.
Далее ближе к правому краю титульного листа указывают фамилию, инициалы и группу учащегося, выполнившего работу, а также фамилию, инициалы преподавателя, принявшего работу.
В нижнем поле листа указывается место выполнения работы и год ее написания (без слова год).
2. Цель работы должна отражать тему работы, а также конкретные задачи, поставленные студенту на период выполнения работы. По объему цель работы в зависимости от сложности и многозадачности работы составляет от нескольких строк до 0,5 страницы.
3. Краткие теоретические сведения. В этом разделе излагается краткое теоретическое описание изучаемой в работе темы. Материал раздела не должен копировать содержание методического пособия или учебника по данной теме, а ограничивается изложением основных понятий, требующихся для дальнейшей обработки полученных результатов. Объем литературного обзора не должен превышать 1/3 части всего отчета.
4. Ход выполнения работы. В данном разделе подробно излагается методика выполнения работы, процесс получения данных и способ их обработки. Если используются стандартные пакеты компьютерных программ для обработки экспериментальных результатов, то необходимо обосновать возможность и целесообразность их применения, а также подробности обработки данных с их помощью.
5. Выводы по работе - кратко излагаются результаты работы, полученные в результате выполнения работы, а также краткий анализ полученных результатов.
Для защиты лабораторной работы студент должен подготовить отчет, провести самостоятельную работу, иметь отметку о проверенном отчете.
Результаты определяются по пятибалльной системе оценок.

Методические рекомендации по выполнению реферата

Реферат – письменная работа объемом 8–10 страниц. Это краткое и точное изложение сущности какого-либо вопроса, темы. Тему реферата студент выбирает из предложенных преподавателем или может предложить свой вариант. В реферате нужны развернутые аргументы, рассуждения, сравнения. Содержание темы излагается объективно от имени автора. Функции реферата. Информативная, поисковая, справочная, сигнальная, коммуникативная. Степень выполнения этих функций зависит от содержательных и формальных качеств реферата и для каких целей их использует. Требования к языку реферата.
Должен отличаться точностью, краткостью, ясностью и простотой.
Структура реферата:
1. Титульный лист
2. Оглавление (на отдельной странице). Указываются названия всех разделов (пунктов плана) реферата и номера страниц, указывающие начало этих разделов в тексте реферата.

3. Введение. Аргументируется актуальность исследования, т.е. выявляется практическое и теоретическое значение данного исследования. Далее констатируется, что сделано в данной области предшественниками, перечисляются положения, которые должны быть обоснованы. Обязательно формулируются цель и задачи реферата.
4. Основная часть. Подчиняется собственному плану, что отражается в разделении текста на главы, параграфы, пункты. План основной части может быть составлен с использованием различных методов группировки материала. В случае если используется чья-либо неординарная мысль, идея, то обязательно нужно сделать ссылку на того автора, у кого взят данный материал.
5. Заключение. Последняя часть научного текста. В краткой и сжатой форме излагаются полученные результаты, представляющие собой ответ на главный вопрос исследования.
6. Приложение. Может включать графики, таблицы, расчеты.
7. Библиография (список литературы). Указывается реально использованная для написания реферата литература. Названия книг располагаются по алфавиту с указанием их выходных данных. Общие требования к построению, содержанию и оформлению».
При проверке реферата оцениваются:
- знание фактического материала, усвоение общих представлений, понятий, идей;
- характеристика реализации цели и задач исследования;
- степень обоснованности аргументов и обобщений;
- качество и ценность полученных результатов;
- использование литературных источников;
- культура письменного изложения материала;
- культура оформления материалов работы.

Методические рекомендации по устному опросу/самоподготовке

После изучения определенной темы по записям в конспекте и учебнику, а также решения достаточного количества соответствующих задач на практических занятиях и самостоятельно студенту рекомендуется, используя лист опорных сигналов, воспроизвести по памяти определения, выводы формул, формулировки основных положений и доказательств. В случае необходимости следует рекомендовать еще раз внимательно разобраться в материале. Иногда недостаточность усвоения того или иного вопроса выясняется только при изучении дальнейшего материала. В этом случае надо вернуться назад и повторить плохо усвоенный материал. Важный критерий усвоения теоретического материала – умение решать задачи или пройти тестирование по пройденному материалу. Однако преподавателю следует помнить, что правильное решение задачи может получиться в результате применения механически заученных формул без понимания сущности теоретических положений.

Методические рекомендации по подготовке к семинарским занятиям

Одним из видов внеаудиторной самостоятельной работы является подготовка к семинарским занятиям. Семинар – форма учебно-практических занятий, при которой студенты обсуждают сообщения, доклады и рефераты, выполненные ими по результатам учебных или научных исследований под руководством преподавателя. Преподаватель в этом случае является координатором обсуждений темы семинара, подготовка к которому является обязательной. Поэтому тема семинара и основные источники обсуждения предъявляются до обсуждения для детального ознакомления, изучения. Цели обсуждений направлены на формирование навыков профессиональной полемики и закрепление обсуждаемого материала. Семинар – это такая форма организации обучения, при которой на этапе подготовки доминирует самостоятельная работа учащихся с
учебной литературой и другими дидактическими средствами над серией вопросов, проблем и задач, а в процессе семинара идут активное обсуждение, дискуссии и выступления учащихся, где они под руководством преподавателя делают обобщающие выводы и заключения. Семинар предназначен для углубленного изучения дисциплины, овладения методологией научного познания, то главная цель семинарских занятий – обеспечить студентам возможность овладеть навыками и умениями использования теоретического знания применительно к особенностям изучаемой отрасли.

Методические рекомендации по подготовке эссе

Одним из видов самостоятельной работы студентов является написание творческой работы по заданной либо согласованной с преподавателем теме. Творческая работа (эссе) представляет собой оригинальное произведение объемом 500-700 слов, посвященное какой-либо значимой классической либо современной проблеме в определенной теоретической и практической области. Творческая работа не является рефератом и не должна носить описательный характер, большое место в ней должно быть уделено аргументированному представлению своей точки зрения студентами, критической оценке рассматриваемого материала и проблематики, что должно способствовать раскрытию творческих и аналитических способностей. Цели написания эссе – научиться логически верно и аргументировано строить устную и письменную речь;
работать над углублением и систематизацией своих философских знаний; овладеть способностью использовать основы знаний для формирования мировоззренческой позиции. Приступая к написанию эссе, изложите в одном предложении, что именно вы будете утверждать и доказывать (свой тезис).

Методические рекомендации по подготовке к докладу

Для подготовки доклада необходимо выбрать актуальную тему. Желательно, чтобы тема была интересна докладчику и вызывала желание качественно подготовить материалы. Подготовка доклада предполагает: определение цели доклада; подбор необходимого материала, определяющего содержание доклада; составление плана доклада, распределение собранного материала в необходимой логической последовательности.
Композиция доклада имеет вступление, основную часть и заключение.
Вступление должно содержать: название доклада; сообщение основной идеи; современную оценку предмета изложения;

краткое перечисление рассматриваемых вопросов; интересную для слушателей форму изложения. Основная часть, в которой необходимо раскрыть суть темы, обычно строится по принципу отчёта. Задача основной части: представить достаточно данных для того, чтобы слушатели заинтересовались темой.
Заключение – чёткое обобщение и краткие выводы по излагаемой теме.

Методические рекомендации по подготовке к собеседованию

Собеседование – средство контроля, организованное как специальная беседа преподавателя с обучающимся на темы, связанные с изучаемой дисциплиной, и рассчитанное на выяснение объема знаний обучающегося по определенному разделу, теме, проблеме и т.п.
Цель собеседования: проверка усвоения знаний; умений применять знания; сформированности профессионально значимых личностных качеств.
Подготовка к собеседованию предполагает повторение пройденного материала и приобретение навыка свободного владения терминологией и фактическими данными по определенному разделу дисциплины.

Методические рекомендации по подготовке к тестированию/ итоговому тестированию

Тестирование/итоговое тестирование – это система стандартизированных заданий, определенного содержания и упорядоченных в рамках определенной стратегии предъявления, позволяющая качественно оценить структуру и эффективно измерить уровень знаний, умений и навыков по учебной дисциплине (разделу). Тестирование/итоговое тестирование позволяет оценить у студентов уровень знаний фактического материала, умения логически мыслить, способности к рефлексии и творческому подходу к решению поставленной задачи на каждом этапе их обучения.
При самостоятельной подготовке к тестированию/итоговому тестированию студенту необходимо:
- четко выяснить все условия тестирования/итогового тестирования заранее (сколько тестов будет предложено, сколько времени отводится на тестирование/итоговое тестирование, какова система оценки результатов);
- проконсультироваться с преподавателем по вопросу выбора учебной литературы;
- проработать информационный материал по разделу дисциплине;
- конспективно выписать из соответствующих разделов учебной дисциплины основные понятия, ключевые идеи, теоретические концепции и подходы;
- повторить теоретический материал лекций, самостоятельно изученной учебной и научной литературы, семинарских занятий.
При выполнении тестовых заданий рекомендуется:
- внимательно полностью прочитать вопрос и предлагаемые варианты ответов, выбрать правильные из них (их может быть несколько);
- при затруднении при ответе на сложный для студента вопрос целесообразно переходить к другим вопросам теста. По завершению ответов на все последующие вопросы теста необходимо вернуться к трудному вопросу и дать на него продуманный ответ;
- по окончании тестирования/итогового тестирования следует проверить правильность данных ответов на вопросы теста, чтобы избежать механических ошибок.

Методические рекомендации по подготовке к экзамену

Подготовка к экзамену 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 экзамене студент демонстрирует то, что он приобрел в процессе обучения по конкретной учебной дисциплине. Экзаменационная сессия – это серия экзаменов, установленных учебным планом. Между экзаменами интервал 2-4 дня, в течение студент систематизирует уже имеющиеся знания. На консультации перед экзаменом студенты должны
быть ознакомлены с основными требованиями и получить ответы на возникающие в процессе подготовки
вопросы. Необходимо ориентировать студентов на систематическую подготовку к занятиям в течение семестра, что позволит использовать время экзаменационной сессии для систематизации знаний.

Методические рекомендации по подготовке к зачету/ зачету с оценкой

В ходе подготовки к зачету/зачету с оценкой студент, в первую очередь, должен систематизировать знания, полученные в ходе изучения дисциплины. К зачету/зачету с оценкой необходимо готовиться целенаправленно, регулярно, систематически и с первых дней обучения по данной дисциплине. В самом начале учебного курса познакомьтесь со следующей учебно- методической документацией:
- программой дисциплины;
- перечнем знаний, умений и навыков, которыми студент должен владеть;
- тематическими планами лекций, семинарских занятий;
- учебниками, учебными пособиями по дисциплине, а также электронными ресурсами;
- перечнем вопросов к зачету/зачету с оценкой.
После этого у обучающихся должно сформироваться четкое представление об объеме и характере знаний и умений, которыми надо будет овладеть по дисциплине. Систематическое выполнение учебной работы на лекциях и лабораторных занятиях позволит успешно освоить дисциплину и создать хорошую базу для сдачи зачета/зачета с оценкой.

Методические рекомендации по выполнению курсовой работы/ курсового проекта

Курсовая работа/курсовой проект - это учебно-исследовательский труд студента, выполняемый по определённой дисциплине под руководством научного руководителя. Её цель - закрепить и углубить теоретические знания, развить

навыки самостоятельного анализа, работы с научной литературой и эмпирическими данными, а также сформировать умения структурировать информацию и формулировать обоснованные выводы.
Курсовая работа/курсовой проект – письменная работа объемом 30–50 страниц, выполняется в несколько этапов:
1. Подготовительный этап
2. Сбор и анализ информации
3. Написание работы
Структура курсовой работы/курсового проекта:
1. Титульный лист – по шаблону вуза.
2. Содержание – разделы с номерами страниц.
3. Введение: актуальность, цель, задачи, объект, предмет, методы, значимость, структура.
4. Основная часть:
о Глава 1 (теоретическая) – анализ концепций, выводы.
о Глава 2 (аналитическая/практическая) – анализ объекта, расчёты, данные, выводы.
о Глава 3 (рекомендательная, если нужно) – предложения, прогнозы, выводы.
5. Заключение – выводы по задачам, итоги, перспективы.
6. Список литературы – группируйте по типам, оформляйте по ГОСТ, указывайте только использованные источники.
7. Приложения (если нужно) – таблицы, графики, анкеты и т. д. (нумеруются, есть ссылки в тексте).
4. Оформление
5. Проверка и доработка
6. Подготовка к защите
• Доклад (5–7 мин.):
• Презентация.

Методические рекомендации по подготовке к зачету/ зачету с оценкой

В ходе подготовки к зачету/зачету с оценкой студент, в первую очередь, должен систематизировать знания, полученные в ходе изучения дисциплины. К зачету/зачету с оценкой необходимо готовиться целенаправленно, регулярно, систематически и с первых дней обучения по данной дисциплине. В самом начале учебного курса познакомьтесь со следующей учебно- методической документацией:
- программой дисциплины;
- перечнем знаний, умений и навыков, которыми студент должен владеть;
- тематическими планами лекций, семинарских занятий;
- учебниками, учебными пособиями по дисциплине, а также электронными ресурсами;
- перечнем вопросов к зачету/зачету с оценкой.
После этого у обучающихся должно сформироваться четкое представление об объеме и характере знаний и умений, которыми надо будет овладеть по дисциплине. Систематическое выполнение учебной работы на лекциях и лабораторных занятиях позволит успешно освоить дисциплину и создать хорошую базу для сдачи зачета/зачета с оценкой.

8. ОРГАНИЗАЦИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ДЛЯ ЛИЦ С ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ ЗДОРОВЬЯ И ИНВАЛИДОВ

В образовательном процессе используются социально-активные и рефлексивные методы обучения, технологии социокультурной реабилитации с целью оказания помощи в установлении полноценных межличностных отношений с другими студентами, создании комфортного психологического климата в студенческой группе.
Студенты с ограниченными возможностями здоровья, в отличие от остальных студентов, имеют свои специфические особенности восприятия, переработки материала. Подбор и разработка учебных материалов производится с учетом индивидуальных особенностей.
Предусмотрена возможность обучения по индивидуальному графику, при составлении которого возможны различные варианты проведения занятий: в академической группе и индивидуально, на дому с использованием дистанционных образовательных технологий.
Основной формой в дистанционном обучении является индивидуальная форма обучения. Главным достоинством индивидуального обучения для инвалидов и лиц с ограниченными возможностями здоровья является то, что оно позволяет полностью индивидуализировать содержание, методы и темпы учебной деятельности инвалида, следить за каждым его действием и операцией при решении конкретных задач; вносить вовремя необходимые коррекции как в деятельность студента-инвалида, так и в деятельность преподавателя. Дистанционное обучение также обеспечивает возможности коммуникаций не только с преподавателем, но и с другими обучаемыми, сотрудничество в процессе познавательной деятельности.
При изучении дисциплины используются следующие организационные мероприятия:

- использование возможностей сети «Интернет» для обеспечения связи с обучающимися, предоставления им необходимых материалов для самостоятельного изучения, контроля текущей успеваемости и проведения тестирования;

- проведение видеоконференций, лекций, консультаций, и т.д. с использованием программ, обеспечивающих дистанционный контакт с обучающимся в режиме реального времени.
- предоставление электронных учебных пособий, включающих в себя основной материал по дисциплинам, включенным в ОП;

- проведение занятий, консультаций, защит курсовых работ и т.д. на базе консультационных пунктов, обеспечивающих условия для доступа туда лицам с ограниченными возможностями;

- предоставление видеолекций, позволяющих изучать материал курса дистанционно.
- использование программного обеспечения и технических средств, имеющих функции адаптации для использования лицами с ограниченными возможностями.

Актуализация с учетом развития науки, техники, культуры, экономики, техники, технологий и социальной сферы

Руководитель ОП _____

Рабочая программа актуализирована, обсуждена и одобрена на заседании ПЦК _____
Протокол от __ _____ 2026 г. № __
Председатель ПЦК _____

==========

Актуализация с учетом развития науки, техники, культуры, экономики, техники, технологий и социальной сферы

Руководитель ОП _____

Рабочая программа актуализирована, обсуждена и одобрена на заседании ПЦК _____
Протокол от __ _____ 2027 г. № __
Председатель ПЦК _____

==========

Актуализация с учетом развития науки, техники, культуры, экономики, техники, технологий и социальной сферы

Руководитель ОП _____

Рабочая программа актуализирована, обсуждена и одобрена на заседании ПЦК _____
Протокол от __ _____ 2028 г. № __
Председатель ПЦК _____

==========

Актуализация с учетом развития науки, техники, культуры, экономики, техники, технологий и социальной сферы

Руководитель ОП _____

Рабочая программа актуализирована, обсуждена и одобрена на заседании ПЦК _____
Протокол от __ _____ 2029 г. № __
Председатель ПЦК _____

==========

ПРИЛОЖЕНИЕ А

ЗАДАНИЯ НА ПРАКТИЧЕСКИЕ РАБОТЫ

РАЗДЕЛ 1. ОСНОВЫ ТЕОРИИ БАЗ ДАННЫХ

Практическая работа №1

Тема: Основные понятия и архитектуры систем баз данных.

Цель: Закрепить теоретические знания об основных понятиях баз данных (БД, СУБД, данные, метаданные, модель данных), компонентах системы баз данных, принципах централизованного управления и архитектуре систем баз данных, а также научиться анализировать и классифицировать элементы реальных информационных систем с точки зрения теории баз данных.

