Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Гриб Владислав Валерьевич
Должность: Ректор
Дата подписания: 14.10.2026 17:08:08
Уникальный программный ключ:
637517d24e103c3db032acf37e839d98ec1c5bb2f5eb89c29abfcd7f43985447

Образовательное частное учреждение высшего образования
«МОСКОВСКИЙ УНИВЕРСИТЕТ ИМЕНИ А. С. ГРИБОЕДОВА»
(ИМПЭ им. А.С. Грибоедова)

МЕЖДУНАРОДНЫЙ ИНСТИТУТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И БИЗНЕС-ИНФОРМАТИКИ

УТВЕРЖДАЮ
Директор Международного института информационных технологий и бизнес-информатики
_____ Т.В. Новикова
«17» декабря 2025г.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

по учебной дисциплине:
БАЗЫ ДАННЫХ

направление подготовки
38.03.05
БИЗНЕС-ИНФОРМАТИКА
(уровень бакалавриат)

направления/профиль:
«Цифровая трансформация бизнеса»

Форма обучения: очная, очно-заочная

Москва

Фонд оценочных средств для дисциплины «Базы данных». Направление подготовки/специальность 38.03.05 Бизнес-информатика, направленность (профиль): «Цифровая трансформация бизнеса»/ О.Ю. Евдокимова – М.: ИМПЭ им. А.С. Грибоедова.

Фонд оценочных средств является неотъемлемой частью рабочей программы дисциплины.

Разработчик: Т.В. Новикова

Заведующий кафедрой /Т.В. Новикова, к. э. н., доцент

1. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИ, ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ

Настоящий Фонд оценочных средств (ФОС) является неотъемлемым приложением к рабочей программе дисциплины (РПД) «Базы данных». На данный ФОС распространяются все реквизиты утверждения, представленные в РПД по данной дисциплине.

2. ПЕРЕЧЕНЬ ОЦЕНОЧНЫХ СРЕДСТВ

Для определения качества освоения обучающимися учебного материала по дисциплине используются следующие оценочные средства:

№ п/п	Оценочное средство	Краткая характеристика оценочного средства	Представление оценочного средства в ФОС
1	Тестирование	Вид контроля, позволяющий оценить изученный теоретический материал.	Вопросы для проведения тестирования
2	Практические задания	Вид контроля, позволяющий оценить умение обучающегося применять осваиваемую компетенцию в практических ситуациях и при решении производственных задач	Задания к практическому занятию
3	Контрольная работа	Вид контроля, позволяющий определить результат освоения компетенций по дисциплине в рамках рассматриваемой темы, оцениваемый с помощью соответствующих индикаторов достижения компетенций	Задания контрольной работы
4	Самостоятельная работа	Вид контроля, позволяющий оценить проработку теоретического материала, изучение рекомендуемой литературы, выполнение практико-ориентированных заданий (заполнение таблиц, проведение сравнительного анализа, составление схем и др.), решение практических задач, создание презентаций, написание рефератов, подборку нормативного и иного материала и выполнение других заданий	Задания самостоятельной работы
5	Курсовая работа	Вид контроля, позволяющий выявить степень владения базовыми знаниями, умениями и навыками, необходимыми для обучения, и определить уровень владения новым материалом	Индивидуальные задания (темы) для курсовой работы
6	Зачет/Зачет с оценкой/Экзамен	Вид контроля, позволяющий выявить степень овладения знаниями, умениями и навыками, необходимыми для дальнейшего освоения образовательной программы подготовки	Вопросы для подготовки к зачету/зачету с оценкой/экзамену

3. ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ ДИСЦИПЛИНЕ

3.1. Сопроводительная информация.

Разработчик	Т.В. Новикова
Кафедра	Бизнес-информатика технологий управления
Наименование дисциплины	Базы данных
Факультет / институт	Международный институт информационных технологий и бизнес-информатики
Направление подготовки / специальность	38.03.05 Бизнес-информатика
Количество вопросов в оценочных заданиях (диапазон)	От 22 до 44
Общее время тестирования(мин)	90
Общее количество вопросов/заданий в ФОС	44
Размещенность на сайте Университета примерного перечня вопросов, заданий ФОС – для подготовки обучающихся к прохождению оценки (да / нет)	Да

3.2. Характеристика оцениваемых компетенций.

Код компетенции	Формулировка компетенции	Индикаторы достижения компетенции
ПК-15.	Способен информировать заказчика на этапе предконтрактных работ о возможностях типовой ИС и вариантах ее модификации	**ИПК-15.1 Знать**: Сетевые протоколы Основы современных операционных систем Основы современных СУБД Устройство и функционирование современных ИС Архитектура мультиарендного программного обеспечения Основы ИБ организации Современные стандарты информационного взаимодействия систем Программные средства и платформы инфраструктуры информационных технологий организаций Современные подходы и стандарты автоматизации организации Основы теории систем и системного анализа **ИПК-15.2 Уметь**: Проводить презентации заинтересованным сторонам в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ИС **ИПК-15.3 Владеть**: Навыками информирования заказчика на этапе предконтрактных работ о возможностях типовой ИС и вариантах ее модификации

4. СОДЕРЖАНИЕ ОЦЕНОЧНЫХ СРЕДСТВ ТЕКУЩЕГО КОНТРОЛЯ

4.1. ТИПОВЫЕ ТЕСТОВЫЕ ЗАДАНИЯ.

Тесты содержат набор вопросов, в полном объеме охватывающие изученный теоретический материал по указанной теме. Выполнение тестов позволяет определить результат освоения компетенций по дисциплине в рамках рассматриваемой темы, оцениваемый с помощью соответствующих индикаторов достижения компетенций. Индивидуальный тестовый сеанс для каждого обучающегося формируется по специальному алгоритму, обеспечивающему заданную тематическую структуру и пропорциональное наличие вопросов разного типа и сложности.

При формировании тестов необходимо использовать задания следующих типов:

Тип задания 1. Задания закрытого типа на установление соответствия.

Тип задания 2. Задания закрытого типа на установление последовательности.

Тип задания 3. Задания комбинированного типа, предполагающие выбор одного правильного ответа из предложенных

Тип задания 4. Задания комбинированного типа, предполагающие выбор нескольких ответов из предложенных

Тип задания 5. Задания открытого типа с развернутым ответом.

<table>
<tr><th>Тема</th><th>Код и формулировка компетенции</th><th>Индикаторы достижения компетенции</th><th>Тест по теме</th></tr>
<tr><td>Тема 1. Основные понятия области баз данных и информационных систем.</td><td>ПК.15</td><td>ИПК.15.1</td><td>
Задания закрытого типа на установление соответствия
Задание 1. Установите соответствие между уровнями архитектуры ANSI/SPARC и их описанием.
<table>
<tr><td>Уровень архитектуры</td><td>Описание</td></tr>
<tr><td>1. Внешний уровень</td><td>A. Описывает физическое хранение данных на носителе</td></tr>
<tr><td>2. Концептуальный уровень</td><td>B. Определяет логическую структуру всей базы данных</td></tr>
<tr><td>3. Внутренний уровень</td><td>C. Представляет данные с точки зрения конкретного пользователя или приложения</td></tr>
</table>
Правильный ответ:
1 — C, 2 — B, 3 — A

Задание 2. Установите соответствие между термином и его определением.
<table>
<tr><td>Термин</td><td>Определение</td></tr>
<tr><td>1. СУБД</td><td>A. Совокупность данных, организованных по определённым правилам</td></tr>
<tr><td>2. База данных</td><td>B. Программное обеспечение для управления базами данных</td></tr>
<tr><td>3. Информационная система</td><td>C. Комплекс средств, обеспечивающий сбор, обработку, хранение и выдачу информации</td></tr>
</table>
Правильный ответ:
1 — B, 2 — A, 3 — C

Задания закрытого типа на установление последовательности
Задание 3. Укажите правильную последовательность этапов проектирования базы данных.
</td></tr>
</table>

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
			A. Физическое проектирование B. Концептуальное проектирование C. Логическое проектирование D. Анализ требований Правильный ответ: D → B → C → A Задание 4. Установите правильную последовательность уровней абстракции в архитектуре базы данных (от наиболее абстрактного к наименее). A. Внутренний уровень B. Внешний уровень C. Концептуальный уровень Правильный ответ: B → C → A **Задания комбинированного типа (выбор одного ответа)** Задание 5. Какой из перечисленных компонентов НЕ является частью архитектуры клиент-серверной СУБД? A. Клиентское приложение B. Сервер базы данных C. Файловая система операционной системы D. Сетевой протокол передачи данных Правильный ответ: C Объяснение: Файловая система ОС используется сервером базы данных как средство хранения, но не является самостоятельным компонентом архитектуры клиент-серверной СУБД. Основные компоненты — клиент, сервер и сетевое взаимодействие между ними. Задание 6. Что из перечисленного лучше всего описывает концептуальную модель базы данных? A. Схема таблиц с указанием типов данных

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
			B. ER-диаграмма C. Физический файл данных D. SQL-скрипт создания таблиц Правильный ответ: B Объяснение: Концептуальная модель отражает сущности предметной области и связи между ними без привязки к конкретной СУБД. ER-диаграмма (Entity-Relationship) — стандартный инструмент представления такой модели. **Задания комбинированного типа (выбор нескольких ответов)** Задание 7. Какие из перечисленных утверждений верны для реляционной модели данных? A. Данные представлены в виде таблиц B. Каждая строка таблицы должна быть уникальной C. Порядок строк в таблице имеет значение D. Атрибуты могут иметь составные значения Правильные ответы: A, B Объяснение: В реляционной модели данные организованы в виде отношений (таблиц), где каждая строка (кортеж) уникальна (благодаря наличию первичного ключа), порядок строк не важен, а атрибуты должны быть атомарными (неделимыми) — это требование первой нормальной формы. Задание 8. Какие функции выполняет СУБД? A. Обеспечение целостности данных B. Хранение исходного кода приложений C. Управление транзакциями D. Поддержка многопользовательского доступа

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
			Правильные ответы: A, C, D Объяснение: СУБД управляет данными: обеспечивает их целостность, безопасность, согласованность (через транзакции) и одновременный доступ множества пользователей. Хранение исходного кода приложений — задача систем контроля версий, а не СУБД. **Задания открытого типа с развёрнутым ответом** Задание 9. Почему при проектировании информационных систем важно разделять логический и физический уровни представления данных. Приведите примеры преимуществ такого подхода. Ожидаемый ответ должен включать: • Разделение логики работы с данными и способа их хранения. • Независимость приложений от изменений в физической структуре (например, смена дисковой системы или индексов). • Упрощение модификации и масштабирования ИС. • Пример: изменение типа индекса не требует переписывания бизнес-логики. Задание 10. Опишите роль концептуального моделирования в жизненном цикле разработки базы данных. Какие ошибки могут возникнуть при его отсутствии? Ожидаемый ответ должен включать: • Концептуальное моделирование — этап анализа предметной области и выявления сущностей, атрибутов и связей. • Позволяет согласовать требования заказчика и разработчиков до реализации. • Без него возможны: дублирование данных, нарушение целостности, несоответствие бизнес-процессам, высокая стоимость последующих изменений. • Пример: отсутствие связи «клиент–заказ» приведёт к невозможности отслеживать историю покупок.
Тема 2. Проектирование	ПК.15	ИПК.15.1	**Задания закрытого типа на установление соответствия** Задание 1. Установите соответствие между этапом проектирования базы данных и

<table>
<tr><th>Тема</th><th>Код и формулировка компетенции</th><th>Индикаторы достижения компетенции</th><th>Тест по теме</th></tr>
<tr><td>баз данных. Элементы математической теории баз данных.</td><td></td><td></td><td>его целью.
<table>
<tr><td>Этап проектирования</td><td>Цель</td></tr>
<tr><td>1. Концептуальное проектирование</td><td>A. Определение физических структур хранения и индексов</td></tr>
<tr><td>2. Логическое проектирование</td><td>B. Построение модели предметной области без привязки к СУБД</td></tr>
<tr><td>3. Физическое проектирование</td><td>C. Преобразование концептуальной модели в реляционную схему</td></tr>
</table>
Правильный ответ:
1 — B, 2 — C, 3 — A

