МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
МИНИСТЕРСТВО НАУКИ, ВЫСШЕГО ОБРАЗОВАНИЯ И ИННОВАЦИЙ КЫРГЫЗСКОЙ РЕСПУБЛИКИ
Межгосударственная образовательная организация высшего образования «Кыргызско-Российский Славянский университет имени первого Президента Российской Федерации Б.Н. Ельцина»

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ПО ДИСЦИПЛИНЕ «БАЗЫ ДАННЫХ»

Уровень высшего образования: БАКАЛАВРИАТ
Направление подготовки: 38.03.01 - РФ, 580100 - КР «Экономика»
Профиль: «Математические методы в экономике»
Квалификация: Бакалавр
Общая трудоемкость: 5 ЗЕТ (160 часов)
Форма обучения: Очная

Бишкек 2026

Фонд оценочных средств предназначен для контроля знаний обучающихся по направлению подготовки 38.03.01 - РФ, 580100 - КР Экономика профиль «Математические методы в экономике».

Фонд оценочных средств рассмотрен и утвержден на заседании Международной бизнес-школы протокол № _1_ от 03.09.2026 г.

Директор Международной бизнес-школы Кочербаева А.А.

Руководитель образовательной программы
Математические методы в экономике Мокроусов Н.В.

Исполнители:

Доцент Мусакулова Ж.А.

1. ПЕРЕЧЕНЬ КОМПЕТЕНЦИЙ, С УКАЗАНИЕМ ЭТАПОВ ИХ ФОРМИРОВАНИЯ В ПРОЦЕССЕ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Освоение дисциплины «Базы данных» направлено на формирование профессиональной компетенции ПК-1 в соответствии с ФГОС ВО по направлению подготовки 38.03.01 / 580100 «Экономика» (профиль «Математические методы в экономике»). В таблице 1.1 приведены планируемые результаты обучения и соответствующие оценочные средства.

Формируемые компетенции	Планируемые результаты обучения по дисциплине (по уровню)	Виды оценочных средств (Шифр блока)
ПК-1: Способен системно анализировать связи между объектами и процессами в экономике и бизнесе, выявлять проблемы, сравнивать альтернативы с учетом рисков и выбирать наиболее оптимальный вариант решения.	ЗНАТЬ: • 3.1.1 Основные понятия и модели баз данных, используемые для представления экономических процессов. • 3.1.2 Принципы проектирования реляционных БД и установления связей между таблицами. • 3.1.3 Средства обеспечения целостности и согласованности данных. • 3.1.4 Основы языка SQL для извлечения и анализа данных. • 3.1.5 Возможности MS Access и PostgreSQL для анализа экономической информации.	Блок A (Контрольные вопросы) Блок D (Тестирование)
	УМЕТЬ: • 3.2.1 Моделировать экономические процессы в виде структурированных баз данных. • 3.2.2 Выявлять связи между объектами предметной области с использованием реляционной модели. • 3.2.3 Формировать SQL-запросы (включая JOIN и агрегатные функции) для анализа данных. • 3.2.4 Сравнивать альтернативные управленческие решения на основе запросов.	Блок B (Практические и домашние задания) Блок D (Тестирование)
	ВЛАДЕТЬ: • 3.3.1 Навыками проектирования и реализации баз данных в MS Access и PostgreSQL. • 3.3.2 Инструментами анализа данных с использованием SQL-запросов, представлений и отчётов. • 3.3.3 Приёмами системного анализа экономических данных, хранимых в базах данных.	Блок C (Лабораторные работы и Исследовательский проект) Блок D (Зачет с оценкой)

2. ТЕХНОЛОГИЧЕСКАЯ КАРТА ДИСЦИПЛИНЫ «БАЗЫ ДАННЫХ»

Курс / семестр: 3 / 6 семестр. Трудоемкость: 5 ЗЕТ (160 часов). Аудиторные занятия: 56 ч (Лекции — 24 ч, Практические — 32 ч). Самостоятельная работа: 68 ч. Часы на контроль: 35,7 ч. Форма промежуточной аттестации: Зачет с оценкой (6 семестр).

