

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«ДАГЕСТАНСКИЙ ГОСУДАРСТВЕННЫЙ ПЕДАГОГИЧЕСКИЙ
УНИВЕРСИТЕТ ИМ. Р. ГАМЗАТОВА»
ПРОФЕССИОНАЛЬНО-ПЕДАГОГИЧЕСКИЙ КОЛЛЕДЖ
КАФЕДРА ПРОФЕССИОНАЛЬНЫХ ДИСЦИПЛИН



ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ПО ДИСЦИПЛИНЕ
ОПЦ.03 БАЗЫ ДАННЫХ

Направление подготовки 09.01.03 Оператор информационных систем и ресурсов

Квалификация: Оператор информационных систем и ресурсов

Срок обучения по ОП: 10 мес. (очное обучение)

Форма обучения: очная

Образовательный стандарт (ФГОС) № 974 от 11.11.2022

Махачкала 2026

ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОСВОЕНИЯ

Освоение междисциплинарного курса должно способствовать формированию общих компетенций:

- ОК 01. Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам.
- ОК 02. Использовать современные средства поиска, анализа и интерпретации информации, и информационные технологии для выполнения задач профессиональной деятельности;
- ОК 04. Эффективно взаимодействовать и работать в коллективе и команде;
- ОК 05. Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста;
- ПК 1.1. Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием.
- ПК 1.6. Разрабатывать модули программного обеспечения для мобильных платформ.
- ПК 1.7. Выполнять операции с объектами базы данных

В результате освоения междисциплинарного курса обучающийся должен получить **практический опыт**:

- Разрабатывать алгоритм решения поставленной задачи и реализовывать его средствами автоматизированного проектирования.
- Разрабатывать код программного продукта на основе готовой спецификации на уровне модуля.
- Разрабатывать мобильные приложения.
- Использовать инструментальные средства на этапе отладки программного продукта.
- Проводить тестирование программного модуля по определенному сценарию.
- Осуществлять разработку кода программного модуля на современных языках программирования.
- Оформлять документацию на программные средства.

В результате освоения дисциплины обучающийся должен **уметь**:

Код умения	Название умения
У1	распознавать задачу и/или проблему в профессиональном и/или социальном контексте;
У2	определять задачи для поиска информации;
У4	организовывать работу коллектива и команды;
У5	грамотно излагать свои мысли и оформлять документы по профессиональной тематике на государственном языке, проявлять толерантность в рабочем коллективе
У6	Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием. Оформлять документацию на программные средства.
У7	Осуществлять разработку кода программного модуля на современных языках программирования. Оформлять документацию на программные средства.
У8	выполнять обновление информации в базах данных

В результате освоения дисциплины обучающийся должен **знать:**

Код знания	Название знания
31	актуальный профессиональный и социальный контекст, в котором приходится работать и жить;
32	номенклатура информационных источников, применяемых в профессиональной деятельности;
34	психологические основы деятельности коллектива, психологические особенности личности; основы проектной деятельности
35	особенности социального и культурного контекста; правила оформления документов и построения устных сообщений.
36	Основные этапы разработки программного обеспечения. Основные принципы технологии структурного и объектно-ориентированного программирования. Актуальная нормативно-правовая база в области документирования алгоритмов.
37	Основные этапы разработки программного обеспечения. Основные принципы технологии структурного и объектно-ориентированного программирования.
38	виды и правила построения запросов к базам данных

Общие компетенции:

Код компетенции	Формулировка компетенции	Знания, умения
ОК 01	Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам	Умения: распознавать задачу и/или проблему в профессиональном и/или социальном контексте; анализировать задачу и/или проблему и выделять её составные части; определять этапы решения задачи; выявлять и эффективно искать информацию, необходимую для решения задачи и/или проблемы; составить план действия; определить необходимые ресурсы; владеть актуальными методами работы в профессиональной и смежных сферах; реализовать составленный план; оценивать результат и последствия своих действий (самостоятельно или с помощью наставника) Знания: актуальный профессиональный и социальный контекст, в котором приходится работать и жить; основные источники информации и ресурсы для решения задач и проблем в профессиональном и/или социальном контексте; алгоритмы выполнения работ в профессиональной и смежных областях; методы работы в профессиональной и смежных сферах; структуру плана для решения задач; порядок оценки результатов решения задач профессиональной деятельности

ОК 02	Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности	Умения: определять задачи для поиска информации; определять необходимые источники информации; планировать процесс поиска; структурировать получаемую информацию; выделять наиболее значимое в перечне информации; оценивать практическую значимость результатов поиска; оформлять результаты поиска; применять средства информационных технологий для решения профессиональных задач; использовать современное программное обеспечение
		Знания: номенклатура информационных источников, применяемых в профессиональной деятельности; приемы структурирования информации; формат оформления результатов поиска информации; современные средства и устройства информатизации; порядок их применения и программное обеспечение в профессиональной деятельности в том числе с использованием цифровых средств.
ОК 04	Эффективно взаимодействовать и работать в коллективе и команде	Умения: организовывать работу коллектива и команды; взаимодействовать с коллегами, руководством, клиентами в ходе профессиональной деятельности
		Знания: психологические основы деятельности коллектива, психологические особенности личности; основы проектной деятельности
ОК 05	Осуществлять устную и письменную коммуникацию на государственном языке с учетом особенностей социального и культурного контекста.	Умения: грамотно излагать свои мысли и оформлять документы по профессиональной тематике на государственном языке, проявлять толерантность в рабочем коллективе
		Знания: особенности социального и культурного контекста; правила оформления документов и построения устных сообщений.

Профессиональные компетенции:

Код и наименование компетенции	Показатели освоения компетенции
ПК 1.1.	Практический опыт в: Разрабатывать алгоритм решения поставленной задачи и реализовывать его средствами автоматизированного проектирования.
	Умения: Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием. Оформлять документацию на программные средства. Оценка сложности алгоритма.
	Знания: Основные этапы разработки программного обеспечения. Основные принципы технологии структурного и объектно-ориентированного программирования.

	Актуальная нормативно-правовая база в области документирования алгоритмов.
ПК 1.6.	Практический опыт: Разрабатывать мобильные приложения.
	Умения: Осуществлять разработку кода программного модуля на современных языках программирования. Оформлять документацию на программные средства.
	Знания: Основные этапы разработки программного обеспечения. Основные принципы технологии структурного и объектно-ориентированного программирования.
ПК 1.7 Выполнять операции с объектами базы данных.	Практический опыт в: ведении и актуализации информационных баз данных;
	Умения: выполнять обновление информации в базах данных;
	Знания: виды и правила построения запросов к базам данных

ФОРМЫ И МЕТОДЫ КОНТРОЛЯ И ОЦЕНИВАНИЯ

Формы текущего контроля по междисциплинарному курсу:

- устный опрос (фронтальный, индивидуальный, комбинированный);
- тестирование (письменное или компьютерное);
- письменная проверка (ответы на вопросы, решение задач и примеров, составление тезисов, рефератов, выполнение схем, выполнение заданий для самостоятельной работы и др.);
- практическая проверка (при проведении практических и лабораторных занятий);
- самоконтроль и взаимопроверка.

Возможны и другие формы текущего контроля успеваемости, в том числе инновационные на основе информационно-коммуникационных технологий.

Преподаватель на одном учебном занятии может использовать одну или несколько форм текущего контроля.

Код и наименование профессиональных и общих компетенций, формируемых в рамках модуля	Критерии оценки	Методы оценки
ПК 1.1. Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием.	Оценка «отлично» - программный модуль разработан по имеющемуся алгоритму в среде разработки на указанном языке программирования методами объектно-ориентированного/структурного программирования и полностью соответствует техническому заданию, соблюдены и пояснены основные этапы разработки; документация на модуль оформлена и соответствует стандартам. Оценка «хорошо» - программный модуль	– Экзамен в форме собеседования; – Практическое задание по разработке программного модуля в соответствии с заданием. – Защита

	разработан по имеющемуся алгоритму в среде разработки на указанном языке программирования методами объектно-ориентированного/структурного программирования и практически соответствует техническому заданию с незначительными отклонениями, пояснены основные этапы разработки; документация на модуль оформлена и соответствует стандартам. Оценка «удовлетворительно» - программный модуль разработан по имеющемуся алгоритму в среде разработки на указанном языке программирования методами объектно-ориентированного/структурного программирования и соответствует техническому заданию; документация на модуль оформлена без существенных отклонений от стандартов	отчетов по лабораторным занятиям – Устный опрос – Тестирование – Интерпретация результатов наблюдений за деятельностью обучающегося
ПК 1.6. Разрабатывать модули программного обеспечения для мобильных платформ.	Оценка «отлично» - разработан модуль для заданного мобильного устройства с соблюдением основных этапов разработки на одном из современных языков программирования; при проверке работоспособности модуля на устройстве или эмуляторе установлено его соответствие спецификации. Оценка «хорошо» - разработан модуль для заданного мобильного устройства с учетом основных этапов разработки на одном из современных языков программирования; при проверке работоспособности модуля на устройстве или эмуляторе установлено соответствие выполняемых функций спецификации с незначительными отклонениями. Оценка «удовлетворительно» - разработан модуль для заданного мобильного устройства на одном из современных языков программирования; при проверке работоспособности модуля на устройстве или эмуляторе установлено соответствие основных выполняемых функций спецификации.	– Экзамен в форме собеседования : – Практическое задание по разработке программного модуля в соответствии с заданием. – Защита отчетов по лабораторным занятиям – Устный опрос – Тестирование – Интерпретация результатов наблюдений за деятельностью обучающегося
ОК 01. Выбирать способы решения задач профессиональной деятельности, применительно к	– Обоснованность постановки цели, выбора и применения методов и способов решения профессиональных задач. – Адекватная оценка и самооценка эффективности и качества выполнения профессиональных задач	– Экспертное наблюдение за выполнением работ

различным контекстам		
ОК 02. Использовать современные средства поиска, анализа и интерпретации информации, и информационные технологии для выполнения задач профессиональной деятельности	– Использование различных источников, включая электронные ресурсы, медиаресурсы, Интернет-ресурсы, периодические издания по специальности для решения профессиональных задач. – Эффективность использования информационно-коммуникационных технологий в профессиональной деятельности согласно формируемым умениям и получаемому практическому опыту	
ОК 04. Эффективно взаимодействовать и работать в коллективе и команде	– Взаимодействовать с обучающимися, преподавателями и мастерами в ходе обучения, с руководителями учебной и производственной практик. – Обоснованность анализа работы членов команды (подчиненных)	
ОК 05. Осуществлять устную и письменную коммуникацию на государственном языке с учетом особенностей социального и культурного контекста	– Демонстрировать грамотность устной и письменной речи, - ясность формулирования и изложения мыслей	

ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ МЕЖДИСЦИПЛИНАРНОГО КУРСА

ОП.03 БАЗЫ ДАННЫХ

№	Наименование темы	Результаты обучения (освоенные умения и знаний)	ПК, ОК	Текущий контроль успеваемости	Промежуточная аттестация	
1	2	3	4	5	6	
Раздел 1. Основные понятия баз данных.						
Тема 1.1. Основные понятия баз данных						
1.	Основные понятия теории БД	У1, У2, У4, У5, У6, У7, У8, 31, 32, 34, 34, 36, 37, 38	ОК 01, ОК 02, ОК 04, ОК 05, ПК 1.1., ПК 1.6., ПК 1.7.	Устный опрос, тестирование	Зачет	
2.	Анализ предметной области			Устный опрос, тестирование		
Тема 1.2. Взаимосвязи в моделях и реляционный подход к построению моделей						
3.	Логическая и физическая независимость данных	У1, У2, У4, У5, У6, У7, У8, 31, 32, 34, 34, 36, 37, 38	ОК 01, ОК 02, ОК 04, ОК 05, ПК 1.1., ПК 1.6., ПК 1.7.	Устный опрос, тестирование		
4.	Типы моделей данных. Реляционная модель данных			Устный опрос, тестирование		
5.	Реляционная алгебра			Устный опрос, тестирование		
Раздел 2. Проектирование баз данных						
Тема 2.1.Этапы проектирования баз данных						
6.	Основные этапы проектирования БД	У1, У2, У4, У5, У6, У7, У8, 31, 32, 34, 34, 36, 37, 38	ОК 01, ОК 02, ОК 04, ОК 05, ПК 1.1., ПК 1.6., ПК 1.7.	Устный опрос, тестирование		
7.	Концептуальное проектирование БД			Устный опрос, тестирование		
8.	Нормализация БД			Устный опрос, тестирование		
9.	Лабораторное занятие № 1. Нормализация реляционной БД, освоение принципов проектирования БД.	У1, У2, У4, У5, У6, У7, У8, 31, 32, 34, 34, 36,		Отчет по лабораторному занятию		
10.	Лабораторное занятие № 2. Проектирование			Отчет по лабораторному		

	реляционной БД. Нормализация таблиц.	37, 38		занятию
Тема 2.2 Проектирование структур баз данных				
11.	Средства проектирования структур БД	У1, У2, У4, У5, У6, У7, У8, 31, 32, 34, 34, 36, 37, 38	ОК 01, ОК 02, ОК 04, ОК 05, ПК 1.1., ПК 1.6., ПК 1.7.	Устный опрос, тестирование
12.	Организация интерфейса с пользователем			Устный опрос, тестирование
13.	Лабораторное занятие № 3. Создание проекта БД. Создание БД. Редактирование и модификация таблиц	У1, У2, У4, У5, У6, У7, У8, 31, 32, 34, 34, 36, 37, 38		Отчет по лабораторному занятию
14.	Лабораторная работа № 4. Создание файла проекта базы данных. Создание интерфейса входной формы. Использование исполняемого файла проекта БД, приемы			Отчет по лабораторному занятию
Тема 2.3. Организация запросов SQL				
15.	Основные понятия языка SQL. Синтаксис операторов, типы данных	У1, У2, У4, У5, У6, У7, У8, 31, 32, 34, 34, 36, 37, 38	ОК 01, ОК 02, ОК 04, ОК 05, ПК 1.1., ПК 1.6., ПК 1.7.	Устный опрос, тестирование
16.	Создание, модификация и удаление таблиц. Операторы манипулирования данными			Устный опрос, тестирование
17.	Организация запросов на выборку данных при помощи языка SQL			Устный опрос, тестирование
18.	Организация запросов на выборку данных при помощи языка SQL			Устный опрос, тестирование
19.	Сортировка и группировка данных в SQL			Устный опрос, тестирование
20.	Лабораторное занятие № 5. Создание и модификация таблиц БД. Выборка данных из БД. Модификация содержимого БД.	У1, У2, У4, У5, У6, У7, У8, 31, 32, 34, 34, 36, 37, 38		Отчет по лабораторному занятию
21.	Лабораторное занятие № 6. Обработка транзакций. Использование функций защиты для БД			Отчет по лабораторному занятию
Самостоятельная работа обучающихся:				
1. Размещение новых объектов в таблице. 2. Создание новых таблиц.				

ОЦЕНОЧНЫЕ СРЕДСТВА ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИ ОП. 03 БАЗЫ ДАННЫХ

Типовые задания для оценки знаний и умений

Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы

Тема 1.1. Основные понятия баз данных

Задание для устного опроса по темам

1. Дайте определения понятиям: информационная система, предметная область.
2. Что называется базой данных и каково ее место в ИС?
3. В чем различие между данными и метаданными?
4. Каково назначение систем управления базами данных?
5. Для чего используется словарь данных?
6. Назовите этапы развития БД.
7. Какую роль в развитии технологии БД сыграло появление ПК?
8. Каковы функции СУБД?
9. Назовите достоинства и недостатки существующих многопользовательских технологий с базами данных.
10. Каким образом прикладные программы взаимодействуют с БД?
11. Чем банк данных отличается от базы данных?
12. Какие компоненты входят в состав банка данных?
13. Что представляет собой трехуровневая архитектура СУБД?
14. В чем особенность уровня внешних моделей?
15. В чем особенность концептуального уровня?
16. В чем особенность физического уровня?
17. Что означает логическая и физическая независимость данных?

Тестовые вопросы

Тема 1.1. Основные понятия баз данных

Вариант 1

1. Выберите правильные ответы

К основным моделям данных относятся модели:

1. основанные на инвертированных списках (реляционные);
2. семейные;
3. иерархические;
4. транзакционные;
5. сетевые.

2. Выберите правильный ответ

В базе данных поддеревья могут иметь любое число корневых деревьев. Такая база данных состоит из набора записей и множества связей между этими записями. Это описание:

1. сетевой модели данных
2. реляционной модели данных
3. иерархической модели данных

3. Выберите правильный ответ

База данных построена таким образом, что таблицы и пути доступа к ним видны пользователям; при этом строки таблиц физически упорядочены в некоторой

последовательности. Это описание:

1. сетевой модели данных
2. реляционной модели данных
3. иерархической модели данных

4. Выберите правильный ответ

Реляционная база «Ученики» данных задана таблицей:

	Фамилия	Город	Школа	Класс	Балл
1	Петрова	Казань	7	11 а	43
2	Толмачев	Самара	3	9 б	47
3	Черепанов	Казань	44	10 г	35

В ней ... полей и ... записей

Реляционная БД "Ученики" данных задана таблицей (см рисунок) . В ней ... полей и ... записей.

1. 5 и 3
2. 3 и 5
3. 5 и 4
4. 4 и 5

5. Выберите правильный ответ

Отметьте все виды информации, которые могут храниться в базах данных

1. текст
2. числа
3. видеоролики
4. запахи

6. Вставьте пропущенное слово

Программное обеспечение для работы с базой данных называется _____ управления базами данных.