Задание 2. Установите соответствие между понятием из реляционной теории и его определением.
<table>
<tr><td>Понятие</td><td>Определение</td></tr>
<tr><td>1. Отношение</td><td>A. Подмножество кортежей, удовлетворяющих заданному условию</td></tr>
<tr><td>2. Кортеж</td><td>B. Множество атрибутов, однозначно идентифицирующих кортеж</td></tr>
<tr><td>3. Первичный ключ</td><td>C. Одна строка в таблице реляционной базы данных</td></tr>
<tr><td>4. Выборка (селекция)</td><td>D. Таблица, соответствующая математическому понятию отношения</td></tr>
</table>
Правильный ответ:
1 — D, 2 — C, 3 — B, 4 — A

Задания закрытого типа на установление последовательности
Задание 3. Укажите правильную последовательность этапов нормализации отношений при</td></tr>
</table>

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
			логическом проектировании БД. A. Приведение к третьей нормальной форме (3НФ) B. Приведение ко второй нормальной форме (2НФ) C. Приведение к первой нормальной форме (1НФ) D. Приведение к нормальной форме Бойса–Кодда (НФБК) Правильный ответ: C → B → A → D Задание 4. Установите правильную последовательность действий при построении ER-диаграммы в рамках концептуального проектирования. A. Определение связей между сущностями B. Выявление сущностей предметной области C. Уточнение кардинальности связей D. Определение атрибутов сущностей Правильный ответ: B → D → A → C **Задания комбинированного типа (выбор одного ответа)** Задание 5. Какое из следующих утверждений верно описывает первую нормальную форму (1НФ)? A. Все неключевые атрибуты зависят только от всего первичного ключа B. В отношении отсутствуют повторяющиеся группы и все атрибуты атомарны C. Отсутствуют транзитивные зависимости неключевых атрибутов D. Каждый неключевой атрибут зависит от части составного ключа Правильный ответ: B Объяснение: Первая нормальная форма требует, чтобы все атрибуты были атомарными (неделимыми) и не содержали повторяющихся групп или списков. Это базовое требование реляционной модели. Задание 6. Что является основной целью нормализации при проектировании реляционных баз данных?

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
			A. Ускорение выполнения SQL-запросов B. Уменьшение объема хранимых данных C. Устранение избыточности и аномалий обновления D. Упрощение создания пользовательского интерфейса Правильный ответ: C Объяснение: Нормализация направлена на устранение избыточности данных и предотвращение аномалий вставки, удаления и обновления, что повышает целостность и согласованность данных. **Задания комбинированного типа (выбор нескольких ответов)** Задание 7. Какие из перечисленных условий являются признаками нарушения второй нормальной формы (2НФ)? A. Наличие составного первичного ключа B. Неключевой атрибут функционально зависит только от части составного ключа C. Все атрибуты атомарны D. Неключевой атрибут зависит от другого неключевого атрибута Правильные ответы: A, B Объяснение: 2НФ применима только к отношениям с составным первичным ключом. Нарушение 2НФ возникает, когда неключевой атрибут зависит не от всего ключа, а от его части. Условие D относится к третьей нормальной форме. Задание 8. Какие из следующих утверждений верны для реляционной алгебры? A. Операция «проекция» уменьшает количество строк в отношении B. Операция «селекция» фильтрует строки по условию C. Операция «соединение» объединяет два отношения по общему атрибуту D. Результат любой операции реляционной алгебры — также отношение Правильные ответы: B, C, D

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
			Объяснение: Проекция уменьшает количество атрибутов (столбцов), а не строк. Селекция — фильтрация строк. Соединение действительно объединяет отношения. Реляционная алгебра замкнута: результат любой операции — отношение. **Задания открытого типа с развёрнутым ответом** Задание 9. Как элементы математической теории реляционных баз данных (например, функциональные зависимости и нормальные формы) способствуют созданию качественного программного обеспечения. Приведите пример, иллюстрирующий последствия игнорирования этих принципов. Ожидаемые элементы ответа: • Функциональные зависимости позволяют выявить избыточность и спроектировать структуру БД без дублирования. • Нормальные формы обеспечивают согласованность и целостность данных. • Пример: таблица Сотрудники(ТабНомер, ФИО, Отдел, РуководительОтдела) — при изменении руководителя отдела нужно обновлять множество строк → аномалия обновления. • Это усложняет поддержку ПО, увеличивает риск ошибок и снижает надёжность системы. Задание 10. Опишите роль концептуального и логического проектирования в обеспечении соответствия базы данных требованиям будущего программного обеспечения. Как математическая теория помогает на этих этапах? Ожидаемые элементы ответа: • Концептуальное проектирование фиксирует бизнес-правила и структуру данных независимо от технологии. • Логическое проектирование преобразует модель в реляционную схему с учётом теоретических ограничений (нормализация, функциональные зависимости). • Математическая теория (реляционная модель, алгебра, нормализация) даёт формальный аппарат для проверки корректности схемы.

<table>
<tr><th>Тема</th><th>Код и формулировка компетенции</th><th>Индикаторы достижения компетенции</th><th>Тест по теме</th></tr>
<tr><td></td><td></td><td></td><td>• Это позволяет избежать проектных ошибок на ранних этапах, что критично для надёжности и масштабируемости ПО.</td></tr>
<tr><td>Тема 3. СУБД MySQL.</td><td>ПК.15</td><td>ИПК.15.1</td><td>Задания закрытого типа на установление соответствия
Задание 1. Установите соответствие между типом данных MySQL и его назначением.
<table>
<tr><td>Тип данных MySQL</td><td>Назначение</td></tr>
<tr><td>1. INT</td><td>A. Хранение даты и времени</td></tr>
<tr><td>2. VARCHAR(255)</td><td>B. Хранение целого числа</td></tr>
<tr><td>3. DATETIME</td><td>C. Хранение текстовой строки переменной длины</td></tr>
<tr><td>4. DECIMAL(10,2)</td><td>D. Хранение точного числового значения с фиксированной дробной частью</td></tr>
</table>
Правильный ответ:
1 — B, 2 — C, 3 — A, 4 — D

Задание 2. Установите соответствие между командой MySQL и её функцией.
<table>
<tr><td>Команда</td><td>Функция</td></tr>
<tr><td>1. CREATE DATABASE</td><td>A. Добавление новой записи в таблицу</td></tr>
<tr><td>2. INSERT INTO</td><td>B. Создание новой базы данных</td></tr>
<tr><td>3. ALTER TABLE</td><td>C. Изменение структуры существующей таблицы</td></tr>
<tr><td>4. DROP TABLE</td><td>D. Удаление таблицы из базы данных</td></tr>
</table>
Правильный ответ:
1 — B, 2 — A, 3 — C, 4 — D</td></tr>
</table>

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
			Задания закрытого типа на установление последовательности Задание 3. Укажите правильную последовательность шагов при создании базы данных в MySQL через командную строку. A. Подключение к серверу MySQL с помощью команды mysql -u root -p B. Создание таблицы с помощью CREATE TABLE C. Выбор базы данных командой USE имя_базы D. Создание базы данных командой CREATE DATABASE имя_базы Правильный ответ: A → D → C → B Задание 4. Установите правильную последовательность действий при настройке внешнего ключа в InnoDB. A. Убедиться, что обе таблицы используют движок InnoDB B. Создать первичный ключ в родительской таблице C. Определить столбец во внешней таблице с тем же типом данных D. Добавить ограничение внешнего ключа с помощью FOREIGN KEY ... REFERENCES Правильный ответ: A → B → C → D **Задания комбинированного типа (выбор одного ответа)** Задание 5. Какой движок хранения в MySQL по умолчанию поддерживает транзакции и внешние ключи? A. MyISAM B. MEMORY C. InnoDB D. CSV Правильный ответ: C Объяснение: InnoDB — это стандартный движок хранения в современных версиях MySQL, поддерживающий ACID-транзакции, внешние ключи и восстановление после сбоев. MyISAM, напротив, не поддерживает транзакции и внешние ключи.

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
			Задание 6. Что произойдёт, если выполнить команду DELETE FROM users; без предложения WHERE? A. Команда вызовет синтаксическую ошибку B. Будут удалены все записи из таблицы users, но сама таблица останется C. Таблица users будет полностью удалена из базы данных D. Система запросит подтверждение перед удалением Правильный ответ: B Объяснение: Команда DELETE FROM без условия WHERE удаляет все строки из указанной таблицы, но структура таблицы (схема) сохраняется. Для полного удаления таблицы используется DROP TABLE. **Задания комбинированного типа (выбор нескольких ответов)** Задание 7. Какие из перечисленных утверждений верны относительно индексов в MySQL? A. Индексы ускоряют операции SELECT B. Индексы замедляют операции INSERT, UPDATE, DELETE C. Первичный ключ автоматически создаёт кластерный индекс в InnoDB D. Индексы можно создавать только на числовых полях Правильные ответы: A, B, C Объяснение: Индексы действительно ускоряют поиск, но требуют обновления при изменении данных, что замедляет модификацию. В InnoDB первичный ключ реализуется как кластерный индекс. Индексы могут создаваться на любых типах данных, включая строки (VARCHAR, TEXT с ограничением). Задание 8. Какие из следующих возможностей предоставляет MySQL Workbench? A. Визуальное проектирование ER-диаграмм B. Генерация SQL-скриптов по модели данных C. Выполнение резервного копирования базы данных D. Автоматическая нормализация таблиц до 3НФ

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
			Правильные ответы: A, B, C Объяснение: MySQL Workbench поддерживает визуальное моделирование, forward/reverse engineering, администрирование (включая бэкапы). Однако автоматическая нормализация до 3НФ — это задача аналитика, а не инструмента; Workbench не выполняет её автоматически. **Задания открытого типа с развёрнутым ответом** Задание 9. Почему при разработке прикладного программного обеспечения важно правильно выбирать типы данных и движки хранения в MySQL. Приведите примеры последствий неправильного выбора. Ожидаемые элементы ответа: • Правильный выбор типов данных влияет на производительность, объём хранимых данных и корректность логики (например, TINYINT vs BIGINT). • Выбор движка (InnoDB vs MyISAM) определяет поддержку транзакций, блокировок, целостности. • Пример 1: использование MyISAM в системе интернет-магазина может привести к потере данных при сбое, так как нет поддержки транзакций. • Пример 2: хранение даты в VARCHAR вместо DATE нарушает целостность, усложняет запросы (WHERE, сортировку) и увеличивает объём данных. Задание 10. Опишите роль MySQL как компонента архитектуры современного программного обеспечения. Как знание особенностей MySQL помогает при проектировании надёжных и масштабируемых приложений? Ожидаемые элементы ответа: • MySQL часто используется как центральный компонент backend-архитектуры (в связке с PHP, Python, Java и др.). • Знание особенностей (движки, индексация, транзакции, репликация) позволяет проектировать масштабируемые и отказоустойчивые системы. • Например, понимание работы InnoDB помогает правильно проектировать схемы с учётом блокировок и изоляции транзакций. • Умение использовать индексы, партиционирование и оптимизацию запросов снижает

<table>
<tr><th>Тема</th><th>Код и формулировка компетенции</th><th>Индикаторы достижения компетенции</th><th>Тест по теме</th></tr>
<tr><td></td><td></td><td></td><td>нагрузку на сервер и улучшает UX.
• Проектирование ПО требует глубокого понимания используемых СУБД как средства управления данными.</td></tr>
<tr><td>Тема 4. Разработка клиентских приложений MySQL. Интерфейс MySQL C API.</td><td>ПК.15</td><td>ИПК.15.1</td><td>Задания закрытого типа на установление соответствия
Задание 1. Установите соответствие между функцией MySQL C API и её назначением.
<table>
<tr><th>Функция MySQL C API</th><th>Назначение</th></tr>
<tr><td>1. mysql_init()</td><td>A. Выполняет SQL-запрос к базе данных</td></tr>
<tr><td>2. mysql_real_connect()</td><td>B. Инициализирует структуру MYSQL</td></tr>
<tr><td>3. mysql_query()</td><td>C. Освобождает память, выделенную под результат запроса</td></tr>
<tr><td>4. mysql_free_result()</td><td>D. Устанавливает соединение с сервером MySQL</td></tr>
</table>
Правильный ответ:
1 — B, 2 — D, 3 — A, 4 — C