Задание:

1. Используя конспект лекций и дополнительные источники, выполните следующие задания.

Дайте определения следующим терминам (по 1-2 предложения на каждый):

- база данных,
- система управления базами данных (СУБД),
- данные,
- метаданные,
- модель данных.

Приведите по одному примеру из реальной жизни для каждого из следующих понятий:

- данные,
- метаданные (например, для файла или таблицы),
- иерархическая модель данных,
- сетевая модель данных,
- реляционная модель данных.

2. Рассмотрите любую знакомую вам информационную систему, использующую базу данных (например, система учёта успеваемости в учебном заведении, интернет-магазин, библиотечная система, система бронирования билетов).

Для выбранной системы перечислите и кратко опишите следующие компоненты системы баз данных (в виде таблицы):

Компонент системы БД	Что (или кто) в вашей системе выполняет эту роль?	Краткое описание
Пользователи (конечные)		
Прикладные программы		
СУБД		
Базы данных (физически)		
Администратор баз данных		

3. На примере той же выбранной системы ответьте письменно на вопросы (по 2–3 предложения на каждый):

1. Какие преимущества даёт централизованное управление данными в этой системе по сравнению с возможным децентрализованным хранением (например, файлами на отдельных компьютерах)?

2. Насколько полно в этой системе реализованы следующие требования к централизованным базам данных (поставьте «+» или «–» и кратко поясните):

- минимизация избыточности (дублирования) данных,
- непротиворечивость (целостность) данных,
- безопасность и разграничение доступа,
- независимость данных от прикладных программ,
- возможность коллективного доступа.

4. Установите соответствие между уровнями архитектуры (внешний, концептуальный, внутренний) и следующими описаниями (напишите название уровня рядом с описанием):

Описание	Уровень архитектуры (выберите)
Как данные физически хранятся на диске (блоки, индексы, файлы)	

Представление данных «глазами» конкретного пользователя или приложения (например, только нужные поля)	
Общее логическое описание всей базы данных без привязки к физическому хранению	

Форма сдачи отчёта: письменный отчёт (в тетради) с выполненными заданиями.

Практическая работа №2

Тема: Сравнительный анализ видов моделей данных.

Цель: Изучить основные виды моделей данных, научиться различать их по структуре, свойствам и областям применения, а также приобрести навыки представления предметной области в виде различных моделей данных.

Задание:

1. Дайте краткие определения (по 1-2 предложения) следующим видам моделей данных:

- иерархическая модель данных,
- сетевая модель данных,
- реляционная модель данных,
- объектно-ориентированная модель данных,
- документоориентированная модель данных (NoSQL).

2. Заполните сравнительную таблицу:

Характеристика	Иерархическая	Сетевая	Реляционная	Объектно-ориентированная
Основная структурная единица				
Форма представления связей между данными				
Поддержка языка запросов (пример)				
Пример СУБД (название)				
Основная область применения				

3. Дана предметная область: «Учебный процесс в университете»

Сущности: СТУДЕНТ, ГРУППА, ПРЕПОДАВАТЕЛЬ, ДИСЦИПЛИНА, ЗАЧЁТНАЯ ВЕДОМОСТЬ

Связи:

- студент учится в одной группе,
- в группе учится много студентов,
- преподаватель ведёт несколько дисциплин,
- дисциплину может вести один преподаватель,
- зачётная ведомость связывает студента, дисциплину и оценку.

3.1. Представьте данную предметную область в виде:

- иерархической модели (нарисуйте дерево, укажите корень и уровни),
- сетевой модели (нарисуйте граф с вершинами и дугами),
- реляционной модели (укажите набор таблиц, их поля и связи между ними, включая первичные и внешние ключи),
- объектно-ориентированной модели (укажите классы, их атрибуты и методы хотя бы для одного из классов).

3.2. Для каждой из четырёх моделей укажите одно преимущество и одно ограничение применительно к данной предметной области.

4. Дана ситуация:

Компания разрабатывает информационную систему для крупного логистического центра. Требуется хранить данные о:

- складских помещениях (ячейки вложены друг в друга: зона → стеллаж → полка → ячейка),
- маршрутах доставки (склады связаны между собой сложной сетью дорог с возможностью объезда),
- товарах (каждый товар имеет множество характеристик, которые различаются для разных категорий),

– клиентах и заказах (стандартные табличные данные).

Какую модель данных (или комбинацию моделей) вы предложите для каждой перечисленной задачи? Обоснуйте свой выбор (2-3 предложения на каждый пункт).

Оформить в виде таблицы:

Компонент системы	Рекомендуемая модель данных	Краткое обоснование
Хранение структуры склада		
Хранение маршрутов доставки		
Хранение характеристик товаров		
Хранение клиентов и заказов		

Форма сдачи отчёта: письменный отчёт (в тетради) с выполненными заданиями.

Практическая работа №3

Тема: Основы реляционной алгебры: традиционные и специальные операции над отношениями.

Цель: Изучить базовые операции реляционной алгебры (традиционные: объединение, пересечение, разность, декартово произведение; специальные: выборка, проекция, соединение, деление), а также операции, дополненные Дейтом, научиться применять их для формулирования запросов к реляционным базам данных.

Задание:

1. Дайте краткие определения (по 1-2 предложения) следующим операциям реляционной алгебры:

– объединение отношений,
– пересечение отношений,
– разность отношений,
– декартово произведение,
– выборка (ограничение, σ),
– проекция (π),
– естественное соединение (⋈),
– деление отношений (÷).

2. Заполните классификационную таблицу:

Операция	Тип (традиционная, специальная, дополненная Дейтом)	Какие отношения совместимы	Результат операции
Объединение			
Пересечение			
Разность			
Декартово произведение			
Выборка			
Проекция			
Соединение			
Деление			

3. Даны два отношения (таблицы):

R1 (СТУДЕНТЫ)

Номер зачётной книжки	Фамилия	Группа
101	Иванов	ИТ-21
102	Петров	ИТ-21
103	Сидорова	ИС-22
104	Козлов	ИТ-21

R2 (ОТЛИЧНИКИ)

Номер зачётной книжки	Фамилия	Группа
101	Иванов	ИТ-21
103	Сидорова	ИС-22
105	Морозова	ИТ-22

3.1. Выполните следующие операции (запишите результирующие таблицы):
– R1 ∪ R2 (объединение),
– R1 ∩ R2 (пересечение),
– R1 \ R2 (разность),
– R2 \ R1 (разность).
3.2. Для операции выборки (σ) выполните:
– выбрать из R1 студентов группы «ИТ-21» (результат – таблица).
3.3. Для операции проекции (π) выполните:
– проецировать R1 на атрибуты «Фамилия, Группа» (результат – таблица).

4. Даны два отношения:

R3 (СТУДЕНТЫ_ДИСЦИПЛИНЫ)

Номер студента	Код дисциплины
101	МАТ
101	ФИЗ
102	МАТ
103	ФИЗ
103	ПРОГ

R4 (ДИСЦИПЛИНЫ_СЕМЕСТРЫ)

Код дисциплины	Семестр
МАТ	1
ФИЗ	1
ПРОГ	2

4.1. Выполните естественное соединение R3 ⋈ R4 (запишите результирующую таблицу).
4.2. Выполните операцию деления R3 на отношение, содержащее только дисциплину «МАТ» (найти студентов, которые сдают математику)

5. Ситуация:

По мнению Дейта, реляционная алгебра может быть дополнена такими операциями, как полусоединение (⋉), антиполусоединение (▷) и внешнее соединение (⟕, ⟖, ⟗).

5.1. Используя R3 и R4, выполните (описательно, без строгой таблицы, но с объяснением):
– какие кортежи войдут в результат полусоединения R3 ⋉ R4 (проекция R3 на те кортежи, которые имеют соответствие в R4)?
5.2. Объясните одним предложением, чем левое внешнее соединение отличается от естественного соединения.

6. Для базы данных «Университет» (отношения: СТУДЕНТЫ, ДИСЦИПЛИНЫ, ЭКЗАМЕНЫ) запишите последовательность операций реляционной алгебры для следующих запросов:
– найти фамилии студентов, сдававших экзамен по дисциплине «Математика»,
– найти номера студентов, которые не сдавали ни одного экзамена,
– найти список дисциплин, которые сдавал студент с номером 101.

Форма сдачи отчёта: письменный отчёт (в тетради) с выполненными заданиями.

РАЗДЕЛ 2. ПРОЕКТИРОВАНИЕ БАЗ ДАННЫХ

Практическая работа №4

Тема: Построение ER-диаграмм в различных нотациях.

Цель: Изучить основы инфологического (концептуального) проектирования баз данных, освоить построение ER-диаграмм в нотациях Чена, Баркера и IDEF1X, а также приобрести навыки преобразования ER-диаграмм в реляционные отношения (таблицы).

Задание:

1. Дайте краткие определения (по 1-2 предложения) следующим понятиям:

- инфологическое (концептуальное) проектирование базы данных,
- ER-диаграмма (Entity-Relationship diagram),
- сущность (entity),
- атрибут сущности,
- связь (relationship) между сущностями,
- ключевой атрибут (идентификатор).

2. Заполните таблицу «Этапы построения ER-диаграммы»

№ этапа	Название этапа	Краткое описание действий
1		
2		
3		
4		
5		

3. Сравнение нотаций ER-диаграмм

Заполните сравнительную таблицу:

Характеристика	Нотация Чена	Нотация Баркера	Нотация IDEF1X
Обозначение сущности			
Обозначение атрибута			
Обозначение ключевого атрибута			
Обозначение связи один-ко-многим (1:N)			
Обозначение обязательности связи			
Где преимущественно используется			

4. Описание предметной области «Библиотека»

Библиотека содержит множество книг.

Каждая книга имеет уникальный инвентарный номер, название, автора, год издания.

Читатель имеет уникальный номер читательского билета, фамилию, адрес, телефон.

Читатель может взять несколько книг (одновременно не более 5).

Книга может быть выдана одному читателю или находиться в библиотеке.

Выдача книги фиксируется с указанием даты выдачи и планируемой даты возврата.

4.1. Выделите все сущности предметной области (не менее 3).

4.2. Для каждой сущности определите атрибуты и укажите ключевой атрибут.

4.3. Определите связи между сущностями, укажите тип связи (1:1, 1:N, N:M) и степень обязательности.

4.4. Нарисуйте ER-диаграмму в нотации Чена (вручную или с помощью инструментов).

4.5. Перерисуйте ER-диаграмму из части 4 в нотации Баркера (с использованием прямоугольников со скруглёнными углами, глагольных фраз, символов «палочка и вилка» и «круг и палочка» для обозначения обязательности).

4.6. Перерисуйте ER-диаграмму из части 4 в нотации IDEF1X (с использованием прямоугольников для независимых и зависимых сущностей, линий с точкой и ромбами на концах).

5. Преобразование ER-диаграммы в реляционные отношения

5.1. На основе ER-диаграммы из части 4 (Библиотека) выполните преобразование в набор реляционных таблиц (отношений) по следующим правилам:

- каждая сущность становится таблицей,
- атрибуты сущности становятся полями таблицы,
- ключевой атрибут становится первичным ключом,
- связи реализуются через внешние ключи.

5.2. Для каждой полученной таблицы укажите:

- имя таблицы,

– список полей с типами данных (на уровне описания: число, строка, дата),
– первичный ключ,
– внешние ключи (если есть).

5.3. Для связи «многие-ко-многим» (если она есть в вашей диаграмме) опишите, какая дополнительная таблица (связующая) создаётся.

6. Дана ситуация:

Разрабатывается база данных для интернет-магазина. Требуется хранить информацию о:

– Товарах (название, цена, остаток на складе, категория),
– Клиентах (логин, пароль, email, адрес доставки),
– Заказах (дата заказа, статус, общая сумма),
– Позициях заказа (какой товар, в каком количестве, по какой цене на момент заказа).

6.1. Выполните инфологическое проектирование (письменно, по шагам):

– перечислите все сущности (не менее 4),
– для каждой сущности перечислите атрибуты и выделите ключевые,
– определите все связи между сущностями и их типы,
– нарисуйте ER-диаграмму в любой нотации (по выбору).

6.2. Выполните преобразование полученной ER-диаграммы в реляционные отношения (перечислите таблицы с полями, первичными и внешними ключами).

7. Выберите любую реальную предметную область из вашего окружения (например, «Расписание занятий», «Учёт домашних питомцев в ветклинике», «Каталог фильмов», «Меню ресторана»). Самостоятельно выполните полный цикл инфологического проектирования:

– словесное описание предметной области,
– список сущностей, атрибутов и связей,
– ER-диаграмма в нотации Чена,
– ER-диаграмма в нотации Баркера,
– преобразование в реляционные отношения (не менее 3 таблиц).

Форма сдачи: отчёт в электронном документе с выполненными заданиями (таблицы, рисунки, схемы ER-диаграмм, текстовые пояснения).

Практическая работа №5

Тема: Метод функционального моделирования SADT (IDEF0). DFD-диаграммы. Введение в UML.

Цель: Изучить методологию функционального моделирования IDEF0, освоить основные элементы IDEF0-диаграмм, ознакомиться с DFD-диаграммами, получить базовые знания о языке UML и его видах диаграмм, а также научиться строить диаграммы вариантов использования.

Задание:

1. Дайте краткие определения (по 1-2 предложения) следующим понятиям:

– методология SADT (Structured Analysis and Design Technique),
– IDEF0 (Integration Definition for Function Modeling),
– контекстная диаграмма (диаграмма A-0),
– декомпозиция в IDEF0,
– DFD-диаграмма (Data Flow Diagram),
– UML (Unified Modeling Language),
– диаграмма вариантов использования (Use Case Diagram).

2. Основные элементы IDEF0-диаграмм

2.1. Изучите семантику IDEF0 и заполните таблицу «Стороны функционального блока»:

Сторона блока	Тип стрелки	Что обозначает (какой аспект функции)	Пример для процесса «приготовление обеда»
Левая сторона			
Верхняя сторона			
Правая сторона			
Нижняя сторона			

2.2. Заполните таблицу «Типы связей между блоками в IDEF0»:

Тип связи	Характер связи	Пример
Связь по входу		
Связь по управлению		

Обратная связь по входу		
Обратная связь по управлению		
Связь выход-механизм		

2.3. Ответьте на вопросы (2–3 предложения каждый):

- Что означает «туннелирование» стрелки в IDEF0 и как оно графически обозначается?
- Для чего используется «дерево узлов» (Node Tree) в IDEF0?
- Что такое ICOM-коды и для чего они служат?

3. Построение IDEF0-диаграммы (контекстной и первого уровня декомпозиции)

Описание предметной области «Обработка заказа в интернет-магазине»:

- покупатель размещает заказ через веб-сайт,
- менеджер проверяет наличие товаров на складе,
- при наличии всех товаров формируется счёт и заказ передаётся в отдел доставки,
- при отсутствии некоторых товаров оформляется заказ поставщику,
- после комплектации заказа товары отгружаются покупателю,
- система управления складом предоставляет данные об остатках,
- бухгалтерия контролирует оплату заказа.

3.1. Выделите следующие элементы для модели IDEF0:

- основную функцию (блок A0) – сформулируйте глагольным оборотом,
- входы (что преобразуется),
- выходы (результат функции),
- управления (правила, стандарты, регламенты),
- механизмы (ресурсы, оборудование, персонал).

3.2. Нарисуйте контекстную диаграмму (A-0), содержащую один функциональный блок со всеми граничными стрелками (входы, выходы, управления, механизмы).

3.3. Выполните декомпозицию контекстного блока на 3–5 дочерних функций. Нарисуйте диаграмму первого уровня декомпозиции (A0), показав связи между дочерними блоками с помощью стрелок (входы, выходы, управления, механизмы, обратные связи).

4. DFD-диаграммы (Data Flow Diagrams)

4.1. Заполните сравнительную таблицу «IDEF0 и DFD»:

Характеристика	IDEF0	DFD
Основной объект моделирования		
Типы стрелок (потоков)		
Какие элементы могут быть декомпозированы		
Основное назначение		

4.2. Для той же предметной области «Обработка заказа в интернет-магазине» перечислите внешние сущности (источники и приёмники данных) и потоки данных между ними и процессами (в виде списка).

4.3. Нарисуйте DFD-диаграмму первого уровня (контекстную) для обработки заказа, используя следующие нотационные элементы:

- процесс (круг или прямоугольник со скруглёнными углами),
- внешняя сущность (квадрат или прямоугольник),
- поток данных (стрелка с подписью),
- хранилище данных (открытый прямоугольник).

5. Введение в UML. Виды UML-диаграмм

5.1. Заполните таблицу «Классификация UML-диаграмм» (укажите не менее 8 видов диаграмм и их назначение):

Вид диаграммы (на русском)	Название на английском	Группа (структурная / поведенческая)	Краткое назначение

5.2. Перечислите основные элементы (сущности) диаграммы классов и кратко опишите каждый.

5.3. Перечислите основные элементы диаграммы последовательности и кратко опишите каждый.

6. Диаграммы вариантов использования

6.1. Изучите нотацию диаграммы вариантов использования и заполните таблицу «Элементы Use Case Diagram»:

Элемент (на русском)	Графическое обозначение (опишите словами)	Краткое описание	Пример для ИС «Библиотека»
Актёр			
Вариант использования			
Ассоциация			
Включение			
Расширение			
Обобщение			

6.2. Ответьте на вопросы (2-3 предложения каждый):

– В чём разница между отношениями include и extend на диаграмме вариантов использования?