Название модулей (разделов) согласно РПД	Вид контроля	Форма контроля	Мин. балл	Макс. балл	График (нед.)
Модуль 1. Объекты БД. СУБД MS Access	Текущий контроль	Лабораторная работа №1-3	5	8	1-4
	Рубежный контроль	Контрольная работа, Тест.	7	12	5
Модуль 2. Создание пользовательского интерфейса в Access	Текущий контроль	Лабораторная работа №4-6	5	8	6-8
	Рубежный контроль	Контрольная работа, Тест.	7	12	9
Модуль 3. Язык SQL и работа с PostgreSQL	Текущий контроль	Лабораторная работа №7-9	5	8	10-13
	Рубежный контроль	Контрольная работа, Тест.	7	12	14
Модуль 4-5. Возможности PostgreSQL и аналитические запросы	Текущий контроль	Лабораторная работа №10-14	4	6	15-17
	Рубежный контроль	Контрольная **работа**	3	4	18
ИТОГО ЗА ТЕКУЩИЙ И РУБЕЖНЫЙ КОНТРОЛЬ			**43**	**70**	
Промежуточная аттестация (Зачет с оценкой)	Промежуточный	Тестирование (30 вопросов)	17	30	Сессия
СЕМЕСТРОВЫЙ РЕЙТИНГ ПО ДИСЦИПЛИНЕ			**60**	**100**	

3. ТИПОВЫЕ КОНТРОЛЬНЫЕ ЗАДАНИЯ И ИНЫЕ МАТЕРИАЛЫ, НЕОБХОДИМЫЕ ДЛЯ ОЦЕНКИ ПЛАНИРУЕМЫХ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ

Данный раздел представляет собой расширенный комплексно-ориентированный блок оценочных средств, сгруппированный по четырем фундаментальным категориям: знания (Блок А), умения (Блок В), владение (Блок С) и компьютерное тестирование (Блок D).

3.1. Блок А: Фонд теоретических контрольных вопросов (Оценка компетенций уровня «ЗНАТЬ»)

Теоретические вопросы используются при проведении устного опроса, коллоквиумов, терминологических диктантов и подготовке к зачету. Вопросы охватывают 4 тематических модуля дисциплины.

Модуль 1. Теоретико-методологические основы реляционных баз данных

1. Дайте определение понятиям «данные», «информация», «база данных» и «система управления базами данных» (СУБД). В чем заключается принципиальное отличие БД от файловых систем?
2. Раскройте концепцию трехуровневой архитектуры БД (ANSI/SPARC): внешний, концептуальный и внутренний уровни. Как она обеспечивает физическую и логическую независимость данных?
3. Охарактеризуйте классические модели данных: иерархическую, сетевую, реляционную и объектно-ориентированную. Каковы их преимущества и сферы применения?
4. В чем суть инфологического (концептуального) и даталогического (логического) проектирования БД? Опишите нотации ER-диаграмм (Чена, IDEF1X, Crow's Foot).
5. Сформулируйте понятия «сущность», «атрибут», «связь», «первичный ключ», «внешний ключ» и «составной ключ». Назовите типы бинарных связей (1:1, 1:M, M:M) и способы их реализации.
6. Раскройте понятие реляционной алгебры Франка Кодда. Опишите теоретико-множественные операции (объединение, пересечение, вычитание, декартово произведение) и специальное реляционные операции (выборка, проекция, соединение, деление).
7. Что такое избыточность данных и аномалии обновления, удаления и вставки (insertion/deletion/update anomalies)? Приведите наглядные экономические примеры.
8. Сформулируйте определения 1-й, 2-й и 3-й нормальных форм (1NF, 2NF, 3NF). В чем отличие 3NF от нормальной формы Бойса-Кодда (BCNF)?
9. Охарактеризуйте требования к 4-й и 5-й нормальным формам (4NF, 5NF). В каких ситуациях на практике оправдана денормализация баз данных?
10. Поясните принципы обеспечения целостности данных в реляционных БД: сущностная целостность (Entity Integrity), ссылочная целостность (Referential Integrity) и доменная целостность (Domain Integrity).

Модуль 2. Архитектура и функциональные возможности настольной СУБД MS Access

11. Опишите особенности архитектуры СУБД MS Access. Для какого класса экономических задач целесообразно использование файл-серверных СУБД?

12. Перечислите основные типы данных в MS Access (Короткий текст, Длинный текст, Числовой, Денежный, Дата/время, Счетчик, Логический, OLE, Гиперссылка, Вложение). Каковы принципы выбора типов данных?