Задание 2. Установите соответствие между этапом работы с MySQL через C API и используемой функцией.
<table>
<tr><th>Этап</th><th>Функция</th></tr>
<tr><td>1. Инициализация соединения</td><td>A. mysql_store_result()</td></tr>
<tr><td>2. Выполнение запроса</td><td>B. mysql_init()</td></tr>
<tr><td>3. Получение результата</td><td>C. mysql_query()</td></tr>
<tr><td>4. Завершение работы</td><td>D. mysql_close()</td></tr>
</table>
Правильный ответ:
1 — B, 2 — C, 3 — A, 4 — D</td></tr>
</table>

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
			Задания закрытого типа на установление последовательности Задание 3. Укажите правильную последовательность вызовов функций MySQL C API для выполнения SELECT-запроса и обработки результата. A. mysql_query(conn, "SELECT * FROM users") B. MYSQL_RES *result = mysql_store_result(conn) C. mysql_init(NULL) и mysql_real_connect(...) D. while ((row = mysql_fetch_row(result))) { ... } E. mysql_free_result(result) Правильный ответ: C → A → B → D → E Задание 4. Установите правильную последовательность действий при разработке клиентского приложения на C для взаимодействия с MySQL. A. Компиляция программы с флагами -lmysqlclient B. Подключение заголовочного файла #include <mysql/mysql.h> C. Обработка ошибок с помощью mysql_error() D. Освобождение ресурсов и закрытие соединения Правильный ответ: B → C → D → A **Задания комбинированного типа (выбор одного ответа)** Задание 5. Какая функция MySQL C API возвращает текстовое описание последней ошибки, произошедшей при работе с соединением? A. mysql_errno() B. mysql_error() C. perror() D. mysql_info() Правильный ответ: B Объяснение: Функция mysql_error(MYSQL *mysql) возвращает строку с описанием последней ошибки,

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
			связанной с данным соединением. mysql_errno() возвращает только числовой код ошибки, а perror() — системные ошибки ОС, не специфичные для MySQL. Задание 6. Почему при разработке клиентского приложения на C с использованием MySQL C API важно вызывать mysql_library_init() в многопоточных приложениях? A. Для ускорения выполнения SQL-запросов B. Для инициализации глобальных переменных библиотеки MySQL перед созданием потоков C. Для подключения к удалённому серверу базы данных D. Для автоматического создания резервной копии базы данных Правильный ответ: B Объяснение: В многопоточных приложениях необходимо вызвать mysql_library_init() до создания потоков, чтобы корректно инициализировать внутренние глобальные структуры клиентской библиотеки MySQL. Это требование документировано в официальной документации MySQL. **Задания комбинированного типа (выбор нескольких ответов)** Задание 7. Какие из перечисленных действий являются обязательными при корректной работе с MySQL через C API? A. Проверка возвращаемых значений функций на NULL или ошибки B. Явное освобождение памяти результата с помощью mysql_free_result() C. Использование только глобальных переменных для хранения соединения D. Закрытие соединения с помощью mysql_close() Правильные ответы: A, B, D Объяснение: Проверка ошибок, освобождение ресурсов (mysql_free_result) и закрытие соединения (mysql_close) — обязательные шаги для предотвращения утечек памяти и неопределённого поведения. Использование глобальных переменных не требуется — соединение может быть локальным или передаваться как параметр. Задание 8. Какие особенности характерны для клиентских приложений, использующих

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
			MySQL C API? A. Требуют линковки с библиотекой libmysqlclient B. Могут выполняться без установки MySQL Server на том же компьютере C. Автоматически генерируют графический интерфейс пользователя D. Позволяют напрямую управлять транзакциями через SQL-команды Правильные ответы: A, B, D Объяснение: Клиентское приложение на C с MySQL C API должно быть скомпоновано с libmysqlclient. Оно может подключаться к удалённому серверу MySQL, поэтому локальная установка СУБД не обязательна. Транзакции управляются через SQL (START TRANSACTION, COMMIT, ROLLBACK). Графический интерфейс не создаётся автоматически — его нужно реализовывать отдельно. **Задания открытого типа с развёрнутым ответом** Задание 9. Как использование MySQL C API способствует созданию эффективных и надёжных клиентских приложений. Какие принципы проектирования ПО следует соблюдать при работе с этим интерфейсом? Ожидаемые элементы ответа: • MySQL C API предоставляет низкоуровневый, производительный доступ к базе данных без промежуточных слоёв. • Позволяет точно контролировать соединения, память и обработку ошибок. • Принципы проектирования: • Обязательная проверка возвращаемых значений всех функций. • Корректное управление ресурсами (освобождение памяти, закрытие соединений). • Обработка ошибок с логированием или восстановлением. • Изоляция логики работы с БД в отдельные модули/функции (принцип единственной ответственности). Задание 10. Сравните подходы к взаимодействию с MySQL через C API и через высокоуровневые языки (например, Python с библиотекой mysql-connector). В каких случаях целесообразно использовать именно C API?

<table>
<tr><th>Тема</th><th>Код и формулировка компетенции</th><th>Индикаторы достижения компетенции</th><th>Тест по теме</th></tr>
<tr><td></td><td></td><td></td><td>Ожидаемые элементы ответа:
C API: максимальная производительность, минимальное потребление памяти, полный контроль над соединением и ресурсами. Требует ручного управления памятью и ошибками.
Высокоуровневые языки: удобство, быстрая разработка, автоматическое управление памятью, встроенные ORM.
Целесообразно использовать C API, когда:
✓ Требуется высокая производительность (встроенные системы, high-load сервисы).
✓ Приложение уже написано на C/C++.
✓ Необходима минимальная зависимость от интерпретаторов или виртуальных машин.
С точки зрения проектирования ПО: выбор инструмента должен соответствовать требованиям к надёжности, производительности и сопровождаемости. C API — мощное, но требовательное средство, подходящее для системного уровня.</td></tr>
<tr><td>Тема 5. Разработка клиентских приложений MySQL. Интерфейс QMYSQL для GUI-приложений MySQL.</td><td>ПК.15</td><td>ИПК.15.1</td><td>Задания закрытого типа на установление соответствия

Задание 1. Установите соответствие между компонентом Qt и его ролью при работе с MySQL через драйвер QMYSQL.<table><tr><td>Компонент Qt</td><td>Роль</td></tr><tr><td>1. QSqlDatabase</td><td>A. Выполняет SQL-запросы и обрабатывает результаты</td></tr><tr><td>2. QSqlQuery</td><td>B. Представляет модель данных для отображения в таблице</td></tr><tr><td>3. QSqlTableModel</td><td>C. Управляет соединением с базой данных</td></tr><tr><td>4. QSqlError</td><td>D. Содержит информацию об ошибке, возникшей при выполнении запроса</td></tr></table>
Правильный ответ:
1 — C, 2 — A, 3 — B, 4 — D</td></tr>
</table>

<table>
<tr><th>Тема</th><th>Код и формулировка компетенции</th><th>Индикаторы достижения компетенции</th><th>Тест по теме</th></tr>
<tr><td></td><td></td><td></td><td>
Задание 2. Установите соответствие между этапом разработки GUI-приложения на Qt и необходимым действием для подключения к MySQL.
<table>
<tr><td>Этап</td><td>Действие</td></tr>
<tr><td>1. Настройка проекта</td><td>A. Вызов QSqlDatabase::addDatabase("QMYSQL")</td></tr>
<tr><td>2. Инициализация соединения</td><td>B. Добавление строки QT += sql в файл .pro</td></tr>
<tr><td>3. Обработка результата запроса</td><td>C. Использование QSqlQuery::next() и value()</td></tr>
<tr><td>4. Отображение данных в интерфейсе</td><td>D. Привязка QSqlTableModel к QTableView</td></tr>
</table>
Правильный ответ:
1 — B, 2 — A, 3 — C, 4 — D

Задания закрытого типа на установление последовательности

Задание 3. Укажите правильную последовательность шагов для подключения GUI-приложения на Qt к базе данных MySQL.
A. Установка параметров соединения: хост, имя БД, логин, пароль
B. Добавление модуля SQL в проект (QT += sql)
C. Проверка успешности подключения через isOpen()
D. Создание объекта соединения: QSqlDatabase::addDatabase("QMYSQL")

Правильный ответ: B → D → A → C

Задание 4. Установите правильную последовательность действий при выполнении SELECT-запроса и отображении результата в виджете QTableView.
A. Создание и настройка модели: QSqlTableModel model; model.setTable("users"); model.select();
B. Привязка модели к виджету: tableView->setModel(&model);
</td></tr>
</table>

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
			C. Установка активного соединения с MySQL D. Настройка внешнего вида таблицы (опционально) Правильный ответ: C → A → B → D **Задания комбинированного типа (выбор одного ответа)** Задание 5. Какой драйвер необходимо указать в методе addDatabase(), чтобы использовать MySQL в приложении на Qt? A. "MYSQL" B. "QMYSQL" C. "MySQL" D. "QtMySQL" Правильный ответ: B Объяснение: В Qt для подключения к MySQL используется драйвер с именем "QMYSQL". Это официальное название плагина SQL-драйвера, предоставляемого фреймворком Qt для работы с MySQL. Задание 6. Что произойдёт, если не добавить строку QT += sql в файл проекта .pro при использовании классов QSqlDatabase и QSqlQuery? A. Приложение скомпилируется, но не сможет подключиться к MySQL B. Компилятор выдаст ошибку «undefined reference» при линковке C. Приложение будет работать, но с пониженной производительностью D. Qt автоматически подключит модуль SQL при обнаружении использования базы данных Правильный ответ: B Объяснение: Модуль SQL в Qt является отдельной библиотекой. Без указания QT += sql в .pro-файле линковщик не найдёт реализации классов QSqlDatabase, QSqlQuery и др., что приведёт к ошибкам линковки (undefined reference).

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
			Задания комбинированного типа (выбор нескольких ответов) Задание 7. Какие из перечисленных условий необходимы для успешной работы GUI-приложения на Qt с MySQL через QMYSQL? A. Наличие установленного сервера MySQL или MariaDB B. Наличие скомпилированного плагина qsqlmysql.dll (Windows) или libqsqlmysql.so (Linux) C. Использование только глобальных переменных для хранения соединения D. Корректная настройка путей к библиотекам MySQL (например, libmysql.dll) Правильные ответы: A, B, D Объяснение: Для работы QMYSQL требуется: Доступ к серверу MySQL (локальному или удалённому); Наличие SQL-драйвера Qt для MySQL (плагин); Доступ к клиентской библиотеке MySQL (libmysqlclient или libmysql). Глобальные переменные не обязательны — соединение может управляться локально или через синглтон. Задание 8. Какие преимущества даёт использование QSqlTableModel по сравнению с ручным выполнением запросов через QSqlQuery? A. Автоматическая генерация SQL-запросов для CRUD-операций B. Встроенная поддержка редактирования данных прямо в QTableView C. Более высокая производительность при работе с большими объёмами данных D. Упрощённая привязка к графическим элементам интерфейса Правильные ответы: A, B, D Объяснение: QSqlTableModel упрощает создание редактируемых табличных представлений: он автоматически формирует SELECT, INSERT, UPDATE, DELETE, поддерживает редактирование в QTableView и легко интегрируется с GUI. Однако при работе с очень большими данными предпочтительнее использовать QSqlQueryModel или пагинацию, так как QSqlTableModel загружает все данные в память.

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
			Задания открытого типа с развёрнутым ответом Задание 9. Как использование интерфейса QMYSQL в рамках фреймворка Qt способствует эффективному проектированию клиентских приложений, взаимодействующих с MySQL. Какие принципы разработки ПО реализуются при этом? Ожидаемые элементы ответа: • QMYSQL интегрирован в архитектуру Qt, что позволяет использовать единый стиль работы с данными (MVC-подход). • Классы QSqlDatabase, QSqlQuery, QSqlTableModel обеспечивают абстракцию над низкоуровневыми операциями, повышая читаемость и сопровождаемость кода. • Поддержка сигналов и слотов упрощает реакцию на изменения данных. • Реализуются принципы: ✓ Инкапсуляции — детали подключения скрыты; ✓ Разделения ответственности — логика БД отделена от GUI; ✓ Повторного использования — модели можно применять в разных частях приложения. Задание 10. Сравните подходы к разработке клиентских приложений для MySQL с использованием «голого» C API и с использованием QMYSQL в Qt. В каких ситуациях предпочтителен каждый из них? Ожидаемые элементы ответа: • C API: Плюсы: максимальная производительность, минимальные зависимости, полный контроль. Минусы: ручное управление памятью, отсутствие встроенной поддержки GUI, сложность обработки ошибок. Применение: системные утилиты, embedded-приложения без GUI, high-performance backend-компоненты. • QMYSQL (Qt): Плюсы: тесная интеграция с GUI, автоматическое управление ресурсами, поддержка MVC, кроссплатформенность. Минусы: накладные расходы фреймворка, зависимость от Qt и плагинов. Применение: десктопные GUI-приложения (админки, аналитические панели, CRM-клиенты).