7. Выберите правильный ответ

Первичным ключом (ключом отношения, ключевым атрибутом) называется

1. совокупность правил порождения структур данных в базе данных, операций с ними, а также ограничений целостности, определяющих допустимые связи и значения данных, последовательность их изменения.
2. атрибут или набор атрибутов отношения, однозначно идентифицирующий каждый из его записей.
3. множество пар (имя атрибута, значение}, которое содержит одно вхождение каждого имени атрибута.
4. совокупность определенным образом организованной информации на какую-то тему (в рамках некоторой предметной области), предназначенная для хранения во внешней памяти компьютера и постоянного применения.

8. Выберите правильный ответ

В таблицу базы данных СКЛАД, содержащую 5 столбцов информации о товаре (наименование, поставщик, количество, дата окончания срока хранения, цена), внесена информация о 25 видах товара. Количество записей в таблице равно

1. 125
2. 5
3. 30
4. 25

9. Выберите правильный ответ

Для таблицы реляционной базы данных ложно утверждение, что

1. все столбцы таблицы содержат однородные по типу данные
2. каждая запись в таблице содержит однородные по типу данные
3. в таблице нет двух одинаковых записей
4. каждый столбец таблицы имеет уникальное имя

10. Выберите правильный ответ

СУБД представляет собой программный продукт, входящий в состав

1. системного программного обеспечения
2. операционной системы
3. прикладного программного обеспечения
4. систем программирования

Вариант 2

1. Выберите правильный ответ

База данных состоит из упорядоченного набора деревьев. Каждое дерево состоит из одного "корневого" (предок) и упорядоченного набора из нуля или более связанных с ним поддеревьев (потомки). Целостность связи между ними поддерживается автоматически. Это описание:

1. сетевой модели данных
2. реляционной модели данных
3. иерархической модели данных

2. Выберите правильные ответы

К основным моделям данных относятся модели:

1. основанные на инвертированных списках (реляционные);
2. семейные;
3. иерархические;
4. транзакционные;
5. сетевые.

3. Выберите правильный ответ

Наиболее точным аналогом реляционной базы данных может служить:

1. неупорядоченное множество данных
2. вектор
3. генеалогическое дерево
4. двумерная таблица

4. Выберите правильный ответ

Записью реляционной базы данных является

1. корень дерева
2. столбец таблицы
3. строка таблицы
4. дерево

5. Выберите правильные ответы

В реляционной базе данных столбец таблицы называется

1. поле
2. строка
3. запись
4. атрибут

6. Окончите предложение

Поле таблицы (или сочетание полей), которое однозначно определяет запись называется

_____.

7. Выберите правильный ответ

Внешние ключи - это

1. множество пар (имя атрибута, значение}, которое содержит одно вхождение каждого имени атрибута.
2. атрибут или набор атрибутов отношения, однозначно идентифицирующий каждый из его кортежей.
3. основной механизм для организации связей между таблицами и поддержания целостности и непротиворечивости информации в базе данных.

8. Выберите правильный ответ

Для первичного ключа ложно утверждение, что

1. в таблице может быть назначен только один первичный ключ
2. первичный ключ может быть простым и составным
3. первичный ключ может принимать нулевое значение
4. первичный ключ однозначно определяет каждую запись в таблице

9. Выберите правильный ответ

Совокупность логически связанных полей, характеризующих типичные свойства реального объекта – это

1. ключ
2. запись
3. отчет
4. поле

10. Выберите правильный ответ

В записи файла реляционной базы данных (БД) может содержаться

1. неоднородная информация (данные разных типов)
2. исключительно однородная информация (данные только одного типа)
3. только текстовая информация
4. исключительно числовая информация

Тема 1.2. Взаимосвязи в моделях и реляционный подход к построению моделей

Задание для устного опроса по темам

1. Что такое модель данных?
2. Для чего строится модель данных?
3. Укажите достоинства и недостатки иерархической модели данных.
4. Как организуется физическое размещение данных в БД иерархического типа?
5. Охарактеризуйте сетевую модель данных.
6. Охарактеризуйте реляционную модель данных.
7. Чем отличается реляционная модель данных от предшествующих ей моделей?
8. Что такое простой ключ и составной ключ?
9. Перечислите виды связей между объектами? Охарактеризуйте их.
10. Как проявляется иерархическая подчиненность в связи «один ко многим»?
11. Сколько реляционных операций образуют реляционную алгебру?
12. Перечислите и охарактеризуйте операции реляционной алгебры. Приведите примеры.

Тестовые вопросы

Тема 1.2. Взаимосвязи в моделях и реляционный подход к построению моделей

Тест «Реляционная модель данных»

Задание №1

Конечное множество упорядоченных пар вида $\langle A, T \rangle$, где A называется именем атрибута, а T обозначает имя некоторого базового типа или ранее определенного домена

Выберите несколько из 5 вариантов ответа:

- 1) Заголовок отношения
- 2) Нг
- 3) Схема отношения
- 4) Тело отношения
- 5) Значение отношения

Задание №2

Сопоставьте термины, используемые в классической реляционной модели, и термины, применяемые в обычных учебниках.

Укажите соответствие для всех 5 вариантов ответа:

- 1) Кортеж
- 2) Атрибут
- 3) Схема отношения

- ☐ Столбец таблицы
- ☐ Строка
- ☐ Запись
- ☐ Заголовок таблицы
- ☐ Поле

Задание №3

Набор пар $\langle \text{имя_VARr}, \text{Нг} \rangle$

Выберите один из 4 вариантов ответа:

- 1) Схема реляционной базы данных
- 2) Реляционная база данных
- 3) Значение отношения
- 4) Переменная отношения

Задание №4

Укажите степень схемы данного отношения

СЛУ_НОМЕР	СЛУ_ИМЯ	СЛУ_ЗАРП	СЛУ_ОТД_НОМЕР
2934	Иванов	22000.00	310
2935	Петров	30000.00	310
2936	Сидоров	18000.00	313
2937	Федоров	20000.00	310
2938	Иванова	22000.00	315

Запишите число:

Задание №5

Сколько кортежей будет в отношении, полученном из исходного при выполнении операции, СЛУЖАЩИЕ REMOVE ПРО_НОМ

СЛУЖАЩИЕ			
СЛУ_НОМЕР	СЛУ_ИМЯ	СЛУ_ЗАРП	ПРО_НОМ
2934	Иванов	22400.00	1
2935	Петров	29600.00	1
2936	Сидоров	18000.00	1
2937	Федоров	20000.00	1
2938	Иванова	22000.00	1
2934	Иванов	22400.00	2
2935	Петров	29600.00	2
2939	Сидоренко	18000.00	2
2940	Федоренко	20000.00	2
2941	Иваненко	22000.00	2

Запишите число:

Задание №6

Описание какой операции алгебры A приведено:

- ☐ $H_s = H_r \text{ minus } \{ \langle A, T \rangle \}$
- ☐ $B_s = \{ ts : \text{exists } tr \text{ exists } v (tr \in Br \text{ and } v \in T \text{ and } \langle A, T, v \rangle \in tr \text{ and } ts = tr \text{ minus } \{ \langle A, T, v \rangle \}) \}$

Выберите один из 4 вариантов ответа:

- 1) r <REMOVE> A
- 2) r <PROJECT> A
- 3) r <RENAME> (A, B)
- 4) r1 <AND> r2

Задание №7

Как будет производиться операция реляционной конъюнкции отношений,
СЛУЖАЩИЕ_В_ПРОЕКТЕ_1 и СЛУЖАЩИЕ_В_ПРОЕКТЕ_2

СЛУЖАЩИЕ В ПРОЕКТЕ 1			
СЛУ_НОМЕР	СЛУ_ИМЯ	СЛУ_ЗАРП	СЛУ_ОТД_НОМЕР
2934	Иванов	22000.00	310
2935	Петров	30000.00	310
2936	Сидоров	18000.00	313
2937	Федоров	20000.00	310
2938	Иванова	22000.00	315

СЛУЖАЩИЕ В ПРОЕКТЕ 2			
СЛУ_НОМЕР	СЛУ_ИМЯ	СЛУ_ЗАРП	СЛУ_ОТД_НОМЕР
2934	Иванов	22000.00	310
2935	Петров	30000.00	310
2939	Сидоренко	18000.00	313
2940	Федоренко	20000.00	310
2941	Иваненко	22000.00	315

ПРОЕКТЫ	
ПРО_НОМ	ПРОЕКТ_РУК
1	Иванов
2	Иваненко

СЛУЖАЩИЕ			
СЛУ_НОМЕР	СЛУ_ИМЯ	СЛУ_ЗАРП	ПРО_НОМ
2934	Иванов	22400.00	1
2935	Петров	29600.00	1
2936	Сидоров	18000.00	1
2937	Федоров	20000.00	1
2938	Иванова	22000.00	1
2934	Иванов	22400.00	2
2935	Петров	29600.00	2
2939	Сидоренко	18000.00	2
2940	Федоренко	20000.00	2
2941	Иваненко	22000.00	2

Выберите один из 4 вариантов ответа:

- 1) Как расширенное декартово произведение этих отношений
- 2) Как пересечение тел отношений-операндов
- 3) Как операция естественного соединения отношений
- 4) Как операция эквисоединения отношений

Задание №8

Для каких операций алгебры Кодда требуется выполнение совместимости по взятию объединения отношений:

Выберите несколько из 5 вариантов ответа:

- 1) UNION
- 2) INTERSECT
- 3) MINUS
- 4) DIVIDE BY
- 5) TIMES

Задание №9

Сколько операций входит в исходный базовый набор операций алгебры А

Запишите число:

Задание №10

Сколько операций входит в алгебру Кодда

Запишите число:

Задание №11

Частным случаем какой операции является операция взятия расширенного декартова произведения TIMES

Запишите ответ:

Задание №12

Операция реляционной дизъюнкции выражается через реляционный вариант операции "штрих Шеффера" как...

Выберите один из 3 вариантов ответа:

1) $\neg A \vee \neg B$

2) $A \vee A$

3) $\neg(A \vee B)$

Задание №13

К какому количеству операций можно свести набор операций алгебры A при использовании реляционных вариантов операций "штрих Шеффера" и "стрелка Пирса"

Запишите число:

Задание №14

Установите приоритет операций алгебры Кодда

Укажите соответствие для всех 9 вариантов ответа:

1) 1

2) 2

3) 3

4) 4

___ Операция переименования

___ Операция объединений отношений

___ Операция взятия разности отношений

___ Операция реляционного деления

___ Операция соединения

___ Операция взятия расширенного декартова произведения отношений

___ Операция пересечения отношений

___ Операция взятия проекции

___ Операция ограничения

Задание №15

Пусть у отношений-операндов операции TIMES имеются одноимённые атрибуты с одинаковыми доменами. Какую операцию алгебры Кодда необходимо произвести для обеспечения совместимости по взятию расширенного декартова произведения.

Запишите ответ:

Задание №16

Сколько кортежей будет в теле отношения СЛУЖАЩИЕ_В_ПРОЕКТЕ_1 MINUS СЛУЖАЩИЕ_В_ПРОЕКТЕ_2

СЛУЖАЩИЕ_В_ПРОЕКТЕ_1			
СЛУ_НОМЕР	СЛУ_ИМЯ	СЛУ_ЗАРП	СЛУ_ОТД_НОМЕР
2934	Иванов	22000.00	310
2935	Петров	30000.00	310
2936	Сидоров	18000.00	313
2937	Федоров	20000.00	310
2938	Иванова	22000.00	315

СЛУЖАЩИЕ_В_ПРОЕКТЕ_2			
СЛУ_НОМЕР	СЛУ_ИМЯ	СЛУ_ЗАРП	СЛУ_ОТД_НОМЕР
2934	Иванов	22000.00	310
2935	Петров	30000.00	310
2939	Сидоренко	18000.00	313
2940	Федоренко	20000.00	310
2941	Иваненко	22000.00	315

Запишите число:

Задание №17

Результатом какой операции алгебры А является нижнее отношение

СЛУЖАЩИЕ			
СЛУ_НОМЕР	СЛУ_ИМЯ	СЛУ_ЗАРП	ПРО_НОМ
2934	Иванов	22400.00	1
2935	Петров	29600.00	1
2936	Сидоров	18000.00	1
2937	Федоров	20000.00	1
2938	Иванова	22000.00	1
2934	Иванов	22400.00	2
2935	Петров	29600.00	2
2939	Сидоренко	18000.00	2
2940	Федоренко	20000.00	2
2941	Иваненко	22000.00	2

СЛУ_НОМЕР	СЛУ_ИМЯ	СЛУ_ЗАРП
2934	Иванов	22400.00
2935	Петров	29600.00
2936	Сидоров	18000.00
2937	Федоров	20000.00
2938	Иванова	22000.00
2939	Сидоренко	18000.00
2940	Федоренко	20000.00
2941	Иваненко	22000.00

Выберите один из 4 вариантов ответа:

- 1) СЛУЖАЩИЕ REMUV ПРО_НОМ
- 2) СЛУЖАЩИЕ REMOVE ПРО_НОМ
- 3) СЛУЖАЩИЕ DELETE ПРО_НОМ
- 4) СЛУЖАЩИЕ ПРОЕКТ {СЛУ_НОМ, СЛУ_ИМЯ, СЛУ_ЗАРП}

Задание №18

Укажите верные высказывания

Выберите несколько из 5 вариантов ответа:

- 1) Если схемы отношений-операндов имеют непустое пересечение, то операция реляционной конъюнкции работает как естественное соединение
- 2) Если пересечение схем операндов пусто, то операция реляционной конъюнкции работает как расширенное декартово произведение
- 3) Если схемы отношений полностью совпадают, то результатом операции реляционной конъюнкции является пересечение двух отношений-операндов
- 4) Если схемы отношений полностью совпадают, то результатом операции реляционной конъюнкции является объединение двух отношений-операндов
- 5) Если схемы отношений-операндов имеют непустое пересечение, то операция реляционной конъюнкции работает как эквисоединение

Задание №19

Сколько кортежей содержится в отношении-результате операции СЛУЖАЩИЕ_В_ПРОЕКТЕ_1 <AND> СЛУЖАЩИЕ_В_ПРОЕКТЕ_2

СЛУЖАЩИЕ_В_ПРОЕКТЕ_1			
СЛУ_НОМЕР	СЛУ_ИМЯ	СЛУ_ЗАРП	СЛУ_ОТД_НОМЕР
2934	Иванов	22000.00	310
2935	Петров	30000.00	310
2936	Сидоров	18000.00	313
2937	Федоров	20000.00	310
2938	Иванова	22000.00	315

СЛУЖАЩИЕ_В_ПРОЕКТЕ_2			
СЛУ_НОМЕР	СЛУ_ИМЯ	СЛУ_ЗАРП	СЛУ_ОТД_НОМЕР
2934	Иванов	22000.00	310
2935	Петров	30000.00	310
2939	Сидоренко	18000.00	313
2940	Федоренко	20000.00	310
2941	Иваненко	22000.00	315

ПРОЕКТЫ	
ПРО_НОМ	ПРОЕКТ_РУК
1	Иванов
2	Иваненко

СЛУЖАЩИЕ			
СЛУ_НОМЕР	СЛУ_ИМЯ	СЛУ_ЗАРП	ПРО_НОМ
2934	Иванов	22400.00	1
2935	Петров	29600.00	1
2936	Сидоров	18000.00	1
2937	Федоров	20000.00	1
2938	Иванова	22000.00	1
2934	Иванов	22400.00	2
2935	Петров	29600.00	2
2939	Сидоренко	18000.00	2
2940	Федоренко	20000.00	2
2941	Иваненко	22000.00	2

Запишите число:

Задание №20

Какова мощность заголовка отношения-результата операции СЛУЖАЩИЕ <AND> ПРОЕКТЫ

СЛУЖАЩИЕ В ПРОЕКТЕ_1			
СЛУ_НОМЕР	СЛУ_ИМЯ	СЛУ_ЗАРП	СЛУ_ОТД_НОМЕР
2934	Иванов	22000.00	310
2935	Петров	30000.00	310
2936	Сидоров	18000.00	313
2937	Федоров	20000.00	310
2938	Иванова	22000.00	315
СЛУЖАЩИЕ В ПРОЕКТЕ_2			
СЛУ_НОМЕР	СЛУ_ИМЯ	СЛУ_ЗАРП	СЛУ_ОТД_НОМЕР
2934	Иванов	22000.00	310
2935	Петров	30000.00	310
2939	Сидоренко	18000.00	313
2940	Федоренко	20000.00	310
2941	Иваненко	22000.00	315
ПРОЕКТЫ			
ПРО_НОМ		ПРОЕКТ_РУК	
1		Иванов	
2		Иваненко	
СЛУЖАЩИЕ			
СЛУ_НОМЕР	СЛУ_ИМЯ	СЛУ_ЗАРП	ПРО_НОМ
2934	Иванов	22400.00	1
2935	Петров	29600.00	1
2936	Сидоров	18000.00	1
2937	Федоров	20000.00	1
2938	Иванова	22000.00	1
2934	Иванов	22400.00	2
2935	Петров	29600.00	2
2939	Сидоренко	18000.00	2
2940	Федоренко	20000.00	2
2941	Иваненко	22000.00	2

Запишите число:

Задание №21

Какова

мощность

заголовка

отношения-результата

операции

СЛУЖАЩИЕ В ПРОЕКТЕ_2<AND> ПРОЕКТЫ

СЛУЖАЩИЕ В ПРОЕКТЕ 1			
СЛУ_НОМЕР	СЛУ_ИМЯ	СЛУ_ЗАРП	СЛУ_ОТД_НОМЕР
2934	Иванов	22000.00	310
2935	Петров	30000.00	310
2936	Сидоров	18000.00	313
2937	Федоров	20000.00	310
2938	Иванова	22000.00	315
СЛУЖАЩИЕ В ПРОЕКТЕ 2			
СЛУ_НОМЕР	СЛУ_ИМЯ	СЛУ_ЗАРП	СЛУ_ОТД_НОМЕР
2934	Иванов	22000.00	310
2935	Петров	30000.00	310
2939	Сидоренко	18000.00	313
2940	Федоренко	20000.00	310
2941	Иваненко	22000.00	315
ПРОЕКТЫ			
ПРО_НОМ		ПРОЕКТ_РУК	
1		Иванов	
2		Иваненко	
СЛУЖАЩИЕ			
СЛУ_НОМЕР	СЛУ_ИМЯ	СЛУ_ЗАРП	ПРО_НОМ
2934	Иванов	22400.00	1
2935	Петров	29600.00	1
2936	Сидоров	18000.00	1
2937	Федоров	20000.00	1
2938	Иванова	22000.00	1
2934	Иванов	22400.00	2
2935	Петров	29600.00	2
2939	Сидоренко	18000.00	2
2940	Федоренко	20000.00	2
2941	Иваненко	22000.00	2

Запишите число:

Задание №22

Выберите верные высказывания

Выберите несколько из 5 вариантов ответа:

- 1) При выполнении операции реляционной дизъюнкции если у операндов нет общих атрибутов, то в тело результирующего отношения входят все такие кортежи ts, которые являются объединением кортежей tr1 и tr2, соответствующих заголовкам отношений-операндов, и хотя бы один из этих кортежей принадлежит телу одного из операндов
- 2) При выполнении операции реляционной дизъюнкции если у операндов имеются общие атрибуты, то в тело результирующего отношения входят все такие кортежи ts, которые являются объединением кортежей tr1 и tr2, соответствующих заголовкам отношений-операндов, если хотя бы один из этих кортежей принадлежит телу одного из операндов, и значения общих атрибутов tr1 и tr2 совпадают
- 3) При выполнении операции реляционной дизъюнкции, если схемы отношений-операндов совпадают, то тело отношения-результата является объединением тел операндов
- 4) При выполнении <OR>, если схемы отношений-операндов совпадают, то тело отношения-результата является пересечением тел операндов
- 5) При выполнении операции реляционной дизъюнкции если у операндов имеются общие атрибуты, то в тело результирующего отношения входят все такие кортежи ts, которые являются пересечением кортежей tr1 и tr2, соответствующих заголовкам отношений-операндов, если хотя бы один из этих кортежей принадлежит телу одного из операндов, и значения общих атрибутов tr1 и tr2 совпадают

Задание №23

Пусть есть домен ДОПУСТИМЫЕ_НОМЕРА_ПРОЕКТОВ = {1, 2, 3, 4, 5, 6, 7},

На нём определён атрибут ПРО_НОМ отношения НОМЕРА_ПРОЕКТОВ с рисунка.

Сколько кортежей будет входить в отношение-результат операции <NOT> НОМЕРА_ПРОЕКТОВ

НОМЕРА_ПРОЕКТОВ	
ПРО_НОМ	
1	
2	

Запишите число:

Задание №24

Какова мощность исходного базового набора операций Алгебры А

Запишите число:

Задание №25

Какое минимальное количество операций Алгебры А обеспечивает все необходимые выразительные возможности

Запишите число:

Задание №26

Вставьте пропущенное слово.

Реляционное исчисление основано на формальном механизме исчисления ... первого порядка

Запишите ответ:

Тема 2.1. Этапы проектирования баз данных

Задание для устного опроса по темам

1. Перечислите этапы, составляющие жизненный цикл БД.
2. Что является целью каждого этапа?
3. Какие работы ведутся на каждом из этапов?
4. Что называется концептуальной моделью?
5. Какие базовые понятия используются на этапе концептуального проектирования?
6. Какие задачи решаются на этапе концептуального проектирования?
7. Перечислите шаги концептуального проектирования.
8. Что называется сущностью и экземпляром сущности?
9. Что называется атрибутом сущности и экземпляром атрибута?
10. Что называется связью между сущностями?
11. Дайте определение понятию «класс принадлежности сущности».
12. На какие факторы опираются правила генерации таблиц из ER-диаграмм?
13. Опишите типовую пошаговую процедуру преобразования диаграммы «сущность - связь» в реляционную схему базы данных.
14. Назовите цели нормализации.
15. Чем опасно избыточное дублирование информации?
16. Назовите основные свойства нормальных форм.
17. Какие ограничения таблиц относят к 1НФ, 2НФ и 3НФ?
18. Приведите примеры таблиц, соответствующих и не соответствующих требованиям нормальных форм.

Тестовые вопросы

Тема 2.1. Этапы проектирования баз данных

1. Что обусловило появление систем управления базами данных:

Выберите один из 4 вариантов ответа:

- а) необходимость повышения эффективности работы прикладных программ
- б) появление современных операционных систем
- в) совместное использование данных разными прикладными программами
- г) большой объем данных в прикладной программе

2. Основное назначение СУБД:

Выберите один из 4 вариантов ответа:

- а) обеспечение независимости прикладных программ и данных
- б) представление средств организации данных одной прикладной программе
- в) поддержка сложных математических вычислений
- г) поддержка интегрированной совокупности данных

3. Что не входит в функции СУБД:

Выберите несколько из 7 вариантов ответа:

- а) создание структуры базы данных
- б) загрузка данных в базу данных
- в) предоставление возможности манипулирования данными
- г) проверка корректности прикладных программ, работающих с базой данных
- д) обеспечение логической и физической независимости данных
- е) защита логической и физической целостности базы данных
- ж) управление полномочиями пользователей на доступ к базе данных

4. Основные цели обеспечения логической и физической целостности базы данных:

Выберите несколько из 5 вариантов ответа:

- а) защита от неправильных действий прикладного программиста
- б) защита от неправильных действий администратора баз данных
- в) защита от возможных ошибок ввода данных
- г) защита от машинных сбоев
- д) защита от возможного появления несоответствия между данными после выполнения операций удаления и корректировки

5. Концептуальная модель...

Выберите один из 4 вариантов ответа:

- а) Интегрированные данные
- б) база данных
- в) обобщенное представление пользователей о данных
- г) описание представления данных в памяти компьютера

6. Как соотносятся понятия логической модели и концептуальной модели:

Выберите один из 4 вариантов ответа:

- а) это разные понятия
- б) логическая модель это вариант представления концептуальной модели
- в) это одно и то же
- г) логическая модель является частью концептуальной модели

7. Уровни архитектуры базы данных:

Выберите несколько из 5 вариантов ответа:

- а) нижний
- б) внешний
- в) концептуальный
- г) внутренний
- д) верхний

8. Какие понятия соответствуют внешнему уровню архитектуры базы данных:

Выберите несколько из 4 вариантов ответа:

- а) концептуальные требования пользователей
- б) внешние представления пользователей

- в) концептуальная модель
- г) обобщенное представление

9. Основные этапы проектирования базы данных:

Выберите несколько из 4 вариантов ответа:

- а) изучение предметной области
- б) проектирование обобщенного концептуального представления
- в) проектирование концептуального представления, специфицированного к модели данных СУБД (логической модели)
- г) разработка прикладных программ

10. Из каких этапов состоит первая стадия концептуального проектирования:

Выберите несколько из 5 вариантов ответа:

- а) изучение предметной области
- б) проектирование обобщенного концептуального представления
- в) проектирование концептуального представления, специфицированного к модели данных СУБД (логической модели)
- г) проектирование представления данных в памяти компьютера (структур хранения)
- д) разработка прикладных программ

Лабораторные занятия № 1-2

- Нормализация реляционной БД, освоение принципов проектирования БД.
- Проектирование реляционной БД. Нормализация таблиц.

Тема 2.2 Проектирование структур баз данных

Задание для устного опроса по темам

1. Приведите классификацию СУБД по различным признакам.
2. Какие требования предъявляются к СУБД?
3. Дайте определение CASE-средствам и CASE-технологии.
4. Назовите признаки классификации CASE-средств.
5. Дайте краткую характеристику СУБД Access.
6. Какие типы данных поддерживаются СУБД Access?
7. Какой режим представления данных обеспечивает максимальную гибкость для просмотра и ввода данных?
8. Какие действия можно выполнять, работая с формой?
9. Перечислите основные типы форм.
10. Перечислите способы создания форм.

Тестовые вопросы

Тема 2.2 Проектирование структур баз данных

Вариант 1

1. Таблицы в базах данных предназначены:

Выберите один из 5 вариантов ответа:

- а) для хранения данных базы
- б) для отбора и обработки данных базы
- в) для ввода данных базы и их просмотра
- г) для автоматического выполнения группы команд
- д) для выполнения сложных программных действий

2. Для чего предназначены запросы:

Выберите один из 6 вариантов ответа:

- а) для хранения данных базы
- б) для отбора и обработки данных базы
- в) для ввода данных базы и их просмотра
- г) для автоматического выполнения группы команд
- д) для выполнения сложных программных действий
- е) для вывода обработанных данных базы на принтер

3. Для чего предназначены макросы:

Выберите один из 6 вариантов ответа:

- а) для хранения данных базы
- б) для отбора и обработки данных базы
- в) для ввода данных базы и их просмотра
- г) для автоматического выполнения группы команд
- д) для выполнения сложных программных действий
- е) для вывода обработанных данных базы на принтер

4. В каком диалоговом окне создают связи между полями таблиц базы данных:

Выберите один из 4 вариантов ответа:

- а) таблица связей
- б) схема связей
- в) схема данных
- г) таблица данных

5. Без каких объектов не может существовать база данных:

Выберите один из 6 вариантов ответа:

- а) без отчетов
- б) без таблиц
- в) без форм
- г) без макросов
- д) без запросов
- е) без модулей

6. В чем состоит особенность поля "мемо":

Выберите один из 5 вариантов ответа:

- а) служит для ввода числовых данных
- б) служит для ввода действительных чисел
- в) многострочный текст
- г) имеет ограниченный размер
- д) имеет свойство автоматического наращивания

7. Система управления базами данных представляет собой программный продукт, входящий в состав:

Выберите один из 5 вариантов ответа:

- а) уникального программного обеспечения
- б) систем программирования
- в) системного программного обеспечения
- г) прикладного программного обеспечения
- д) операционной системы

8. Примером иерархической базы данных является:

Выберите один из 4 вариантов ответа:

- а) страница классного журнала
- б) каталог файлов, хранимых на диске
- в) расписание поездов
- г) электронная таблица

9. Информационная система, в которой БД и СУБД находятся на одном компьютере называется

Выберите один из 3 вариантов ответа:

- а) локальная
- б) файл-серверные
- в) клиент-серверные

10. Для первичного ключа ложно утверждение, что ...

Выберите один из 4 вариантов ответа:

- а) первичный ключ может принимать нулевое значение
- б) в таблице может быть назначен только один первичный ключ
- в) первичный ключ может быть простым и составным
- г) первичный ключ однозначно определяет каждую запись в таблице

11. Особенность поля «Счетчик» состоит в том, что ...

Выберите один из 4 вариантов ответа:

- а) оно имеет свойство автоматического наращивания
- б) данные хранятся не в самом поле, а в другом месте, а в поле хранится только указатель
- в) максимальный размер числа, хранящегося в нем, не может превышать 255
- г) оно предназначено для ввода целых чисел

12. Выбрать необходимые данные из одной или нескольких взаимосвязанных таблиц в MS Access, отобрать нужные поля, произвести вычисления и получить результат в виде новой таблицы можно с помощью ...

Выберите один из 4 вариантов ответа:

- а) запроса
- б) схемы данных
- в) главной кнопочной формы
- г) составной формы

13. Основными объектами СУБД MS Access являются ...

Выберите один из 4 вариантов ответа:

- а) таблица, форма, отчет, запрос
- б) конструктор, мастер, шаблон, схема данных
- в) таблица, поле, запись, ключ
- г) схема данных, ключ, шаблон, отчет

14. База данных - это:

Выберите один из 4 вариантов ответа:

- а) совокупность данных, организованных по определенным правилам;
- б) совокупность программ для хранения и обработки больших массивов информации;
- в) интерфейс, поддерживающий наполнение и манипулирование данными;
- г) определенная совокупность информации.

15. Наиболее распространенными в практике являются:

Выберите один из 4 вариантов ответа:

- а) распределенные базы данных
- б) иерархические базы данных
- в) сетевые базы данных
- г) реляционные базы данных

Вариант 2

1. Для первичного ключа ложно утверждение, что ...

Выберите один из 4 вариантов ответа:

- а) первичный ключ может принимать нулевое значение
- б) в таблице может быть назначен только один первичный ключ
- в) первичный ключ может быть простым и составным
- г) первичный ключ однозначно определяет каждую запись в таблице

2. Особенность поля «Счетчик» состоит в том, что ...

Выберите один из 4 вариантов ответа:

- а) оно имеет свойство автоматического наращивания
- б) данные хранятся не в самом поле, а в другом месте, а в поле хранится только указатель

указатель

- в) максимальный размер числа, хранящегося в нем, не может превышать 255
- г) оно предназначено для ввода целых чисел

3. Выбрать необходимые данные из одной или нескольких взаимосвязанных таблиц в MS Access, отобрать нужные поля, произвести вычисления и получить результат в виде новой таблицы можно с помощью ...

Выберите один из 4 вариантов ответа:

- а) запроса
- б) схемы данных
- в) главной кнопочной формы
- г) составной формы

4. Основными объектами СУБД MS Access являются ...

Выберите один из 4 вариантов ответа:

- а) таблица, форма, отчет, запрос
- б) конструктор, мастер, шаблон, схема данных
- в) таблица, поле, запись, ключ
- г) схема данных, ключ, шаблон, отчет

5. База данных - это:

Выберите один из 4 вариантов ответа:

- а) совокупность данных, организованных по определенным правилам;
- б) совокупность программ для хранения и обработки больших массивов информации;
- в) интерфейс, поддерживающий наполнение и манипулирование данными;
- г) определенная совокупность информации.

6. Наиболее распространенными в практике являются:

Выберите один из 4 вариантов ответа:

- а) распределенные базы данных
- б) иерархические базы данных
- в) сетевые базы данных
- г) реляционные базы данных

7. Таблицы в базах данных предназначены:

Выберите один из 5 вариантов ответа:

- а) для хранения данных базы
- б) для отбора и обработки данных базы
- в) для ввода данных базы и их просмотра
- г) для автоматического выполнения группы команд
- д) для выполнения сложных программных действий

8. Для чего предназначены запросы:

Выберите один из 6 вариантов ответа:

- а) для хранения данных базы
- б) для отбора и обработки данных базы
- в) для ввода данных базы и их просмотра
- г) для автоматического выполнения группы команд
- д) для выполнения сложных программных действий
- е) для вывода обработанных данных базы на принтер

9. Для чего предназначены макросы:

Выберите один из 6 вариантов ответа:

- а) для хранения данных базы
- б) для отбора и обработки данных базы
- в) для ввода данных базы и их просмотра
- г) для автоматического выполнения группы команд
- д) для выполнения сложных программных действий
- е) для вывода обработанных данных базы на принтер

10. В каком диалоговом окне создают связи между полями таблиц базы данных:

Выберите один из 4 вариантов ответа:

- а) таблица связей
- б) схема связей
- в) схема данных
- г) таблица данных

11. Без каких объектов не может существовать база данных:

Выберите один из 6 вариантов ответа:

- а) без отчетов
- б) без таблиц
- в) без форм
- г) без макросов
- д) без запросов
- е) без модулей

12. В чем состоит особенность поля "мемо":

Выберите один из 5 вариантов ответа:

- а) служит для ввода числовых данных
- б) служит для ввода действительных чисел
- в) многострочный текст
- г) имеет ограниченный размер
- д) имеет свойство автоматического наращивания

13. Система управления базами данных представляет собой программный продукт, входящий в состав:

Выберите один из 5 вариантов ответа:

- а) уникального программного обеспечения
- б) систем программирования
- в) системного программного обеспечения
- г) прикладного программного обеспечения
- д) операционной системы

14. Примером иерархической базы данных является:

Выберите один из 4 вариантов ответа:

- а) страница классного журнала
- б) каталог файлов, хранимых на диске
- в) расписание поездов
- г) электронная таблица

15. Информационная система, в которой БД и СУБД находятся на одном компьютере:

Выберите один из 3 вариантов ответа:

- а) локальная
- б) файл-серверные

в) клиент-серверные

Лабораторное занятие № 3-4

- Создание проекта БД. Создание БД. Редактирование и модификация таблиц
- Создание файла проекта базы данных. Создание интерфейса входной формы. Использование исполняемого файла проекта БД, приемы создания и управления.