– Может ли один актёр быть обобщением другого актёра? Приведите пример.

– Что такое «абстрактный вариант использования»?

7. Описание предметной области «Система онлайн-бронирования билетов в кинотеатр».

– гость может просмотреть расписание сеансов, выбрать фильм и зарегистрироваться в системе,

– зарегистрированный пользователь (зритель) может авторизоваться, выбрать место в кинозале, забронировать билет, оплатить билет (банковской картой или через электронный кошелёк), отменить бронирование, просмотреть историю своих бронирований,

– администратор может добавлять новые фильмы в расписание, изменять цены на билеты, просматривать отчёты о продажах,

– при оплате билета система обязательно выполняет проверку платёжных данных (общая подфункция «Проверить оплату»),

– при попытке забронировать билет на уже занятое место система предлагает выбрать другое место (расширение основного сценария бронирования).

7.1. Выделите всех актёров системы (не менее 2).

7.2. Выделите все варианты использования (не менее 6).

7.3. Определите связи между вариантами использования:

– Какие варианты использования связаны отношением include? Укажите базовый и включаемый варианты.

– Какие варианты использования связаны отношением extend? Укажите базовый и расширяющий варианты.

– Какие варианты использования связаны отношением обобщения?

7.4. Нарисуйте диаграмму вариантов использования для системы бронирования билетов в кинотеатр со всеми актёрами, вариантами использования и отношениями между ними.

8. Выбор типа диаграммы для ситуации.

Разрабатывается информационная система для больницы. Необходимо решить следующие задачи:

– показать, какие роли пользователей (врач, медсестра, пациент, администратор) и какие функции они могут выполнять в системе,

– описать, как система обрабатывает запись пациента к врачу (пошаговый сценарий взаимодействия объектов во времени),

– показать статическую структуру данных: какие классы есть в системе (Пациент, Врач, Запись, График) и как они связаны между собой,

– описать бизнес-процесс «Проведение лабораторного анализа» на верхнем уровне абстракции (что входит в процесс, что выходит, какие механизмы используются).

8.1. Какой тип UML-диаграммы (или иной нотации) наиболее подходит для решения каждой из перечисленных задач? Заполните таблицу:

Задача	Рекомендуемый тип диаграммы	Краткое обоснование
Показать роли пользователей и их функции		
Описать сценарий записи к врачу (в динамике)		
Показать статическую структуру		

классов		
Описать бизнес-процесс на верхнем уровне		

Форма сдачи: отчёт в электронном документе с выполненными заданиями (таблицы, диаграммы, рисунки, текстовые пояснения).

ЗАДАНИЯ НА ЛАБОРАТОРНЫЕ РАБОТЫ

РАЗДЕЛ 1. ОСНОВЫ ТЕОРИИ БАЗ ДАННЫХ

Лабораторная работа №1

Тема: Создание простейших баз данных с помощью СУБД Ms Access

Цель: Научиться создавать простейшие базы данных с помощью СУБД Ms Access, состоящих из одной, двух и трёх таблиц, а также приобрести практические навыки построения пользовательских форм для ввода/редактирования данных.

Задание:

1. Создание базы данных из одной таблицы

Задание № 1. Создайте новую базу данных и изготовьте структуру таблицы с информацией о студентах «Компьютерной школы».

Порядок выполнения:

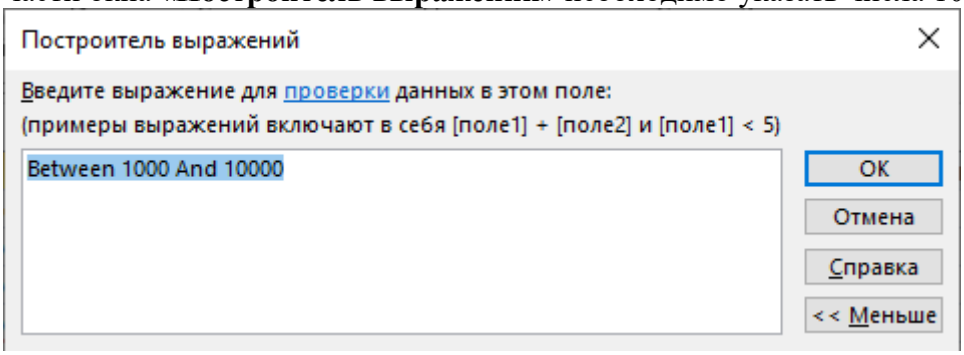
1. Создать файл БД в программе Ms Access *(при сохранении указать путь к папке, где БД будет храниться, и указать имя файла данной практической работы)*.
2. Перевести таблицу в режим **«Конструктор»** и указать имя таблицы **«Список»**.
3. Сформировать **структуру таблицы**, которая будет состоять из следующих полей (с указанием типа данных):

Имя поля	*Тип данных*	*Примечание*
Код	Счетчик	Ключевое поле
Фамилия	Текстовый	Размер поля – 20
Имя	Текстовый	Размер поля – 10
Отчество	Текстовый	Размер поля – 20
Год рождения	Числовой	Размер поля – Целое
Школа	Числовой	Размер поля – Байт
Класс	Числовой	Размер поля – Байт

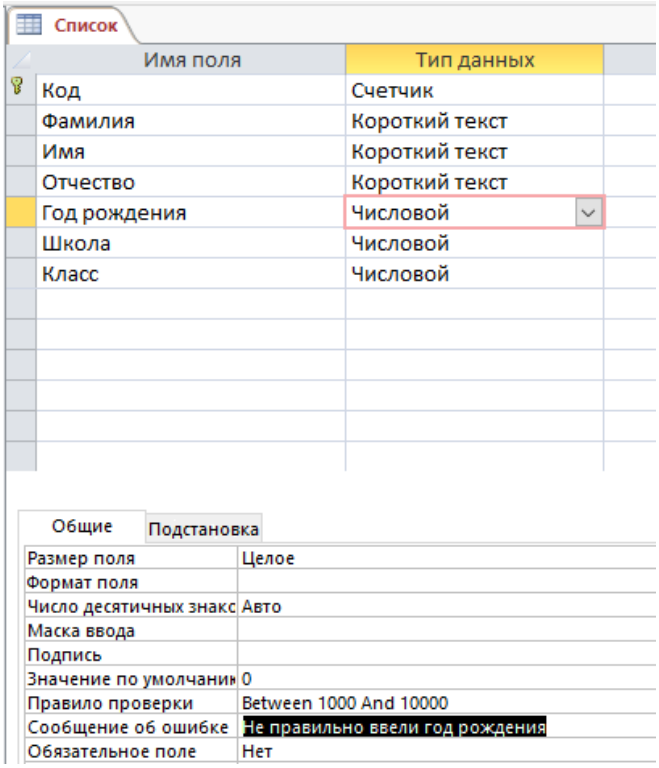
4. Для поля **«Год рождения»** необходимо установить **проверку для ввода данных**. Пользователь должен вводить числа в диапазоне от 1000 до 10000. Для этого используется свойство **«Правило проверки»**, которое удобнее заполнять через кнопку **«Три точки»**.

В диалоговом окне необходимо выбрать оператор **Between**, который находится в нижней части окна, в разделе **«Операторы»**.

В верхней части окна **«Построитель выражений»** необходимо указать числа 1000 и 10000:



5. Для свойства **«Сообщение об ошибке»** поля **«Год рождения»** необходимо указать значение **«Не правильно ввели год рождения»**, которое будет появляться в случае, когда пользователь введет год рождения, которые не соответствует диапазону от 1000 до 10000:



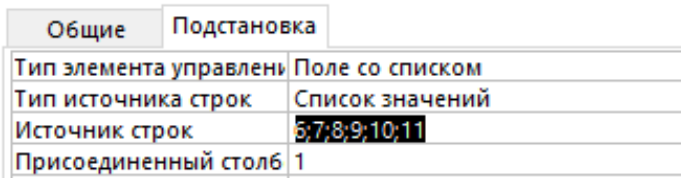
6. Для заполнения поля **«Класс»** пользователь должен выбирать значение из раскрывающегося списка. Для этого необходимо:

– открыть вкладку **«Подстановка»**,

– для свойства **«Тип элемента управления»** указать значение **«Поле со списком»**,

– для свойства **«Тип источника строк»** указать значение **«Список значений»**,

– для свойства **«Источник строк»** указать значение **«6;7;8;9;10;11»**



– для свойства **«Ограничиться списком»** указать значение **«Да»** (пользователь может только выбирать те значения, которые есть в списке).

7. Сохраните изменения в таблице.

8. Открыть таблицу в режиме **«Режим таблица»** и заполнить её в соответствии с таблицей:

Код	*Фамилия*	*Имя*	*Отчество*	*Год рождения*	*Школа*	*Класс*
1	Иванникова	Анна	Ивановна	1984	1	9
2	Баранова	Ирина	Алексеевна	1983	3	10
3	Корнилова	Ольга	Владимировна	1984	5	9
4	Воробьев	Алексей	Петрович	1983	1	10
5	Воробьев	Алексей	Иванович	1984	3	9
6	Воробьев	Олег	Григорьевич	1985	5	8
7	Скоркин	Александр	Евгеньевич	1982	1	11
8	Володина	Анна	Алексеевна	1984	3	9
9	Новоселов	Алексей	Антонович	1983	5	10
10	Александрова	Елена	Алексеевна	1984	1	9

Задание №2. Выполнить редактирование ячеек.

Прядок выполнения:

1. Заменить фамилию **Иванникова** на **Иванова**.
2. Для записи, содержащий информацию про **Иванову**, изменить год рождения на **1983**.

Задание №3. Отсортируйте значения таблицы.

Порядок работы:

1. Выполните сортировку по алфавиту для **фамилий** *(предварительно поставьте курсор в любую ячейку столбца «Фамилия»)*.
2. Отсортируйте строки таблицы по **номеру школы** – по убыванию.

Задание №4. Выполните поиск записей.

Порядок работы:

1. Найти строку таблицы, в которой встречается фамилия **Баранова** *(предварительно поставьте курсор в любую ячейку столбца «Фамилия»)*.

2. Найти строку таблицы, в которой встречается фамилия **Воробьев**.

Задание №5. Подготовьте к печати таблицу «Список».

Порядок работы:

1. Выбрать команду **«Предварительный просмотр»**. Если таблица не уместилась на листе или не видны все данные, то необходимо выйти из закрытого просмотра и изменить размер столбов таблицы.

2. Измените **ориентацию листа** (с книжного на альбомную) в предварительном просмотре.

2. Создание базы данных, состоящей из двух таблиц

Задание №1. Создайте копию файла базы данных, созданную при выполнении предыдущего задания и переименуйте его.

Задание №2. Удалите таблицу «Список».

Задание №3. Создайте таблицу «Группы».

Порядок работы:

1. Откройте вкладку **«Создание»** и нажмите на кнопку **«Конструктор таблиц»**.

2. Сформировать структуру таблицы, которая будет состоять из следующих полей (с указанием типа данных):

Имя поля	*Тип данных*	*Примечание*
Учебная группа	Числовой	Ключевое поле
Преподаватель	Текстовый	Размер поля – 20

3. Сохраните таблицу с именем **«Группы»**.

Задание №4. Заполните таблицу «Группы» через форму.

Порядок работы:

1. Закройте таблицу **«Группы»** и выделите её (если она не выделена).
2. Откройте вкладку **«Создание»** и нажмите на кнопку **«Форма»**.
3. Сохраните форму с именем **«Группы»** и откройте её в режиме **«Форма»**.
4. **Заполнить** таблицу «Группы» **через форму «Группы»** следующими значениями:

Учебная группа	*Преподаватель*
101	Верзаков С.А.
102	Белоусов А.И.
103	Масалова В.А.
104	Новикова Е.В.
105	Зачесова Т.П.

Задание №5. Создайте таблицу «Список».

Порядок работы:

1. Сформировать структуру таблицы, которая будет состоять из следующих полей (с указанием типа данных):

Имя поля	*Тип данных*	*Примечание*
Код	Счетчик	Уникальное поле
Фамилия	Текстовый	Размер поля – 20
Имя	Текстовый	Размер поля – 10
Отчество	Текстовый	Размер поля – 20
Год рождения	Числовой	Размер поля – Целое
Школа	Числовой	Размер поля – Целое
Класс	Числовой	Размер поля – Байт
Учебная группа	Числовой	

2. Установите **ограничение для ввода данных** в поле **«Год рождения»** (пользователь должен вводить данные в диапазоне от 1000 до 9999). Если пользователь неправильно введет значение, то должно появиться соответствующее сообщение.

3. Поле **«Класс»** должно **заполняться числовыми значениями из** раскрывающегося **списка**. Список должен состоять из значений **от 6 до 11** включительно.

4. Значение поля **«Учебная группа»** пользователь должен выбирать из раскрывающегося **списка** (список находится в таблице **«Группы»**), а не вводить вручную. Для этого:

– в свойствах поля на **вкладке «Подстановка»** необходимо изменить значение **«Поле»** свойства **«Тип элемента управления»** на значение **«Поле со списком»**,

– для свойства **«Источник строк»** указать значение – таблицу, из которой будут подставляться данные при заполнении записи (таблица **«Группы»**).

5. Сохраните изменения в таблице **«Список»**.

Задание №6. Создайте связи между таблицами.

Порядок работы:

1. Откройте вкладку **«Работа с базами данных»** и нажмите на кнопку **«Схема данных»**.
2. Добавьте таблицы **«Список»** и **«Группы»** в схему данных.
3. Сформировать связь:

– переместить **поле «Учебная группа»** таблицы **«Группы»** на **поле «Учебная группа»** таблицы **«Список»**,

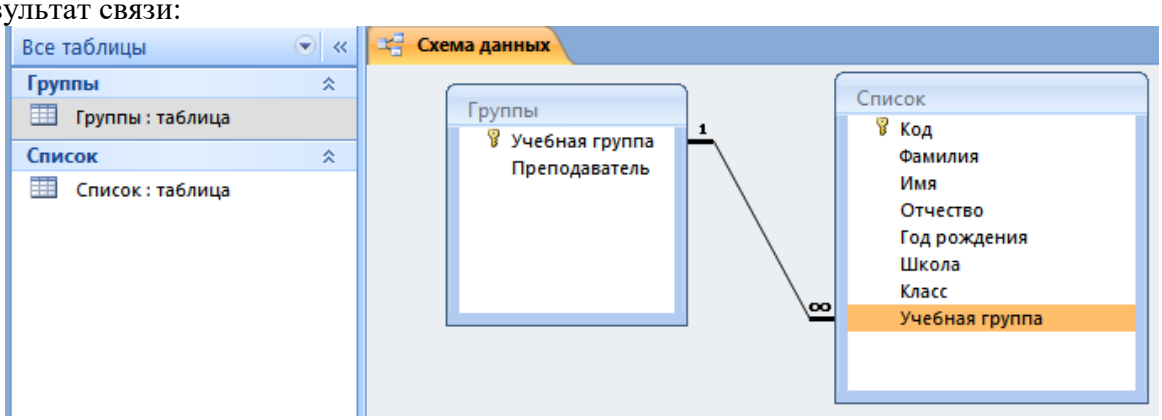
– в диалоговом **окне «Связи»** установить галочки:

a) обеспечение целостности данных;

b) каскадное обновление связанных полей (приведет к тому, что при изменении номера группы в таблице **«Группы»** автоматически изменится соответствующий номер в таблице **«Список»**);

c) *каскадное удаление связанных полей* (приведет к тому, что при удалении записи с номером группы в таблице **«Группы»** будут удалены все записи из таблицы **«Список»**).

Результат связи:



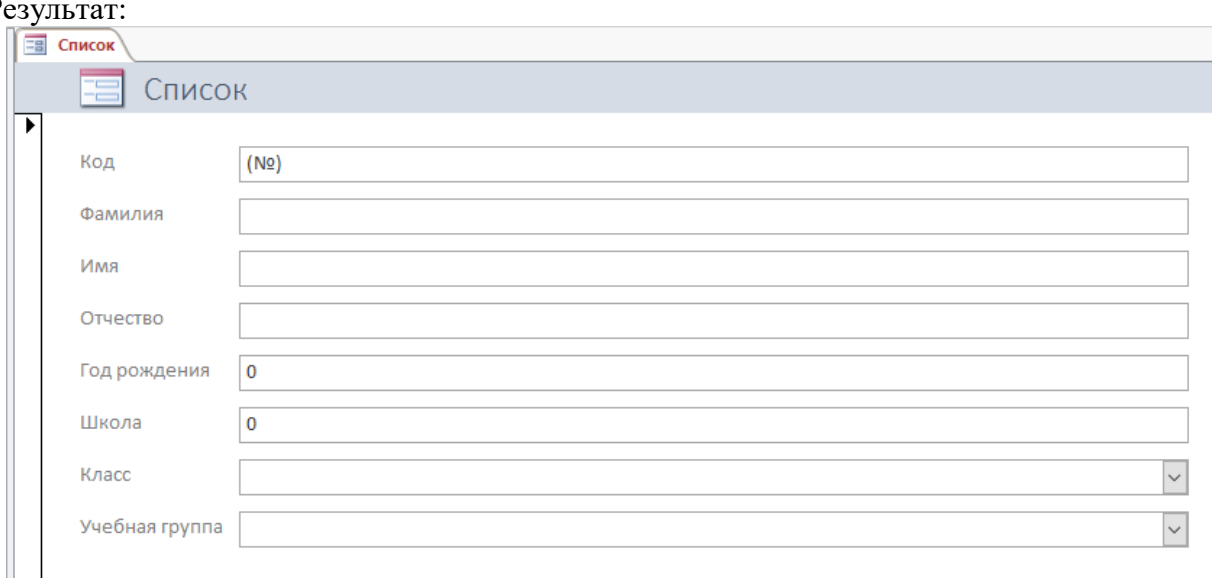
4. Сохранить изменения в **схеме данных** и закрыть её.