<table>
<tr><th>Тема</th><th>Код и формулировка компетенции</th><th>Индикаторы достижения компетенции</th><th>Тест по теме</th></tr>
<tr><td></td><td></td><td></td><td>
С точки зрения проектирования GUI-приложений:
Использование QMYSQL соответствует современным практикам — ускоряет разработку, снижает количество ошибок, обеспечивает согласованность архитектуры. Для прикладной информатики, где важна связь между интерфейсом и данными, Qt + QMYSQL — оптимальный выбор.
</td></tr>
<tr><td>Тема 6. Некоторые вопросы общей теории баз данных. Использование баз данных.</td><td>ПК.15</td><td>ИПК.15.1</td><td>
Задания закрытого типа на установление соответствия

Задание 1. Установите соответствие между свойством базы данных и его описанием.
<table>
<tr><td>Свойство</td><td>Описание</td></tr>
<tr><td>1. Независимость данных</td><td>A. Возможность одновременной работы нескольких пользователей без конфликтов</td></tr>
<tr><td>2. Целостность данных</td><td>B. Логическая и физическая структура данных не влияет на приложения</td></tr>
<tr><td>3. Безопасность</td><td>C. Данные удовлетворяют заданным бизнес-правилам и ограничениям</td></tr>
<tr><td>4. Многопользовательский доступ</td><td>D. Защита данных от несанкционированного доступа и изменений</td></tr>
</table>

Правильный ответ:
1 — B, 2 — C, 3 — D, 4 — A

Задание 2. Установите соответствие между типом использования БД и примером применения.
<table>
<tr><td>Тип использования</td><td>Пример</td></tr>
<tr><td>1. OLTP (оперативная обработка транзакций)</td><td>A. Анализ продаж за последние 5 лет в BI-системе</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Тема</th><th>Код и формулировка компетенции</th><th>Индикаторы достижения компетенции</th><th>Тест по теме</th></tr>
<tr><td></td><td></td><td></td><td>
<table>
<tr><td>2. OLAP (аналитическая обработка)</td><td>B. Регистрация нового заказа в интернет-магазине</td></tr>
<tr><td>3. Хранилище данных (Data Warehouse)</td><td>C. Единое хранилище исторических данных из разных источников</td></tr>
<tr><td>4. Встраиваемая база данных</td><td>D. Использование SQLite в мобильном приложении для хранения заметок</td></tr>
</table>

Правильный ответ:
1 — B, 2 — A, 3 — C, 4 — D

Задания закрытого типа на установление последовательности

Задание 3. Укажите правильную последовательность этапов жизненного цикла базы данных.
A. Эксплуатация и сопровождение
B. Проектирование (концептуальное, логическое, физическое)
C. Анализ требований к данным
D. Реализация (создание структур, загрузка данных)

Правильный ответ: C → B → D → A

Задание 4. Установите правильную последовательность действий при выполнении транзакции в СУБД.
A. Фиксация изменений (COMMIT)
B. Начало транзакции (BEGIN или START TRANSACTION)
C. Выполнение SQL-операций (INSERT, UPDATE, DELETE)
D. Проверка условий целостности и возможных ошибок

Правильный ответ: B → C → D → A
Задания комбинированного типа (выбор одного ответа)

Задание 5. Какое из перечисленных утверждений лучше всего отражает принцип логической
</td></tr>
</table>

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
			независимости данных? A. При изменении физического расположения файла БД приложения продолжают работать без изменений B. При добавлении нового столбца в таблицу существующие запросы не ломаются C. Пользователи видят только ту часть данных, к которой у них есть доступ D. Все операции над данными выполняются через транзакции Правильный ответ: B Объяснение: Логическая независимость означает, что изменения в логической структуре БД (например, добавление атрибутов) не требуют переделки существующих приложений, если они не используют новые элементы. Это достигается за счёт внешних уровней представления (views, подмножества схемы). Задание 6. Для чего в системах управления базами данных используется журналирование (logging)? A. Для ускорения выполнения SELECT-запросов B. Для обеспечения восстановления данных после сбоев C. Для шифрования конфиденциальной информации D. Для автоматического резервного копирования раз в сутки Правильный ответ: B Объяснение: Журнал транзакций (transaction log) фиксирует все изменения в БД. В случае сбоя СУБД использует журнал для отката незавершённых транзакций и доведения до конца зафиксированных, обеспечивая свойство durability (устойчивость) ACID. **Задания комбинированного типа (выбор нескольких ответов)** Задание 7. Какие из перечисленных функций выполняет современная СУБД? A. Обеспечение параллельного доступа к данным B. Компиляция исходного кода клиентских приложений C. Поддержка декларативных ограничений целостности (PRIMARY KEY, FOREIGN KEY и

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
			др.) D. Автоматическое проектирование ER-диаграмм по текстовому описанию Правильные ответы: A, C Объяснение: СУБД управляет конкурентным доступом (через блокировки/MVCC), контролирует целостность данных средствами SQL. Компиляция кода и автоматическое проектирование — задачи IDE или специализированных CASE-средств, а не СУБД. Задание 8. Какие из следующих утверждений верны относительно использования баз данных в современных информационных системах? A. Базы данных позволяют централизованно управлять данными и минимизировать их дублирование B. Все приложения обязательно должны использовать реляционные СУБД C. Использование БД повышает надёжность и безопасность хранения информации D. Базы данных не нужны в небольших приложениях с локальным хранением Правильные ответы: A, C Объяснение: Централизованное управление и снижение избыточности — ключевые преимущества БД. Безопасность и надёжность обеспечиваются СУБД (резервное копирование, права доступа, транзакции). Однако не все приложения используют реляционные БД (есть NoSQL), а даже небольшие приложения могут использовать встраиваемые БД (SQLite), что всё равно считается использованием СУБД. **Задания открытого типа с развёрнутым ответом** Задание 9. Почему знание общей теории баз данных (включая понятия целостности, независимости, транзакций) важно для разработчика прикладного программного обеспечения. Приведите пример, когда игнорирование этих принципов привело к серьёзной проблеме в реальном проекте. Ожидаемые элементы ответа:

<table>
<tr><th>Тема</th><th>Код и формулировка компетенции</th><th>Индикаторы достижения компетенции</th><th>Тест по теме</th></tr>
<tr><td></td><td></td><td></td><td>

Разработчик должен понимать, как данные хранятся, изменяются и защищаются, чтобы проектировать корректное ПО.
Пример: отсутствие транзакций при переводе денег между счетами → при сбое деньги списались, но не зачислились → потеря средств.
Игнорирование целостности (например, отсутствие внешних ключей) → «висячие» записи, некорректная аналитика.
Незнание независимости данных затрудняет модификацию схемы БД без переписывания приложения.

Задание 10. Опишите роль баз данных в архитектуре современного программного обеспечения. Как выбор типа СУБД (реляционная, документ-ориентированная, графовая и др.) влияет на проектирование приложения?

Ожидаемые элементы ответа:

БД — центральный компонент большинства ИС: хранение состояния, обеспечение согласованности, поддержка бизнес-логики.
Выбор СУБД определяется характером данных и требованиями:

Реляционные (MySQL, PostgreSQL) — для структурированных данных с чёткими связями (банки, ERP).
Документ-ориентированные (MongoDB) — для гибких, иерархических данных (каталоги, пользовательские профили).
Графовые (Neo4j) — для сложных связей (соцсети, рекомендательные системы).

Неправильный выбор приводит к:

Сложностям масштабирования,
Неэффективным запросам,
Увеличению сложности кода (например, эмуляция связей в NoSQL).
</td></tr>
<tr><td>Тема 7. Информационные системы в сетях.</td><td>ПК.15</td><td>ИПК.15.1</td><td>
Задания закрытого типа на установление соответствия

Задание 1. Установите соответствие между архитектурой информационной системы и её характеристикой.
<table>
<tr><td>Архитектура</td><td>Характеристика</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Тема</th><th>Код и формулировка компетенции</th><th>Индикаторы достижения компетенции</th><th>Тест по теме</th></tr>
<tr><td></td><td></td><td></td><td>
<table>
<tr><td>1. Клиент-сервер (2-tier)</td><td>A. Бизнес-логика вынесена в отдельный слой между клиентом и СУБД</td></tr>
<tr><td>2. Трёхзвенная (3-tier)</td><td>B. Клиент напрямую взаимодействует с сервером базы данных</td></tr>
<tr><td>3. Веб-ориентированная</td><td>C. Доступ к системе осуществляется через веб-браузер; сервер обрабатывает HTTP-запросы</td></tr>
<tr><td>4. Пиринговая (P2P)</td><td>D. Все узлы равноправны; данные и ресурсы распределены между участниками сети</td></tr>
</table>

Правильный ответ:
1 — B, 2 — A, 3 — C, 4 — D

Задание 2. Установите соответствие между компонентом сетевой информационной системы и его функцией.
<table>
<tr><td>Компонент</td><td>Функция</td></tr>
<tr><td>1. Веб-сервер</td><td>A. Хранение и управление структурированными данными</td></tr>
<tr><td>2. Сервер приложений</td><td>B. Обработка бизнес-логики и выполнение транзакций</td></tr>
<tr><td>3. СУБД</td><td>C. Обслуживание HTTP-запросов и передача статического/динамического контента</td></tr>
<tr><td>4. Балансировщик нагрузки</td><td>D. Распределение входящих запросов между несколькими серверами</td></tr>
</table>

Правильный ответ:
1 — C, 2 — B, 3 — A, 4 — D

Задания закрытого типа на установление последовательности
</td></tr>
</table>

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
			Задание 3. Укажите правильную последовательность этапов взаимодействия пользователя с веб-приложением, обращающимся к базе данных. A. Веб-сервер передаёт запрос серверу приложений B. Пользователь вводит данные в браузере и отправляет форму C. Сервер приложений формирует SQL-запрос и отправляет его СУБД D. Результат отображается в браузере пользователя E. СУБД выполняет запрос и возвращает данные Правильный ответ: B → A → C → E → D Задание 4. Установите правильную последовательность действий при обеспечении отказоустойчивости сетевой информационной системы. A. Настройка репликации базы данных на резервный сервер B. Размещение системы в нескольких географических зонах C. Внедрение механизма автоматического переключения (failover) D. Мониторинг доступности основных компонентов Правильный ответ: D → A → C → B **Задания комбинированного типа (выбор одного ответа)** Задание 5. Какая архитектура наиболее подходит для корпоративной информационной системы с централизованным управлением данными и строгой безопасностью? A. Пиринговая (P2P) B. Одноуровневая файл-серверная C. Трёхзвенная клиент-серверная D. Монолитное настольное приложение Правильный ответ: C Объяснение:

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
			Трёхзвенная архитектура разделяет представление, логику и данные, что позволяет централизованно управлять бизнес-правилами, контролировать доступ к данным и обеспечивать безопасность на уровне сервера приложений и СУБД. P2P и файл-серверная архитектуры не обеспечивают достаточного уровня контроля и масштабируемости. Задание 6. Что является основным преимуществом использования RESTful API в сетевых информационных системах? A. Полная замена необходимости в базе данных B. Упрощение взаимодействия между компонентами за счёт стандартизированных HTTP-методов C. Автоматическое шифрование всех передаваемых данных D. Исключение необходимости в сервере приложений Правильный ответ: B Объяснение: RESTful API использует стандартные HTTP-методы (GET, POST, PUT, DELETE) для операций над ресурсами, что обеспечивает простоту, масштабируемость и независимость клиентов и серверов. Это упрощает интеграцию, тестирование и развитие распределённых систем. Шифрование (например, через HTTPS) — отдельный вопрос, не встроенный в REST по умолчанию. **Задания комбинированного типа (выбор нескольких ответов)** Задание 7. Какие из перечисленных требований характерны для современных сетевых информационных систем? A. Поддержка многопользовательского доступа B. Наличие графического интерфейса только на стороне сервера C. Масштабируемость при росте числа пользователей D. Обеспечение целостности и безопасности данных в распределённой среде Правильные ответы: A, C, D