13. Как в MS Access задаются маски ввода (Input Masks), условия на значение (Validation Rules) и текст ошибки (Validation Text)?

14. Раскройте механизм схемы данных (Data Model Window) в MS Access: настройка связей, параметры обеспечения целостности данных, каскадное обновление и каскадное удаление записей.

15. Охарактеризуйте виды запросов в MS Access: запросы на выборку, параметрические запросы, перекрестные запросы (Crosstab), вычисляемые поля и итоговые запросы.

16. Опишите функциональное назначение и методы конструирования форм в MS Access. Как создаются главные и подчинённые формы (Master-Detail)?

17. Каково назначение отчетов в MS Access? Опишите структуру отчета (Заголовок отчета, Верхний/Нижний колонтитулы, Заголовок группы, Область данных, Примечание отчета).

18. Каковы принципы работы с макросами и модулями VBA в MS Access для автоматизации пользовательского интерфейса?

Модуль 3. Язык реляционных запросов SQL и клиент-серверная СУБД PostgreSQL

19. Опишите архитектуру клиент-серверной СУБД PostgreSQL. В чем заключаются ее преимущества перед файл-серверными СУБД при обработке больших массивов экономических данных?

20. Дайте классификацию команд языка SQL: DDL (Data Definition Language), DML (Data Manipulation Language), DQL (Data Query Language) и DCL (Data Control Language).

21. Детально опишите синтаксис команды CREATE TABLE в PostgreSQL: определение типов данных (VARCHAR, NUMERIC, TIMESTAMP, JSONB), ограничений (PRIMARY KEY, FOREIGN KEY, CHECK, NOT NULL, UNIQUE).

22. Объясните назначение и синтаксис предложение WHERE в операторе SELECT. Каковы правила использования логических операторов (AND, OR, NOT), операторов BETWEEN, IN, LIKE/ILIKE, IS NULL?

23. В чем заключается разница между агрегатными функциями (SUM, AVG, COUNT, MIN, MAX) с группировкой GROUP BY и фильтрацией в предложениях WHERE и HAVING?

24. Подробно опишите механизмы соединения таблиц в SQL: INNER JOIN, LEFT OUTER JOIN, RIGHT OUTER JOIN, FULL OUTER JOIN, CROSS JOIN и SELF JOIN. Приведите диаграммы Эйлера-Венна для каждого типа.

25. Раскройте концепцию вложенных подзапросов (Subqueries) в SQL: скалярные, однострочные, многострочные (IN, ANY, ALL) и коррелированные (EXISTS) подзапросы.

26. Что такое обобщенные табличные выражения (CTE — Common Table Expressions, конструкция WITH)? В чем их преимущество при написании сложных аналитических запросов?

27. Опишите назначение и синтаксис оконных функций (Window Functions) в PostgreSQL: OVER(PARTITION BY ... ORDER BY ...), функции `ROW_NUMBER()`, `DENSE_RANK()`, `LEAD()`, `LAG()`.

28. Сформулируйте понятия транзакции и требования ACID (Atomicity, Consistency, Isolation, Durability). Опишите команды управления транзакциями (BEGIN, COMMIT, ROLLBACK, SAVEPOINT).

29. Перечислите уровни изоляции транзакций по стандарту SQL (Read Uncommitted, Read Committed, Repeatable Read, Serializable) и проблемы параллельного доступа (грязное чтение, неповторяемое чтение, фантомное чтение).

30. Раскройте назначение индексов в PostgreSQL (B-Tree, Hash, GIN, GiST). Как индексы ускоряют выборку данных и каково их влияние на операции INSERT/UPDATE/DELETE?

Модуль 4-5. Проектирование, безопасность и администрирование аналитических БД

31. В чем заключаются принципиальные отличия транзакционных систем (OLTP — Online Transaction Processing) от аналитических систем и хранилищ данных (OLAP / Data Warehouse)?

32. Что такое представления (VIEWS) и материализованные представления (MATERIALIZED VIEWS) в PostgreSQL? В каких экономических задачах они применяются?

33. Опишите архитектуру процедурного языка PL/pgSQL: структура блока (DECLARE, BEGIN, END), переменные, управляющие конструкции (IF, CASE, LOOP, FOR), обработка исключений.