Тема 2.3. Организация запросов SQL

Задание для устного опроса по темам

1. Дайте определение понятию SQL.
2. Перечислите категории команд языка SQL.
3. В чем состоит основное достоинство SQL?
4. Из каких слов состоит оператор SQL?
5. Какие ограничения накладываются на формат идентификатора?
6. Дайте определение понятию «метаязык».
7. Какие символы применяются в нотации БНФ? Что они обозначают?
8. Какие типы данных поддерживаются в SQL?
9. Приведите общий синтаксис SQL-оператора для создания таблицы.
10. Приведите общий синтаксис SQL-оператора для добавления столбца в таблицу.
11. Приведите общий синтаксис SQL-оператора для модификации столбца.
12. В каких случаях модификация столбца невозможна?
13. Приведите общий синтаксис SQL-оператора для удаления столбца.
14. В каких случаях удаление столбца невозможно?
15. Приведите общий синтаксис SQL-оператора для добавления записи в таблицу.
16. Почему список столбцов в данной команде не является обязательным параметром?
17. Приведите общий синтаксис SQL-оператора для модификации записи.
18. Какие операции допустимы в логических выражениях условия?
19. Приведите общий синтаксис SQL-оператора для удаления записи.
20. Приведите общий синтаксис SQL-оператора SELECT.
21. Для чего используются ключевые слова ALL и DISTINCT?
22. Для чего используется ключевое слово FROM?
23. Для чего используется ключевое слово WHERE?
24. Как произвести выборку данных из нескольких связанных таблиц?
25. С помощью какого ключевого слова осуществляется группировка данных в операторе SELECT?
26. Для чего используется ключевое слово HAVING?
27. С помощью какого ключевого слова осуществляется сортировка данных в операторе SELECT?
28. Как отсортировать данные по возрастанию (убыванию)?
29. Какие операции и функции можно выполнять над данными в SQL?
30. Что такое агрегатные функции? Какие функции входят в эту группу?
31. Какие функции для работы со строками в SQL вам известны

Тестовые вопросы

Тема 2.3. Организация запросов SQL

1. Для создания новой таблицы в существующей базе данных используют команду:
 1. NEW TABLE
 2. CREATE TABLE
 3. MAKE TABLE
2. Имеются элементы запроса: 1. *SELECT employees.name, departments.name*; 2. *ON employees.department_id=departments.id*; 3. *FROM employees*; 4. *LEFT JOIN departments*. В каком порядке их нужно расположить, чтобы выполнить поиск имен всех работников со всех отделов
 1. 1, 4, 2, 3
 2. 1, 2, 4, 3
 3. 1, 3, 4, 2

3. Как расшифровывается SQL

1. structured query language
2. strict question line
3. strong question language

4. Запрос для выборки всех значений из таблицы «Persons» имеет вид:

1. SELECT ALL Persons
2. SELECT * FROM Persons
3. SELECT .[Persons]

5. Какое выражение используется для возврата только разных значений:

1. SELECT DISINCT
2. SELECT DIFFERENT
3. SELECT UNIQUE

6. Для подсчета количества записей в таблице «Persons» используется команда:

1. COUNT ROW IN Persons
2. SELECT COUNT(*) FROM Persons
3. SELECT ROWS FROM Persons

7. Наиболее распространенным является тип объединения:

1. INNER JOIN
2. FULL JOIN
3. LEFT JOIN

8. Что возвращает запрос SELECT * FROM Students:

1. Все записи из таблицы «Students»
2. Рассчитанное суммарное количество записей в таблице «Students»
3. Внутреннюю структуру таблицы «Students»

9. Запрос «SELECT name ____ Employees WHERE age ____ 35 AND 50» возвращает имена работников, возраст которых от 35 до 50 лет. Заполните пропущенные места в запросе.

1. INTO, IN
2. FROM, IN
3. FROM, BETWEEN

10. Какая агрегатная функция используется для расчета суммы

1. SUM
2. AVG
3. COUNT

11. Запрос для выборки первых 14 записей из таблицы «Users» имеет вид:

1. SELECT * FROM Users LIMIT 14
2. SELECT * LIMIT 14 FROM Users
3. SELECT * FROM USERS

12. Выберите верное утверждение:

- SQL чувствителен к регистру при написании запросов
- SQL чувствителен к регистру в названиях таблиц при написании запросов
- SQL нечувствителен к регистру

13. Заполните пробелы в запросе «SELECT ____, Country FROM ____ », который возвращает имена заказчиков и страны, где они находятся, из таблицы «Customers».

1. *, Customers
2. NULL, Customers
3. Name, Customers

14. Запрос, возвращающий все значения из таблицы «Countries», за исключением страны с ID=8, имеет вид:

1. SELECT * FROM Countries EXP ID=8
2. SELECT * FROM Countries WHERE ID !=8
3. SELECT ALL FROM Countries LIMIT 8

15. Выберите запрос для выборки данных из таблицы «Customers», где условием является проживание заказчика в городе Москва

1. SELECT * FROM Customers WHERE City="Moscow"
2. SELECT City="Moscow" FROM Customers
3. SELECT Customers WHERE City="Moscow"

16. Выберите запрос, возвращающий имена, фамилии и даты рождения сотрудников

(таблица «Employees»). Условие – в фамилии содержится сочетание «se».

1. SELECT FirstName, LastName, BirthDate from Employees WHERE LastName="se"
2. PEKJAMA
3. SELECT * from Employees WHERE LastName like "_se_"
4. SELECT FirstName, LastName, BirthDate from Employees WHERE LastName like "%se%"

17. Какая функция позволяет преобразовать все буквы в выбранном столбце в верхний регистр:

1. TOP
2. UPPER
3. UP

18. Выберите запрос, позволяющий переименовать столбец LastName в Surname в таблице «Employees».

1. RENAME LastName into Surname FROM Employees
2. ALTER TABLE Employees CHANGE LastName Surname varchar(50)
3. ALTER TABLE Surname(LastName) FROM Employees

19. Для создания новой виртуальной таблицы, которая базируется на результатах сделанного ранее SQL запроса, используется команда:

1. CREATE VIRTUAL TABLE
2. CREATE VIEW
3. ALTER VIEW

20. В таблице «Employees» содержатся данные об именах, фамилиях и зарплате сотрудников. Выберите запрос, который изменит значение зарплаты с 2000 на 2500 для сотрудника с ID=7.

1. SET Salary=2500 FROM Salary=2000 FOR ID=7 FROM Employees
2. ALTER TABLE Employees Salary=2500 FOR ID=7
3. UPDATE Employees SET Salary=2500 WHERE ID=7

21. К какому результату приведет выполнение запроса DROP DATABASE Users:

1. Полное удаление базы данных «Users»
2. Блокировка на внесение изменений в базу данных «Users»
3. Удаление таблицы «Users» из текущей базы данных

22. В таблице «Animals» базы данных зоопарка содержится информация обо всех обитающих там животных, в том числе о лисах: red fox, grey fox, little fox. Выберите запрос, возвращающий информацию о возрасте лис.

1. SELECT %fox age FROM Animals
2. SELECT age FROM Animals WHERE Animal LIKE «%fox»
3. SELECT age FROM %Fox.Animals

23. Что возвращает запрос SELECT FirstName, LastName, Salary FROM Employees Where Salary<(Select AVG(Salary) FROM Employees) ORDER BY Salary DESC

1. Имена, фамилии и зарплаты сотрудников, значения которых соответствуют среднему значению среди всех сотрудников
2. Имена, фамилии сотрудников и их среднюю зарплату за весь период работы, с выполнением сортировки по убыванию
3. Имена, фамилии и зарплаты сотрудников, для которых справедливо условие, что их зарплата ниже средней, с выполнением сортировки зарплаты по убыванию

24. Выберите запрос, возвращающий значения из колонки «FirstName» таблицы «Users».

1. SELECT FirstName FROM Users
2. SELECT FirstName.Users
3. SELECT * FROM Users.FirstName

25. Выберите запрос, возвращающий информацию о заказчиках, проживающих в одном из городов: Москва, Тбилиси, Львов.

1. SELECT Moscow, Tbilisi, Lvov FROM Customers
2. SELECT * FROM Customers WHERE City IN ('Moscow', 'Tbilisi', 'Lvov')
3. SELECT City IN ('Moscow', 'Tbilisi', 'Lvov') FROM Customers

26. Какая команда используется для объединения результатов запроса без удаления дубликатов

1. UNION
2. UNION ALL

3. FULL JOIN

27. Оператор REVOKE предназначен для:

1. Предоставления пользователю или группе пользователей прав на осуществление определенных операций;
2. Задавания пользователю или группе пользователей запрета, который является приоритетным по сравнению с разрешением;
3. Отзыва у пользователя или группы пользователей выданных ранее разрешений

28. Для чего в SQL используются aliases

1. Для назначения имени источнику данных в запросе при использовании выражения в качестве источника данных или для упрощения структуры запросов
2. Для переименования полей
3. Для более точного указания источника данных, если в базе данных содержатся таблицы с одинаковыми названиями полей

29. Выберите запрос, который будет возвращать значения городов из таблицы «Countries».

1. SELECT * FROM Countries WHERE ID="City"
2. SELECT City FROM Countries
3. SELECT City.Countries

30. Имеются элементы запроса: 1. ORDER BY Name; 2. WHERE Age<19; 3. FROM Students; 4. SELECT FirstName, LastName. В каком порядке их нужно расположить, чтобы выполнить поиск имен и фамилий студентов в возрасте до 19 лет с сортировкой по имени:

1. 1, 4, 2, 3
2. 4, 2, 3, 1
3. 4, 3, 2, 1

31. Для чего в SQL используется оператор PRIVILEGE:

1. Для наделения суперпользователя правами администратора
2. Для выбора пользователей с последующим наделением их набором определенных прав
3. Такого оператора не существует

32. Напишите запрос, который будет возвращать текущую дату.

1. SELECT GetDate()
2. SELECT TodayDate()
3. SELECT Date(Today)

33. Какой оператор используется для выборки значений в пределах заданного диапазона:

1. WITHIN
2. IN
3. BETWEEN

Лабораторное занятие № 5-6

- Создание и модификация таблиц БД. Выборка данных из БД. Модификация содержимого БД.
- Обработка транзакций. Использование функций защиты для БД

**Самостоятельная работа обучающихся
СОЗДАНИЕ НОВЫХ ТАБЛИЦ.**

. Создайте таблицу «СписокПациентов», которая содержит следующие поля:

- 1) Для поля «МедПолис» создайте маску ввода по образцу: «АБ-1234».
 - 2) Для поля «Телефон» создайте маску ввода как в задании 3.
 - 3) Для поля «Пол» во вкладке Общие в строке «Значение по умолчанию» введите: муж. Это значение будет указано по умолчанию. В случае необходимости его можно заменить.
 - 4) Заполните таблицу данными (не менее 15 записей).
- 4.2. Создайте таблицу «НаименованиеУслуг», которая содержит следующие поля:

Заполните таблицу данными по примеру:

- 4.3. Создайте таблицу «Журнал», которая содержит следующие поля:

18

- 1) Для поля «НомерКарточки» используйте подстановку:

SELECT СписокПациентов.НомерКарточки, СписокПациентов.[ФИО Пациента] FROM СписокПациентов;

2) Поля «ФИО пациента» и «Должность врача» не заполняйте. Для их заполнения будет использован запрос на обновление. (Задание № 11)

3) Для поля «ФИО врача» используйте подстановочное поле из таблицы «Врачи».

5) Для поля «Услуги» используйте подстановку из таблиц услуг (таблица «НаименованиеУслуг»)

5.2. Критерии оценивания

5.2.1. Критерии оценивания устного ответа

При оценке устного ответа, обучающегося учитывается:

- 1) полнота и правильность ответа;
- 2) степень осознанности, понимания изученного;

Отметка «5»: ответ правильный, полный в соответствии с изученным материалом; материал изложен в определенной логической последовательности, литературным языком.

Отметка «4»: ответ правильный, полный в соответствии с изученным материалом; материал изложен в определенной логической последовательности; возможны отдельные затруднения в формулировке выводов.

Отметка «3»: ответ, в котором в основном правильно, но схематично или с отклонениями от последовательности изложения раскрыт материал или неполный, несвязный ответ, изложенный нелогично

Отметка «2»: при ответе обнаружено непонимание обучающимся основного содержания учебного материала, неумение его анализировать допущены существенные ошибки, которые обучающийся не смог исправить при наводящих вопросах преподавателя, отсутствует логика в изложении материала, нет необходимых обобщений и самостоятельной оценки фактов; недостаточно сформированы навыки устной речи.

Критерии оценивания выполнения заданий на лабораторных и практических занятиях

- **Отметка «5»:** работа выполнена полностью и правильно; сделаны правильные выводы.
- **Отметка «4»:** работа выполнена правильно с учетом 1-2 несущественных ошибок, исправленных самостоятельно по требованию преподавателя.
- **Отметка «3»:** работа выполнена правильно не менее чем на половину или допущены 3-4 существенные ошибки.
- **Отметка «2»:** допущены 5 и более существенные ошибки в ходе работы, которые обучающийся не может исправить даже по требованию преподавателя.

Задания к лабораторным занятиям представлены в методических указаниях к лабораторным занятиям по МДК.01.03. Разработка мобильных приложений

5.2.3. Критерии оценивания тестовых заданий

Оценка в баллах	Степень выполнения задания
Неуд.	Выполнено от 0 до 49,9 % предложенных заданий
Удов.	Выполнено от 50 до 69,9% предложенных заданий
Хор.	Выполнено от 70 до 89,9% предложенных заданий
Отл.	Выполнено от 90 до 100% предложенных заданий

5.2.4. Общая классификация ошибок

При оценке знаний и умений учитываются ошибки и недочёты в работе.

Грубыми считаются ошибки:

- незнание определения основных понятий, законов, общепринятых символов обозначений величин;
- неумение выделить в ответе главное; обобщить результаты изучения;

- неумение применить знания для решения задач;
- неумение использовать полученные данные для выводов;
- неумение пользоваться первоисточниками, учебником, справочником;
- нарушение техники безопасности, небрежное отношение к оборудованию

Негрубыми считаются ошибки:

- неточность формулировок, определений, понятий, законов, вызванная неполнотой охвата основных признаков определяемого понятия или заменой 1-3 из этих признаков второстепенными;
- ошибки, вызванные несоблюдением условий проведения лабораторных занятий;
- недостаточно продуманный план устного ответа (нарушение логики изложения, подмена отдельных основных вопросов второстепенными);
- нерациональные методы работы со справочной литературой;
- неумение выполнять лабораторные задания в общем виде.

ОЦЕНОЧНЫЕ СРЕДСТВА ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО МЕЖДИСЦИПЛИНАРНОМУ КУРСУ ОП. 03 БАЗЫ ДАННЫХ

Промежуточная аттестация по дисциплине проводится в форме экзамена.

Вопросы к зачету по ОП. 03 БАЗЫ ДАННЫХ за I семестр

1. Дайте определения понятиям: информационная система, предметная область.
2. Что называется, базой данных и каково ее место в ИС
3. В чем различие между данными и метаданными
4. Каково назначение систем управления базами данных
5. Для чего используется словарь данных?
6. Назовите этапы развития БД.
7. Какую роль в развитии технологии БД сыграло появление ПК
8. Каковы функции СУБД
9. Назовите достоинства и недостатки существующих многопользовательских технологий с базами данных.
10. Каким образом прикладные программы взаимодействуют с БД
11. Чем банк данных отличается от базы данных?
12. Какие компоненты входят в состав банка данных?
13. Что представляет собой трехуровневая архитектура СУБД?
14. В чем особенность уровня внешних моделей?
15. В чем особенность концептуального уровня?
16. В чем особенность физического уровня?
17. Что означает логическая и физическая независимость данных?
18. Что такое модель данных?
19. Для чего строится модель данных?
20. Укажите достоинства и недостатки иерархической модели данных.
21. Как организуется физическое размещение данных в БД иерархического типа?
22. Охарактеризуйте сетевую модель данных.
23. Охарактеризуйте реляционную модель данных.
24. Чем отличается реляционная модель данных от предшествующих ей моделей?
25. Что такое простой ключ и составной ключ?
26. Перечислите виды связей между объектами? Охарактеризуйте их.
27. Как проявляется иерархическая подчиненность в связи «один ко многим»?
28. Сколько реляционных операций образуют реляционную алгебру?
29. Перечислите и охарактеризуйте операции реляционной алгебры. Приведите примеры.
30. Перечислите этапы, составляющие жизненный цикл БД.
31. Что является целью каждого этапа?
32. Какие работы ведутся на каждом из этапов?
33. Что называется, концептуальной моделью?
34. Какие базовые понятия используются на этапе концептуального

35. проектирования?
36. Какие задачи решаются на этапе концептуального проектирования?
37. Перечислите шаги концептуального проектирования.
38. Что называется, сущностью и экземпляром сущности?
39. Что называется, атрибутом сущности и экземпляром атрибута?
40. Что называется, связью между сущностями?
41. Дайте определение понятию «класс принадлежности сущности».
42. На какие факторы опираются правила генерации таблиц из ER-диаграмм?
43. Опишите типовую пошаговую процедуру преобразования диаграммы «сущность - связь» в реляционную схему базы данных.
44. Назовите цели нормализации.
45. Чем опасно избыточное дублирование информации?
46. Назовите основные свойства нормальных форм.
47. Какие ограничения таблиц относят к 1НФ, 2НФ и 3НФ?
48. Приведите примеры таблиц, соответствующих и не соответствующих
49. требованиям нормальных форм.
50. Приведите классификацию СУБД по различным признакам.
51. Какие требования предъявляются к СУБД?
52. Дайте определение CASE-средствам и CASE-технологии.
53. Назовите признаки классификации CASE-средств.
54. Дайте краткую характеристику СУБД Access.
55. Какие типы данных поддерживаются СУБД Access?
56. Какой режим представления данных обеспечивает максимальную гибкость для просмотра и ввода данных?
57. Какие действия можно выполнять, работая с формой?
58. Перечислите основные типы форм.
59. Перечислите способы создания форм.
60. Дайте определение понятию SQL.
61. Перечислите категории команд языка SQL.
62. В чем состоит основное достоинство SQL?
63. Из каких слов состоит оператор SQL?
64. Какие ограничения накладываются на формат идентификатора?
65. Дайте определение понятию «метаязык».
66. Какие символы применяются в нотации БНФ? Что они обозначают?
67. Какие типы данных поддерживаются в SQL?
68. Приведите общий синтаксис SQL-оператора для создания таблицы.
69. Приведите общий синтаксис SQL-оператора для добавления столбца в таблицу.
70. Приведите общий синтаксис SQL-оператора для модификации столбца.
71. В каких случаях модификация столбца невозможна?
72. Приведите общий синтаксис SQL-оператора для удаления столбца.
73. В каких случаях удаление столбца невозможно?
74. Приведите общий синтаксис SQL-оператора для добавления записи в таблицу.
75. Почему список столбцов в данной команде не является обязательным параметром?
76. Приведите общий синтаксис SQL-оператора для модификации записи.
77. Какие операции допустимы в логических выражениях условия?
78. Приведите общий синтаксис SQL-оператора для удаления записи.
79. Приведите общий синтаксис SQL-оператора SELECT.
80. Для чего используются ключевые слова ALL и DISTINCT?
81. Для чего используется ключевое слово FROM?
82. Для чего используется ключевое слово WHERE?
83. Как произвести выборку данных из нескольких связанных таблиц?
84. С помощью какого ключевого слова осуществляется группировка данных в операторе SELECT?
85. Для чего используется ключевое слово HAVING?
86. С помощью какого ключевого слова осуществляется сортировка данных в операторе SELECT?