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
			Объяснение: Современные ИС должны обслуживать множество пользователей, расти по нагрузке и защищать данные. Графический интерфейс обычно реализуется на стороне клиента (браузер, мобильное приложение), а не на сервере. Поэтому утверждение B — неверно. Задание 8. Какие технологии или подходы используются для повышения производительности сетевых информационных систем? A. Кэширование часто запрашиваемых данных B. Использование stored procedures в СУБД C. Хранение всех данных в оперативной памяти без персистентности D. Горизонтальное масштабирование (шардинг) базы данных Правильные ответы: A, B, D Объяснение: Кэширование (Redis, Memcached), хранимые процедуры (уменьшают сетевой трафик и ускоряют выполнение) и шардинг — стандартные методы оптимизации. Полное хранение данных только в RAM без сохранения на диск (вариант C) нарушает принцип durability и неприемлемо для большинства ИС, где важна надёжность. **Задания открытого типа с развёрнутым ответом** Задание 9. Как архитектура информационной системы влияет на выбор методов и средств проектирования программного обеспечения. Приведите пример, когда неправильный выбор архитектуры привёл к проблемам в эксплуатации системы. Ожидаемые элементы ответа: • Архитектура определяет модульность, масштабируемость, безопасность и сложность сопровождения. • Например, использование файл-серверной архитектуры для многопользовательской системы учёта приведёт к: Конфликтам при одновременной записи, Отсутствию контроля целостности,

<table>
<tr><th>Тема</th><th>Код и формулировка компетенции</th><th>Индикаторы достижения компетенции</th><th>Тест по теме</th></tr>
<tr><td></td><td></td><td></td><td>Проблемам с производительностью при росте числа пользователей.
В то же время трёхзвенная архитектура позволяет отделить логику, использовать пулы соединений, внедрять аутентификацию и авторизацию.
Задание 10. Опишите роль базы данных в распределённой сетевой информационной системе. Как обеспечивается согласованность данных при их репликации между узлами? Какие компромиссы при этом возникают?

Ожидаемые элементы ответа:
БД в распределённой ИС — центральный (или децентрализованный) источник достоверных данных.
Для обеспечения доступности и отказоустойчивости данные реплицируются (мастер-реплика, multi-master).
Согласованность может поддерживаться через:
Синхронную репликацию (гарантирует согласованность, но снижает производительность),
Асинхронную репликацию (повышает производительность, но возможны временные рассогласования),
Использование протоколов вроде Paxos, Raft или двухфазного коммита (2PC).
Возникает компромисс CAP-теоремы: невозможно одновременно обеспечить согласованность (Consistency), доступность (Availability) и устойчивость к разделению сети (Partition tolerance) — приходится выбирать два из трёх.
Например, банковские системы жертвуют доступностью ради согласованности, а соцсети — наоборот.</td></tr>
<tr><td>Тема 8. Распределенные базы данных.</td><td>ПК.15</td><td>ИПК.15.1</td><td>Задания закрытого типа на установление соответствия

Задание 1. Установите соответствие между типом распределённой базы данных и её характеристикой.<table><tr><td>Тип распределённой БД</td><td>Характеристика</td></tr><tr><td>1. Гомогенная</td><td>A. Узлы используют разные СУБД и модели данных</td></tr><tr><td>2. Гетерогенная</td><td>B. Все узлы используют одну и ту же</td></tr></table></td></tr>
</table>

<table>
<tr><th>Тема</th><th>Код и формулировка компетенции</th><th>Индикаторы достижения компетенции</th><th>Тест по теме</th></tr>
<tr><td></td><td></td><td></td><td>
<table>
<tr><td></td><td>СУБД и схему данных</td></tr>
<tr><td>3. Реплицированная</td><td>C. Каждый фрагмент данных хранится в одном узле</td></tr>
<tr><td>4. Фрагментированная (разделённая)</td><td>D. Один и тот же набор данных хранится на нескольких узлах</td></tr>
</table>

Правильный ответ:
1 — B, 2 — A, 3 — D, 4 — C

Задание 2. Установите соответствие между понятием и его определением в контексте распределённых баз данных.
<table>
<tr><td>Понятие</td><td>Определение</td></tr>
<tr><td>1. Прозрачность расположения</td><td>A. Пользователь не знает, сколько копий данных существует</td></tr>
<tr><td>2. Прозрачность репликации</td><td>B. Пользователь не указывает, на каком узле находятся данные</td></tr>
<tr><td>3. Прозрачность фрагментации</td><td>C. Пользователь работает с данными как с единым целым, не зная об их разделении</td></tr>
<tr><td>4. Прозрачность сетевых сбоев</td><td>D. Система автоматически обрабатывает временные разрывы связи</td></tr>
</table>

Правильный ответ:
1 — B, 2 — A, 3 — C, 4 — D

Задания закрытого типа на установление последовательности

Задание 3. Укажите правильную последовательность этапов обработки глобального запроса в гомогенной распределённой базе данных.

A. Передача подзапросов на соответствующие узлы</td></tr>
</table>

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
			B. Анализ запроса и определение необходимых фрагментов/реплик C. Объединение результатов от узлов и формирование итогового ответа D. Оптимизация плана выполнения запроса с учётом стоимости передачи данных Правильный ответ: B → D → A → C Задание 4. Установите правильную последовательность действий при выполнении распределённой транзакции по протоколу двухфазной фиксации (2PC). A. Координатор отправляет всем участникам сообщение COMMIT или ABORT B. Участники выполняют транзакцию локально и отправляют координатору готовность (VOTE COMMIT) C. Координатор получает все голоса и принимает решение D. Инициация транзакции координатором Правильный ответ: D → B → C → A **Задания комбинированного типа (выбор одного ответа)** Задание 5. Какой из перечисленных принципов лежит в основе CAP-теоремы применительно к распределённым базам данных? A. Невозможно одновременно обеспечить согласованность, доступность и устойчивость к сетевым разделениям B. Все распределённые системы должны использовать реляционную модель данных C. Репликация всегда гарантирует полную согласованность данных D. Производительность распределённой БД всегда выше централизованной Правильный ответ: A Объяснение: CAP-теорема (теорема Брюера) утверждает, что в условиях сетевого разделения (Partition tolerance) система может обеспечить либо согласованность (Consistency), либо доступность (Availability), но не оба свойства одновременно. Это фундаментальное ограничение при проектировании распределённых систем.

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
			Задание 6. Что такое «прозрачность» в распределённых базах данных? A. Данные хранятся в открытом виде без шифрования B. Пользователь не замечает распределённой природы системы и работает с ней как с локальной C. Все узлы сети имеют равный доступ ко всем данным D. Система автоматически удаляет устаревшие реплики Правильный ответ: B Объяснение: Прозрачность — ключевой принцип распределённых БД: пользователь или приложение взаимодействует с системой так, будто она централизована. Это включает прозрачность расположения, репликации, фрагментации и сбоев, что упрощает разработку прикладного ПО. **Задания комбинированного типа (выбор нескольких ответов)** Задание 7. Какие из перечисленных стратегий используются для обеспечения согласованности в распределённых базах данных? A. Двухфазная фиксация (2PC) B. Асинхронная репликация без подтверждения C. Кворумное чтение/запись (quorum-based protocols) D. Полное отключение репликации Правильные ответы: A, C Объяснение: 2PC — классический протокол для обеспечения атомарности и согласованности в распределённых транзакциях. Кворумные протоколы (например, в Dynamo, Cassandra) обеспечивают согласованность через правило: R + W > N (где R — число чтений, W — записей, N — реплик). Асинхронная репликация без подтверждения не гарантирует согласованности (eventual consistency), а отключение репликации устраняет проблему, но лишает систему отказоустойчивости.

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
			Задание 8. Какие преимущества даёт использование распределённых баз данных в крупных информационных системах? A. Повышение отказоустойчивости за счёт репликации B. Улучшение локальной производительности за счёт размещения данных ближе к пользователям C. Упрощение проектирования по сравнению с централизованными БД D. Поддержка масштабируемости при росте объёмов данных и числа пользователей Правильные ответы: A, B, D Объяснение: Распределённые БД повышают надёжность (репликация), снижают задержки (локальные узлы), легко масштабируются (горизонтальное масштабирование). Однако они усложняют проектирование из-за необходимости управления согласованностью, транзакциями и сетевыми сбоями — поэтому утверждение C неверно. **Задания открытого типа с развёрнутым ответом** Задание 9. Почему при проектировании современного программного обеспечения важно учитывать особенности распределённых баз данных. Как знание CAP-теоремы помогает принимать архитектурные решения? Ожидаемые элементы ответа: • Современные приложения (веб, мобильные, облачные) часто требуют высокой доступности, масштабируемости и географического распределения. • Распределённые БД — естественный выбор, но они вносят сложность: сетевые задержки, сбои, согласованность. • CAP-теорема помогает осознанно выбирать архитектуру: Банковские системы → выбирают CP (согласованность + устойчивость к разделению), жертвуя мгновенной доступностью. Социальные сети, мессенджеры → выбирают AP (доступность + устойчивость), допуская временную несогласованность. Задание 10. Опишите, как обеспечивается целостность и надёжность данных в

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Тест по теме
			распределённой базе данных. Какие механизмы используются для обработки сбоев узлов или сетевых разделений? Ожидаемые элементы ответа: • Целостность поддерживается через: Глобальные ограничения (сложнее реализуются, чем в централизованных БД), Распределённые транзакции (2PC, Saga-паттерн), Согласованные реплики (кворумы, синхронная репликация). • Надёжность обеспечивается: Репликацией данных на несколько узлов, Автоматическим переключением (failover) при сбое узла, Журналированием (logs) и восстановлением после сбоев. • При сетевом разделении: Системы CP блокируют операции до восстановления связи, Системы AP продолжают работать, но могут возвращать устаревшие данные (eventual consistency). Пример: в Apache Cassandra используется gossip-протокол для обнаружения сбоев и hinted handoff для восстановления данных после восстановления узла.

Критерии оценивания тестового задания:

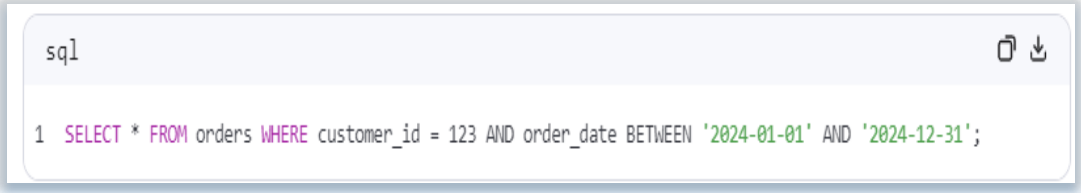
Оценка	Критерии оценивания
отлично	от 90 до 100 % правильно выполненных заданий
хорошо	от 70 до 89 % правильно выполненных заданий
удовлетворительно	от 50 до 69 % правильно выполненных заданий
неудовлетворительно	менее 50 % правильно выполненных заданий

4.2 ТИПОВЫЕ ПРАКТИЧЕСКИЕ ЗАДАНИЯ

Практические заданиядолжны отражать умение обучающегося применять осваиваемую компетенцию в практических ситуациях и при решении производственных задач.

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Задания к практическому занятию
Тема 1. Основные понятия области баз данных и информационных систем.	ПК.15	ИПК.15.2	Задание 1.1. Проанализируйте предложенную предметную область «Университет» (студенты, группы, преподаватели, курсы, оценки). Выделите сущности, их атрибуты и связи между ними. Постройте ER-диаграмму в нотации Чена или UML. Задание 1.2. Сравните файловую систему и СУБД как средства хранения данных на примере приложения «Личный архив документов». Приведите 3 преимущества и 2 недостатка каждого подхода. Сформулируйте, при каких условиях целесообразно использовать СУБД. Задание 1.1. Для предметной области «Медицинская клиника» (пациенты, врачи, приёмы, диагнозы, анализы) выделите сущности, атрибуты и связи. Спроектируйте логическую структуру данных в виде списка сущностей с их атрибутами и типами (например: Пациент(ID: int, ФИО: string, Дата_рождения: date)). Задание 1.2. Сравните реляционную и иерархическую структуры данных на примере хранения информации о подразделениях и сотрудниках компании. Постройте обе структуры и укажите, какая лучше подходит для частых запросов вида «Показать всех сотрудников отдела и его подотделов».
Тема 2. Проектирование баз данных. Элементы математической теории баз данных.	ПК.15	ИПК.15.2	Задание 2.1. Дана таблица Заказы(Номер_заказа, Дата, Клиент, Товар, Цена, Категория_товара, Менеджер). – Приведите её к первой нормальной форме (1НФ). – Выявите функциональные зависимости. Задание 2.2.