Задание №7. Заполните таблицу «Список» через форму.

Порядок работы:

1. Закройте все таблицы и выделите таблицу **«Список»**.
2. Откройте вкладку **«Создание»** и нажмите на кнопку **«Форма»**.

Результат:

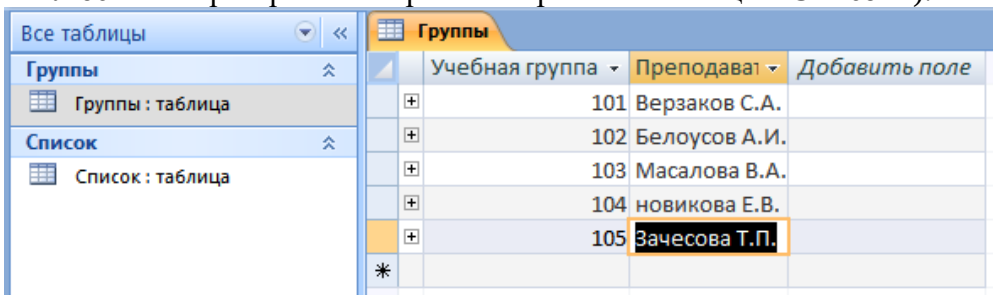


3. Сохраните форму с именем **«Список»** и откройте её в режиме **«Форма»**.
4. **Заполнить** таблицу «Список» **через форму «Список»** следующими значениями:

Код	*Фамилия*	*Имя*	*Отчество*	*Год рождения*	*Школа*	*Класс*	*Учебная группа*
1	Иванникова	Анна	Ивановна	1984	1	9	101
2	Баранова	Ирина	Алексеевна	1983	3	10	102
3	Корнилова	Ольга	Владимировна	1984	5	9	103
4	Воробьев	Алексей	Петрович	1983	1	10	101
5	Воробьев	Алексей	Иванович	1984	3	9	104
6	Воробьев	Олег	Григорьевич	1985	5	8	105
7	Скоркин	Александр	Евгеньевич	1982	1	11	101

8	Володина	Анна	Алексеевна	1984	3	9	102
9	Новоселов	Алексей	Антонович	1983	5	10	103
10	Александрова	Елена	Алексеевна	1984	1	9	101

5. Просмотрите таблицу **«Группы»** – открыть в режиме **«Режим таблицы»** (для каждой строки появится символ **«плюс»** и его раскрытие открывает строки из таблицы **«Список»**):



Задание №8. Перенесите содержимое таблиц «Список» и «Группы» в файлы Ms Excel.

Порядок работы:

1. Выделите таблицу **«Группы».**
2. Откройте вкладку **«Внешние данные»** и нажмите кнопку **«Экспорт в таблицу Excel»**.
3. Аналогичные действия проделайте с таблицей **«Список»**.

Задание №9. Проверить каскадное обновление связанных полей.

Порядок работы:

1. Откройте таблицу **«Группы»** и измените название учебной группы **101** на **202**.
2. Просмотрите изменения в таблице **«Список»**.

Задание №10. Проверить каскадное удаление связанных полей.

Порядок работы:

1. Откройте таблицу **«Группы»** и удалите учебную группу **202**.
2. Просмотрите изменения в таблице **«Список»**.

3. Создание базы данных, состоящей из трёх таблиц

Задание №1. Создайте копию файла базы данных, созданную при выполнении предыдущего задания и переименуйте его.

Задание №2. Откорректируйте данные в таблице **«Группы»**.

Прядок выполнения:

1. Откройте таблицу **«Группы»** через форму **«Группы»**.
2. Введите в таблицу то значение, которое было удалено в **практической работе №2** (учебная группа **101** с преподавателем **Верзаков С.А.**).

Задание №3. Удалите все записи из таблицы **«Список»**, оставив её структуру.

Задание №4. Создайте таблицу **«Личные данные»**.

Порядок работы:

1. Откройте вкладку **«Создание»** и нажмите на кнопку **«Конструктор таблиц»**.
2. Сформировать структуру таблицы, которая будет состоять из полей (с указанием типа данных), представленных в таблице:

Имя поля	*Тип данных*	*Примечание*
Код студента	Числовой	Уникальное поле
Адрес	Текстовый	Размер поля – 20
Номер телефона	Текстовый	Размер поля – 20
Word	Числовой	Размер поля – Байт
Excel	Числовой	Размер поля – Байт
Access	Числовой	Размер поля – Байт

3. Для полей «Word», «Excel» и «Access» необходимо задать свойство **«Правило проверки»** *(пользователь может вводить только числовые значения в диапазоне от 2 до 5)* и свойство **«Сообщение об ошибке»**.
4. Сохраните таблицу с именем **«Личные данные»**.

Задание №5. Измените связи между таблицами.

Порядок работы:

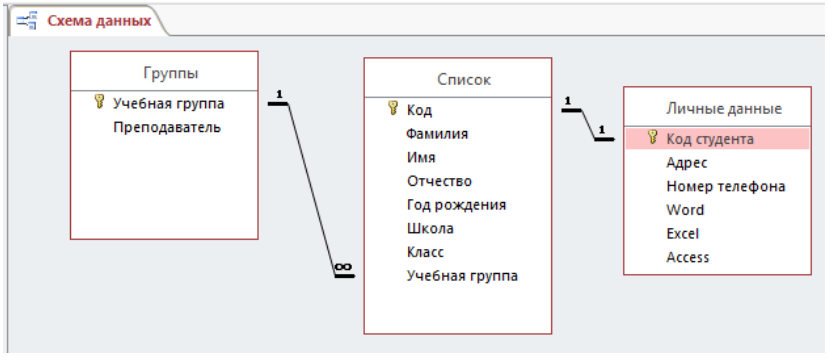
1. Откройте вкладку **«Работа с базами данных»** и нажмите на кнопку **«Схема данных»**.
2. Добавьте таблицу **«Личные данные»** в схему данных.
3. Сформировать связь:

– переместить **поле «Код студента»** таблицы **«Личные данные»** на **поле «Код»** таблицы

«Список»,

– в диалоговом **окне «Связи»** установить галочку *обеспечение целостности данных.*

Результат связи:



4. Сохранить изменения в **схеме данных** и закрыть её.

Задание №6. Заполните таблицы «Список» и «Личные данные» через общую форму.

Порядок работы:

1. Закройте все таблицы (если они открыты).
2. Откройте вкладку **«Создание»** и нажмите на кнопку **«Мастер форм»** *(позволяет достаточно быстро создать форму на основе выбранных для неё данных)*.
3. В поле **«Таблицы и запросы»** выберите таблицу **«Список»** и перенесите все доступные поля таблицы в правую колонку.
4. В поле **«Таблицы и запросы»** выберите таблицу **«Личные данные»** и перенесите все доступные поля таблицы в правую колонку.
5. Нажимаем кнопку **«Далее»** для перехода к следующему этапу построения формы.
6. Выбираем **внешний вид** формы – *в один столбец.*
7. Нажимаем кнопку **«Далее»** для перехода к заключительному этапу построения формы.
8. Задаем имя формы **«Общая форма»** и нажимаем кнопку **«Готово»**.
9. Откройте форму в режиме **«Форма»**.
10. **Заполнить** таблицы «Список» и «Личные данные» **через форму «Общая форма»** следующими значениями:

«Список»

Код	*Фамилия*	*Имя*	*Отчество*	*Год рождения*	*Школа*	*Класс*	*Учебная группа*
1	Иванникова	Анна	Ивановна	1984	1	9	101
2	Баранова	Ирина	Алексеевна	1983	3	10	102
3	Корнилова	Ольга	Владимировна	1984	5	9	103
4	Воробьев	Алексей	Петрович	1983	1	10	101
5	Воробьев	Алексей	Иванович	1984	3	9	104
6	Воробьев	Олег	Григорьевич	1985	5	8	105
7	Скоркин	Александр	Евгеньевич	1982	1	11	101
8	Володина	Анна	Алексеевна	1984	3	9	102
9	Новоселов	Алексей	Антонович	1983	5	10	103
10	Александрова	Елена	Алексеевна	1984	1	9	101

«Личные данные»

Код студента	*Адрес*	*Номер телефона*	*Word*	*Excel*	*Access*
1	Центральная 11-5	51-17-22	5	5	5
2	Солнечная 8-117	51-18-22	4	4	4
3	Сиреневый 7-16	51-19-22	3	4	5
4	Центральная 14-81	51-20-22	5	5	4
5	Сиреневый 7-16	51-21-22	5	4	4
6	Солнечная 2-121	51-22-22	4	4	4
7	Школьная 5-34	51-23-22	5	5	5
8	Центральная 30-7	51-24-22	3	3	3
9	Сиреневый 7-16	51-25-22	3	4	4
10	Солнечная 6-34	51-26-22	5	5	5

11. Откройте таблицу **«Учебные группы»** и убедитесь, что в ней появились данные.

Задание №7. Добавить новое поле в таблицу «Список».

Порядок работы:

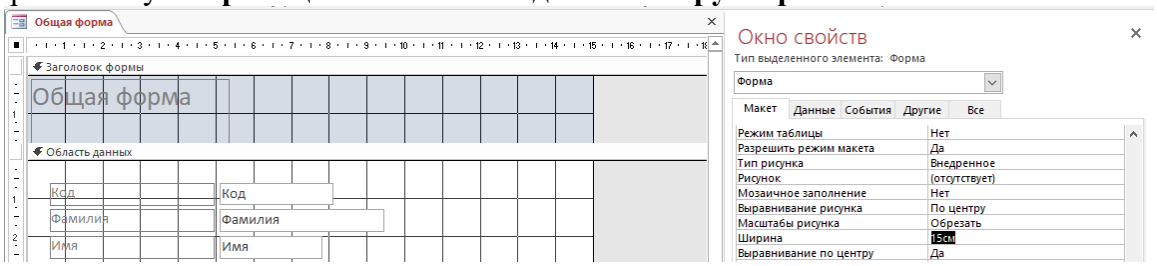
1. Откройте таблицу **«Список»** в режиме **«Конструктор»**.
2. Добавьте поле **«Фото»** с типом данных **«Поле объекта OLE»**.

3. Сохраните изменения в таблице.

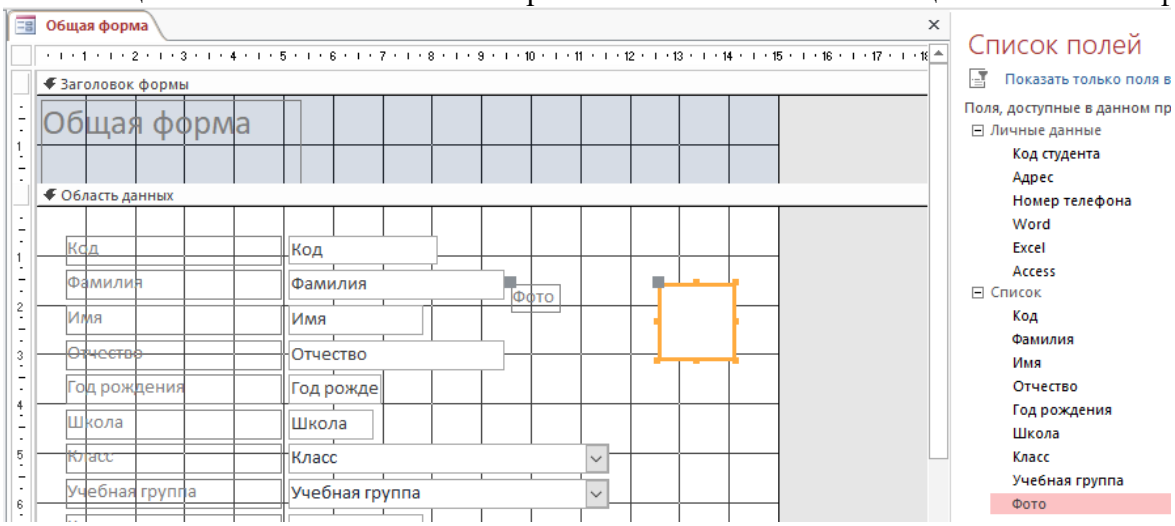
Задание №8. Добавить созданное поле таблицы «Список» на форму «Общая форма».

Порядок работы:

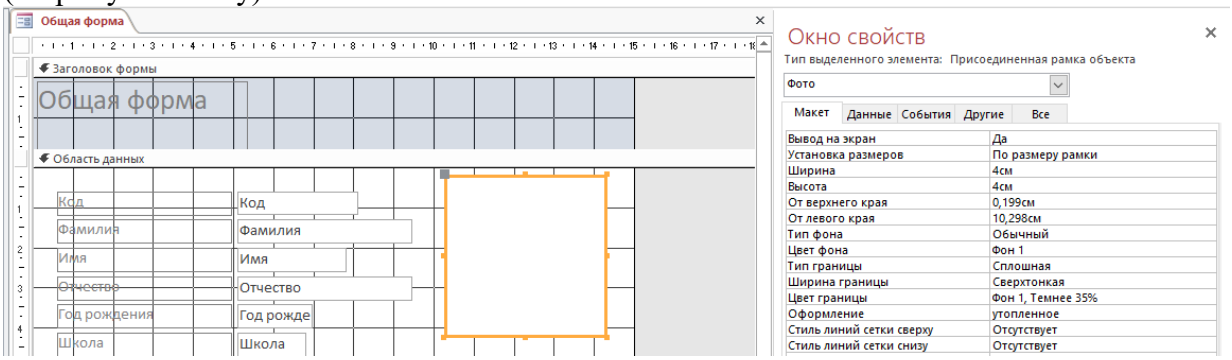
1. Откройте форму **«Общая форма»** в режиме **«Конструктор»**.
2. Измените ширину формы на значение 15 см (свойство «Ширина»). Окно свойств можно открыть через кнопку **«Страница свойств»** вкладки **«Конструктор»**.



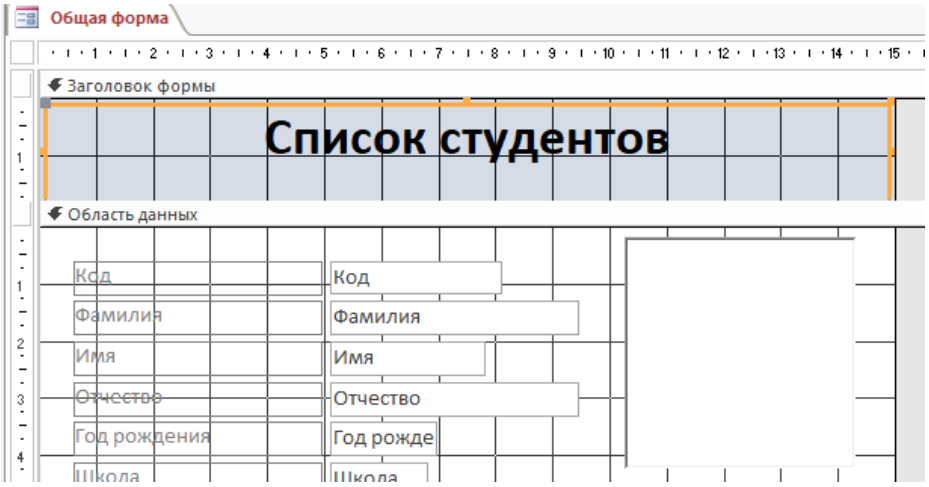
3. На вкладке **«Конструктор»** откройте окно **«Добавить поля»**, в котором надо раскрыть все поля таблицы через ссылку **«Показать все таблицы»**.
4. С помощью левой клавиши мыши перенесите поле **«Фото»** таблицы **«Список»** на форму:



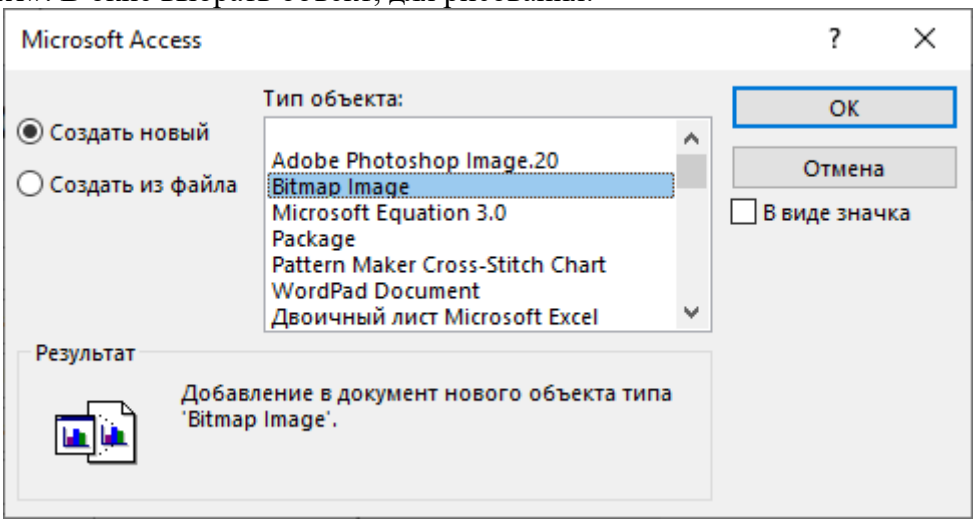
5. Для данного компонента необходимо **удалить метку** *(щелкнуть на ней левой клавиши мыши, чтобы она выделилась, а затем нажать клавишу Delete)*, а размер самого компонента изменить на 4 см (ширину и высоту).