34. Поясните механизм триггеров в PostgreSQL: события (BEFORE, AFTER, INSTEAD OF), урони срабатывания (FOR EACH ROW, FOR EACH STATEMENT) и специальная переменная `NEW`/`OLD`.

35. Раскройте вопросы обеспечения безопасности БД: система ролей в PostgreSQL, назначение и отзыв прав доступа (GRANT / REVOKE), аудит действий пользователей и резервное копирование (pg_dump).

3.2. Блок В: Комплект практических контрольных заданий (Оценка компетенций уровня «УМЕТЬ»)

Блок В содержит практические задания различной сложности, выполняемые в рамках лабораторных работ №1–4 и рубежного контроля. Каждое задание включает предметную постановку задачи, требуемый результат и эталонные SQL-скрипты / решения.

Практическое задание №1 (Модуль 1). Нормализация отношения и устранение аномалий

Условие задачи: Задана ненормализованная таблица учета выданных коммерческих кредитов коммерческого банка со следующими атрибутами:

КредитныйДоговор (НомерДоговора, ДатаЗаключения, КодКлиента, ФИОКлиента, ИННКлиента, ТелефонКлиента, КодСотрудника, ФИОСотрудника, СуммаКредита, ПроцентнаяСтавка, КодСуммыПогашения, ДатаПлатежа, СуммаПлатежа)

Требуется:

1. Выявить функциональные зависимости между атрибутами и указать возможные аномалии обновления, вставки и удаления.

2. Выполнить последовательную нормализацию отношения, приведя схему к 1NF, 2NF и 3NF.

3. Представить итоговую даталогическую схему в виде набора связанных реляционных таблиц с указанием первичных (PK) и внешних (FK) ключей.

Эталонное решение Задания №1:

1. Таблица находится в 1NF, так как все атрибуты атомарны, но присутствуют повторы по клиентам и платежам.
2. Переход к 2NF (устранение частичной функциональной зависимости от составного ключа):
   - Таблица Клиенты (КодКлиента [PK], ФИОКлиента, ИННКлиента, ТелефонКлиента)
   - Таблица Сотрудники (КодСотрудника [PK], ФИОСотрудника)
   - Таблица КредитныеДоговоры (НомерДоговора [PK], ДатаЗаключения, КодКлиента [FK], КодСотрудника [FK], СуммаКредита, ПроцентнаяСтавка)
   - Таблица ГрафикПлатежей (КодПлатежа [PK], НомерДоговора [FK], ДатаПлатежа, СуммаПлатежа)
3. Переход к 3NF: Все транзитивные зависимости (например, зависимость ФИОКлиента от КодКлиента при наличии НомерДоговора) вынесены в отдельные сущности. Схема находится в 3NF.

Практическое задание №2 (Модуль 2). DDL и DML в PostgreSQL — Создание структуры и заполнение данными

Условие задачи: Разработать SQL-скрипт на языке PostgreSQL для создания реляционной структуры базы данных «Учет продаж торговой компании». Структура должна содержать сущности `categories`, `products` и `sales_orders` с обеспечением ссылочной целостности.

Эталонный SQL-скрипт DDL/DML:

```
CREATE TABLE categories (
    category_id SERIAL PRIMARY KEY,
    category_name VARCHAR(100) NOT NULL UNIQUE,
    description TEXT
);

CREATE TABLE products (
    product_id SERIAL PRIMARY KEY,
    product_name VARCHAR(150) NOT NULL,
    category_id INT NOT NULL REFERENCES categories(category_id) ON DELETE RESTRICT,
    price NUMERIC(12, 2) NOT NULL CHECK (price > 0),
```

```
    stock_quantity INT NOT NULL DEFAULT 0 CHECK (stock_quantity >= 0)
);

CREATE TABLE sales_orders (
    order_id SERIAL PRIMARY KEY,
    product_id INT NOT NULL REFERENCES products(product_id) ON DELETE CASCADE,
    order_date DATE NOT NULL DEFAULT CURRENT_DATE,
    quantity INT NOT NULL CHECK (quantity > 0),
    unit_price NUMERIC(12, 2) NOT NULL
);

-- Заполнение тестовыми данными
INSERT INTO categories (category_name) VALUES ('Электроника'), ('Бытовая техника');
INSERT INTO products (product_name, category_id, price, stock_quantity) VALUES
('Ноутбук Dell XPS', 1, 85000.00, 15),
('Смартфон Samsung S23', 1, 65000.00, 25),
('Кофемашина DeLonghi', 2, 42000.00, 8);

INSERT INTO sales_orders (product_id, order_date, quantity, unit_price) VALUES
(1, '2025-10-01', 2, 85000.00),
(2, '2025-10-03', 5, 65000.00),
(1, '2025-10-15', 1, 85000.00);
```