87. Как отсортировать данные по возрастанию (убыванию)?
88. Какие операции и функции можно выполнять над данными в SQL?
89. Что такое агрегатные функции? Какие функции входят в эту группу?
90. Какие функции для работы со строками в SQL вам известны

6.2. Критерии оценивания ответов на экзамене

- оценка **«отлично»**, если студент обладает глубокими и прочными знаниями программного материала; при ответе на вопросы продемонстрировал исчерпывающее, последовательное и логически стройное изложение; правильно сформулировал понятия и закономерности по вопросам; сделал вывод по излагаемому материалу;
- оценка **«хорошо»**, если студент обладает достаточно полным знанием программного материала; его ответ представляет грамотное изложение учебного материала; но имеются существенные неточности в формулировании понятий и закономерностей по вопросам; не полностью сделаны выводы по излагаемому материалу;
- оценка **«удовлетворительно»**, если студент имеет общие знания основного материала без усвоения некоторых существенных положений; формулирует основные понятия с некоторой неточностью; затрудняется в приведении примеров, подтверждающих теоретические положения;
- оценка **«неудовлетворительно»**, если студент не знает значительную часть программного материала; допустил существенные ошибки в процессе изложения; не умеет выделить главное и сделать вывод; приводит ошибочные определения; ни один вопрос не рассмотрен до конца, наводящие вопросы не помогают.

7. . ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ

Основные печатные издания

1. Мартишин, С. А. Базы данных. Практическое применение СУБД SQL и NOSQL-типа для проектирования информационных систем: учеб. пособие / С. А. Мартишин, В. Л. Симонов, М. В. Храпченко. - М.: ИД "ФОРУМ: ИНФРА-М", 2018.-368 с.
2. Федорова, Г. Н. Основы проектирования баз данных: учебник / Г. Н. Федорова. Изд. 4-е, стер. - М.: ИЦ "Академия", 2020.-224 с.
- 7.

Основные электронные издания

1. Голицына, О. Л. Основы проектирования баз данных [Электронный ресурс]: учебное пособие / О. Л. Голицына, Т. Л. Партыка, И. И. Попов. — 2-е изд., перераб. и доп. — М.: ФОРУМ: ИНФРА-М, 2021. — 416 с. - Режим доступа: <https://znanium.com/catalog/product/1190668>
2. Грошев, А. С. Основы работы с базами данных: учебное пособие для СПО / А. С. Грошев. — Саратов: Профобразование, 2021. — 255 с. — ISBN 978-5-4488-1006-0. — Текст: электронный // Электронный ресурс цифровой образовательной среды СПО PROФобразование: [сайт]. — URL: <https://profspo.ru/books/102199>
3. Молдованова, О. В. Информационные системы и базы данных: учебное пособие для СПО / О. В. Молдованова. — Саратов: Профобразование, 2021. — 177 с. — ISBN 978-5-4488-1177-7. — Текст: электронный // Электронный ресурс цифровой образовательной среды СПО PROФобразование: [сайт]. — URL: <https://profspo.ru/books/106617>
4. Швецов, В. И. Базы данных: учебное пособие для СПО / В. И. Швецов. — Саратов: Профобразование, 2019. — 219 с. — ISBN 978-5-4488-0357-4. — Текст: электронный // Электронный ресурс цифровой образовательной среды СПО PROФобразование : [сайт]. — URL: <https://profspo.ru/books/86192>
5. Волк, В. К. Базы данных. Проектирование, программирование, управление и администрирование / В. К. Волк. — 2-е изд., стер. — Санкт-Петербург : Лань, 2022. — 340 с. — ISBN 978-5-8114-9682-2. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/198584>

КЛЮЧИ

ОП.03 БАЗЫ ДАННЫХ				
Тема 1.1. Основные понятия баз данных				
№ Вопросы - Ответ	№ Вопросы - Ответ	№ Вопросы - Ответ	№ Вопросы - Ответ	№ Вопросы - Ответ
Вариант 1				
1 - 1, 3, 5	3 - 2	5 - 1, 2, 3	7 - 2	9 - 2
2 - 1	4 - 1	6 - Система(ой)	8 - 4	10 - 3
Вариант 2				
1 - 3	3 - 4	5 - 1, 4	7 - 3	9 - 2
2 - 1, 3, 5	4 - 3	6 – Ключевым (первичным)	8 - 3	10 - 1
Тема 1.2. Взаимосвязи в моделях и реляционный подход к построению моделей				
Тест «Реляционная модель данных»				
1 - 1, 2, 3	7 - 2	13 - 3	19 - 2	25 - 2
2 - 2, 1, 1, 3, 2	8 - 1, 2, 3	14 - 4, 1, 1, 2, 2, 2, 3, 3	20 - 5	26 - "предикатов".
3 - 1	9 - 5	15 - "RENAME"	21 - 6	
4 - 4	10 - 10	16 - 3	22 - 1, 2, 3	
5 - 8	11 - "AND".	17 - 2	23 - 5	
6 - 1	12 - 1	18 - 1, 2, 3	24 - 5	
Тема 2.1. Этапы проектирования баз данных				
1 - в	3 - г, ж	5 - в	7 - б, в, г	9 - а, б, в
2 - г	4 - в, г, д	6 - б	8 - а, б	10 - а, б
Тема 2.2 Проектирование структур баз данных				
Вариант 1				
1 - а	4 - в	7 - г	10 - а	13 - а
2 - б	5 - б	8 - б	11 - а	14 - а
3 - г	6 - в	9 - а	12 - а	15 - г
Вариант 2				
1 - а	4 - а	7 - а	10 - в	13 - г
2 - а	5 - а	8 - б	11 - б	14 - б
3 - а	6 - г	9 - г	12 - в	15 - а
Тема 2.3. Организация запросов SQL				
1 - 2	8 - 1	15 - 1	22 - 2	29 - 2
2 - 3	9 - 3	16 - 3	23 - 3	30 - 3
3 - 1	10 - 1	17 - 2	24 - 1	31 - 3
4 - 2	11 - 1	18 - 2	25 - 2	32 - 1
5 - 1	12 - 3	19 - 2	26 - 2	33 - 3
6 - 2	13 - 3	20 - 3	27 - 3	
7 - 1	14 - 2	21 - 1	28 - 1	

КЛЮЧИ К ОТКРЫТЫМ ЗАДАНИЯМ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ ДИСЦИПЛИНЫ ОП.03 БАЗЫ ДАННЫХ

Тема 1.1. Основные понятия баз данных

Задание для устного опроса по темам

1. Дайте определения понятиям: информационная система, предметная область

Информационная система - это совокупность взаимосвязанных элементов, предназначенных для сбора, хранения, обработки и передачи информации с целью

поддержки принятия решений и управления.

Предметная область - это определенная сфера знаний, предметов или явлений, которая изучается или описывается в рамках конкретной дисциплины или информационной системы. Например, предметной областью может быть медицина, экономика, технологии или другая сфера деятельности.

2. Что называется базой данных и каково ее место в ИС?

База данных (БД) — именованная совокупность данных, отражающая состояние объектов и их отношений в рассматриваемой предметной области.

База данных - это организованная коллекция данных, хранящаяся в централизованном или распределенном виде и доступная для использования нескольким пользователям или приложениям. База данных включает в себя структуры хранения и методы доступа к данным, позволяющие эффективно хранить, обрабатывать и извлекать информацию.

Место базы данных в информационной системе заключается в том, что она является центральным элементом, который хранит и обрабатывает данные, необходимые для функционирования и обеспечения работы всей системы. База данных предоставляет доступ к информации, позволяет выполнять операции анализа данных, создавать отчеты, обрабатывать запросы пользователей и многое другое. Она является основой для всех операций с данными в информационной системе.

3. В чем различие между данными и метаданными?

Данные - это фактическая информация, представленная в виде символов, цифр, текста, звуков и т.д. Это конкретные значения или факты, которые обрабатываются и используются в информационных системах.

Метаданные - это независимые от программ пользователя данные. Они описывают характеристики данных, их структуру, форматы, местоположение, правила доступа и другие атрибуты. Метаданные позволяют лучше понять и организовать данные, обеспечивают их качество, управляемость и адекватное использование.

Таким образом, основное различие между данными и метаданными заключается в том, что данные представляют собой конкретную информацию, а метаданные представляют собой информацию о данных.

4. Каково назначение систем управления базами данных?

Система управления базами данных (СУБД) предназначена для хранения, организации, управления и обработки данных в базе данных. Основное назначение СУБД заключается в том, чтобы обеспечить доступ к данным пользователям и приложениям, эффективное управление данными, обеспечение целостности и безопасности информации.

5. Для чего используется словарь данных?

Словарь данных - это структурированное описание данных, используемых в информационной системе или базе данных. Он содержит информацию о всех элементах данных, их атрибутах, ограничениях, связях, форматах и других метаданных.

Словарь данных используется для следующих целей:

- Документация данных: словарь данных служит как справочник для описания структуры данных, их связей и атрибутов, что помогает пользователям лучше понимать и использовать данные.
- Согласованность и точность: словарь данных помогает обеспечить согласованность и единообразие данных, предотвращая дублирование и ошибки в данных.
- Управление данными: словарь данных поддерживает централизованное управление структурой и описанием всех данных, упрощая их администрирование и обеспечивая целостность.
- Разработка информационных систем: словарь данных используется при проектировании баз данных и информационных систем, помогая уточнять требования, определять структуры данных и связи между ними.

- Анализ данных и отчетность: словарь данных помогает аналитикам и разработчикам понять структуру данных, создавать запросы и отчеты, проводить анализ данных

6. Назовите этапы развития БД.

Развитие базы данных (БД) обычно проходит через несколько этапов, каждый из которых представляет определенные изменения и улучшения.

Основные этапы развития базы данных включают:

- Проектирование базы данных: на этом этапе определяются структура данных, связи между таблицами, атрибуты для каждой сущности, правила целостности и другие характеристики. В результате проектирования создается схема базы данных.
- Создание и заполнение базы данных: после проектирования базы данных происходит создание самой базы данных и заполнение ее данными. Данные могут быть введены вручную или импортированы из других источников.
- Эксплуатация базы данных: на этом этапе база данных начинает использоваться для различных целей, таких как хранение, доступ и обработка данных для конечных пользователей и приложений.
- Обслуживание базы данных: в течение эксплуатации базы данных требуется ее регулярное обслуживание, включающее резервное копирование, оптимизацию запросов, обновление структуры данных, обеспечение безопасности и др.
- Развитие базы данных: по мере изменения потребностей и требований организации, база данных должна быть развиваема и адаптироваться для новых задач и целей. На этом этапе вносятся изменения в структуру данных, расширяются возможности системы и улучшается производительность.
- Оптимизация и масштабирование: при активном использовании базы данных могут возникнуть проблемы с производительностью или масштабируемостью. В таком случае проводится оптимизация запросов, индексирование данных, повышение масштабируемости системы и т.д.

7. Какую роль в развитии технологии БД сыграло появление ПК?

Появление персональных компьютеров (ПК) сыграло значительную роль в развитии технологии баз данных. Вот несколько ключевых моментов, как появление ПК повлияло на развитие технологии баз данных:

- Доступность и удобство использования: ПК сделали технологию баз данных более доступной и удобной для пользователей обычного уровня. Ранее использование баз данных ограничивалось большими мэйнфреймами и мини-компьютерами, доступ к которым имели только специализированные специалисты. С появлением ПК пользователи могли работать с базами данных прямо с рабочего стола.
- Распространение реляционных баз данных: появление ПК способствовало распространению реляционных баз данных. Реляционные базы данных стали более популярными благодаря своей простоте и эффективности, что отразилось на их широком использовании на ПК.
- Разработка специализированных программ: разработчики программного обеспечения стали создавать специализированные программы для работы с базами данных на ПК. Это позволило пользователям без специальных знаний баз данных управлять информацией и создавать отчеты.
- Интеграция баз данных с офисными приложениями: с развитием ПК стали появляться программы для работы с офисными документами (текстовыми документами, таблицами, презентациями), которые умеют взаимодействовать с базами данных. Это сделало процессы обработки данных более

эффективными и удобными.

- Расширение возможностей анализа данных: с ПК стали появляться инструменты и программы для анализа и визуализации данных, что повысило эффективность работы с большими объемами информации.

Появление ПК существенно расширило доступность, удобство использования и функциональные возможности технологии баз данных, способствуя их широкому распространению и развитию.

8. Каковы функции СУБД?

Система управления базами данных (СУБД) – это программное обеспечение, разработанное для управления хранением, обновлением, доступом и анализом данных в базе данных. СУБД выполняет ряд ключевых функций, необходимых для эффективного управления данными. Основные функции СУБД включают:

- Создание и определение структуры данных: СУБД предоставляет средства для создания базы данных и определения структуры данных, включая таблицы, поля, связи между таблицами и ограничения целостности.
- Хранение и организация данных: СУБД обеспечивает механизмы для хранения данных в базе данных, оптимизируя их структуру и распределение на диске для улучшения производительности.
- Поиск и извлечение данных: СУБД предоставляет возможность выполнять запросы к данным, осуществляя поиск, фильтрацию и сортировку информации в базе данных.
- Обеспечение безопасности данных: СУБД обеспечивает механизмы контроля доступа к данным, управление пользователями и правами доступа, шифрование и другие средства для обеспечения безопасности и конфиденциальности информации.
- Поддержка целостности данных: СУБД поддерживает выполнение правил целостности данных, таких как уникальность значений, ссылочная целостность, ограничения целостности и прочие, чтобы обеспечить согласованность данных в базе.
- Транзакционная обработка: СУБД позволяет работать с транзакциями – наборами операций с данными, которые либо выполняются целиком, либо откатываются в случае ошибки, обеспечивая целостность данных.
- Оптимизация запросов и производительность: СУБД предоставляет оптимизатор запросов, который анализирует запросы пользователей и выбирает наилучшие пути выполнения, чтобы обеспечить высокую производительность системы.
- Резервное копирование и восстановление данных: СУБД позволяет выполнять резервное копирование данных и восстановление базы данных в случае сбоев, сбоев или потери данных. Эти и другие функции СУБД служат для обеспечения эффективного управления данными, обеспечения их целостности и безопасности, а также оптимизации производительности и доступа к информации пользователям и приложениям.

9. Назовите достоинства и недостатки существующих многопользовательских технологий с базами данных.

Достоинства:

- Разделение доступа к данным: возможность предоставления разного уровня доступа к данным различным пользователям.
- Совместная работа: возможность одновременного доступа к базе данных нескольким пользователям, что облегчает совместное редактирование и обновление информации.
- Удобство использования: пользователи могут работать с базой данных из

любого места, где есть доступ к интернету.

- Автоматизация процессов: совместное использование баз данных позволяет автоматизировать процессы сбора, анализа и обработки информации.

Недостатки:

- Безопасность: риск несанкционированного доступа к данным при многопользовательском использовании базы данных.
- Совместная работа: возможность конфликтов при одновременном редактировании данных несколькими пользователями.
- Цена: использование многопользовательских технологий с базами данных может требовать дополнительных затрат на лицензии и обслуживание.
- Производительность: при большом количестве пользователей база данных может ст и работать медленнее из-за перегрузки сервера.

10. Каким образом прикладные программы взаимодействуют с БД?

Прикладные программы взаимодействуют с базами данных через специальные языки программирования и запросы к базе данных, которые позволяют получать, изменять или удалять данные в базе.

Программы могут использовать различные технологии для взаимодействия с базой данных, такие как:

- SQL (Structured Query Language) - язык запросов, который позволяет программам выполнять операции с данными в базе (например, выборку данных, обновление, удаление и т.д.).
- API (Application Programming Interface) - набор интерфейсов и протоколов, которые позволяют приложениям обмениваться данными с базой данных.
- ORM (Object-Relational Mapping) - технология, которая обеспечивает преобразование данных между объектами в программе и реляционной базой данных.
- Драйверы баз данных - специальные программы, которые обеспечивают взаимодействие прикладных программ с конкретной базой данных, переводя запросы и ответы между форматами, используемыми программой и базой данных.

Таким образом, прикладные программы могут отправлять запросы к базе данных для получения необходимой информации, а также обновлять или изменять данные в базе в соответствии с логикой программы.

11. Чем банк данных отличается от базы данных?

Термины "банк данных" и "база данных" часто используются как синонимы, но они имеют некоторые различия в смысловом контексте: 1. База данных (database): это организованная коллекция данных, хранящаяся в централизованном репозитории и организованная таким образом, что позволяет эффективно управлять и получать доступ к информации. База данных обычно используется для хранения информации, которая имеет определенную структуру и связи между собой, и предназначена для управления данными в информационной системе. 2. Банк данных (data bank): термин "банк данных" более широкий и может относиться к любому хранилищу или коллекции данных, включая базы данных, документы, файлы и другие формы информации. Банк данных может содержать неструктурированные и разнородные данные, которые могут быть хранены и использованы для различных целей. Таким образом, база данных является частью банка данных и представляет собой структурированное хранилище данных, используемое для конкретного назначения, в то время как банк данных может включать в себя различные формы данных и информации.