6. Измените заголовок на форме (изменить текст **«Общая форма»** на текст **«Список студентов»**) и оформите его по центру, полужирным начертанием и чёрным цветом. Для заголовка установить размер текста, равным 26 пт. Для работы с текстом можно использовать вкладку **«Главная»**.



7. Сохраните изменения формы и откройте её в режиме «Режим формы».
8. Для первого студента заполните поле **«Фото»** через контекстное меню выбрав команду **«Вставить объект»**. В окне выбрать объект, для рисования:



9. Нарисуйте изображение и закройте окно.
10. Заполните поле для остальных студентов.

Задание №9. Откорректировать форму «Общая форма».

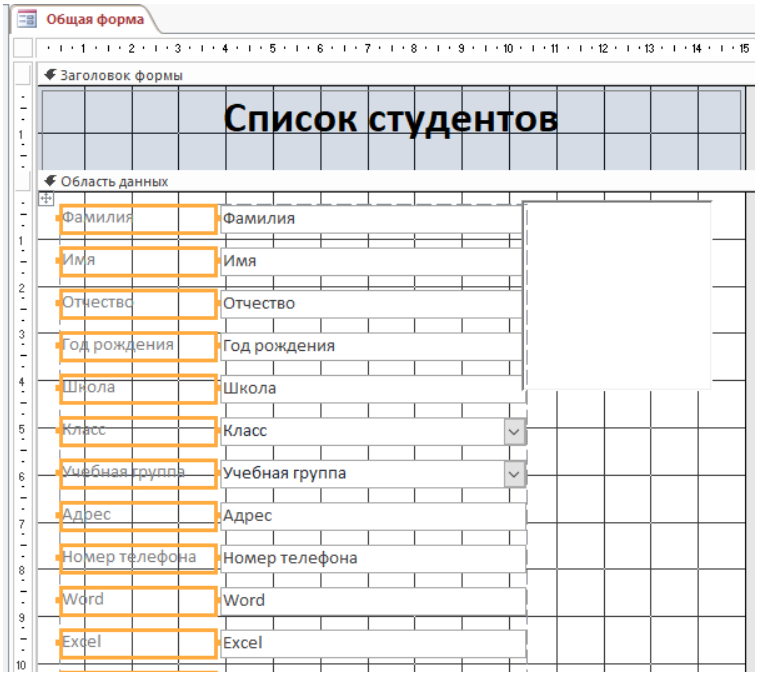
Порядок работы:

1. Откройте форму **«Общая форма»** в режиме **«Конструктор»**.
2. Убрать **область выделения**. Для этого необходимо:

– открыть окно **«Свойства формы»,**

– для свойства **«Область выделения»** (вкладка **«Макет»**) установить значение **«Нет»**.

3. Сохранить изменения и проверить изменения на форме для пользователя.
4. Убрать **кнопки навигации**. Для этого свойству **«Кнопки навигации»** (вкладка **«Макет»**) установить значение **«Нет»**.
5. Сохранить изменения и проверить изменения на форме для пользователя.
6. Убрать с формы компоненты **«Код»** и **«Код студента»** *(выделить их и нажать клавишу Delete).*
7. Выделить все компоненты формы, кроме компонента, связанного с полем **«Фото»** и оформить их **«В столбик»** через соответствующую форму, которая находится на вкладке **«Упорядочить»**.
8. Выделить только метки компонентов и изменить их размеры:



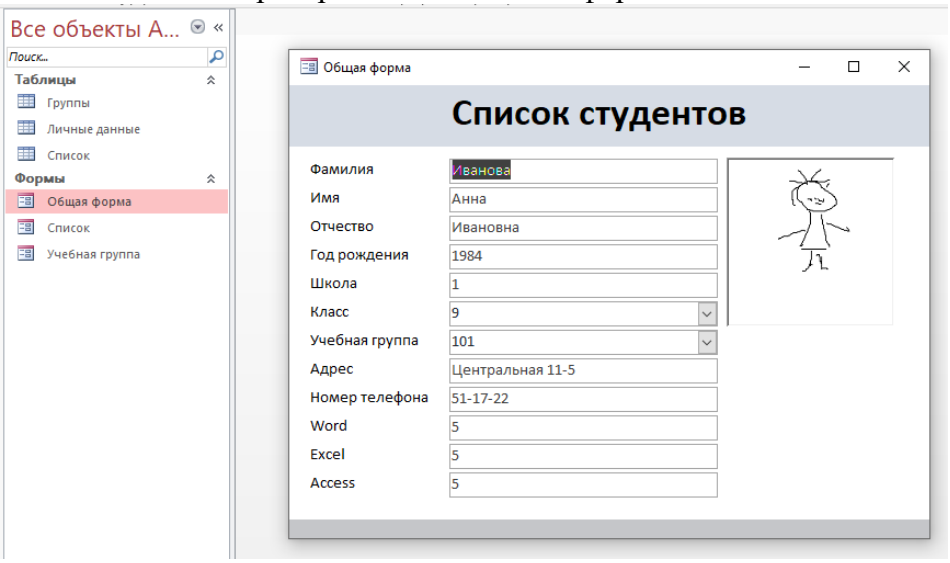
9. Для меток установить черный цвет текста.

10. Выделить все компоненты формы, кроме компонента, связанного с полем **«Фото»** и на вкладке **«Упорядочить»** параметр **«Внутренние поля»** установить в значение **«Узкое»**. Изменить размеры формы (уменьшить высоту формы).

11. Изменить расположение компонента, связанного с полем **«Фото»**, таким образом, чтобы данный компонент не накладывался на остальные компоненты.

12. Через вкладку **«Другие»** свойству **«Всплывающее окно»** установить значение **«Да»** (форма будет открываться в собственном окне).

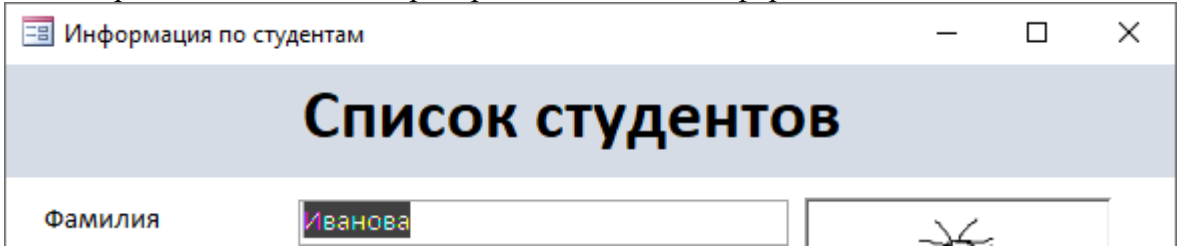
13. Сохранить изменения и проверить изменения на форме для пользователя.



14. Запретить пользователю изменять размеры формы. Для этого свойству **«Тип границ»** (вкладка **«Макет»**) установить значение **«Тонкая»**.

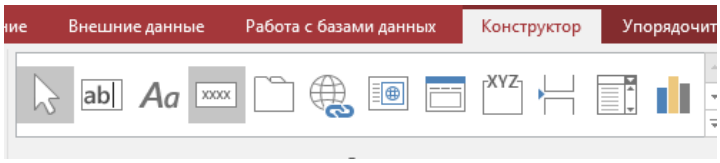
15. Изменить текст в заголовке формы. Для этого свойству **«Подпись»** (вкладка **«Макет»**) установить значение **«Информация по студентам»**.

16. Сохранить изменения и проверить изменения на форме для пользователя.

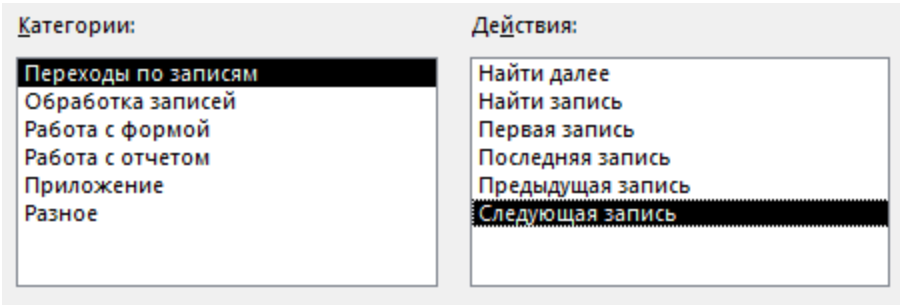


17. Разместить на форме **кнопки** для навигации по записям в таблице базы данных. Для этого:

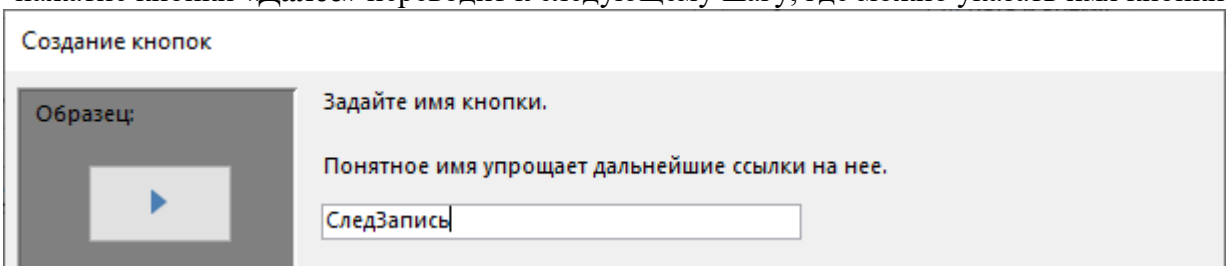
– на вкладке **«Конструктор»** выбрать кнопку *(на ней изображено «XXXX»)* и разместить её на форму



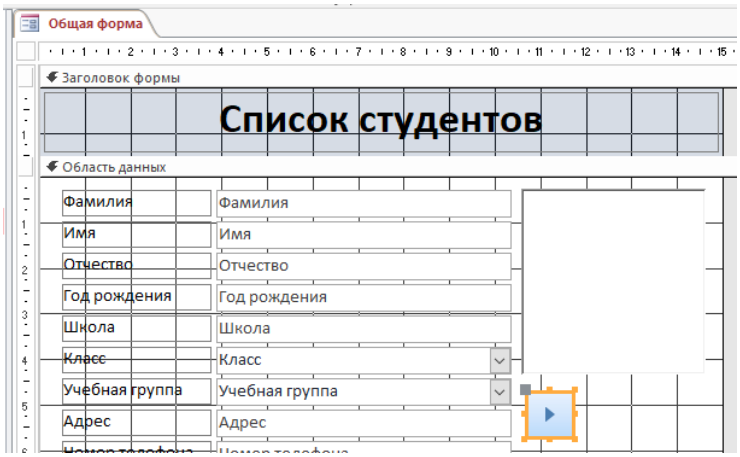
– в диалоговом окне выбрать категорию **«Переходы по записям»**, а затем команду **«Следующая запись»**



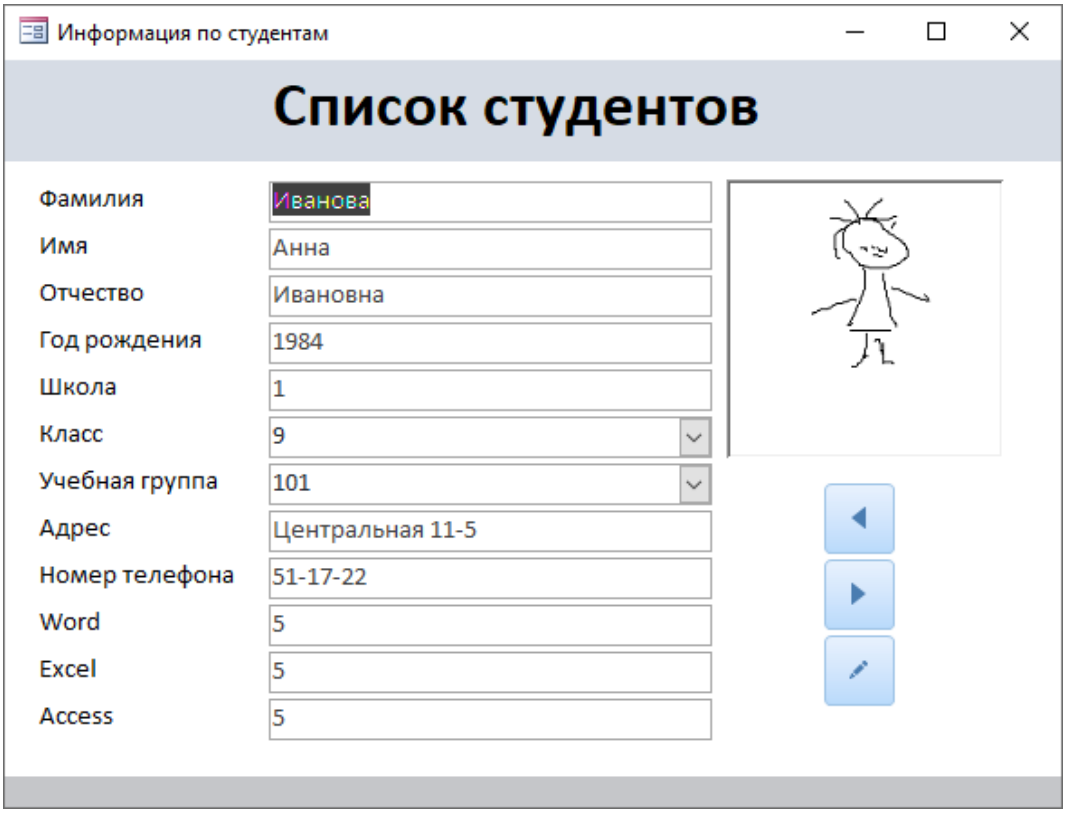
– после нажатия на кнопку **«Далее»** можно выбрать рисунок кнопки,
– нажатие кнопки **«Далее»** переводит к следующему шагу, где можно указать имя кнопки



– кнопка **«Готово»** завершает процесс добавления кнопки на форму.
Результат:



18. Сохранить изменения и проверить изменения на форме для пользователя.
19. Аналогичным образом разместить кнопку **«Предыдущая запись»**.
20. Добавить кнопки из категории **«Обработка записей»** для добавления новой записи.



21. Для кнопок изменить стиль через вкладку **«Формат»** (предварительно выделить кнопки). Через эту вкладку можно изменить форму кнопки (команда **«Изменить фигуру»**).

22. Добавить на форму кнопку **«Закрыть»**. Данная кнопка должна закрывать текущую форму. При создании кнопки использовать категорию **«Работа с формой»**.

23. Для кнопки установить рисунок через свойство **«Рисунок»** (вкладка **«Макет»**). Можно для кнопки изменить свойство **«Расположение подписи к рисунку»**.

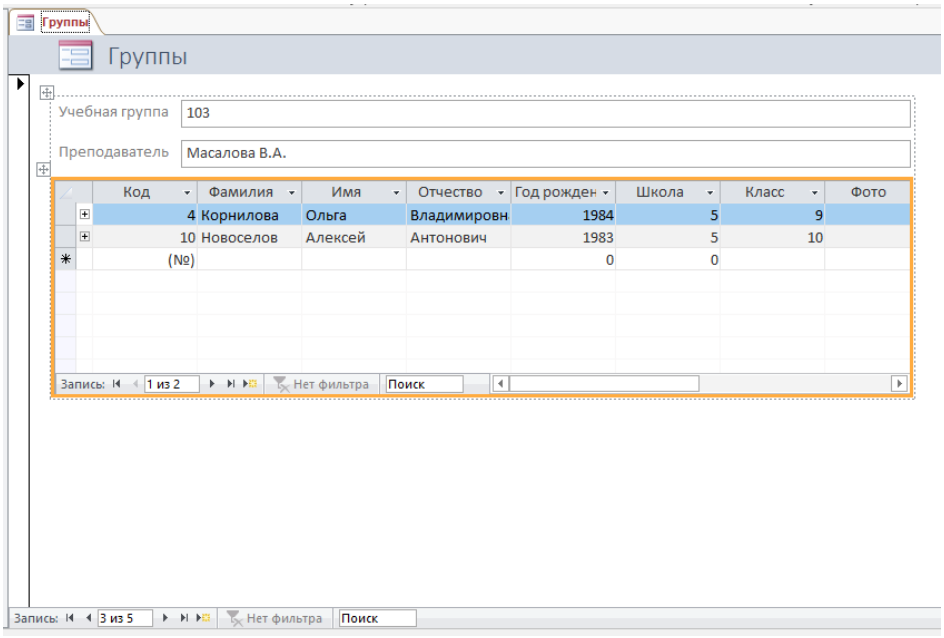
24. Сохранить изменения и проверить изменения на форме для пользователя.

Задание №10. Создать разные формы.

Порядок работы:

1. Выделите таблицу **«Группы».**
2. Откройте вкладку **«Создание»** и нажмите на кнопку **«Форма»**.