Практическое задание №3 (Модуль 3). Написание аналитических SQL-запросов высокой сложности

Условие задачи: На основе базы данных «Учет продаж» написать SQL-запросы повышенной сложности:

А) Вывести список категорий товаров с указанием суммарной выручки, количества проданных единиц и среднего чека за 4 квартал 2025 года. Отфильтровать категории с суммарной выручкой менее 100 000 сом.

Б) Сформировать аналитический запрос с использованием оконных функций: для каждой продажи вывести дату, название товара, количество, стоимость, а также накопительный итог выручки по соответствующей категории товара за все время.

Эталонный SQL-скрипт аналитических запросов:

```
-- Запрос А: Группировка, агрегирование и фильтрация HAVING
SELECT
    c.category_name,
    SUM(so.quantity * so.unit_price) AS total_revenue,
    SUM(so.quantity) AS total_units_sold,
    ROUND(AVG(so.quantity * so.unit_price), 2) AS avg_order_value
FROM categories c
JOIN products p ON c.category_id = p.category_id
```

```
JOIN sales_orders so ON p.product_id = so.product_id
WHERE so.order_date BETWEEN '2025-10-01' AND '2025-12-31'
GROUP BY c.category_id, c.category_name
HAVING SUM(so.quantity * so.unit_price) >= 100000.00
ORDER BY total_revenue DESC;

-- Запрос Б: Аналитический запрос с оконной функцией SUM() OVER
SELECT
    c.category_name,
    p.product_name,
    so.order_date,
    so.quantity,
    (so.quantity * so.unit_price) AS order_cost,
    SUM(so.quantity * so.unit_price) OVER (
        PARTITION BY c.category_id
        ORDER BY so.order_date
        ROWS BETWEEN UNBOUNDED PRECEDING AND CURRENT ROW
    ) AS running_category_revenue
FROM categories c
JOIN products p ON c.category_id = p.category_id
JOIN sales_orders so ON p.product_id = so.product_id
ORDER BY c.category_name, so.order_date;
```

Практическое задание №4-5 (Модуль 4-5). Разработка процедурной бизнес-логики и триггеров в PostgreSQL

Условие задачи: Создать хранимую функцию на языке PL/pgSQL и триггер для автоматического контроля товарных остатков при оформлении новой продажи. Если заказываемое количество превышает остаток на складе (`stock_quantity`), система должна генерировать исключение (RAISE EXCEPTION). В противном случае остаток должен автоматически уменьшаться на количество проданного товара.

Эталонный код PL/pgSQL триггера:

```
CREATE OR REPLACE FUNCTION process_sales_order_trigger()
RETURNS TRIGGER AS $$
DECLARE
    v_current_stock INT;
BEGIN
    -- Получаем текущий остаток товара на складе
    SELECT stock_quantity INTO v_current_stock
    FROM products
    WHERE product_id = NEW.product_id
    FOR UPDATE; -- Блокировка строки для предотвращения состояния гонки (Race Condition)
```

```
    -- Проверка достаточности товара
    IF v_current_stock < NEW.quantity THEN
        RAISE EXCEPTION 'Ошибка продажи: недостаточно товара на складе! Запрошено: %, В
наличии: %',
            NEW.quantity, v_current_stock;
    END IF;

    -- Уменьшаем складской остаток
    UPDATE products
    SET stock_quantity = stock_quantity - NEW.quantity
    WHERE product_id = NEW.product_id;

    RETURN NEW;
END;
$$ LANGUAGE plpgsql;

CREATE TRIGGER trg_check_and_update_stock
BEFORE INSERT ON sales_orders
FOR EACH ROW
EXECUTE FUNCTION process_sales_order_trigger();
```

3.3. Блок С: Комплект сквозных ситуационных задач и мини-проектов (Оценка компетенций уровня «ВЛАДЕТЬ»)

Блок С представляет собой совокупность комплексных практико-ориентированных проектов (кейсов), защищаемых студентами на 13–16 неделях семестра. Каждому студенту высылается индивидуальное техническое задание.