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Задания к практическому занятию
			Для отношения R(A, B, C, D) заданы функциональные зависимости: A → B, B → C, C → D. Определите ключ отношения, проверьте, находится ли оно в НФБК, и если нет — выполните декомпозицию на отношения, удовлетворяющие НФБК. Задание 2.1. Дана неструктурированная таблица: Сотрудники(Табельный_номер, ФИО, Отдел, Руководитель_отдела, Проект, Роль_в_проекте, Ставка) – Выявите функциональные зависимости. – Приведите структуру к третьей нормальной форме (3НФ). – Нарисуйте схему полученных отношений и укажите внешние ключи. Задание 2.2. Для отношения R(A, B, C, D, E) заданы ФЗ: A → B, B → C, C → D, D → E. – Найдите замыкание $\{A\}^+$. – Определите все возможные ключи. – Проверьте, находится ли отношение в НФБК, и если нет — выполните декомпозицию на отношения, удовлетворяющие НФБК, сохранив зависимости.
Тема 3. СУБД MySQL.	ПК.15	ИПК.15.2	Задание 3.1. Создайте в MySQL базу данных library со следующими таблицами: – authors(id, name) – books(id, title, author_id, year, isbn) – readers(id, full_name, email) – loans(book_id, reader_id, issue_date, return_date) Настройте первичные и внешние ключи, используя движок InnoDB. Напишите SQL-скрипт, включающий создание БД, таблиц и ограничений. Задание 3.2. Напишите SQL-запросы для базы library: 1. Вывести список всех книг с именами авторов.

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Задания к практическому занятию
			2. Найти читателей, у которых есть невозвращённые книги (return_date IS NULL). 3. Создать индекс по полю isbn в таблице books и указать, как он повлияет на производительность. Задание 3.1. Спроектируйте структуру базы данных online_store для интернет-магазина. Определите таблицы: – users, products, categories, orders, order_items. Для каждой таблицы укажите: – поля, – типы данных , – первичные и внешние ключи. Реализуйте структуру в MySQL. Задание 3.2. Проанализируйте запрос:  – Предложите оптимальную структуру индекса для ускорения этого запроса. – Почему составной индекс (customer_id, order_date) эффективнее двух отдельных.
Тема 4. Разработка клиентских приложений MySQL. Интерфейс MySQL C API.	ПК.15	ИПК.15.2	Задание 4.1. Напишите программу на языке C, которая подключается к локальной базе library (из Темы 3), выполняет запрос SELECT title FROM books WHERE year > 2020 и выводит результат в консоль. Обработайте возможные ошибки подключения и выполнения запроса с помощью mysql_error(). Задание 4.2. Модифицируйте программу из задания 4.1 так, чтобы она принимала параметр года из

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Задания к практическому занятию
			командной строки и безопасно вставляла его в SQL-запрос с использованием экранирования (mysql_real_escape_string()). Протестируйте на попытке SQL-инъекции. Задание 4.1. Спроектируйте структуру данных на языке C для хранения информации о книге (ID, название, год, ISBN). Напишите функцию, которая принимает MYSQL_RES* и заполняет массив таких структур данными из таблицы books. Обеспечьте корректное управление памятью. Задание 4.2. Разработайте структуру данных для кэширования часто запрашиваемых записей из БД (например, справочник категорий). Реализуйте простой кэш в виде связного списка на C, который загружается один раз при старте программы и используется вместо повторных SQL-запросов.
Тема 5. Разработка клиентских приложений MySQL. Интерфейс QMYSQL для GUI-приложений MySQL.	ПК.15	ИПК.15.2	Задание 5.1. Создайте GUI-приложение на Qt (C++), которое подключается к базе library через драйвер QMYSQL. Реализуйте отображение списка книг в QTableView с использованием QSqlTableModel. Обеспечьте корректную настройку проекта (.pro-файл). Задание 5.2. Добавьте в GUI-приложение на Qt (C++), которое подключается к базе library через драйвер QMYSQL форму для добавления новой книги: поля title, year, выпадающий список авторов (загружается из таблицы authors). После сохранения данные должны появиться в таблице без перезапуска приложения. Задание 5.1. Создайте пользовательскую модель данных на основе QAbstractTableModel для отображения списка студентов с дополнительными вычисляемыми полями (например, «Средний балл»). Спроектируйте внутреннюю структуру данных (например, QList<Student>), где Student — ваш класс с полями. Задание 5.2.

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Задания к практическому занятию
			Спроектируйте структуру данных для хранения дерева подразделений компании (иерархия). Реализуйте отображение в QTreeView с использованием QStandardItemModel. Обеспечьте возможность добавления/удаления узлов и сохранения структуры в БД через самоссылающееся отношение (id, parent_id, name).
Тема 6. Некоторые вопросы общей теории баз данных. Использование баз данных.	ПК.15	ИПК.15.2	Задание 6.1. Разработайте сценарий использования базы данных в мобильном приложении «Заметки». Определите: – Тип СУБД (встраиваемая/серверная), – Необходимость транзакций, – Требования к целостности и безопасности. Задание 6.2. Смоделируйте ситуацию: при одновременном редактировании одной записи двумя пользователями в веб-приложении произошла потеря изменений. Предложите два технических решения (например, optimistic/pessimistic locking) и опишите их реализацию на уровне приложения и СУБД. Задание 6.1. Для мобильного приложения «Личный бюджет» спроектируйте структуру данных, поддерживающую: – категории расходов, – транзакции (доход/расход), – привязку к счёту. Учтите необходимость локального хранения (SQLite) и возможную синхронизацию с облаком. Предложите формат хранения временных меток и валют. Задание 6.2. Спроектируйте структуру данных для журналирования операций в системе (audit log): кто, что, когда изменил. Сделайте выбор между: – хранением в отдельной таблице, – использованием триггеров, – ведением лога на уровне приложения.

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Задания к практическому занятию
			Укажите, какие поля должны быть включены для восстановления состояния.
Тема 7. Информационные системы в сетях.	ПК.15	ИПК.15.2	Задание 7.1. Спроектируйте архитектуру трёхзвенной информационной системы «Электронный журнал школы»: – клиент (веб-браузер), – сервер приложений (на Python/Node.js), – СУБД (MySQL). Опишите взаимодействие компонентов при операции «Выставить оценку ученику», включая форматы передачи данных (например, JSON) и протоколы (HTTP, SQL). Задание 7.2. Реализуйте RESTful API-эндпоинт (на любом языке) для получения списка студентов группы по её ID. Эндпоинт должен: – принимать GET-запрос, – обращаться к MySQL, – возвращать JSON с данными или ошибку 404. Продемонстрируйте вызов через curl или Postman. Задание 7.1. Спроектируйте структуру данных для RESTful API сервиса «Управление задачами» (task manager). Определите: – JSON-схемы для сущностей Task, User, Project, – связи между ними, – поля для пагинации и фильтрации. Обеспечьте согласованность имён и типов (например, task_id: integer, created_at: ISO 8601 string). Задание 7.2. Спроектируйте структуру кэша на стороне сервера для хранения часто запрашиваемых профилей пользователей. Определите: – ключ кэша (например, user:123), – формат значения (JSON или сериализованный объект),

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Задания к практическому занятию
			– политику инвалидации при обновлении данных в БД.
Тема 8. Распределенные базы данных.	ПК.15	ИПК.15.2	Задание 8.1. Рассмотрите систему «Онлайн-магазин с филиалами в разных странах». – Предложите схему горизонтального шардинга данных по регионам. – Опишите, как будет обрабатываться запрос «Показать все заказы клиента», если клиент делал покупки в разных регионах. – Укажите, какой тип согласованности (strong/eventual) вы выберете и почему. Задание 8.2. Смоделируйте отказ одного узла в распределённой системе с репликацией (мастер + 2 реплики). – Опишите последовательность действий механизма failover. – Как будет восстановлен узел после возврата в строй? – Какие данные могут быть потеряны при асинхронной репликации? Задание 8.1. Для системы «Глобальная социальная сеть» спроектируйте структуру данных с учётом географического шардинга: – как будут распределены пользователи по регионам, – как будет храниться информация о друзьях из разных регионов, – предложите схему глобального ID (например, UUID или составной ключ). Задание 8.2. Спроектируйте структуру данных для хранения событий в системе с eventual consistency (например, лайки, комментарии). Учтите: – необходимость идемпотентности, – временные метки, – идентификаторы источника события. Предложите формат события, позволяющий безопасно применять его в любом порядке.

Критерии оценивания практических занятий:

Оценка	Критерии оценивания
отлично	Выставляется, если обучающийся умеет увязывать теорию с практикой (решает задачи, формулирует выводы, умеет пояснить полученные результаты), владеет понятийным аппаратом, полно и глубоко овладел материалом по заданной теме, обосновывает свои суждения и даёт правильные ответы на вопросы преподавателя
хорошо	Выставляется, если обучающийся умеет увязывать теорию с практикой (решает задачи и формулирует выводы, умеет пояснить полученные результаты), владеет понятийным аппаратом, полно и глубоко овладел материалом по заданной теме, но содержание ответов имеют некоторые неточности и требуют уточнения и комментария со стороны преподавателя
удовлетворительно	Выставляется, если обучающийся знает и понимает материал по заданной теме, но изложение неполное, непоследовательное, допускаются неточности в определении понятий, студент не может обосновать свои ответы на уточняющие вопросы преподавателя
неудовлетворительно	Выставляется, если обучающийся допускает ошибки в определении понятий, искажающие их смысл, беспорядочно и неуверенно излагает материал. Делает ошибки в ответах на уточняющие вопросы преподавателя

4.3. ТИПОВЫЕ ЗАДАНИЯ ДЛЯ КОНТРОЛЬНЫХ РАБОТ

Контрольные работы содержат несколько практических заданий по индивидуальным вариантам, в полном объеме охватывающих изученный материал по указанной теме. Выполнение контрольных работ позволяет определить результат освоения компетенций по дисциплине в рамках рассматриваемой темы, оцениваемый с помощью соответствующих индикаторов достижения компетенций.