Результат:



3. Сохраните форму с именем **«Студенты по группам»**.
4. Откройте форму в режиме **«Режим форма»**.
5. Переведите форму в режим **«Режим макета»**.
6. Через контекстное меню скрыть поле «Фото».
7. Сохранить изменения и открыть форму в режиме **«Режим форма»**.
8. Закройте форму **«Студенты по группам»**.

9. Выделите таблицу **«Список».**

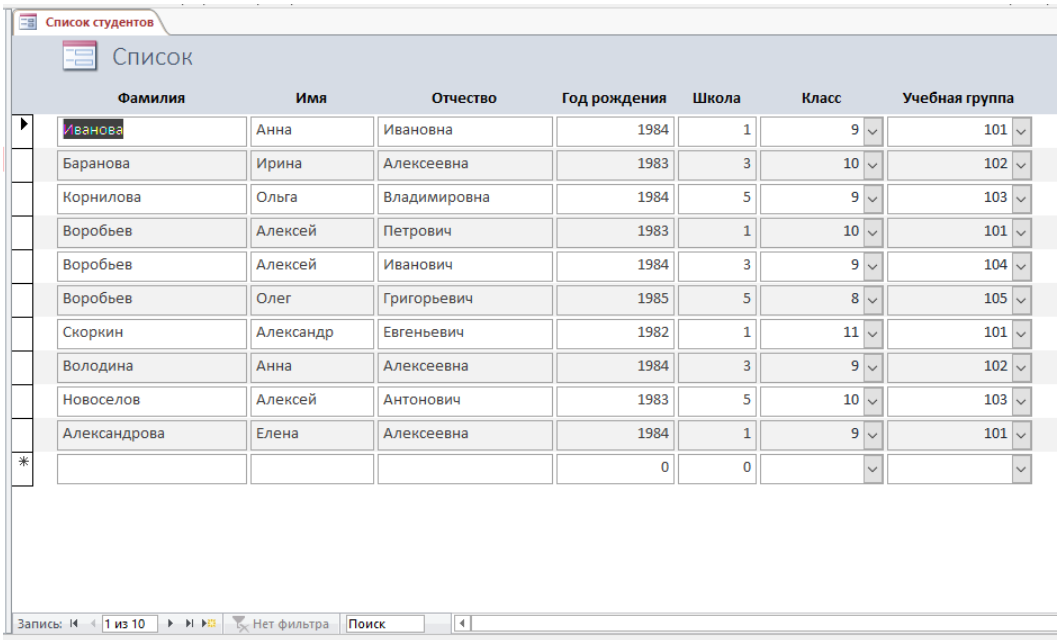
10. Откройте вкладку **«Создание»** и нажмите на кнопку **«Другие формы» → «Несколько элементов»**.

11. Отредактировать форму в режиме **«Режим макета»**: убрать поле **«Код»** *(выделить заголовок и с клавишей Shift любую ячейку в данном столбце)*; изменить ширину столбцов; выделить строку заголовка и изменить начертание на полужирное, установить чёрный цвет текста и выравнивание по центру.

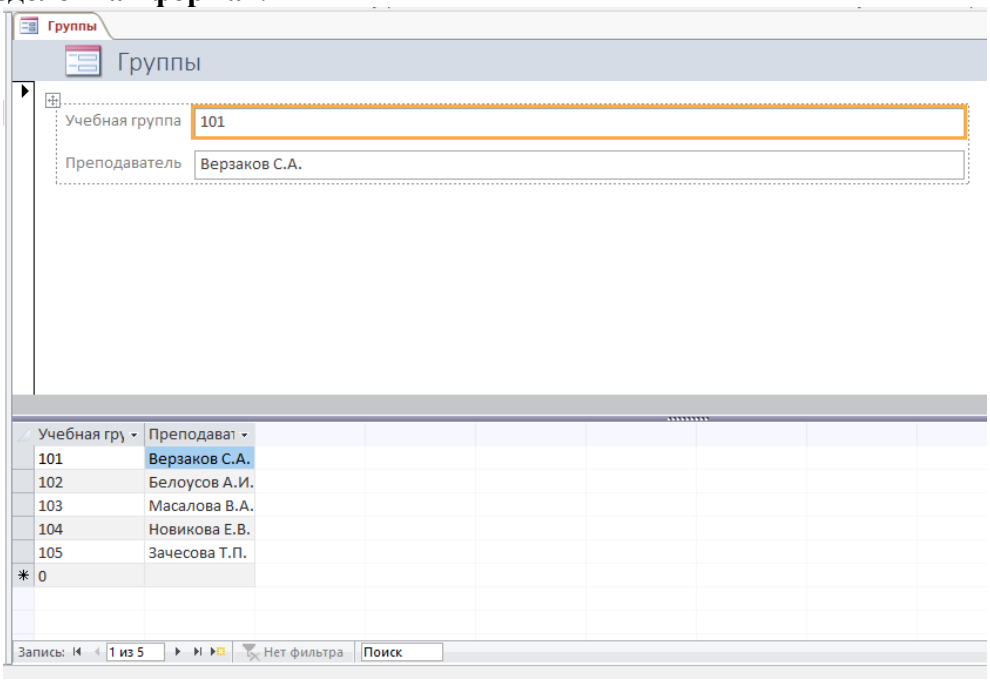
12. Изменить высоту строк.

13. Сохранить форму **«Список студентов».**

Результат:



14. Создать для таблицы «Группы» разделенную форму через вкладку **«Создание» → «Другие формы» → «Разделенная форма»**:



15. Закрыть форму не сохраняя.

Задание №11. Добавить на форму «Общая форма» поле.

На форму добавить поле, которое позволят устанавливать для студента: поступает он в институт или не поступает – пользователь для текущей записи должен установить или снять галочку (флажок) – **логическое поле**.

Задание №12. Создать основное меню (кнопочная форма) для базы данных.

Лабораторная работа №2

Тема: Проектирование реляционной базы данных: от концептуальной модели до физической

реализации в СУБД (на примере Microsoft Access).

Цель: Закрепить теоретические знания и сформировать практические навыки в области проектирования реляционных баз данных, включая: анализ предметной области и выделение сущностей; определение атрибутов, первичных и внешних ключей; установление связей между таблицами (1:1, 1:N, M:N); применение принципов нормализации (до 3-й нормальной формы); физическую реализацию спроектированной структуры в СУБД MS Access, включая создание форм для ввода данных и отчётов для вывода информации.

Задание:

Спроектировать и реализовать в СУБД MS Access базу данных для автоматизации учёта в следующей предметной области (выберите один из вариантов или укажите свой):

Вариант 1. Библиотека (Книги – Читатели – Выдача книг).

Вариант 2. Склад товаров (Товары – Поставщики – Поступления товаров).

Вариант 3. Успеваемость студентов (Студенты – Дисциплины – Оценки).

Вариант 4. Отдел кадров (Сотрудники – Должности – Назначения на должность).

Вариант 5. Заказы в интернет-магазине (Клиенты – Товары – Заказы).

Вариант 6. Укажите свой вариант по согласованию с преподавателем.

Этапы выполнения:

Часть 1. Проектирование (в электронном виде)

- провести анализ предметной области и выделить не менее 3 сущностей,
- для каждой сущности определить атрибуты (не менее 4-х на каждую), указать типы данных,
- определить первичные ключи для каждой таблицы,
- выявить связи между таблицами (обосновать выбор типа связи),
- привести структуру БД к 3-й нормальной форме (3НФ).

Часть 2. Реализация в MS Access

- создать новую базу данных (.accdb) с именем [Фамилия_ЛР1],
- создать три таблицы в режиме конструктора, задав имена полей, типы данных и ключевые поля,
- установить связи между таблицами (обеспечить целостность данных, каскадное обновление/удаление),
- заполнить каждую таблицу не менее чем 5 записями (реальными или правдоподобными тестовыми данными),
- создать одну форму (с помощью мастера форм или вручную) для удобного ввода данных в главную таблицу,
- создать один отчёт (с помощью мастера отчётов) для вывода содержимого любой таблицы с группировкой по одному из полей.

Часть 3. Контрольные запросы (для отчёта)

Сформулировать и выполнить с помощью обычного или перекрёстного запроса следующий аналитический запрос (пример для варианта «Библиотека»): вывести список читателей, у которых на руках более 2 книг, с указанием даты выдачи.

Для своего варианта сформулируйте аналогичный запрос с условием группировки или подсчёта.

РАЗДЕЛ 3. ОРГАНИЗАЦИЯ БАЗ ДАННЫХ

Лабораторная работа №3

Тема: Индексы, связи между таблицами и объединение таблиц.

Цель: Изучить механизмы оптимизации доступа к данным в реляционных базах данных, включая принципы последовательного и индексного поиска, методы сортировки и фильтрации записей; освоить практические навыки создания и использования различных типов индексов (первичных, уникальных, составных) в СУБД MS Access; научиться применять псевдонимы для таблиц и полей, а также реализовывать различные типы объединения таблиц (INNER JOIN, LEFT JOIN, RIGHT JOIN) для извлечения связанных данных из нескольких источников.

Задание:

Создать базу данных, содержащую три таблицы, связанные между собой. Состав таблиц и полей определяется в соответствии с индивидуальным вариантом.

Примерная структура (для варианта «Торговля»):

Таблица	Поля

Товары	Код товара (счётчик, первичный ключ), Название (текст), Категория (текст), Цена (числовой), Остаток (числовой)
Клиенты	Код клиента (счётчик, первичный ключ), Фамилия (текст), Город (текст), Телефон (текст)
Продажи	Код продажи (Счетчик, Первичный Ключ), Код Товара (Числовой, Внешний Ключ). Код Клиента (Числовой, Внешний Ключ), Дата, (Дата/Время), Количество (числовой)

Заполнить каждую таблицу не менее 10 записями.

Индивидуальные варианты заданий

Выберите один из вариантов или получите у преподавателя:

Вариант	Предметная область
1	Библиотека (Книги – Читатели – Выдача)
2	Торговля (Товары – Клиенты – Продажи)
3	Успеваемость (Студенты – Дисциплины – Оценки)
4	Кадры (Сотрудники – Отделы – Назначения)
5	Склад (Товары – Поставщики – Поступления)
6	Поликлиника (Пациенты – Врачи – Приемы)

Этапы выполнения

Часть 1. Создание структуры БД и первичных индексов

Создать новую базу данных MS Access с именем [Фамилия_ЛР_Индексы].

Создать три таблицы в режиме конструктора, задав для каждого поля: Имя поля, Тип данных, Описание (по желанию).

Для каждого первичного ключа автоматически создается кластеризованный уникальный индекс. Убедиться, что поле первичного ключа имеет свойство Индексированное поле = Да (Совпадения не допускаются).

Часть 2. Создание дополнительных индексов

Создать следующие индексы (способ – по выбору: через конструктор таблиц или SQL-запрос):

№	Таблица	Поле(я)	Тип индекса	Обоснование
1	Товары	Название	Обычный	Ускорение поиска по названию
2	Клиенты	Фамилия	Обычный	Частая фильтрация по фамилиям
3	Клиенты	Город	Обычный	Группировка и фильтрация по городу
4	Продажи	Дата	Обычный	Сортировка и отбор по датам
5	Продажи	Код товара	Обычный	Ускорение JOIN с таблицей Товары
6	Продажи	Код клиента	Обычный	Ускорение JOIN с таблицей Клиенты
7	Товары	(Категория, Цена)	Составной	Частые запросы по категории с сортировкой по цене

Для составного индекса в Access: открыть окно «Индексы» (кнопка на панели инструментов в режиме конструктора) и добавить строку с двумя полями.

Часть 3. Сравнение последовательного и индексного поиска

Выполнить эксперимент, демонстрирующий эффективность индексов:

1. До создания индекса (если индексы уже созданы — временно отключить их или использовать новое неиндексированное поле):

– выполнить запрос на поиск записи по неиндексированному полю (например, SELECT * FROM Товары WHERE Название = 'XXX'),

– засечь время выполнения (можно использовать Timer в VBA или приблизительно оценить).

2. После создания индекса:

– создать индекс по полю поиска,

– выполнить тот же запрос,

– сравнить скорость выполнения.

3. Сделать вывод в отчёте: насколько ускорился поиск (в разах).

Часть 4. Сортировка и фильтрация

Выполнить следующие запросы (каждый оформить в виде SQL-запроса и сохранить в БД):

№	Запрос	тип
1	Вывести все товары, отсортированные по категории (по возрастанию) и внутри категории – по цене (по убыванию)	Сортировка
2	Вывести клиентов из города «Москва» с фамилиями, начинающимися на букву «И»	Фильтрация (LIKE)

3	Вывести продажи за период с 01.01.25 по 31.12.25 с суммой чека (Количество × Цена) больше 5000	Фильтрация (BEETWEN, вычисляемое поле)
4	Вывести список товаров с остатком меньше 5 единиц	Фильтрация
5	Вывести 3 самых дорогих товара	Сортировка и Топ

РАЗДЕЛ 4. УПРАВЛЕНИЕ БАЗОЙ ДАННЫХ С ПОМОЩЬЮ SQL

Лабораторная работа №4

Тема: Структурированный язык запросов SQL.

Цель: Изучить синтаксис и возможности структурированного языка запросов SQL; освоить практические навыки применения команд определения данных (CREATE, ALTER, DROP) и манипулирования данными (INSERT, UPDATE, DELETE); научиться формировать сложные выборки данных с использованием команды SELECT, включая фильтрацию (WHERE), сортировку (ORDER BY), группировку (GROUP BY) и применение групповых функций (COUNT, SUM, AVG, MAX, MIN); овладеть навыками использования символьных функций, функций даты и времени для обработки текстовых и временных данных в запросах.

Задание:

Создать базу данных в СУБД MS Access (или любой другой реляционной СУБД по согласованию с преподавателем), содержащую три-четыре таблицы, связанные между собой. Допускается использовать базу данных, созданную в предыдущих лабораторных работах, либо разработать новую.

Рекомендуемая структура (на примере предметной области «Учебный процесс»):

Таблица	Поля
Студенты	Код студента (счетчик, ключ), Фамилия (текст), Имя (текст), Дата рождения (дата/время), Город (текст), Стипендия (числовой)
Дисциплины	Код дисциплины (счетчик, ключ), Название (текст), Часы (числовой), Семестр (числовой)
Успеваемость	Код записи (счетчик, ключ), Код студента (числовой, ключ), Код дисциплины (числовой, ключ), Дата экзамена (дата/время), Оценка (числовой)

Заполнить каждую таблицу не менее 10-15 записями.

Альтернативные предметные области: Торговля, Библиотека, Кадры, Склад (по согласованию с преподавателем).

Каждый студент выполняет задания на основе своей базы данных. Допускается работа в парах с разными вариантами наполнения данных.

Этапы выполнения

Часть 1. Команды определения данных (DDL)

Выполнить в SQL-режиме (в MS Access: Создание → Запрос → Конструктор → SQL-режим).

Создание таблицы (если таблицы ещё не созданы):

```
CREATE TABLE Преподаватели (
    КодПреподавателя COUNTER PRIMARY KEY,
    Фамилия TEXT(50) NOT NULL,
    Должность TEXT(50),
    Кафедра TEXT(100)
);
```

Изменение структуры таблицы (ALTER TABLE):

- добавить поле Телефон (тип TEXT(20)) в таблицу Студенты,
- изменить тип данных поля Стипендия на числовой (если был текстовым),
- добавить ограничение CHECK (где поддерживается) или настроить правило проверки в Access:

```
ALTER TABLE Успеваемость
ADD CONSTRAINT chk_Оценка CHECK (Оценка BETWEEN 2 AND 5);
```

Удаление таблицы (DROP TABLE):

- создать временную таблицу (например, Temp), затем удалить её командой:

```
DROP TABLE Temp;
```

Часть 2. Команды манипулирования данными (DML)

Вставка данных (INSERT):

- добавить одного нового студента:

INSERT INTO Студенты (Фамилия, Имя, ДатаРождения, Город, Стипендия)
VALUES ('Иванов', 'Петр', #15.05.2000#, 'Москва', 1200);

– добавить несколько записей в таблицу Успеваемость одной командой (в Access через UNION, в других СУБД – многострочный INSERT)

Обновление данных (UPDATE):

– повысить стипендию всем студентам из Москвы на 15%:

UPDATE Студенты
SET Стипендия = Стипендия * 1.15
WHERE Город = 'Москва';

– заменить оценку «2» на «3» для всех записей в Успеваемости

Удаление данных (DELETE):

– удалить студентов, у которых нет ни одной оценки (с использованием подзапроса):

DELETE FROM Студенты
WHERE КодСтудента NOT IN (SELECT DISTINCT КодСтудента FROM Успеваемость);

Часть 3. Команда SELECT – базовые конструкции

Простая выборка (все поля, все записи):

SELECT * FROM Студенты;

Выборка с фильтрацией по условию (WHERE):

– студенты, родившиеся после 01.01.2024:

SELECT Фамилия, Имя, ДатаРождения
FROM Студенты
WHERE ДатаРождения > #01.01.2001#;

– студенты с фамилией на букву «С»:

SELECT * FROM Студенты WHERE Фамилия LIKE 'С%';

Сортировка (ORDER BY):

SELECT Фамилия, Имя, Стипендия
FROM Студенты
WHERE Стипендия > 1000
ORDER BY Стипендия DESC, Фамилия ASC;

Часть 4. Групповые (агрегатные) функции

Выполнить запросы с использованием следующих групповых функций:

Функция	Задание
COUNT(*)	Подсчитать общее количество студентов
COUNT(DISTINCT поле)	Подсчитать количество различных городов, из которых есть студенты
SUM	Найти общую сумму стипендии всех студентов
AVG	Найти среднюю стипендию студентов
MAX / MIN	Найти максимальную и минимальную оценку в группе

Часть 5. Символьные функции

Выполнить запросы с использованием символьных функций (в MS Access они имеют специфические названия; для совместимости с ANSI SQL указаны аналоги):

Функция	Задание
Len()	Вывести фамилию и длину фамилии (количество символов)
UCase()	Вывести фамилию, преобразованную в верхний регистр
LCase()	Вывести фамилию, преобразованную в нижний регистр
Mid()	Вывести первые 3 буквы фамилии
& или Concatenate()	Вывести полное имя (фамилия и имя)
Trim()	Удалить лишние пробелы (если есть)

Часть 6. Функции даты и времени

Выполнить запросы с функциями даты и времени:

Функция	Задание
Date()	Вывести текущую дату
Year()	Вывести год рождения каждого студента
Month()	Вывести месяц рождения
Day()	Вывести день рождения
DateDiff()	Вычислить возраст студента (количество полных лет)
DateAdd()	Найти дату окончания семестра (текущая дата + 30 дней)

Format()	Вывести дату в формате «ДД.ММ.ГГГГ»

Часть 7. Сложные запросы (комбинация изученных элементов)

Создать и выполнить не менее 5 сложных запросов, комбинирующих:

- несколько таблиц (JOIN),
- агрегатные функции (GROUP BY + HAVING),
- подзапросы,
- символьные и/или функции даты.