12. Какие компоненты входят в состав банка данных?

Банк данных (data bank) включает в себя различные компоненты, которые обеспечивают хранение, управление и доступ к данным. Основные компоненты банка данных включают: 1. Базы данных (databases): это центральный компонент банка данных,

в котором хранятся структурированные данные в соответствии с определенной схемой. Каждая база данных может содержать несколько таблиц или коллекций данных, организованных для эффективного доступа и управления. 2. Метаданные (metadata): информация о структуре, организации и характеристиках данных, хранящихся в базе данных. Метаданные могут включать описания таблиц, полей, связей между данными, прав доступа и другие атрибуты данных. 3. Программное обеспечение (software): специализированное программное обеспечение, которое обеспечивает управление и доступ к данным в базе данных. Программное обеспечение может включать в себя системы управления базами данных (СУБД), средства разработки приложений, средства администрирования и т.д. 4. Пользователи (users): люди или приложения, которые имеют доступ к данным в банке данных и используют их для различных целей. Пользователи могут выполнять запросы к базе данных, обновлять и извлекать информацию, а также управлять данными в соответствии с их ролями и правами доступа. 5. Хранилища данных (data storage): физические носители, на которых хранятся данные, такие как жесткие диски, серверы, облачные хранилища и т.д. Хранилища данных обеспечивают надежное и безопасное сохранение информации, доступ к которой осуществляется через программное обеспечение и метаданные. Эти компоненты в совокупности позволяют эффективно управлять данными в банке данных, обеспечивать их сохранность, доступность и конфиденциальность, а также обеспечивать удобный и быстрый доступ к информации для пользователей.

13. Что представляет собой трехуровневая архитектура СУБД?

Трехуровневая архитектура системы управления базами данных (СУБД) - это концепция организации компонентов и функций СУБД на три уровня с целью улучшения модульности, управляемости и производительности системы. Трехуровневая архитектура состоит из следующих компонентов: 1. Внешний уровень (external level): этот уровень предназначен для пользователей конечных приложений, которые используют данные из базы данных. На этом уровне определяются представления данных, запросы и отчеты, а также интерфейсы для взаимодействия с базой данных. Пользовательские приложения работают с внешним уровнем через наборы функций и процедур, предоставляемых СУБД. 2. Концептуальный уровень (conceptual level): этот уровень определяет структуру и организацию данных в базе данных независимо от конкретных приложений. На концептуальном уровне разрабатывается логическая модель данных, в которой определяются таблицы, отношения и атрибуты данных. Этот уровень представляет собой абстракцию для внешних приложений и обеспечивает унифицированный доступ к данным для всех пользователей. 3. Внутренний уровень (internal level): это самый низкий уровень в трехуровневой архитектуре, в котором определяется физическая организация данных на уровне хранилища. На этом уровне учитываются технические аспекты хранения данных, такие как структура индексов, алгоритмы поиска, оптимизация запросов и распределение данных в хранилище. Внутренний уровень обеспечивает преобразование логических структур данных в физические структуры, оптимизированные для быстрого доступа и обработки. Трехуровневая архитектура позволяет разделять данные, логику и физическое хранилище данных, что облегчает разработку, поддержку и расширение систем управления базами данных. Кроме того, такая архитектура позволяет изолировать изменения на одном уровне от других и обеспечивает гибкость и масштабируемость баз данных.

14. В чем особенность уровня внешних моделей?

Уровень внешних моделей в трехуровневой архитектуре систем управления базами данных (СУБД) представляет собой интерфейс между конечными пользователями и базой данных. Основные особенности уровня внешних моделей включают: 1. Предоставление пользовательского доступа: уровень внешних моделей обеспечивает пользователям конечных приложений доступ к данным в базе данных через удобные и простые интерфейсы. Пользователи могут выполнять запросы, получать информацию и

взаимодействовать с данными, не зная подробностей физической организации базы данных. 2. Поддержка специфических потребностей: внешние модели данных на этом уровне могут быть специализированы под конкретные потребности различных групп пользователей. Например, разные пользователи могут иметь доступ к различным видам данных, представленным в удобной для них форме. 3. Управление представлениями данных: на уровне внешних моделей пользователи могут создавать, модифицировать и управлять различными представлениями данных в базе данных. Представления могут объединять данные из разных таблиц, фильтровать и группировать информацию и представлять ее в удобном формате для конечных пользователей. 4. Обеспечение безопасности доступа: уровень внешних моделей также отвечает за управление правами доступа пользователей к данным в базе данных. Различные пользователи могут иметь разные уровни доступа к данным в соответствии с их ролями и задачами, что позволяет обеспечить конфиденциальность и защиту информации. Особенность уровня внешних моделей заключается в том, что он создает абстракцию между структурой и организацией данных на более низких уровнях (концептуальном и внутреннем) и конечными пользователями, что упрощает работу с базой данных и повышает эффективность использования информации.

15. В чем особенность концептуального уровня?

Концептуальный уровень в трехуровневой архитектуре систем управления базами данных (СУБД) играет ключевую роль в организации данных и обеспечении их целостности. Основные особенности концептуального уровня включают: 1. Логическая модель данных: на концептуальном уровне разрабатывается логическая модель данных, которая определяет структуру и отношения между данными в базе данных. Логическая модель представляет собой абстракцию реального мира, в которой выделены сущности, их атрибуты и связи между ними. 2. Независимость данных от приложений: концептуальный уровень обеспечивает независимость данных от конкретных приложений, позволяя разрабатывать унифицированную модель данных, которая может использоваться различными приложениями. Это упрощает разработку приложений, поскольку изменения в структуре данных не требуют изменений во всех связанных приложениях. 3. Обеспечение целостности данных: на концептуальном уровне определяются ограничения целостности данных, такие как уникальность значений ключей, ссылочная целостность и правила валидации данных. Эти ограничения гарантируют целостность и согласованность данных в базе данных. 4. Разграничение доступа к данным: концептуальный уровень позволяет определить различные уровни доступа к данным для различных групп пользователей и приложений. Это обеспечивает безопасность данных и защиту от несанкционированного доступа. 5. Оптимизация запросов: на концептуальном уровне разрабатываются оптимальные структуры данных для ускорения выполнения запросов к базе данных. Это включает выбор подходящих индексов, оптимизацию запросов и структуры таблиц для улучшения производительности системы. Особенность концептуального уровня заключается в том, что он служит важным звеном между внешним уровнем (пользователями) и внутренним уровнем (физической организацией данных), обеспечивая логическое представление данных и их эффективное управление в базе данных.

16. В чем особенность физического уровня?

Физический уровень в трехуровневой архитектуре систем управления базами данных (СУБД) описывает способ хранения и организации данных на самом низком уровне, близком к железу. Основные особенности физического уровня включают: 1. Физическая организация данных: на физическом уровне определяется структура хранения данных, включая способы организации таблиц, индексов, файлов и блоков данных. Это включает выбор алгоритмов хранения, физическое распределение данных на диске и оптимизацию для ускорения доступа к данным. 2. Управление памятью и дисковым пространством: физический уровень включает в себя управление памятью для

кэширования данных и управление дисковым пространством для хранения данных на диске. Оптимизация использования памяти и управление дисковым пространством являются важными аспектами этого уровня. 3. Реализация структур данных: на физическом уровне происходит конкретная реализация логических структур данных, определенных на концептуальном уровне, в виде файлов, таблиц, индексов и других структур хранения данных. Это включает в себя выбор способа хранения данных (например, последовательный доступ или индексированный доступ) и оптимизацию производительности хранения и доступа к данным. 4. Управление транзакциями и блокировками: физический уровень включает в себя управление транзакциями и механизмами блокировки данных для обеспечения целостности и консистентности данных при многопользовательском доступе к базе данных. Эти механизмы обеспечивают согласованность операций над данными. Особенность физического уровня заключается в том, что он предоставляет конкретные реализации структур данных и методов доступа к данным, оптимизирует работу с физическим хранилищем данных и обеспечивает высокую производительность и эффективность работы системы управления базами данных.

17. Что означает логическая и физическая независимость данных?

Логическая и физическая независимость данных представляют собой два важных концепта в области баз данных, которые обеспечивают гибкость и удобство в управлении данными. 1. Логическая независимость данных: Логическая независимость данных означает, что логическая структура данных (логическая модель данных) может быть изменена без изменения приложений, которые используют эти данные. Это означает, что при изменении структуры базы данных (например, добавление новых таблиц или полей) приложения, работающие с этими данными, остаются незатронутыми. Логическая независимость достигается на уровне концептуальной модели данных в трехуровневой архитектуре СУБД. 2. Физическая независимость данных: Физическая независимость данных означает, что физическая реализация хранения данных на диске или в памяти компьютера может быть изменена без влияния на логическую структуру данных и приложения, которые используют эти данные. Система управления базами данных обеспечивает абстракцию логической структуры данных от физической организации данных, что позволяет изменять способы физического хранения данных без необходимости вносить изменения на высоких уровнях абстракции. Таким образом, логическая и физическая независимость данных позволяют разработчикам и администраторам баз данных вносить изменения в структуру и физическую организацию данных, не затрагивая работу приложений и обеспечивая гибкость и удобство в управлении данными.

Тема 1.2. Взаимосвязи в моделях и реляционный подход к построению моделей

Задание для устного опроса по темам

1. Что такое модель данных?

Модель данных - это абстрактное представление о структуре, организации и связях между данными в базе данных. Модель данных определяет способы хранения, обработки, доступа и управления данными в базе данных. Она описывает, как данные представлены, структурированы и связаны друг с другом.

Существует несколько типов моделей данных, включая:

- Иерархическая модель данных: данные представляются в виде иерархии, где каждый элемент данных связан с одним или несколькими элементами данных уровнем выше. Пример - модель IMS (Information Management System).
- Сетевая модель данных: данные представляются через сеть связей между записями. Элементы данных могут иметь несколько родителей или детей. Пример - модель CODASYL.
- Реляционная модель данных: данные представляются в виде таблиц, где каждая таблица представляет собой набор строк и столбцов. Связи между

таблицами устанавливаются с помощью ключевых полей. Пример - модель реляционной базы данных.

- Объектно-ориентированная модель данных: данные представляются в виде объектов, которые содержат как данные, так и методы обработки этих данных. Пример - модель Unified Modeling Language (UML).

Каждая модель данных обладает своими уникальными характеристиками, преимуществами и недостатками, что позволяет выбирать наиболее подходящую модель в зависимости от конкретных требований при проектировании базы данных. Модель данных играет важную роль в определении структуры и организации данных в базе данных и обеспечивает основу для построения информационных систем.

2. Для чего строится модель данных?

Модель данных строится для целого ряда целей, которые включают в себя:

- Проектирование базы данных: Модель данных используется для определения структуры данных, их типов и отношений между ними при разработке базы данных. Это помогает организовать данные таким образом, чтобы они максимально соответствовали требованиям бизнеса или прикладной задачи.
- Оптимизация запросов: Модель данных может использоваться для оптимизации запросов к базе данных. Путем правильного проектирования модели данных можно ускорить выполнение запросов и повысить производительность базы данных.
- Обеспечение целостности данных: Модель данных определяет правила и ограничения, соблюдение которых гарантирует целостность данных. Например, определение первичных и внешних ключей помогает обеспечить связи между данными и предотвратить ошибки при их обновлении.
- Понимание структуры данных: Модель данных позволяет пользователям или разработчикам лучше понимать структуру данных и взаимосвязи между ними. Это облегчает работу с базой данных, а также обеспечивает ее документацию.
- Интеграция систем: Модель данных позволяет определить структуру и формат данных для их обмена или интеграции между различными информационными системами. Строить модель данных необходимо на этапе проектирования базы данных, что позволяет четко определить структуру данных и отношения между ними, обеспечивая эффективное и удобное управление информацией.

3. Укажите достоинства и недостатки иерархической модели данных.

Иерархическая модель данных является одним из первых подходов к структурированию данных в базе данных. Вот преимущества и недостатки данной модели:

Достоинства иерархической модели данных:

- Простота и понятность: Иерархическая модель легко понимается и проста в реализации, что делает ее привлекательной для начинающих разработчиков.
- Эффективность чтения: Поскольку в иерархической модели все данные организованы в виде иерархии, доступ к данным и выполнение запросов по структуре иерархии могут быть быстрыми и эффективными.
- Предполагает жесткую структуру: Структура иерархической модели четко определена и ограничена, что облегчает управление данными и обеспечивает ее целостность.

Недостатки иерархической модели данных:

- Ограниченные возможности запросов: Иерархическая модель ограничивает возможности выполнения сложных запросов, так как доступ к данным возможен только по пути иерархии.

- Неудобство при изменении структуры: Изменение структуры данных в иерархической модели может быть сложным и затратным процессом, особенно при перестроении связей между элементами.
- Неудобство добавления новых связей: Добавление новых связей между элементами данных или реорганизация иерархии может быть затруднительным и приводить к дублированию данных.
- Недостаточная гибкость: Иерархическая модель не обладает такой гибкостью и расширяемостью, как реляционная модель, что ограничивает возможности работы с данными. В целом, иерархическая модель данных хорошо подходит для простых иерархических структур данных, но может быть неэффективной для сложных и разветвленных баз данных, где требуется гибкость и простота выполнения запросов.

4. Как организуется физическое размещение данных в БД иерархического типа?

В иерархической модели данных физическое размещение данных в базе данных организуется следующим образом:

- Блоки данных: Данные в иерархической базе данных хранятся в блоках данных, которые содержат информацию об элементах данных и их отношениях в структуре иерархии.
- Уровни иерархии: Каждый уровень иерархии может быть физически размещен на различных блоках данных в базе данных. Элементы данных на более высоких уровнях иерархии будут ссылаться на элементы данных на более низких уровнях.
- Связи и указатели: Для установления отношений между элементами данных на разных уровнях иерархии используются указатели или ссылки. Таким образом, каждый элемент данных может содержать указатель на своих потомков или родителей.
- Индексирование: Для ускорения доступа к данным в иерархической модели часто используется индексирование. Индексы помогают быстро находить элементы данных по заданным критериям или путям в структуре иерархии.
- Кластеризация данных: Для улучшения производительности и уменьшения фрагментации данных в базе данных могут применяться методы кластеризации. Кластеризация позволяет группировать данные, которые часто используются вместе, на одном или близко расположенных дисках.

Физическое размещение данных в иерархической базе данных играет важную роль в обеспечении быстрого доступа к данным и эффективной работы базы данных в целом. Организация данных таким образом позволяет оптимизировать процессы чтения и записи информации, а также обеспечить целостность и надежность хранения данных.

5. Охарактеризуйте сетевую модель данных.

Сетевая модель данных - это одна из первых моделей данных, представленных после иерархической модели, и представляет собой более гибкую и расширяемую структуру данных.

Вот основные характеристики сетевой модели данных:

- Структура основана на сети: В сетевой модели данные организованы в виде нескольких типов записей, которые соединены между собой, образуя сетевую структуру. В этой структуре каждая запись может быть связана с несколькими другими записями.
- Типы записей: В сетевой модели присутствуют два основных типа записей - сущностей (entity record) и связи (set record). Сущности представляют собой независимые объекты данных, а связи определяют отношения между сущностями.

- **Гибкость в отношениях:** Одной из особенностей сетевой модели является возможность установления сложных и многозначных связей между различными типами записей. Это делает модель более гибкой и подходящей для моделирования сложных структур данных.
- **Наличие указателей:** Для установления связей между записями в сетевой модели часто используются указатели или ссылки. Это позволяет эффективно обрабатывать запросы, требующие доступа к связанным записям.
- **Каскадное удаление:** Сетевая модель поддерживает каскадное удаление, что означает, что при удалении одной записи автоматически удаляются все связанные с ней записи.
- **Сложность разработки:** Из-за сложности в установлении связей и обработке запросов сетевая модель данных может быть труднее в разработке и обслуживании по сравнению с другими моделями, например, реляционной.

Сетевая модель данных предлагает более гибкий подход к организации данных, нежели иерархическая модель, но имеет свои особенности и недостатки, которые нужно учитывать при выборе модели данных для конкретного проекта.

6. Охарактеризуйте реляционную модель данных.

Реляционная модель данных - это одна из наиболее распространенных моделей данных, используемая в современных базах данных. Вот основные характеристики реляционной модели данных:

- **Таблицы:** Данные в реляционной модели организованы в виде таблиц, которые состоят из строк и столбцов. Каждая таблица представляет отдельный тип сущности или отношение.
- **Ключи:** Ключи используются для однозначной идентификации каждой строки в таблице. В реляционной модели часто используется первичный ключ (primary key) для уникальной идентификации записи и внешний ключ (foreign key) для установления связей между таблицами.
- **Нормализация:** Реляционная модель поддерживает процесс нормализации данных, который позволяет минимизировать избыточность информации и обеспечить целостность базы данных.
- **Ограничения и целостность:** Реляционная модель позволяет задавать различные ограничения и правила целостности данных, например, уникальность значений, целостность ссылочной целостности и другие.
- **SQL:** Для работы с данными в реляционной модели используется язык структурированных запросов SQL (Structured Query Language), который позволяет выполнять запросы к данным, обновлять информацию, создавать и модифицировать структуру базы данных.
- **Транзакции и ACID:** Реляционная модель баз данных поддерживает принципы ACID (Atomicity, Consistency, Isolation, Durability) для обеспечения надежности и целостности транзакций.
- **Отношения между таблицами:** В реляционной модели таблицы могут быть связаны между собой через общие атрибуты, что позволяет эффективно моделировать сложные структуры данных.