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Типовой вариант контрольной работы
Тема 1. Основные понятия области баз данных и информационных систем.	ПК.15	ИПК.15.2	Задание 1.1 Для предметной области «Библиотека» (читатели, книги, авторы, выдачи) выделите сущности, их атрибуты и связи. Постройте ER-диаграмму (в произвольной нотации). Задание 1.2 Сравните использование файловой системы и СУБД для хранения данных в приложении «Учёт склада». Приведите 3 преимущества СУБД и 2 ситуации, где файловая система может быть допустима. Задание 1.1 Для предметной области «Автомобильный сервис» (мастера, клиенты, автомобили, заказ-наряды, запчасти) выделите сущности, их атрибуты и связи. Спроектируйте логическую структуру данных в виде

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Типовой вариант контрольной работы
			списка сущностей с указанием типов атрибутов (например: Клиент(ID: int, ФИО: varchar(100), Телефон: varchar(20))). Задание 1.2 Сравните реляционную и документ-ориентированную структуры данных на примере хранения информации о курсах и студентах. Постройте обе структуры и укажите, какая лучше подходит для запросов вида «Показать все курсы студента и его оценки».
Тема 2. Проектирование баз данных. Элементы математической теории баз данных.	ПК.15	ИПК.15.2	Задание 2.1 Дана таблица: Сессия(Студент, Группа, Дисциплина, Преподаватель, Оценка, Дата_экзамена) – Выявите функциональные зависимости. – Приведите к третьей нормальной форме (3НФ). – Укажите, какие аномалии устранены. Задание 2.2 Для отношения R(A, B, C, D) заданы функциональные зависимости: A → B, B → C, C → D. – Найдите ключ отношения. – Проверьте, находится ли оно в НФБК. – Если нет — выполните декомпозицию на отношения, удовлетворяющие НФБК. Задание 2.1 Дана таблица: Продажи(Номер_чека, Дата, Кассир, Товар, Цена, Количество, Скидка, Итог) – Выявите функциональные зависимости. – Приведите структуру к третьей нормальной форме (3НФ). – Нарисуйте схему полученных отношений и укажите внешние ключи. Задание 2.2 Для отношения R(A, B, C, D, E) заданы ФЗ:

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Типовой вариант контрольной работы
			A → B, B → C, C → D, D → E. – Найдите замыкание $\{A\}^+$. – Определите все возможные ключи. – Проверьте, находится ли отношение в НФБК, и если нет — выполните декомпозицию на отношения, удовлетворяющие НФБК, сохранив зависимости.
Тема 3. СУБД MySQL.	ПК.15	ИПК.15.2	Задание 3.1 Напишите SQL-скрипт для создания базы данных university со следующими таблицами: students(id, name, group_id) groups(id, name) courses(id, title, teacher) enrollments(student_id, course_id, grade) Настройте первичные и внешние ключи, используя движок InnoDB. Задание 3.2 Для таблицы products(id, name, price, category_id, created_at) напишите: Запрос на выбор всех товаров категории 5, отсортированных по цене. Запрос на обновление цены на 10% для товаров, добавленных до 2023 года. Создайте составной индекс, ускоряющий оба запроса. Задание 3.1 Спроектируйте структуру базы данных fitness_club для фитнес-клуба. Определите таблицы: – members, trainers, subscriptions, workouts, attendance. Для каждой таблицы укажите: – поля, – типы данных с обоснованием (например, DECIMAL(8,2) для стоимости), – первичные и внешние ключи. Реализуйте структуру в виде SQL-скрипта. Задание 3.2

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Типовой вариант контрольной работы
			Проанализируйте запрос: sql 1 SELECT * FROM orders 2 WHERE user_id = 100 AND status = 'active' AND created_at > '2025-01-01'; – Предложите оптимальную структуру составного индекса. – Как изменится структура индекса, если часто фильтровать только по status?
Тема 4. Разработка клиентских приложений MySQL. Интерфейс MySQL C API.	ПК.15	ИПК.15.2	Задание 4.1 Напишите фрагмент программы на языке C, который: – подключается к базе university, – выполняет запрос SELECT name FROM students WHERE group_id = 2, – выводит результат в консоль, – корректно освобождает ресурсы. Обработайте возможные ошибки. Задание 4.2 Напишите функцию на языке C, которая безопасно вставляет новую запись в таблицу logs(event_type, description, timestamp) с использованием экранирования входных данных (mysql_real_escape_string). Продемонстрируйте защиту от SQL-инъекции. Задание 4.1 Спроектируйте структуру данных на языке C для хранения информации о тренировке: – ID, название, дата, ID тренера, продолжительность (в минутах). Напишите функцию, которая принимает результат запроса (MYSQL_RES*) и заполняет динамический массив таких структур. Обеспечьте корректное управление памятью. Задание 4.2 Разработайте структуру данных для кэширования справочника категорий тренировок (ID, название). Реализуйте простой кэш в виде связного списка на C, который загружается один раз при старте программы и используется вместо повторных SQL-запросов.

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Типовой вариант контрольной работы
Тема 5. Разработка клиентских приложений MySQL. Интерфейс QMYSQL для GUI-приложений MySQL.	ПК.15	ИПК.15.2	Задание 5.1 Опишите шаги по настройке проекта на Qt (C++) для подключения к MySQL через QMYSQL. Приведите содержимое .pro-файла и код инициализации соединения. Как проверить, успешно ли установлено соединение? Задание 5.2 Создайте в Qt приложении форму на основе QFormLayout для добавления нового студента: поля «Имя», «Группа» (выпадающий список, загружаемый из БД). После нажатия кнопки «Сохранить» данные должны быть добавлены в таблицу students и отображены в QTableView без перезагрузки. Задание 5.1 Создайте пользовательскую модель данных на основе QAbstractTableModel для отображения расписания тренировок. Спроектируйте внутреннюю структуру данных (например, QList<Workout>), где Workout — ваш класс с полями. Задание 5.2 Спроектируйте структуру данных для хранения иерархии упражнений (например: «Силовые → Ноги → Приседания»). Реализуйте отображение в QTreeView с использованием QStandardItemModel. Предложите схему хранения в БД (самоссылающееся отношение) и внутреннюю структуру в памяти.
Тема 6. Некоторые вопросы общей теории баз данных. Использование баз данных.	ПК.15	ИПК.15.2	Задание 6.1 В приложении «Электронный дневник» два учителя одновременно выставляют оценки одному ученику. Опишите, как может возникнуть проблема гонки данных. Предложите два способа её решения: один на уровне СУБД (например, блокировки), другой — на уровне приложения. Задание 6.2 Зачем в СУБД используются транзакции. Приведите пример из практики (например, банковский перевод), где отсутствие транзакций приведёт к нарушению целостности данных. Укажите, какие свойства ACID здесь критичны.

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Типовой вариант контрольной работы
			Задание 6.1 Для мобильного приложения «Личный трекер привычек» спроектируйте структуру данных, поддерживающую: – категории привычек, – ежедневные отметки выполнения, – статистику за неделю/месяц. Учтите необходимость локального хранения (SQLite) и возможную синхронизацию. Предложите формат хранения дат и булевых значений. Задание 6.2 Спроектируйте структуру данных для журналирования операций в системе (audit log): кто, что, когда изменил. Учтите необходимость восстановления состояния. Укажите обязательные поля (например, entity_id, old_value, new_value, timestamp, user_id)
Тема 7. Информационные системы в сетях.	ПК.15	ИПК.15.2	Задание 7.1 Спроектируйте архитектуру трёхзвенной системы «Сервис бронирования отелей». Укажите: – компоненты (клиент, сервер приложений, СУБД), – протоколы взаимодействия, – как передаётся запрос «Показать свободные номера в отеле X на даты Y–Z». Задание 7.2 Разработайте RESTful API-эндпоинт (описание на псевдокоде или любом языке) для получения списка заказов пользователя по его ID. Укажите: – HTTP-метод, – формат запроса и ответа (JSON), – как обрабатывается ошибка «пользователь не найден». Задание 7.1 Спроектируйте структуру данных для RESTful API сервиса «Управление задачами» (task manager). Определите JSON-схемы для:

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Типовой вариант контрольной работы
			– Task(id: integer, title: string, status: enum, created_at: ISO 8601), – User(id, email, name), – связи между задачами и пользователями. Обеспечьте согласованность имён, типов и форматов. Задание 7.2 Спроектируйте структуру кэша на стороне сервера для хранения профилей пользователей. Определите: – ключ кэша (например, user:profile:123), – формат значения (JSON или сериализованный объект), – политику инвалидации при обновлении данных в БД.
Тема 8. Распределенные базы данных.	ПК.15	ИПК.15.2	Задание 8.1 Система «Международный банк» хранит данные клиентов в региональных узлах (Европа, Азия, Америка). – Предложите схему репликации для обеспечения отказоустойчивости. – Как будет обрабатываться перевод средств между клиентами из разных регионов? – Какой тип согласованности (strong/eventual) вы выберете и почему? Задание 8.2 Система «Глобальный мессенджер» использует распределённую БД с узлами в разных странах. – Как обеспечивается доступность при сетевом разделении между Европой и Азией? – Какие данные могут быть несогласованными в этот период? – Как восстанавливается согласованность после восстановления связи? Привяжите ответ к CAP-теореме. Задание 8.1 Для системы «Глобальный фитнес-сервис» с узлами в Европе, Азии и США спроектируйте структуру данных с учётом географического шардинга: – как распределяются пользователи, – как хранятся данные о посещениях в разных странах,

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Типовой вариант контрольной работы
			– предложите схему глобального ID (например, UUID или составной ключ). Задание 8.2 Спроектируйте структуру данных для хранения событий в системе с eventual consistency (например, лайки, комментарии). Учтите: – необходимость идемпотентности, – временные метки, – идентификаторы источника события. Предложите формат события, позволяющий безопасно применять его в любом порядке (например, с использованием векторных часов или уникальных ID).

Критерии оценивания контрольной работы:

Оценка	Критерии оценивания
отлично	Выставляется, если обучающийся умеет увязывать теорию с практикой (решает задачи, формулирует выводы, умеет пояснить полученные результаты), владеет понятийным аппаратом, полно и глубоко овладел материалом по заданной теме, обосновывает свои суждения и даёт правильные ответы на вопросы преподавателя
хорошо	Выставляется, если обучающийся умеет увязывать теорию с практикой (решает задачи и формулирует выводы, умеет пояснить полученные результаты), владеет понятийным аппаратом, полно и глубоко овладел материалом по заданной теме, но содержание ответов имеют некоторые неточности и требуют уточнения и комментария со стороны преподавателя.
удовлетворительно	Выставляется, если обучающийся знает и понимает материал по заданной теме, но изложение неполное, непоследовательное, допускаются неточности в определении понятий, студент не может обосновать свои ответы на уточняющие вопросы преподавателя
неудовлетворительно	Выставляется, если обучающийся допускает ошибки в определении понятий, искажающие их смысл, беспорядочно и неуверенно излагает материал. Делает ошибки в ответах на уточняющие вопросы преподавателя

4.4. ТИПОВЫЕ ЗАДАНИЯ ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ

Самостоятельная работа включает в себя проработку теоретического материала, изучение рекомендуемой литературы, выполнение практико-ориентированных заданий (заполнение таблиц, проведение сравнительного анализа, составление схем и др.), решение практических задач, создание презентаций, написание рефератов, подборка нормативного и иного материала и выполнение других заданий.

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Типовое задание для самостоятельной работы
Тема 1. Основные понятия области баз данных и информационных систем.	ПК.15	ИПК.15.1 ИПК.15.2	Задание 1.1 Тип: Реферат + сравнительный анализ Напишите реферат на тему: «Эволюция моделей данных: от иерархической к реляционной и NoSQL» (объём — 5–7 страниц). Выполните сравнительный анализ файловой системы и СУБД по критериям: целостность, безопасность, многопользовательский доступ, масштабируемость. Результат оформите в виде таблицы. Подготовьте краткую презентацию (5 слайдов) для защиты реферата. Задание 1.2 Тип: Практико-ориентированное проектирование

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Типовое задание для самостоятельной работы
			Выберите одну из предметных областей: «Учёт студентов в университете» «Медицинская карта пациента» «Управление проектами в IT-компании» На основе рекомендаций из ГОСТ Р ИСО/МЭК 12207 определите требования к данным. Спроектируйте концептуальную модель в виде ER-диаграммы (используйте draw.io, Lucidchart или аналог). Оформите пояснительную записку (2–3 стр.), описывающую сущности, атрибуты, связи и обоснование кардинальности.
Тема 2. Проектирование баз данных. Элементы математической теории баз данных.	ПК.15	ИПК.15.1 ИПК.15.2	Задание 2.1 Тип: Теоретический анализ + практическое нормализация Для заданной таблицы Расписание(Преподаватель, Предмет, Группа, Аудитория, День, Время): – выявите функциональные зависимости, – определите ключ, – приведите к 1НФ, 2НФ, 3НФ. Оформите результат в виде схемы отношений с указанием первичных и внешних ключей. Напишите краткий отчёт (1–2 стр.) о том, какие аномалии устранены. Задание 2.2 Тип: Реферат + исследовательская работа Напишите реферат на тему: «Нормальные формы и их роль в проектировании надёжных информационных систем». Включите в работу: – определения 1НФ–5НФ и НФБК, – примеры нарушений, – сравнение подходов: полная нормализация vs. денормализация для производительности. Подготовьте библиографию (не менее 5 источников, включая научные статьи из eLibrary или IEEE Xplore).