Примеры сложных запросов:

- топ-3 студентов по среднему баллу,
- студенты, сдавшие экзамены на отлично (оценка 5) по всем дисциплинам,
- количество студентов по городам, отсортированное по убыванию,
- средняя оценка по каждому семестру (используя связь с Дисциплинами),
- инициалы студентов (формат: И. Иванов).

РАЗДЕЛ 5. СИСТЕМЫ УПРАВЛЕНИЯ РАСПРЕДЕЛЕННЫМИ БАЗАМИ ДАННЫХ

Лабораторная работа №5

Тема: Архитектуры распределенных баз данных.

Цель: Изучить архитектурные принципы построения распределенных баз данных (РБД), включая модели фрагментации (горизонтальная, вертикальная, смешанная) и репликации данных; освоить практические навыки создания фрагментов распределенной базы данных на основе централизованной БД; исследовать особенности выполнения распределенных запросов с объединением данных из различных фрагментов.

Задание:

В рамках работы моделируется распределенная база данных, физические фрагменты которой размещаются на трех различных файлах (базах данных) MS Access, имитирующих удаленные узлы. Центральное приложение (главная база) с помощью связанных таблиц объединяет данные из всех фрагментов.

Предметная область (вариант по умолчанию): «Торговое предприятие с филиалами».

Логическая схема централизованной БД:

Таблица	Поля	Описание
Сотрудники	Код сотрудника (PK), Фамилия, Должность, Филиал ID, Зарплата	Данные о сотрудниках
Товары	Код товара (PK), Наименование, Категория, Цена, Остаток	Справочник товаров
Продажи	Код продажи (PK), Филиал ID, Дата, Код товара, Количество, Сумма	Данные о продажах
Филиалы	Филиал ID (PK), Название, Город, Адрес	Справочник филиалов

Узлы (фрагменты) распределенной системы:

Узел	Размещаемые данные	Файл БД
Узел А (Москва)	Сотрудники московского филиала + все продажи по Москве	Фрагмент_Москва.accdb
Узел Б (Санкт-Петербург)	Сотрудники петербургского филиала + все продажи по Санкт-Петербургу	Фрагмент_СПб.accdb
Узел В (Центральный)	Справочник товаров + справочник филиалов (реплицируемые данные)	Фрагмент_Центр.accdb

Выберите один из вариантов предметной области или получите у преподавателя:

Вариант	Предметная область	Критерий распределения
1	Торговая сеть	По филиалам (географический)
2	Университет	По факультетам или кампусам
3	Банковская система	По региональным отделениям
4	Медицинская сеть	По поликлиникам или районам
5	Логистическая компания	По складам или регионам
6	Сеть ресторанов	По заведениям (географический)
7	Свой вариант	По согласованию с преподавателем

Этапы выполнения

Часть 1. Создание фрагментов распределенной БД (горизонтальная фрагментация)

1.1. Создание централизованной базы-источника

Создать базу данных Центральная_БД.accdb с четырьмя таблицами (Сотрудники, Товары, Продажи, Филиалы). Заполнить таблицы тестовыми данными:

- филиалы: не менее 2-3 филиалов,
- сотрудники: не менее 10 записей, распределенных по филиалам,
- товары: не менее 10 наименований,
- продажи: не менее 20 записей.

1.2. Горизонтальное фрагментирование

Создать отдельные базы данных для каждого узла:

Узел	База данных	Фрагменты (таблицы)	Условие фрагментации
Узел А	Фрагмент_УзелА.accdb	Сотрудники_А, Продажи_А	Филиал ID = 1
Узел Б	Фрагмент_УзелБ.accdb	Сотрудники_Б, Продажи_Б	Филиал ID = 2
Узел В	Фрагмент_УзелВ.accdb	Товары, Филиалы	Полная копия (без фрагментации)

Способ создания:

- экспортировать данные из центральной БД с фильтром,
- или создать пустые таблицы в узлах и выполнить запросы INSERT INTO ... SELECT ... WHERE ...

Пример для узла А:

```
-- В базе Фрагмент_УзелА.accdb
SELECT * INTO Сотрудники_А
FROM Сотрудники IN 'Центральная_БД.accdb'
WHERE ФилиалID = 1;
```

1.3. Репликация справочных данных

В узле В создать полные копии таблиц Товары и Филиалы (без фрагментации). Это демонстрирует полную репликацию.

Часть 2. Создание связанной распределенной системы

2.1. Создание главной базы данных

Создать новую базу данных Распределенная_БД.accdb, которая будет служить единым интерфейсом для работы с распределенными данными.

2.2. Подключение внешних таблиц (связывание)

В главной БД выполнить связывание таблиц из фрагментов:

Локальное имя (в главной БД)	Источник	Узел
Сотрудники_Москва	Таблица Сотрудники_А	Фрагмент_УзелА.accdb
Сотрудники_СПб	Таблица Сотрудники_Б	Фрагмент_УзелБ.accdb
Продажи_Москва	Таблица Продажи_А	Фрагмент_УзелА.accdb
Продажи_СПб	Таблица Продажи_Б	Фрагмент_УзелБ.accdb
Товары	Таблица Товары	Фрагмент_УзелВ.accdb
Филиалы	Таблица Филиалы	Фрагмент_УзелВ.accdb

В MS Access: Внешние данные → Новый источник данных → Из базы данных → Файл Access → указать файл фрагмента → выбрать таблицу → выбрать «Связать».

2.3. Создание унифицированного представления

Создать запросы-представления, объединяющие данные из всех узлов:

```
-- Унифицированное представление сотрудников
CREATE VIEW ВсеСотрудники AS
SELECT * FROM Сотрудники_Москва
UNION ALL
SELECT * FROM Сотрудники_СПб;
-- Унифицированное представление продаж
CREATE VIEW ВсеПродажи AS
SELECT * FROM Продажи_Москва
UNION ALL
SELECT * FROM Продажи_СПб;
```

Часть 3. Выполнение распределенных запросов

Реализовать следующие типы распределенных запросов:

3.1. Локальный запрос (данные из одного узла)

```
-- Запрос только к московскому узлу
SELECT * FROM Сотрудники_Москва WHERE Зарплата > 50000;
```

3.2. Глобальный запрос без объединения (UNION)

```
-- Общее количество продаж по всем филиалам
SELECT 'Москва' AS Филиал, COUNT(*) AS КоличествоПродаж
FROM Продажи_Москва
UNION ALL
SELECT 'СПб', COUNT(*)
FROM Продажи_СПб;
```

3.3. Распределенное соединение (JOIN данных из разных узлов)

```
-- Для каждого товара – общее количество продаж по всем филиалам
SELECT
    Товары.Наименование,
    (SELECT SUM(Количество) FROM Продажи_Москва
     WHERE Продажи_Москва.КодТовара = Товары.КодТовара) +
    (SELECT SUM(Количество) FROM Продажи_СПб
     WHERE Продажи_СПб.КодТовара = Товары.КодТовара) AS ИтогоПродано
FROM Товары;
```

3.4. Запрос с фрагментацией по условию

```
-- Вывести всех сотрудников с указанием филиала
SELECT Фамилия, Должность, 'Москва' AS Филиал FROM Сотрудники_Москва
UNION ALL
SELECT Фамилия, Должность, 'Санкт-Петербург' FROM Сотрудники_СПб
ORDER BY Филиал;
```

Часть 4. Исследование репликации данных

4.1. Создание реплицируемой таблицы

В узле В (центральный узел) находится таблица Товары. Создать её копию в узле А для имитации репликации.

4.2. Обновление реплики (синхронизация)

Смоделировать ситуацию:

- изменить цену товара в узле В (основная копия),
- обновить реплику в узле А с помощью запроса на вставку/обновление.

```
-- Обновление реплики в узле А на основе данных из узла В
UPDATE Товары_Реплика_А
INNER JOIN Товары IN 'Фрагмент_УзелВ.accdb'
ON Товары_Реплика_А.КодТовара = Товары.КодТовара
SET Товары_Реплика_А.Цена = Товары.Цена;
```

4.3. Анализ конфликта репликации

Создать ситуацию конфликта (одни и те же данные изменены в двух узлах) и предложить стратегию разрешения (например, приоритет узла В).

Лабораторная работа №6

Тема: Клиентская часть распределенной базы данных.

Цель: Изучить принципы проектирования и разработки клиентской части распределенной базы данных; освоить методы планирования пользовательского интерфейса с учетом требований эргономики и эффективности доступа к распределенным данным; приобрести практические навыки создания форм и отчетов с использованием мастеров и конструкторов MS Access; научиться разрабатывать и модифицировать различные типы меню (главное меню, контекстное меню, лента); освоить основы программирования на VBA для реализации бизнес-логики клиентского приложения; ознакомиться с технологиями доступа к данным (DAO, ADO) и методами оптимизации производительности клиент-серверного взаимодействия в распределенной среде.

Задание:

В качестве основы используется распределенная база данных, созданная в лабораторной работе №5 по теме «Архитектуры распределенных баз данных».

Выберите один из вариантов предметной области или получите у преподавателя:

Вариант	Предметная область	Особые требования к клиентской части

1	Торговая сеть	Формы для оформления заказов, отчеты по продажам
2	Университет	Формы для ввода оценок, отчеты успеваемости
3	Медицинская клиника	Формы для записи пациентов, отчеты по приемам
4	Библиотека	Формы выдачи книг, отчеты по задолжникам
5	Складской учет	Формы прихода/расхода, отчеты по остаткам
6	Кадровый учет	Формы для сотрудников, отчеты по персоналу
7	Свой вариант	По согласованию с преподавателем

Этапы выполнения

Часть 1. Планирование клиентского приложения

1.1. Анализ требований

Составить документ «План клиентского приложения», содержащий:

Раздел	Содержание
Назначение системы	Краткое описание целей создания клиентского приложения
Категории пользователей	Администраторы, операторы, руководители, гости
Сценарии использования	Основные действия каждой категории (не менее 3 на категорию)
Функциональные требования	Перечень функций, которые должно реализовать приложение
Нефункциональные требования	Быстродействие, удобство интерфейса, надежность
Макеты экранов	Эскизы основных форм

Пример сценария использования (вариант «Торговая сеть»):

Категория	Сценарий	Описание
Оператор	Оформление продажи	Выбор товара, указание количества, расчет суммы, печать чека
Оператор	Поиск товара	Поиск по наименованию, категории, штрих-коду
Руководитель	Отчет по продажам	Выбор периода, филиала, формирование сводного отчета
Администратор	Добавление товара	Ввод нового товара в справочник с репликацией на узлы

1.2. Проектирование навигационной структуры

Создать схему навигации (в виде блок-схемы или таблицы).

Часть 2. Создание форм с помощью мастера и конструктора

2.1. Создание формы в режиме мастера

С помощью мастера форм создать 3 формы:

Тип формы	Источник данных	Назначение
Простая форма (в столбец)	Товары (из узла В)	Просмотр и редактирование справочника товаров
Ленточная форма	Сотрудники_Москва (из узла А)	Список сотрудников московского филиала
Форма с подчиненной формой	Филиалы + Сотрудники (из узлов А, Б, В)	Информация о филиалах и их сотрудниках

Для создания формы с подчиненной формой:

Главная форма на основе таблицы «Филиалы» (узел В)

Подчиненная форма на основе объединения данных сотрудников из всех узлов

2.2. Модификация формы в режиме конструктора

Для каждой созданной формы выполнить модификацию:

Элемент	Действие
Заголовок формы	Изменить цвет фона, добавить логотип (можно текстовый)
Поля	Изменить ширину, высоту, цвет, шрифт
Кнопки	Добавить кнопки: «Сохранить», «Отмена», «Обновить», «Закрыть»
Поля со списком	Заменить обычное поле на комбинированный список (ComboBox) для выбора значений из связанной таблицы
Условное форматирование	Настроить подсветку полей (например, товары с остатком < 10 – красным)

2.3. Создание диалоговых форм

Создать 2 диалоговые формы:

Форма	Назначение	Элементы
dlgПоискТовара	Поиск товара по критериям	Поле ввода «Название», ComboBox «Категория», кнопки «Найти», «Сброс», «Закрыть»
dlgВыборПериода	Выбор периода для отчетов	Два поля «Дата начала», «Дата окончания», кнопки «Сформировать», «Отмена»

Часть 3. Создание отчетов с помощью мастера и конструктора

Источник данных	Тип	Назначение
Товары	Простой (ленточный)	Список всех товаров с ценами
Продажи (объединение узлов А и Б)	С группировкой по филиалам	Отчет по продажам в разрезе филиалов
Сотрудники (объединение узлов А и Б)	С группировкой по должностям	Штатная ведомость

3.3. Создание отчета с помощью конструктора (вручную)

Создать отчет «Сводный отчет по продажам» с нуля в режиме конструктора.

Часть 4. Создание и настройка меню

Главное меню (Switchboard / Навигационная форма)

Создать главное меню для навигации по приложению.

Создать форму frmГлавноеМеню со следующей структурой:

Кнопка	Действие	Цвет фона
Справочники	Открыть форму frmТовары	Синий
Продажи	Открыть форму frmПродажи	Зеленый
Сотрудники	Открыть форму frmСотрудники	Желтый
Отчеты	Открыть форму frmОтчетыМеню	Оранжевый
Администрирование	Открыть форму frmАдминистрирование (только для администратора)	Красный
Выход	Закрыть приложение	Серый

РАЗДЕЛ 6. АДМИНИСТРИРОВАНИЕ И БЕЗОПАСНОСТЬ БАЗ ДАННЫХ

Лабораторная работа №7

Тема: Обеспечение целостности, достоверности и непротиворечивости данных.

Цель: Освоить механизмы поддержки целостности, достоверности и непротиворечивости данных в реляционных СУБД на примере реализации первичных и внешних ключей, ограничений (CHECK, NOT NULL, UNIQUE), триггеров и транзакций. Научиться анализировать и предотвращать возникновение аномалий при модификации данных.

Задание:

Дана схема базы данных «Университет», содержащая три таблицы:

– Students (student_id INTEGER PRIMARY KEY, name VARCHAR(50), group_id INTEGER, scholarship DECIMAL(8,2), email VARCHAR(100))

– Groups (group_id INTEGER PRIMARY KEY, group_name VARCHAR(20), faculty VARCHAR(50))

– Exams (exam_id INTEGER PRIMARY KEY, student_id INTEGER, subject VARCHAR(50), score INTEGER, exam_date DATE)

Таблицы содержат тестовые данные (по 5-10 строк). В настоящий момент все ограничения, кроме PRIMARY KEY, отсутствуют.

Этапы выполнения

Часть 1. Декларативная целостность

1. Первичный ключ – проверить и при необходимости доопределить.
2. Внешние ключи – добавить связь Students.group_id → Groups.group_id и Exams.student_id → Students.student_id с каскадным обновлением и ограничением на удаление (ON DELETE RESTRICT).
3. Ограничения CHECK:

– scholarship >= 0

– score BETWEEN 0 AND 100

– exam_date <= CURRENT_DATE

4. NOT NULL – наложить на поля: name, group_name, subject, exam_date.

5. UNIQUE – на поле email в таблице Students (при наличии дублей – предварительно очистить данные).

6. Демонстрация – попытаться вставить заведомо некорректные записи (отрицательная стипендия, оценка 150, дата в будущем, email-дубликат) и зафиксировать сообщения об ошибках СУБД.

Часть 2. Поддержка достоверности данных через транзакции

1. Сымитировать сценарий перевода студента из одной группы в другую. Выполнить как единую транзакцию:

– Уменьшить количество студентов в старой группе (условный счётчик – добавить поле student_count в таблицу Groups и заполнить его триггером, либо просто выполнить UPDATE).

– Изменить group_id у студента.

– Увеличить количество студентов в новой группе.

2. Продемонстрировать ROLLBACK при возникновении ошибки (например, указать несуществующую новую группу).