Реляционная модель данных широко применяется в различных областях, благодаря своей простоте и эффективности. Это позволяет эффективно организовывать данные, проводить сложные запросы, обеспечивать целостность информации и обеспечивать надежную работу базы данных.

7. Чем отличается реляционная модель данных от предшествующих ей моделей?

Реляционная модель данных отличается от предшествующих ей моделей, таких как иерархическая и сетевая, по ряду ключевых особенностей:

- Структура данных: В иерархической модели данные организованы в виде древовидной структуры, где каждый элемент имеет один родительский элемент. В сетевой модели данные представлены в виде сети, где каждый элемент может иметь несколько связей. В реляционной модели данные организованы в виде таблиц, состоящих из строк и столбцов, что позволяет представлять сущности и их связи более гибко.
- Ключи и отношения: Реляционная модель данных использует понятие ключей для однозначной идентификации строк в таблицах и установления отношений между данными. Это позволяет эффективно моделировать комплексные структуры данных и устанавливать связи между таблицами.
- Нормализация: Реляционная модель поддерживает процесс нормализации данных для минимизации избыточности информации и обеспечения целостности базы данных. Иерархическая и сетевая модели часто имеют проблемы с избыточными данными из-за своей ограниченной структуры.
- Язык запросов: В реляционной модели для работы с данными используется язык SQL, который позволяет выполнять разнообразные запросы к базе данных. В предшествующих моделях использовались собственные языки запросов, что делало их менее удобными и стандартизированными.
- Независимость данных: Реляционная модель данных обладает более высоким уровнем независимости данных, по сравнению с иерархической и сетевой моделями. Это упрощает процессы разработки, сопровождения и обновления баз данных.

В целом, реляционная модель данных представляет собой более гибкую, эффективную и удобную для работы структуру данных, чем предшествующие ей модели, что привело к ее широкому распространению и популярности в сфере баз данных.

8. Что такое простой ключ и составной ключ?

Простой ключ и составной ключ - это два основных типа ключей в реляционной модели данных, которые используются для однозначной идентификации записей в таблице. Вот их определения и различия:

- Простой ключ (Primary Key): - Простой ключ состоит из одного атрибута (одного столбца) в таблице. - Этот ключ должен быть уникальным для каждой записи в таблице, то есть значение ключа должно однозначно идентифицировать каждую запись. - Простой ключ обеспечивает уникальность данных в таблице и используется для обеспечения ссылочной целостности в связанных таблицах. - Простой ключ обычно отмечается как первичный ключ (Primary Key) в базе данных.

Пример простого ключа:

Таблица "Пользователи" - Идентификатор (ID) - Простой ключ

- Составной ключ (Composite Key): - Составной ключ состоит из двух или более атрибутов (столбцов) в таблице. - Этот ключ также должен быть уникальным для каждой записи, но для его идентификации требуется комбинация нескольких значений атрибутов. - Составной ключ используется, когда одного атрибута недостаточно для уникальной идентификации записи, и несколько атрибутов должны быть использованы вместе. - Составной ключ обеспечивает уникальность данных в таблице и используется для обеспечения ссылочной целостности в связанных таблицах.

Пример составного ключа: Таблица "Заказы" - Номер заказа (Order Number) + Идентификатор клиента (Customer ID) - Составной ключ.

Оба типа ключей играют важную роль в реляционной модели данных, обеспечивая

надежность и целостность информации в базе данных.

9. Перечислите виды связей между объектами? Охарактеризуйте их.

Существует несколько видов связей между объектами в базах данных. Вот основные виды связей и их характеристики:

- Один к одному (One-to-One): - Каждая запись в одной таблице связана с одной записью в другой таблице. - Эта связь используется, когда каждый объект первой таблицы имеет единственное соответствие с объектом второй таблицы, и наоборот. - Например, сущность "Студент" может иметь однозначное соответствие с сущностью "Учебная группа".
- Один ко многим (One-to-Many): - Каждая запись в одной таблице связана с несколькими записями в другой таблице. - Эта связь используется, когда объект из одной таблицы может быть связан с несколькими объектами из другой таблицы. - Например, сущность "Отдел" может иметь несколько сотрудников в таблице "Сотрудники".
- Многие ко многим (Many-to-Many): - Множество записей в одной таблице связано с множеством записей в другой таблице. - Эта связь требует дополнительной таблицы-связи, которая устанавливает соответствие между записями двух таблиц. - Например, сущности "Студент" и "Курс" могут иметь связь "Студенты-Курсы", где каждый студент может изучать несколько курсов, и каждый курс может посещаться несколькими студентами.
- Само-соотношение (Self-Referencing): - Объект в таблице может быть связан с другим объектом в той же таблице. - Такие связи используются, когда объекты имеют отношение с другими объектами того же типа. - Например, сущность "Сотрудник" может содержать поле "Руководитель", которое ссылается на другую запись сотрудника в той же таблице.

Эти виды связей являются основополагающими в реляционной модели данных и позволяют связывать данные между разными таблицами для обеспечения целостности и связности информации.

10. Как проявляется иерархическая подчиненность в связи «один ко многим»?

Иерархическая подчиненность в связи «один ко многим» проявляется следующим образом:

- Один ко многим (One-to-Many): - В этой типичной связи одна запись в одной таблице связана с несколькими записями в другой таблице. - Это означает, что у каждой записи в одной таблице существует отношение с несколькими записями в другой таблице, при этом каждая из этих связанных записей принадлежит только одной записи. - Например, один поставщик (таблица "Поставщики") может иметь несколько продуктов (таблица "Продукты"), и каждый продукт принадлежит только одному поставщику.

Иерархическая подчиненность в связи «один ко многим» означает, что сущности их связанных таблиц находятся в иерархии, где одна сущность является главной (родительской), а другая - подчиненной (дочерней). Это говорит о том, что зависимые отношения между данными формируют структуру, похожую на дерево или иерархию.

Тема 2.1. Этапы проектирования баз данных

Задание для устного опроса по темам

1. Перечислите этапы, составляющие жизненный цикл БД.

Жизненный цикл базы данных (БД) состоит из нескольких этапов, каждый из которых играет важную роль в процессе создания, использования и управления базой данных.

Вот основные этапы жизненного цикла базы данных:

- Постановка задачи и планирование: - Определение требований и целей для базы данных. - Анализ бизнес-процессов для выявления основных данных и их связей. - Разработка плана по созданию базы данных с указанием ресурсов, сроков и бюджета.
- Проектирование базы данных: - Определение структуры данных, включая таблицы, поля, связи и ограничения. - Создание схемы базы данных, оптимизированной для хранения, доступа и обработки данных. - Разработка логической и физической модели данных.
- Разработка и реализация: - Создание базы данных на основе разработанной схемы и модели. - Заполнение базы данных начальными данными, если необходимо. - Тестирование и проверка правильности работы базы данных.
- Эксплуатация и поддержка: - Режим эксплуатации базы данных, включая обеспечение доступности данных, производительности и безопасности. - Резервное копирование и восстановление данных. - Обеспечение поддержки пользователям, обновлений и оптимизации базы данных.
- Мониторинг и оптимизация: - Отслеживание и анализ производительности базы данных. - Оптимизация запросов и индексов для повышения скорости работы. - Внесение изменений в структуру базы данных в зависимости от потребностей.
- Удаление или замена: - Принятие решения об удалении базы данных в случае устаревания или невостребованности. - Планирование и выполнение процесса удаления или замены базы данных. - Обеспечение сохранности данных и переноса их в другое хранилище при необходимости.

Жизненный цикл базы данных является непрерывным процессом, который включает в себя все эти этапы для эффективного управления данными и обеспечения их актуальности, надежности и доступности.

2. Что является целью каждого этапа?

- Постановка задачи и планирование: - Цель этого этапа - определить требования и цели для создания базы данных, выявить основные бизнес-процессы и данные, а также разработать план действий для реализации проекта.
- Проектирование базы данных: - Цель этого этапа - определить структуру базы данных, включая таблицы, поля, связи и ограничения, создать схему логической и физической модели данных, которая будет оптимизирована для хранения, доступа и обработки данных.
- Разработка и реализация: - Цель этого этапа - создание базы данных на основе разработанной схемы и модели, заполнение базы данных начальными данными, при необходимости тестирование и проверка правильности работы базы данных.
- Эксплуатация и поддержка: - Цель этого этапа - обеспечить бесперебойную работу базы данных, в том числе обеспечить доступность данных, производительность и безопасность, обеспечить поддержку пользователей, внесение изменений и обновлений, а также обеспечить резервное копирование и восстановление данных.
- Мониторинг и оптимизация: - Цель этого этапа - отслеживать производительность базы данных, оптимизировать работу базы данных для повышения скорости запросов, вносить изменения в структуру базы данных для улучшения эффективности и производительности.
- Удаление или замена: - Цель этого этапа - принять решение об удалении базы данных в случае устаревания или невостребованности, планировать и выполнить процесс удаления или замены базы данных, обеспечить

сохранность данных и их перенос в другое хранилище при необходимости.

Каждый этап жизненного цикла базы данных имеет свою цель и выполняется для обеспечения эффективного управления данными, обеспечения их надежности, доступности и актуальности.

3. Какие работы ведутся на каждом из этапов?

- Постановка задачи и планирование: - Определение требований к базе данных. - Анализ бизнес-процессов и основных данных. - Разработка плана проекта создания базы данных.
- Проектирование базы данных: - Создание логической и физической модели данных. - Определение структуры данных, таблиц, полей и связей. - Разработка схемы базы данных.
- Разработка и реализация: - Создание самой базы данных. - Заполнение базы данных их начальными данными. - Тестирование и проверка правильности функционирования базы данных.
- Эксплуатация и поддержка: - Обеспечение доступности и безопасности данных. - Поддержка пользователей. - Резервное копирование и восстановление данных. - Внесение обновлений и изменений в базу данных.
- Мониторинг и оптимизация: - Мониторинг и анализ производительности базы данных. - Оптимизация запросов и индексов. - Внесение изменений в структуру базы данных для улучшения производительности.
- Удаление или замена: - Принятие решения об удалении или замене базы данных. - Планирование и выполнение процесса удаления или замены базы данных. - Перенос данных в другое хранилище или их удаление.

На каждом из этапов жизненного цикла базы данных проводятся определенные работы, направленные на создание, развитие, поддержку и оптимизацию базы данных, а также на решение проблем, возникающих в процессе использования и управления данными.

4. Что называется концептуальной моделью?

Концептуальная модель - это абстрактная модель, которая описывает основные концепции и отношения между ними в конкретной предметной области. Она представляет собой упрощенное представление реальности, которое помогает понять структуру и функционирование системы или процесса. Концептуальная модель обычно используется в информационных технологиях для проектирования баз данных или разработки программных продуктов.

5. Какие базовые понятия используются на этапе концептуального проектирования?

На этапе концептуального проектирования используются следующие базовые понятия:

- Цель проекта - определение основной задачи, которую необходимо решить с помощью проектирования.
- Требования к проекту - определение функциональных и нефункциональных требований, которым должен соответствовать проект.
- Концепция проекта - определение общего подхода к решению задачи, включая выбор архитектуры, технологии и ресурсы.
- Характеристики пользователей - определение целевой аудитории проекта и их потребностей.
- Анализ рисков - выявление потенциальных проблем и опасностей, которые могут возникнуть при реализации проекта.
- Модель данных - создание структуры данных и их взаимосвязей для обеспечения эффективного хранения и обработки информации.
- Прототипирование - создание простого прототипа проекта для визуализации

и проверки концепции перед началом реальной разработки.

- Оценка ресурсов - определение необходимых материальных и человеческих ресурсов для реализации проекта.

6. Какие задачи решаются на этапе концептуального проектирования?

На этапе концептуального проектирования решаются следующие задачи:

1. Определение основных потребностей и требований заказчика.
2. Создание концепций и идейных решений для разработки продукта или системы.
3. Проектирование общей структуры и функциональности продукта.
4. Выбор технологий и методов, которые будут использоваться при разработке.
5. Определение основных компонентов, интерфейсов и взаимодействий между ними.

6. Разработка концептуальной модели, описывающей ключевые концепции и отношения в предметной области.

7. Оценка рисков и возможных проблем, которые могут возникнуть при реализации проекта.

8. Подготовка основы для последующего технического проектирования и разработки.

7. Перечислите шаги концептуального проектирования.

Шаги концептуального проектирования могут включать в себя следующие этапы:

1. Сбор информации и анализ требований заказчика.
2. Формулирование основных целей проекта.
3. Создание концепции проекта и определение ключевых идейных решений.
4. Разработка общей структуры проекта и определение основных компонентов.
5. Выбор технологий и методов, которые будут использоваться при реализации проекта.

6. Проектирование интерфейсов и взаимодействий между компонентами.

7. Разработка концептуальной модели или концепции продукта.

8. Оценка рисков и возможных проблем.

9. Подготовка документации и презентация концепции заказчику или заинтересованным сторонам.

10. Проработка дальнейших этапов технического проектирования и разработки.

8. Что называется сущностью и экземпляром сущности?

Сущность (entity) в концептуальном проектировании баз данных представляет собой абстрактный объект, о котором должна храниться информация в базе данных. Сущность может иметь атрибуты (properties), которые описывают характеристики этой сущности. Например, если рассматривать базу данных для хранения информации о студентах, то "студент" может быть сущностью, а атрибутами этой сущности могут быть имя, фамилия, номер студенческого билета и т.д. Экземпляр сущности (entity instance) - это конкретный объект, который соответствует определенной сущности и содержит определенные значения атрибутов. Например, если "студент" - это сущность, то каждый конкретный студент (например, Иванов Иван с номером студенческого билета 12345) будет экземпляром этой сущности.

9. Что называется атрибутом сущности и экземпляром атрибута?

Атрибут сущности (entity attribute) - это характеристика или свойство, которое описывает определенную сущность в базе данных. Например, если рассматривать сущность "студент", то атрибутами этой сущности могут быть Имя, Фамилия, Номер студенческого билета и т.д. Экземпляр атрибута (attribute instance) - это конкретное значение атрибута, которое относится к определенному экземпляру сущности. Например, если у студента существуют атрибуты "Имя" и "Фамилия", то конкретные значения этих атрибутов для студента Иванова Ивана будут экземплярами атрибутов: "Иван" для атрибута "Имя" и "Иванов" для атрибута "Фамилия".

10. Что называется связью между сущностями?

Связь между сущностями (entity relationship) - это отношение, которое устанавливается между двумя или более сущностями в базе данных. Связь определяет, каким образом две сущности взаимодействуют друг с другом и как они связаны между собой.

Связь между сущностями может иметь различные типы, такие как:

1. Один к одному (one-to-one): каждая запись одной сущности соответствует одной записи другой сущности.
2. Один ко многим (one-to-many): каждая запись одной сущности соответствует нескольким записям другой сущности.
3. Многие ко многим (many-to-many): множество записей одной сущности соответствует множеству записей другой сущности.

Связи между сущностями помогают организовать данные в базе данных и позволяют работать с данными более эффективно и структурированно.

11. Дайте определение понятию «класс принадлежности сущности».

Понятие "класс принадлежности сущности" (entity class) относится к типу или категории сущности, которая описывает объект или предмет, с которым мы хотим работать в базе данных. Класс принадлежности сущности определяет набор атрибутов (полей) и методов (операций), которые принадлежат этой сущности. Например, если у нас есть сущность "Студент", класс принадлежности этой сущности будет включать в себя атрибуты такие как имя, фамилия, номер студенческого билета и методы для работы с этими атрибутами, например, добавление нового студента, удаление студента и т.д. Класс принадлежности сущности является логическим представлением этой сущности в программе или базе данных, которое определяет её структуру, свойства и поведение. Он является основой для создания экземпляров (объектов) этой сущности при работе с базой данных.

12. На какие факторы опираются правила генерации таблиц из ER-диаграмм?

Правила генерации таблиц из ER-диаграмм обычно опираются на следующие факторы:

1. Сущности (Entities): Каждая сущность в ER-диаграмме становится отдельной таблицей в базе данных. Атрибуты сущностей превращаются в столбцы этих таблиц.
2. Атрибуты (Attributes): Каждый атрибут сущности становится отдельным столбцом в таблице, соответствующей этой сущности. Тип данных атрибутов ER-диаграммы определяет тип данных столбцов в таблице.
3. Отношения (Relationships): Отношения между сущностями определяют связи между таблицами в базе данных. Атрибуты-ключи в таблицах используются для установления этих связей.
4. Ключи (Keys): В ER-диаграммах обычно выделяются первичные и внешние ключи. Первичные ключи становятся уникальными индексами в таблицах, а внешние ключи используются для связывания таблиц между собой.
5. Спецификации логического дизайна: Дополнительные правила и спецификации логического дизайна могут быть заданы для преобразования ER-диаграммы в таблицы, учитывая особенности конкретной базы данных или требования проекта.

13. Опишите типовую пошаговую процедуру преобразования диаграммы

1. Изучение и анализ исходной диаграммы для определения целей преобразования и необходимых изменений.
2. Выбор инструментов и программного обеспечения для преобразования диаграммы.
3. Открытие исходной диаграммы в выбранном программном обеспечении.
4. Выделение основных элементов диаграммы и их группировка для удобства работы.
5. Проектирование новой структуры диаграммы в соответствии с поставленными

целями преобразования.

6. Перемещение, добавление или удаление элементов диаграммы в соответствии с новой структурой.

7. Изменение формы, цвета, шрифта и других свойств элементов диаграммы для улучшения визуального представления.

8. Добавление текстовой информации, комментариев или подписей для более подробного описания диаграммы.