Кейс-проект №1: «Разработка аналитической базы данных коммерческого банка»

Студент должен спроектировать инфологическую схему (ERD) системы банковского кредитования (сущности: Заемщики, Кредитные продукты, Договоры, График платежей, Залоги). В СУБД PostgreSQL реализовать схемы DDL, заполнить таблицы синтетическими данными (не менее 500 записей), написать 5 сложных SQL-запросов для расчета просроченной задолженности и построить триггер аудита измененных процентных ставок.

Кейс-проект №2: «Оптимизация производительности и индексация базы данных интернет-ритейлера»

На основе базы данных с миллионом записей заказов провести профилирование запросов с помощью оператора `EXPLAIN ANALYZE`. Разработать оптимальную стратегию индексации (B-Tree, Partial Indexes, Composite Indexes) для ликвидации операций `Seq Scan` (последовательное сканирование) в пользу `Index Scan`, добившись ускорения выполнения SQL-запросов не менее чем в 10 раз.

Кейс-проект №3: «Проектирование базы данных автоматизированного кадрового учета (MS Access)»

Спроектировать и полностью имплементировать в СУБД MS Access реляционную систему для отдела кадров предприятия. Создать схемы связей с каскадным удалением, построить эргономичные формы с кнопочным управлением на VBA, сформировать аналитические групповые отчеты с вычислением фонда оплаты труда по филиалам.

Критерии оценивания выполнения и защиты мини-проектов (Блок С):

Оценка / Баллы	Содержательная характеристика выполнения проекта	Процент выполнения
Отлично / 18–20 баллов	Проект выполнен в полном объеме, БД находится в 3NF, SQL-код структурирован и оптимизирован, реализована процедурная бизнес-логика. На защите студент демонстрирует глубокие знания.	90–100%
Хорошо / 14–17 баллов	Проект выполнен без существенных недочетов, но присутствуют мелкие погрешности в индексации или оформлении ER-схем. Студент уверенно отвечает на большинство вопросов.	75–89%
Удовлетворительно / 10–13 баллов	БД работает, но нормализация не доведена до 3NF (присутствует избыточность), SQL-запросы содержат синтаксические оптимизационные изъяны. Ответы на защите поверхностные.	60–74%
Неудовлетворительно / 0–9 баллов	Проект не соответствует техническому заданию, отсутствуют внешние ключи, SQL-скрипты выдают ошибки исполнения. Студент не владеет терминологией.	0–59%

3.4. Блок D: Компьютерное тестирование и банк оценочных средств промежуточной аттестации (Зачет с оценкой)

Блок D включает банк тестовых заданий для рубежного контроля и зачета с оценкой. Тест содержит вопросы 4-х типов: с выбором одного верного ответа, с множественным выбором, на установление соответствия и на определение верной последовательности / SQL-ввод.

Банк тестовых заданий (25 репрезентативных вопросов с ответами):

1. Какое из следующих утверждений наилучшим образом описывает сущностную целостность (Entity Integrity) в реляционной модели?

а) Внешний ключ не может содержать значение NULL.

б) Первичный ключ должен быть уникальным и не может принимать значение NULL.

в) Все атрибуты таблицы должны иметь тип VARCHAR.

г) Таблица должна состоять минимум из трех столбцов.

Ответ: б) Первичный ключ должен быть уникальным и не может принимать значение NULL.

2. Укажите оператор языка SQL, относящийся к группе DDL (Data Definition Language):
а) SELECT
б) UPDATE
в) ALTER TABLE
г) GRANT

Ответ: в) ALTER TABLE

3. Отношение находится во Второй нормальной форме (2NF), если оно находится в 1NF и:
а) Отсутствуют транзитивные зависимости.
б) Каждый неключевой атрибут функционально полно зависит от всего первичного ключа.
в) Все поля содержат массив значений.
г) Таблица имеет физический индекс.

Ответ: б) Каждый неключевой атрибут функционально полно зависит от всего первичного ключа.