3. Установить уровень изоляции READ COMMITTED и показать, что незафиксированные изменения не видны другому сеансу (имитировать параллельное подключение через два окна SQL).

Часть 3. Обеспечение непротиворечивости с помощью триггеров.

1. Создать триггер before_insert_on_exams, который запрещает добавление экзамена, если у студента нет стипендии (scholarship = 0), при этом оценка ниже 60. (Правило: «неуспевающий студент без стипендии не может сдавать экзамен, пока не исправит долги» – проверка по наличию других экзаменов со score < 60).

2. Создать триггер after_delete_on_students, который при удалении студента автоматически удаляет все его записи об экзаменах (альтернатива каскадному удалению, если явный внешний ключ не был создан с CASCADE).

3. Протестировать триггеры:

– вставить экзамен для студента без стипендии с плохими предыдущими оценками (должно быть отклонено),

– удалить студента и проверить, что записи в Exams удалены.

Лабораторная работа №8

Тема: Реализация механизмов защиты информации в СУБД.

Цель: Изучить основные механизмы защиты информации в СУБД (аутентификация, управление доступом, шифрование). Освоить методы резервного копирования (логическое и физическое), переноса данных между базами и сценарии восстановления данных из резервных копий после сбоев или ошибочных действий пользователей.

Задание:

Дана учебная база данных «Клиника», содержащая три таблицы:

– Patients (patient_id INTEGER PRIMARY KEY, full_name VARCHAR(100), birth_date DATE, diagnosis VARCHAR(200), phone VARCHAR(20), insurance_number VARCHAR(50))

– Doctors (doctor_id INTEGER PRIMARY KEY, full_name VARCHAR(100), specialization VARCHAR(50), salary DECIMAL(10,2))

– Appointments (appointment_id INTEGER PRIMARY KEY, patient_id INTEGER, doctor_id INTEGER, appointment_date DATE, diagnosis_notes TEXT)

Таблицы содержат тестовые данные (по 10-15 строк). В настоящий момент:

– отсутствует разграничение прав доступа (все пользователи имеют полные права),

– не настроено резервное копирование,

– нет механизмов аудита и шифрования конфиденциальных данных (диагнозы, страховые номера).

Этапы выполнения

Часть 1. Управление доступом и защита информации

1. Создание ролей и пользователей:

– Создать роль doctor_role с правами: SELECT на Patients (только имя, телефон, но не диагноз), SELECT/INSERT/UPDATE на Appointments, SELECT на свою запись в Doctors.

– Создать роль admin_role с полными правами на все таблицы.

– Создать роль auditor_role с правом SELECT на все таблицы, но без права изменения.

– Создать по одному пользователю для каждой роли (например, user_doctor, user_admin, user_auditor).

2. Разграничение доступа на уровне строк (RLS):

– Включить Row Level Security для таблицы Patients.

– Создать политику, чтобы врач видел только полные данные тех пациентов, у которых есть хотя бы один приём к этому врачу.

– Создать политику, чтобы администратор видел всех пациентов.

3. Шифрование конфиденциальных данных:

– Добавить в таблицу Patients поле passport_data VARCHAR(200).

– Реализовать шифрование поля passport_data с использованием встроенных функций СУБД (например, pgcrypto в PostgreSQL, AES_ENCRYPT в MySQL) с мастер-ключом.

– Создать представление Patients_public, которое показывает все поля, кроме passport_data, с расшифровкой только для роли admin_role.

4. Демонстрация:

– Подключиться под разными пользователями и показать, какие данные доступны, а какие – нет.

– Попытаться выполнить запрещённую операцию (например, врач пытается удалить запись о приёме) и зафиксировать отказ.

Часть 2. Логическое резервное копирование и перенос данных

1. Экспорт данных (логическое копирование):

– Выполнить логический дамп (backup) базы данных «Клиника» с помощью утилиты pg_dump (для PostgreSQL) или mysqldump (для MySQL) в файл clinic_backup_дата.sql.

– Извлечь из дампа только таблицу Patients в отдельный файл patients_only.sql.

2. Перенос данных между базами:

– Создать новую базу данных Clinic_Analytics.

– Импортировать в неё только таблицу Appointments из созданного дампа.

– Создать в базе Clinic_Analytics таблицу DailyStats (date DATE, appointments_count INTEGER) и заполнить её агрегированными данными из Appointments (количество приёмов по дням).

– Выгрузить таблицу DailyStats в CSV-файл (для последующего анализа во внешних инструментах).

– Использование утилит для переноса: показать команды для копирования таблицы между разными схемами в одной БД:

Часть 3. Физическое резервное копирование и восстановление

1. Создание физической резервной копии:

– Остановить СУБД (если допустимо в учебных целях) или использовать горячее резервное копирование (например, pg_basebackup для PostgreSQL).

– Скопировать каталог данных (datadir) в безопасное место: /backup/clinic_data_физическая/.

– Запустить СУБД обратно.

2. Симуляция сбоя и восстановление:

– Сценарий А (ошибочное удаление): Пользователь user_doctor случайно удалил всех пациентов с диагнозом «Грипп». Выполнить восстановление только этих записей из логического дампа (без потери остальных данных).

– Сценарий В (повреждение таблицы): Симулировать повреждение таблицы Appointments (например, удалить файл таблицы из datadir). Восстановить всю таблицу из физической копии.

– Сценарий С (point-in-time recovery): Восстановить базу данных на состояние 15 минут назад (до ошибочного массового обновления), используя WAL-журналы (PostgreSQL) или бинарные логи (MySQL).

3. Автоматизация резервного копирования:

– Написать простой shell-скрипт (или batch-файл), который: создаёт ежедневный логический дамп с датой в имени файла; удаляет дампы старше 7 дней; сжимает дамп (gzip или zip).

– Запланировать выполнение скрипта через cron (или планировщик задач Windows) – имитировать настройку (без реального запуска).

Часть 4. Аудит и журналирование операций

ПРИЛОЖЕНИЕ Б

Тестовые задания (контрольная работа)

Индикатор ЗНАТЬ:

ПК 1.1. Производить установку и настройку компонентов автоматизированных (информационных) систем в защищенном исполнении в соответствии с требованиями эксплуатационной документации

1. Установить правильную последовательность этапов проектирования базы данных:

А) Физическое проектирование

Б) Реализация и наполнение БД

В) Логическое проектирование

Г) Концептуальное проектирование

2. Установите правильную последовательность создания объекта в MS Access с использованием конструктора.

А) Выбрать тип создаваемого объекта (например, «Конструктор запросов»)

Б) Задать свойства и параметры объекта в конструкторе

В) Выбрать вкладку "Создание" на ленте меню

Г) Сохранить объект с уникальным именем

3. Установите правильную последовательность действий при нормализации таблицы (устранение повторяющихся данных).

А) Проанализировать повторяющиеся группы данных в таблице

Б) Вынести повторяющиеся данные в отдельную таблицу

В) Создать связь между основной и новой таблицей

Г) Определить первичный ключ для новой таблицы

4. Установить правильную последовательность создания объектов в БД:

А) Создание связей

Б) Создание таблиц

В) Настройка ограничений

Г) Определение первичных ключей

5. Установить правильную последовательность видов моделей данных (от простых к сложным):

А) Объектно-ориентированная

Б) Иерархическая

В) Реляционная

Г) Сетевая

Установите правильную последовательность этапов проектирования простой базы данных.

А) Определение связей между таблицами

Б) Создание таблиц и определение первичных ключей

В) Определение цели базы данных и необходимых данных

Г) Определение полей для каждой таблицы

6. Установите правильную последовательность создания запроса в Access

А) Выбрать таблицы для запроса

Б) Добавить условия в «Условие отбора»

В) Нажать «Выполнить»

Г) Открыть «Конструктор запросов»

7. Установите правильную последовательность написания простого запроса на выборку данных по условию.

А) Написать FROM НазваниеТаблицы

Б) Запустить запрос на выполнение

В) Написать SELECT *

Г) Написать условие через WHERE

8. Установите соответствие между типами моделей данных и их описаниями.

Модель данных	Описание модели данных
А) Реляционная модель	1) Данные организованы в виде дерева с

	отношениями «родитель-потомок»
Б) Иерархическая модель	2) Данные представлены в виде таблиц с атрибутами и связями
В) Сетевая модель	3) Данные представлены в виде объектов с методами и свойствами
Г) Объектно-ориентированная модель	4) Данные организованы в виде графа с произвольными связями

9. Установите соответствие между ключевыми словами SQL и их назначением

Ключевые слова SQL	Назначение
А) SELECT	1. Выборка данных
Б) UPDATE	2.Изменение данных
В) DELETE	3. Удаление данных
Г) INSERT	4. Объединение таблиц
	5. Добавление данных

10. Установите соответствие между функциями SQL и их результатом

Функции SQL	Результат
А) COUNT	1. Максимальное значение
Б) SUM	2. Сумма значений
В) AVG	3. Среднее значение
Г) MAX	4. Количество записей
	5. Конкатенация строк

11. Установите соответствие между видами связей в базах данных и их описанием.

Виды связей в БД	Описание
А) Один-ко-многим	1. Каждой записи в одной таблице соответствует только одна запись в другой
Б) Многие-ко-многим	2. Обобщающий термин для всех типов связей между таблицами
В) Один-к-одному	3. Одна запись в таблице А связана с несколькими записями в таблице Б, и наоборот
Г) Реляционная связь	4. Одна запись в главной таблице связана с несколькими записями в подчиненной таблице

12. Установите соответствие между типами запросов в SQL и их функциями.

Типы запросов в SQL	Функции
А) Запрос на выборку	1. CREATE TABLE
Б) Запрос на обновление	2. INSERT INTO
В) Запрос на добавление	3. SELECT
Г) Запрос на удаление	4. UPDATE
	5. DELETE

13. Установите соответствие между объектами Ms Access и их функциями

Объекты Ms Access	Функции
А) Таблицы	1. Хранение данных
Б) Запросы	2. Выборка данных с условиями
В) Формы	3. Ввод данных через интерфейс
Г) Отчеты	4. Настройка прав доступа
	5. Визуальное представление данных для печати

14. Установите соответствие между типами данных в СУБД и их описанием

Типы данных СУБД	Описание
А) TEXT	1. Целое число
Б) INTEGER	2. Дата и время
В) BOOLEAN	3. Текст
Г) DATETIME	4. Логический тип
	5. Десятичное число

15. Установите соответствие между объектом базы данных и его назначением.

Объект БД	Назначение
А) Представление	1. Объект для генерации уникальных числовых

	значений
Б) Индекс	2. Виртуальная таблица, основанная на результате SQL-запроса
В) Последовательность	3. Набор SQL-инструкций, сохраняемый и выполняемый на стороне СУБД.
Г) Хранимая процедура	4. Структура, ускоряющая поиск и извлечение данных из таблицы

Индикатор УМЕТЬ:

ПК 1.1. Производить установку и настройку компонентов автоматизированных (информационных) систем в защищенном исполнении в соответствии с требованиями эксплуатационной документации

1. Выберите правильный ответ из предложенных вариантов и обоснуйте свой выбор. Какой этап проектирования БД предполагает создание ER-диаграмм и выявление сущностей?

А) Физическое проектирование

Б) Логическое проектирование

В) Концептуальное проектирование

Г) Реализация БД

2. Выберите правильный ответ из предложенных вариантов и обоснуйте свой выбор: Что такое первичный ключ (Primary Key) в таблице базы данных?

А) Поле, которое не может содержать букв, только цифры.

Б) Поле, значения которого должны совпадать со значениями в другой таблице.

В) Поле или набор полей, однозначно идентифицирующих каждую запись в таблице.

Г) Поле, которое используется для сортировки данных по умолчанию.

3. Выберите правильный ответ из предложенных вариантов и обоснуйте свой выбор: Какой SQL-оператор используется для извлечения данных из базы данных?

А) UPDATE

Б) SELECT

В) GET

Г) EXTRACT

4. Выберите правильный ответ из предложенных вариантов и обоснуйте свой выбор. Что такое индекс в базе данных?

А) Это копия таблицы для ускорения запросов

Б) Это структура данных, ускоряющая операции извлечения данных по заданным атрибутам

В) Это оглавление базы данных

Г) Это специальная таблица, содержащая часто используемые данные

5. Выберите правильный ответ из предложенных вариантов и обоснуйте свой выбор. Какой основной принцип отличает распределенную БД от централизованной?

А) Физическое размещение данных на разных серверах

Б) Наличие нескольких копий данных

В) Использование разных СУБД для хранения данных

Г) Автоматическое резервное копирование

6. Выберите правильный ответ из предложенных вариантов и обоснуйте свой выбор. Какой компонент ER-диаграммы обозначается ромбом?

А) Сущность

Б) Атрибут

В) Слабая сущность

Г) Тип связи

7. Выберите правильный ответ из предложенных вариантов и обоснуйте свой выбор. Как реализуется связь «многие ко многим» в реляционной модели?

А) Добавлением внешнего ключа в одну из таблиц

Б) Созданием промежуточной таблицы

В) Использованием составного первичного ключа

Г) Путем дублирования данных

Индикатор ИМЕТЬ ПРАКТИЧЕСКИЙ ОПЫТ:

ПК 1.1. Производить установку и настройку компонентов автоматизированных (информационных) систем в защищенном исполнении в соответствии с требованиями эксплуатационной документации

1. Прочитайте текст и запишите развернутый обоснованный ответ. В табличной форме представлен фрагмент базы данных о результатах тестирования учащихся (используется стобалльная шкала).

Фамилия	Баллы	Математика	Химия	Информатика	Биология
Аганян	ж	57	93	43	62
Воронин	м	34	64	74	58
Григорчук	м	46	57	64	63
Роднина	ж	37	74	67	86
Сергеенко	ж	94	78	36	48
Черепанова	ж	74	83	82	92

Сколько записей в данном фрагменте удовлетворяют условию:

(Пол = «ж») И (Химия < Биология)?

В ответе укажите одно число – искомое количество записей.

2. Рассчитайте и запишите ответ. База данных службы доставки состоит из двух связанных таблиц:

Покупатель	Улица	Дом	Квартира	Подъезд	Этаж	Лифт
Андреева Б.В.	Цветочная	12	68	1	12	есть
Борисова В.Г.	Полевая	4	21	2	4	есть
Васильев Д.Е.	Луговая	1	7	1	2	есть
Дмитриева Е.И.	Цветочная	16	4	1	2	нет
Егоров И.К.	Луговая	5	79	3	3	есть
Захаров К.Л.	Полевая	16	14	2	3	нет

№ заказа	Покупатель	Наименование	Кол-во	Общий	Цена	Сумма
1	Васильев Д.Е.	Мед липовый	1 банка	350	243	243
2	Дмитриева Е.И.	Сахар-песок	1 уп.	900	29	29
3	Захаров К.Л.	Мед липовый	2 банки	700	243	486
4	Андреева Б.В.	Конфеты «Ассорти»	1 кор.	600	196	196
5	Андреева Б.В.	Халва	1 уп.	400	79	79
6	Борисова ВГ.	Печенье клубничное	4 пач.	800	42	168

Каков общий вес товаров, которые курьер должен доставить на ул. Полевая? В ответе укажите целое число.

3. Рассчитайте и запишите ответ. В фрагменте базы данных представлены сведения о родственных отношениях. На основании приведенных данных определите, сколько дочерей и внучек Бунько А.С. упомянуто в таблицах? В ответе укажите целое число.

Таблица 1

ID	Фамилия_И.О.	Пол
916	Демченко И.М.	Ж
926	Вейко А.В.	М
927	Вейко В.А.	М
928	Вейко В.В.	М
836	Алонсо Т.А.	Ж
837	Алонсо Б.Г.	Ж
838	Алонсо Г.Г.	М
546	Бунько А.С.	Ж
547	Бунько В.А.	М
548	Айсберг К.Г.	Ж
549	Айсберг И.К.	М
456	Ломако Н.В.	Ж
166	Канаян Г.В.	Ж

Таблица 2

ID_Родителя	ID_Ребенка
926	927
546	927
927	928
166	928
926	836
546	836
836	837
838	837
916	838
836	548
838	548
927	456
166	456

4. Прочитайте текст и запишите развернутый обоснованный ответ. Определите, какой атрибут необходимо добавить в сущность Автомобиль (VIN, Модель), для обеспечения связи с сущностью Владелец (Паспорт, ФИО) и связью 1:N.

5. Прочитайте текст и запишите развернутый обоснованный ответ. В проекте базы данных для университета имеются сущности «Студент» и «Курс». Студент может записываться на несколько курсов, а курс может посещаться многими студентами. Опишите, из каких таблиц правильно реализовать эту связь в реляционной модели данных.

6. Рассчитайте и запишите ответ. Какой объем памяти (в байтах) потребуется для хранения таблицы из 1 000 строк, если каждая строка содержит 5 полей: INT (4 байта), FLOAT (4 байта), DATE (3 байта), VARCHAR (в среднем 25 байт), BOOLEAN (1 байт)? В ответе запишите целую часть числа. Разрешается использовать калькулятор.

7. Рассчитайте и запишите ответ. Сколько страниц (по 8 КБ каждая) будет занято в кластерном индексе, если таблица содержит 500 000 записей, а размер одной записи – 200 байт? В ответе запишите целую часть числа. Разрешается использовать калькулятор.

8. Прочитайте текст и запишите развернутый обоснованный ответ. Определите первичные ключи для таблиц: Заказ (Номер_заказа, Дата, Клиент_ID); Клиент (ID, ФИО, Телефон).