9. Повторное анализирование и проверка результатов преобразования диаграммы.

10. Сохранение и экспорт окончательной версии преобразованной диаграммы в нужном формате.

14. «сущность - связь» в реляционную схему базы данных.

Преобразование диаграммы «сущность-связь» в реляционную схему базы данных является процессом перевода концептуальной модели базы данных в структуру таблиц, связей и атрибутов для конкретной СУБД.

Вот шаги, которые следует выполнить для этого преобразования:

1. Идентификация основных сущностей, атрибутов и связей на диаграмме «сущность-связь».

2. Присвоение каждой сущности структуры реляционной таблицы, определяя основной ключ для каждой таблицы.

3. Преобразование связей между сущностями во внешние ключи, чтобы создать связи между таблицами.

4. Определение атрибутов каждой сущности и их соответствующих столбцов в таблице.

5. Проверка нормализации схемы для уменьшения избыточности и поддержания целостности данных.

6. Создание индексов для улучшения производительности запросов к базе данных.

7. Решение возможных проблем с производительностью, учетом объема данных, частоты запросов и потенциальной денормализации таблиц.

8. Отображение результирующей реляционной схемы базы данных на основе преобразованных сущностей и связей.

9. Создание скрипта или схемы для создания базы данных с преобразованной структурой и импорта данных из диаграммы «сущность-связь».

10. Тестирование и проверка функциональности базы данных на основе преобразованной реляционной схемы.

Эти шаги помогут преобразовать диаграмму «сущность-связь» в реляционную схему базы данных с учетом необходимых атрибутов, связей и структур данных.

15. Назовите цели нормализации. Чем опасно избыточное дублирование информации?

Основные цели нормализации в базах данных:

1. Уменьшение избыточности данных.

2. Исключение возможности вставки аномалий.

3. Улучшение структуры базы данных.

4. Оптимизация производительности запросов.

5. Обеспечение целостности данных.

Избыточное дублирование информации опасно тем, что:

1. Усложняет обновление данных и поддержание их целостности.

2. Может привести к противоречивым данным.

3. Увеличивает объем хранимой информации, что увеличивает затраты на хранение данных.

4. Уменьшает производительность базы данных из-за большого объема дублированных данных.

5. Затрудняет анализ и поиск информации из-за различных версий данных.

Тема 2.2 Проектирование структур баз данных

Задание для устного опроса по темам

1. Приведите классификацию СУБД по различным признакам.

Существует несколько способов классификации систем управления базами данных (СУБД) по различным признакам. Ниже приведены основные классификации СУБД:

1. По модели данных: - Реляционные СУБД (например, MySQL, PostgreSQL, Oracle). - Иерархические СУБД (например, IBM Db2, IMS). - Сетевые СУБД (например, IDMS, RDM). - Объектно-ориентированные СУБД (например, ObjectDB, Objectivity/DB). - Мульти-модельные СУБД (например, ArangoDB, OrientDB).

2. По способу доступа: - Клиент-серверные СУБД. - Встроенные СУБД. - Облачные СУБД.

3. По способу хранения данных: - Файловые СУБД. - СУБД с поддержкой стандартов SQL.

4. По функциональному назначению: - Отношенческие (для работы с реляционными данными). - Специализированные (для определенных отраслей или задач, например, геоинформационные СУБД).

5. По типу лицензирования: - Бесплатные (Open Source). - Коммерческие.

Это основные способы классификации СУБД по различным признакам, однако можно выделить и другие аспекты, такие как поддержка транзакций, масштабируемость, безопасность и т.д.

2. Какие требования предъявляются к СУБД?

1. Надежность: СУБД должна обеспечивать сохранность данных, обеспечивать защиту от потери информации и гарантировать целостность базы данных.

2. Производительность: СУБД должна обеспечивать быстрый доступ к данным, эффективное выполнение запросов и обработку больших объемов информации.

3. Масштабируемость: СУБД должна быть способна управлять большими объемами данных, поддерживать работу со множеством пользователей и обеспечивать расширяемость системы.

4. Безопасность: СУБД должна обеспечивать защиту данных от несанкционированного доступа, обеспечивать контроль доступа пользователей к информации и предотвращать утечки данных.

5. Совместимость: СУБД должна быть совместима с другими программными продуктами и обеспечивать возможность интеграции с различными приложениями.

6. Простота использования: СУБД должна быть удобной для администрирования, настройки и управления, обладать интуитивным интерфейсом для пользователей и администраторов.

7. Поддержка стандартов: СУБД должна соответствовать стандартам и нормативам баз данных, обеспечивать совместимость с языками запросов и другими технологиями.

8. Расширенные функциональные возможности: СУБД должна обладать широким набором функций, включая поддержку транзакций, многопользовательский доступ, репликацию данных и другие возможности.

3. Дайте определение CASE-средствам и CASE-технологии.

CASE (Computer-Aided Software Engineering) - это набор средств, методов и технологий, которые используются для автоматизированного проектирования и разработки программного обеспечения. В рамках CASE-средств разрабатываются инструменты для работы с диаграммами, моделями, базами данных, кодом и другими элементами, необходимыми для создания программного продукта. CASE-технология включает в себя программные средства для упрощения и автоматизации процесса разработки программного обеспечения, такие как CASE-системы, интегрированные средства и т.д. CASE-технология способствует повышению продуктивности разработчиков, улучшению качества кода и облегчает процесс управления проектом.

4. Назовите признаки классификации CASE-средств.

CASE-средства могут быть классифицированы по различным признакам, включая:

1. Функциональность: CASE-средства могут быть классифицированы по функциональным возможностям, таким как моделирование, проектирование, анализ, генерация кода, управление конфигурацией и др.

2. Применение: CASE-средства могут быть разделены на общего назначения и специализированные, например, CASE-средства для разработки веб-приложений, мобильных приложений, баз данных и т.д.

3. Поддерживаемые методологии: CASE-средства могут быть адаптированы для работы с определенными методологиями разработки, такими как Waterfall, Agile, Scrum и другими.

4. Вид доступа: CASE-средства могут быть классифицированы по способу доступа к данным и функциональности, например, локальные, веб-приложения, облачные сервисы.

5. Совместимость и интеграция: CASE-средства могут быть классифицированы по возможности интеграции с другими инструментами разработки, системами управления проектами и т.д.

5. Дайте краткую характеристику СУБД Access.

Microsoft Access - программный продукт от компании Microsoft, представляющий собой систему управления базами данных. Он предназначен для создания и управления базами данных, а также для создания отчетов и форм для работы с данными. Access позволяет пользователям легко создавать связанные таблицы, запросы, формы и отчеты без необходимости знания специализированного языка запросов. Отличительной особенностью Access является интеграция с другими продуктами Microsoft, такими как Excel, Outlook и SharePoint.

6. Какие типы данных поддерживаются СУБД Access?

СУБД Access поддерживает различные типы данных, включая:

1. Текстовые данные (Text)
2. Числовые данные (Number)
3. Дата и время (Date/Time)
4. Логический (Yes/No)
5. Валютные данные (Currency)
6. Автоинкремент (AutoNumber)
7. Hyperlink
8. Мемо (Memo)
9. Однобайтовое (Byte)
10. Определяемое пользователем (Attachment)

Это лишь некоторые из типов данных, которые поддерживаются в СУБД Access. каждый тип данных предназначен для хранения конкретного вида информации и имеет свои особенности и ограничения.

7. Какой режим представления данных обеспечивает максимальную гибкость для просмотра и ввода данных?

Для максимальной гибкости просмотра и ввода данных в СУБД Access рекомендуется использовать формы (Forms). Формы позволяют создавать пользовательские интерфейсы для ввода данных, отображения информации и выполнения различных операций с данными.

С использованием форм можно легко организовать удобный и интуитивно понятный интерфейс для пользователя, что делает процесс работы с данными более удобным и эффективным. Формы позволяют настраивать внешний вид и расположение элементов, добавлять кнопки управления, создавать комбинированные поля для выбора данных из списка, устанавливать ограничения на ввод информации и многое другое.

Таким образом, использование форм обеспечивает максимальную гибкость для

просмотра и ввода данных в СУБД Access.

8. Какие действия можно выполнять, работая с формой?

При работе с формой в СУБД Access можно выполнять множество действий, в том числе:

1. Ввод и редактирование данных: на форме можно вводить новые данные, редактировать существующие записи, удалять или копировать информацию.
2. Поиск и фильтрация: с помощью формы можно выполнять поиск по базе данных с использованием различных критериев и фильтров.
3. Сортировка данных: данные на форме можно сортировать по различным полям в порядке возрастания или убывания.
4. Создание отчетов: через форму можно создавать отчеты на основе данных из базы данных.
5. Выполнение расчетов и агрегирование данных: с помощью формы можно выполнять различные вычисления, суммировать данные, создавать статистические отчеты и диаграммы.
6. Навигация: с использованием формы можно быстро перемещаться по записям базы данных, переходить между различными формами и отчетами.
7. Взаимодействие с другими объектами базы данных: с формой можно взаимодействовать с другими объектами базы данных, такими как таблицы, запросы, отчеты и макросы.

Эти возможности делают работу с формами в СУБД Access удобной и эффективной для управления данными и выполнения различных операций.

9. Перечислите основные типы форм.

1. Форма для отправки информации
2. Форма для регистрации пользователя
3. Форма для заказа товаров или услуг
4. Форма обратной связи
5. Форма опроса или анкеты
6. Форма подписки на рассылку или новости
7. Форма для загрузки файлов
8. Форма оплаты заказа
9. Форма авторизации/входа на сайт

10. Перечислите способы создания форм.

1. HTML и CSS
2. JavaScript
3. jQuery
4. PHP
5. Python Django
6. Ruby on Rails
7. ASP.NET
8. AngularJS
9. ReactJS
10. Vue.js

Тема 2.3. Организация запросов SQL

Задание для устного опроса по темам

1. Дайте определение понятию SQL.

SQL (Structured Query Language) - это язык программирования, используемый для управления реляционными базами данных. SQL позволяет создавать, изменять и управлять данными в базе данных, извлекать информацию и выполнять различные операции, такие как добавление, удаление, обновление и фильтрацию данных. SQL состоит из различных команд и операторов, позволяющих взаимодействовать с базами

данных и выполнять различные запросы.

2. Перечислите категории команд языка SQL.

1. Команды управления данными (Data Manipulation Language, DML) - SELECT, INSERT, UPDATE, DELETE.
2. Команды определения данных (Data Definition Language, DDL) - CREATE, ALTER, DROP, TRUNCATE.
3. Команды управления доступом (Data Control Language, DCL) - GRANT, REVOKE.
4. Команды управления транзакциями (Transaction Control Language, TCL) - COMMIT, ROLLBACK, SAVEPOINT.
5. Команды управления метаданными (Data Manipulation Language, DML) - SHOW, DESCRIBE.

3. В чем состоит основное достоинство SQL?

Основное достоинство SQL заключается в его универсальности и простоте. SQL является стандартным языком запросов для взаимодействия с реляционными базами данных, что делает его удобным инструментом для работы с данными. SQL позволяет легко создавать сложные запросы для извлечения и обработки информации из баз данных, выполнять различные операции с данными, управлять структурой базы данных и управлять доступом пользователей к данным. Благодаря широкому распространению и поддержке SQL, разработчики и аналитики могут легко обмениваться и работать с данными в различных системах и приложениях.

4. Из каких слов состоит оператор SQL?

Оператор SQL состоит из ключевых слов, выражений, операндов и операторов. Операторы SQL используются для выполнения различных операций на данных в базе данных, таких как выборка данных, добавление и обновление записей, удаление данных и другие манипуляции с данными. Ключевые слова, такие как SELECT, INSERT, UPDATE, DELETE, используются для указания типа операции, которую нужно выполнить. Выражения и операнды используются для указания условий выборки и изменения данных. Операторы, такие как WHERE, AND, OR, NOT, используются для комбинирования условий и выполнения логических операций.

5. Какие ограничения накладываются на формат идентификатора?

В SQL идентификаторы (названия таблиц, столбцов, индексов и др.) подчиняются определенным правилам и ограничениям:

1. Идентификаторы могут содержать буквы (как строчные, так и заглавные), цифры и символ подчеркивания "_", но не могут содержать другие специальные символы или пробелы.
2. Идентификаторы могут начинаться только с буквы или символа подчеркивания "_", но не могут начинаться с цифры.
3. Максимальная длина идентификатора зависит от конкретной СУБД, но обычно она составляет от 64 до 128 символов.
4. Идентификаторы чувствительны к регистру в некоторых системах управления базами данных (СУБД), таких как PostgreSQL и MySQL, в то время как в других СУБД, таких как Oracle и SQL Server, они могут быть нерегистрозависимыми.
5. Некоторые ключевые слова языка SQL, например, SELECT, FROM, WHERE, не могут быть использованы в качестве идентификаторов.

Соблюдение этих ограничений поможет избежать ошибок при создании объектов в базе данных и обеспечит корректную работу запросов и операций с данными.

6. Какие символы применяются в нотации БНФ? Что они обозначают?

В нотации БНФ (бэк-несформальное описание) применяются следующие символы:

1. ::= - двоеточие и равно обозначают определение правила (нетерминала).
2. | - вертикальная черта обозначает выбор одного из нескольких вариантов.
3. {} - фигурные скобки обозначают повторение нуля или более раз.
4. [] - квадратные скобки обозначают опциональность символа.

5. <> - угловые скобки обычно используются для обозначения пользовательских вводов или аргументов.

Таким образом, символы в нотации БНФ используются для определения грамматики языка или формата данных, описывая правила синтаксиса и структуру, которую должна иметь последовательность символов.

7. Какие типы данных поддерживаются в SQL?

В SQL (Structured Query Language) поддерживаются следующие типы данных:

1. Целые числа (INTEGER, SMALLINT, BIGINT)
2. Числа с плавающей запятой (FLOAT, REAL, DOUBLE)
3. Символьные строки (CHAR, VARCHAR, TEXT)
4. Дата и время (DATE, TIME, TIMESTAMP)
5. Булевы значения (BOOLEAN)
6. Бинарные данные (BLOB)
7. Другие специализированные типы, например, географические данные (GEOMETRY)

Каждая СУБД может поддерживать дополнительные типы данных в зависимости от своих особенностей и реализации.

8. Какие операции и функции можно выполнять над данными в SQL?

В SQL можно выполнять различные операции и функции над данными, включая:

1. Выборка данных с использованием оператора SELECT.
2. Вставка новых данных с помощью оператора INSERT.
3. Изменение существующих данных с помощью оператора UPDATE.
4. Удаление данных с использованием оператора DELETE.
5. Объединение данных из разных таблиц с помощью операторов JOIN.
6. Сортировка данных с использованием оператора ORDER BY.
7. Группировка данных с использованием оператора GROUP BY.
8. Фильтрация данных с использованием оператора WHERE.
9. Выборка уникальных значений с использованием оператора DISTINCT.
10. Выполнение агрегатных функций, таких как MIN, MAX, AVG, COUNT, SUM.
11. Преобразование данных с помощью функций преобразования (например, CAST, CONVERT).
12. Работа с датами и временем с использованием специальных функций (например, DATEPART, DAY, MONTH, YEAR).
13. Обработка строк с использованием функций работы со строками (например, CONCAT, SUBSTRING, REPLACE).
14. Математические операции (например, +, -, *, /).
15. Логические операции (например, AND, OR, NOT).

Это лишь некоторые из операций и функций, которые можно выполнять в SQL. Конкретный набор операций и функций может отличаться в зависимости от типа СУБД и её реализации.

9. Что такое агрегатные функции? Какие функции входят в эту группу?

Агрегатные функции в SQL - это функции, которые обрабатывают множество строк данных и возвращают единое значение на основе этих строк. Такие функции позволяют проводить вычисления по группам строк или по всему набору данных и получать суммы, средние значения, количество элементов и другие агрегированные результаты.

Некоторые из наиболее часто используемых агрегатных функций в SQL:

1. COUNT() - возвращает количество строк в результате.
2. SUM() - вычисляет сумму значений в столбце.
3. AVG() - вычисляет среднее значение числовых данных.
4. MIN() - находит минимальное значение в столбце.
5. MAX() - находит максимальное значение в столбце.
6. GROUP_CONCAT() - объединяет значения в строку с разделителем.

Эти функции обычно используются с оператором `GROUP BY`, который позволяет группировать данные по определенному столбцу или выражению. Агрегатные функции применяются для получения сводных результатов из базы данных, что делает их очень важными для анализа данных и формирования отчетов.

10. Какие функции для работы со строками в SQL вам известны

В SQL существует ряд функций для работы со строками, которые помогают обрабатывать и извлекать нужную информацию из текстовых данных. Некоторые из наиболее распространенных функций работы со строками в SQL:

1. `CONCAT(str1, str2, ...)` - объединяет строки в одну.
2. `SUBSTRING(str, start, length)` - извлекает подстроку из строки, начиная с указанной позиции и заданной длины.
3. `UPPER(str)` - преобразует строку в верхний регистр.
4. `LOWER(str)` - преобразует строку в нижний регистр.
5. `LENGTH(str)` - возвращает длину строки.
6. `TRIM(str)` - удаляет пробелы или другие символы с начала и конца строки.
7. `REPLACE(str, from\_str, to\_str)` - заменяет в строке все вхождения подстроки `from\_str` на `to\_str`.
8. `CHARINDEX(substring, str)` - возвращает позицию первого вхождения подстроки в строку.
9. `LEFT(str, length)` - возвращает указанное количество символов слева от строки.
10. `RIGHT(str, length)` - возвращает указанное количество символов справа от строки.

Это лишь небольшой список из доступных функций для работы со строками в SQL, их можно комбинировать и использовать для решения различных задач обработки данных в базах данных.