4. Какая агрегатная функция SQL подсчитывает количество строк в таблице, включая строки со значениями NULL?
а) COUNT(DISTINCT column)
б) SUM(*)
в) COUNT(*)
г) TOTAL()

Ответ: в) COUNT(*)

5. Какое ключевое слово используется в операторе SELECT для устранения дубликатов из результирующего набора строк?
а) UNIQUE
б) DISTINCT
в) GROUP BY
г) NO REPEAT

Ответ: б) DISTINCT

6. Какой тип соединения (JOIN) возвращает все строки из левой таблицы и совпадающие строки из правой таблицы, заполняя отсутствующие значения NULL?
а) INNER JOIN
б) LEFT OUTER JOIN
в) RIGHT OUTER JOIN
г) CROSS JOIN

Ответ: б) LEFT OUTER JOIN

7. Что произойдет при попытке удалить строку из родительской таблицы, если внешнее ограничение определено как `ON DELETE RESTRICT` и существуют связанные строки в дочерней таблице?
а) Автоматически удалятся все дочерние строки.
б) СУБД сгенерирует ошибку нарушении ссылочной целостности и заблокирует удаление.

в) Дочерние строки получат значение NULL.
г) Родительская строка удалится, а дочерние останутся орфанными.

Ответ: б) СУБД сгенерирует ошибку нарушении ссылочной целостности и заблокирует удаление.

8. Какое предложение в SQL применяется для фильтрации результатов группировки, сформированных оператором GROUP BY?
а) WHERE
б) HAVING
в) ORDER BY
г) LIKE

Ответ: б) HAVING

9. Что гарантирует свойство Атомарности (Atomicity) в концепции ACID?
а) Транзакция выполняется либо полностью, либо не выполняется вообще.
б) Данные сохраняются на диск при сбое питания.
в) Параллельные транзакции не изолированы друг от друга.
г) Результат транзакции доступен всем пользователям до COMMIT.

Ответ: а) Транзакция выполняется либо полностью, либо не выполняется вообще.

10. Какой индекс в СУБД PostgreSQL является индексом по умолчанию при создании строкового первичного ключа?
а) Hash
б) B-Tree
в) GIN
г) BRIN

Ответ: б) B-Tree

11. Установите соответствие между понятием и его определением:
1) СУБД
2) Первичный ключ
3) Денормализация
А) Намеренный ввод избыточности в БД для повышения скорости чтения.
Б) Программный комплекс для создания, ведения и совместного использования БД.
В) Минимальный набор атрибутов, однозначно идентифицирующий кортеж.

Ответ: 1-Б, 2-В, 3-А

12. Выберите ВСЕ операторы, относящиеся к манипулированию данными (DML):
а) INSERT
б) DROP TABLE
в) UPDATE
г) DELETE
д) CREATE INDEX

Ответ: а, в, г

13. Какой оператор SQL используется для изменения существующих записей в таблице?
а) CHANGE
б) MODIFY
в) UPDATE
г) ALTER

Ответ: в) UPDATE

14. Для чего используется конструкция WITH (CTE) в PostgreSQL?
а) Для создания физических таблиц на диске.
б) Для определения временных именованных результирующих наборов в рамках одного запроса.
в) Для шифрования сетевого трафика СУБД.
г) Для удаленного подключения к серверу.

Ответ: б) Для определения временных именованных результирующих наборов в рамках одного запроса.

15. В чем разница между типом данных CHAR(10) и VARCHAR(10)?
а) CHAR(10) хранит строки переменной длины, VARCHAR(10) — фиксированной.
б) CHAR(10) дополняет строку пробелами до 10 символов, VARCHAR(10) хранит фактическую длину строки.
в) VARCHAR(10) не поддерживает символы кириллицы.
г) Они абсолютно идентичны.

Ответ: б) CHAR(10) дополняет строку пробелами до 10 символов, VARCHAR(10) хранит фактическую длину строки.

16. Какая процедурная переменная в триггере PL/pgSQL типа AFTER UPDATE содержит строки таблицы ПОСЛЕ выполнения изменения?
а) OLD
б) CURRENT
в) NEW
г) NEXT

Ответ: в) NEW

17. Укажите правильный порядок выполнения секций SQL запроса СУБД (логическая последовательность обработки):
а) SELECT -> FROM -> WHERE -> GROUP BY -> HAVING -> ORDER BY
б) FROM -> WHERE -> GROUP BY -> HAVING -> SELECT -> ORDER BY
в) WHERE -> FROM -> SELECT -> GROUP BY -> HAVING -> ORDER BY
г) SELECT -> WHERE -> FROM -> GROUP BY -> ORDER BY

Ответ: б) FROM -> WHERE -> GROUP BY -> HAVING -> SELECT -> ORDER BY

18. Какой оператор SQL используется для объединения результатов двух SELECT-запросов с исключением дубликатов?
а) JOIN
б) UNION

в) UNION ALL
г) MERGE

Ответ: б) UNION

19. Ограничение CHECK (age >= 18) относится к следующему уровню целостности данных:
а) Сущностная целостность
б) Ссылочная целостность
в) Доменная целостность
г) Пользовательский уровень безопасности

Ответ: в) Доменная целостность

20. Какой тип данных в PostgreSQL предназначен для эффективного хранения и индексации полуструктурированных JSON документов?
а) TEXT
б) BLOB
в) JSONB
г) XML

Ответ: в) JSONB

21. Какой уровень изоляции транзакций предотвращает «фантомное чтение» в стандарте SQL?
а) Read Committed
б) Repeatable Read
в) Serializable
г) Read Uncommitted

Ответ: в) Serializable

22. Какая команда SQL отозвать ранее предоставленные права доступа у пользователя `economist_user`?
а) DENY SELECT ON orders FROM economist_user;
б) REVOKE SELECT ON orders FROM economist_user;
в) REMOVE PERMISSION FROM economist_user;
г) DROP USER economist_user;

Ответ: б) REVOKE SELECT ON orders FROM economist_user;

23. Какая оконная функция присваивает уникальный последовательный номер каждой строке внутри окна без пропусков при совпадении значений?
а) RANK()
б) DENSE_RANK()
в) ROW_NUMBER()
г) NTILE()

Ответ: в) ROW_NUMBER()

24. В СУБД MS Access объект, предназначенный для наглядного вывода сгруппированных данных на печать или в PDF, называется:
а) Форма
б) Таблица

в) Запрос
г) Отчет

Ответ: г) Отчет

25. Укажите утилиту командной строки PostgreSQL, применяемую для создания дампа (резервной копии) базы данных:
а) pg_ctl
б) psql
в) pg_dump
г) pg_restore

Ответ: в) pg_dump

4. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНИВАНИЯ ЗНАНИЙ, УМЕНИЙ И НАВЫКОВ

Оценивание результатов обучения по дисциплине «Базы данных» производится в соответствии с балльно-рейтинговой системой КРСУ. Оценка формируется на основе критериев, представленных в таблице 4.1.

Низкий (0-30)	**Фрагментарный (31-59)**	**Поверхностный (60-69)**	**Достаточный (70-84)**	**Высокий (85-100)**
Неудовлетворительно (0-30 баллов) Знания отсутствуют, задания не выполнены, грубые ошибки.	Неудовлетворительно (31-59 баллов) Фрагментарные знания, ошибки в SQL-запросах и проектировании.	Удовлетворительно (60-69 баллов) Базовые знания, простые запросы выполняет верно, есть неточности.	Хорошо (70-84 балла) Уверенное владение MS Access и SQL, верное проектирование таблиц и связей.	Отлично (85-100 баллов) Глубокое понимание теории, сложные аналитические SQL-запросы в PostgreSQL.

5. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

Самостоятельная работа студентов по дисциплине «Базы данных» включает изучение теоретического материала по лекциям и учебникам, выполнение лабораторных работ в MS Access и PostgreSQL, подготовку к текущему и рубежному контролю, а также работу над мини-проектом.

Рекомендации по выполнению практических заданий и мини-проектов:
1. При работе в MS Access обращать внимание на корректность задания типов данных,

первичных ключей и параметров связи в окне «Схема данных».
2. При написании SQL-запросов в PostgreSQL использовать среду pgAdmin, отлаживать запросы поэтапно: сначала предложение FROM и JOIN, затем WHERE, GROUP BY и HAVING.
3. При подготовке отчетов по лабораторным работам и мини-проекту предоставлять исходные SQL-скрипты и скриншоты результатов с пояснительной запиской.