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Типовое задание для самостоятельной работы
			Создайте инфографику (в Canva или PowerPoint), иллюстрирующую этапы нормализации.
Тема 3. СУБД MySQL.	ПК.15	ИПК.15.1 ИПК.15.2	Задание 3.1 Тип: Практическая работа + документирование Установите MySQL Server и MySQL Workbench. Создайте базу данных ecommerce со следующими таблицами: users, products, orders, order_items. Настройте: – первичные и внешние ключи, – ограничения (NOT NULL, UNIQUE), – индексы для полей, участвующих в JOIN и WHERE. Напишите SQL-скрипт создания БД и отчёт (2 стр.). Задание 3.2 Тип: Сравнительный анализ + презентация Изучите официальную документацию MySQL и MariaDB. Выполните сравнительный анализ MySQL и PostgreSQL по критериям: – поддержка стандартов SQL, – транзакции, – репликация, – производительность при OLTP/OLAP. Оформите результат в виде таблицы и диаграмм. Подготовьте презентацию (7–10 слайдов) с выводами о целесообразности выбора СУБД для разных типов приложений.
Тема 4. Разработка клиентских приложений MySQL. Интерфейс MySQL C API.	ПК.15	ИПК.15.1 ИПК.15.2	Задание 4.1 Тип: Практическое программирование + отчёт Изучите документацию по MySQL C API (dev.mysql.com). Напишите программу на C, которая: – подключается к БД, – выполняет SELECT-запрос, – выводит результат, – корректно обрабатывает ошибки и освобождает память. Протестируйте программу на примере базы library (из Темы 3).

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Типовое задание для самостоятельной работы
			Напишите отчёт (2–3 стр.), включающий: – схему взаимодействия с API, – описание обработки ошибок, – рекомендации по безопасному использованию. Задание 4.2 Тип: Исследовательская работа + сравнение Изучите альтернативы MySQL C API: – SQLite C API, – PostgreSQL libpq, – ODBC. Сравните их по критериям: простота использования, производительность, кроссплатформенность, документация. Оформите результат в виде таблицы и аналитической справки (2 стр.). Подготовьте вывод: в каких случаях предпочтителен MySQL C API.
Тема 5. Разработка клиентских приложений MySQL. Интерфейс QMYSQL для GUI-приложений MySQL.	ПК.15	ИПК.15.1 ИПК.15.2	Задание 5.1 Тип: Практическое проектирование GUI-приложения Установите Qt и настройте QMYSQL-драйвер. Создайте GUI-приложение на Qt (C++), которое: – подключается к MySQL, – отображает данные из таблицы в QTableView через QSqlTableModel, – позволяет добавлять/редактировать записи. Оформите пользовательскую документацию (1–2 стр.) и скриншоты интерфейса. Напишите отчёт о трудностях настройки драйвера и способах их решения. Задание 5.2 Тип: Реферат + сравнительный анализ Напишите реферат: «Подходы к созданию GUI-приложений с поддержкой баз данных: Qt/QMYSQL vs. Java/JDBC vs. Python/PyQt+SQLAlchemy». Включите: – архитектурные схемы, – сравнение по трудозатратам, производительности, кроссплатформенности.

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Типовое задание для самостоятельной работы
			Подготовьте презентацию (6–8 слайдов) с рекомендациями по выбору технологии для прикладной информатики.
Тема 6. Некоторые вопросы общей теории баз данных. Использование баз данных.	ПК.15	ИПК.15.1 ИПК.15.2	Задание 6.1 Тип: Кейс-анализ + проектирование Изучите кейс: «Потеря данных в мобильном приложении из-за отсутствия транзакций». Проанализируйте причины сбоев. Спроектируйте структуру данных для приложения «Личный бюджет» с учётом: – локального хранения (SQLite), – синхронизации с облаком, – поддержки транзакций. Оформите схему БД и пояснительную записку (2 стр.). Задание 6.2 Тип: Реферат + нормативный анализ Напишите реферат: «Обеспечение целостности и безопасности данных в современных информационных системах». Включите: – ссылки на ФЗ-152 (РФ), GDPR, – методы защиты (шифрование, права доступа, аудит), – роль СУБД в обеспечении безопасности. Подготовьте перечень нормативных требований к хранению персональных данных в ИС.
Тема 7. Информационные системы в сетях.	ПК.15	ИПК.15.1 ИПК.15.2	Задание 7.1 Тип: Проектирование архитектуры + схема Спроектируйте архитектуру трёхзвенной ИС «Сервис онлайн-обучения». Нарисуйте схему взаимодействия компонентов (клиент, сервер приложений, СУБД). Опишите: – протоколы (HTTP, SQL), – форматы данных (JSON), – механизмы аутентификации. Оформите архитектурную диаграмму (в draw.io) и пояснительную записку (2 стр.).

Тема	Код и формулировка компетенции	Индикаторы достижения компетенции	Типовое задание для самостоятельной работы
			Задание 7.2 Тип: Анализ + презентация Изучите архитектуры: – классическая клиент-сервер, – микросервисная, – serverless. Сравните их по критериям: масштабируемость, отказоустойчивость, сложность развёртывания. Подготовьте презентацию (8–10 слайдов) с примерами использования в реальных проектах (Netflix, Uber и др.).
Тема 8. Распределенные базы данных.	ПК.15	ИПК.15.1 ИПК.15.2	Задание 8.1 Тип: Исследовательская работа + схема Изучите CAP-теорему и её последствия. Проанализируйте архитектуры: – Apache Cassandra (AP), – Google Spanner (CP). Спроектируйте структуру данных для глобального сервиса «Социальная сеть» с учётом шардинга и репликации. Нарисуйте схему распределения данных и напишите отчёт (2–3 стр.). Задание 8.2 Тип: Реферат + сравнительный анализ Напишите реферат: «Распределённые базы данных: вызовы, подходы и современные решения». Включите: – сравнение реляционных (CockroachDB) и NoSQL (MongoDB, Cassandra) решений, – описание протоколов согласования (2PC, Paxos, Raft), – примеры использования в промышленности. Подготовьте библиографию (не менее 6 источников, включая англоязычные).

Критерии оценивания самостоятельной работы:

Оценка	Критерии оценивания
отлично	выставляется если работа носит научно-исследовательский характер, проанализирован и сделан сравнительный анализ нескольких литературных источников, приведены примеры
хорошо	выставляется если проанализирован и сделан сравнительный анализ нескольких литературных источников, приведены примеры
удовлетворительно	выставляется если проведен сравнительный анализ научно-методической литературы, приведены примеры
неудовлетворительно	выставляется если работа прошла проверку на антиплагиат и соответствует требованиям оформления

4.5. ИНДИВИДУАЛЬНЫЕ ЗАДАНИЯ ДЛЯ КУРСОВЫХ РАБОТ

Курсовая работа позволяет выявитьстепень владения базовыми знаниями,умениями и навыками, необходимымидля обучения, и определить уровеньвладения новым материалом.

Примерные индивидуальные задания (темы)для курсовых работ:

Курсовая работа по учебному плану не предусмотрена

Критерии оценивания курсовой работы:

Оценка	Критерии оценивания
отлично	Содержание курсовой работы полностью соответствует заданию, содержащемуся в методических указаниях, и плану. Представлены результаты структурированного и логически последовательного обзора литературных и иных источников по теме исследования. Структура курсовой работы логически и методически выдержана. Верно определены исходные данные для расчетов. Все аналитические расчеты выполнены верно, корректно применены методы экономического анализа, не нарушена методика анализа предмета исследования. Все выводы и предложения убедительно аргументированы. При защите курсовой работы обучающийся правильно и уверенно отвечает на вопросы преподавателя, демонстрирует глубокое знание теоретического материала, способен аргументировать собственные утверждения и выводы
хорошо	Содержание курсовой работы полностью соответствует заданию, содержащемуся в методических указаниях, и плану. Представлены результаты структурированного и логически последовательного обзора литературных и иных источников по теме исследования. Структура курсовой работы логически и методически выдержана. Верно определены исходные данные для расчетов. В расчетах допускаются незначительные (не искажающие общего итога оценки) погрешности/ошибки. Большинство выводов и предложений аргументировано, корректно применены методы экономического анализа, не нарушена методика анализа предмета исследования. Оформление курсовой работы и полученные результаты в целом отвечают требованиям, изложенным в методических указаниях. Имеются одна-две несущественные ошибки в использовании терминов, в построенных диаграммах и

	схемах, в оформлении таблиц. Наличествует незначительное количество грамматических и/или стилистических ошибок. При защите курсовой работы обучающийся правильно и уверенно отвечает на большинство вопросов преподавателя, демонстрирует хорошее знание теоретического материала, но не всегда способен аргументировать собственные утверждения и выводы. При наводящих вопросах преподавателя исправляет ошибки в ответе
удовлетворительно	Содержание курсовой работы полностью соответствует заданию, содержащемуся в методических указаниях, и плану. Результаты обзора литературных и иных источников представлены недостаточно полно, недостаточно логично и последовательно. Верно определены исходные данные для расчетов, но имеются грубые ошибки в расчетах. Аргументация выводов и предложений слабая или отсутствует. Экономические выводы носят констатирующий (описательный) характер. Имеются одно-два существенных отклонений от требований в оформлении курсовой работы. Полученные результаты в целом отвечают требованиям, изложенным в методических указаниях. Имеются одна-две существенных ошибки в использовании терминов, в построенных диаграммах и схемах. Много грамматических и/или стилистических ошибок. При защите курсовой работы обучающийся допускает грубые ошибки при ответах на вопросы преподавателя, демонстрирует слабое знание теоретического материала, в большинстве случаев не способен уверенно аргументировать собственные утверждения и выводы
неудовлетворительно	Содержание курсовой работы не соответствует заданию, содержащемуся в методических указаниях, и плану. Неверно определены исходные данные для расчетов, неверно и не корректно применены методы экономического анализа. Экономические выводы содержат неверную экономическую оценку. Имеются более двух существенных отклонений от требований в оформлении курсовой работы. Большое количество существенных ошибок по сути работы, много грамматических и стилистических ошибок и др. Полученные результаты не отвечают требованиям, изложенным в методических указаниях. При защите курсовой работы обучающийся демонстрирует слабое понимание программного материала, студент не может защитить свои решения, допускает грубые фактические ошибки при ответах на поставленные вопросы или вовсе не отвечает на них. Курсовая работа не представлена преподавателю. Обучающийся не явился на защиту курсовой работы

5. МАТЕРИАЛЫ ДЛЯ ОЦЕНКИ РЕЗУЛЬТАТОВ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

Промежуточная аттестация проводится в форме зачета с оценкой.

5.1. Вопросы к зачету с оценкой:

1. Общие понятия баз данных. Информационная система, банк данных, база данных, Система управления базами данных. Их характеристики и виды.
2. Информационная модель данных и ее состав.
3. Типы логических моделей данных.
4. Архитектура системы баз данных. Централизованные и распределенные базы данных.

5. Реляционная модель данных. Определения реляционной модели.
6. Понятие первичного и внешнего ключа. Связывание таблиц. Обеспечение
непрерывности и целостности данных.
7. Типы взаимосвязей в модели данных. Обеспечение непротиворечивости и целостности данных в базе.
8. Типы взаимосвязей в модели данных. Средства ускоренного доступа к данным.
9. Этапы проектирования баз данных. Нормализация баз данных.
10. Этапы проектирования баз данных. Модель сущность-связь.
11. Структурированный язык запросов SQL. Общие сведения. Основные операторы языка SQL.
12. Структурированный язык запросов SQL. Общие сведения. Запросы в SQL. Общий синтаксис.
13. Запросы SQL с использованием единственной таблицы. Исключение дубликатов, выбор вычисляемых значений.
14.Запросы SQL с использованием единственной таблицы. Выборка с использованием фразы WHERE. Использование в запросах операторов IN, LIKE, BETWEEN AND, IS NULL.
15. Запросы SQL с использованием единственной таблицы. Выборка с использованием фразы WHERE. Выборка с упорядочиванием. Использование в запросах операторов IS NULL.
16.Запросы SQL с использованием единственной таблицы. Специальные SQL – функции. SQL функции без использования фразы GROUP BY.
17.SQL функции с использованием фразы GROUP BY. Использование фразы HAVING.
18. Запросы в SQL. Общий синтаксис. Параметрический запрос в SQL.
19.Запросы действий в SQL. Запрос на добавление, запрос на обновление.
20.Запросы действий в SQL. Запрос на удаление, запрос на создание новой таблицы.
21. Запросы SQL с использованием нескольких таблиц. Запросы, использующие соединения (Декартово произведение таблиц, Естественное соединение таблиц, Композиция таблиц)
22. Запросы SQL с использованием нескольких таблиц. Виды вложенных подзапросов.
23. Запросы SQL с использованием нескольких таблиц. Простые вложенные подзапросы.
24.Запросы SQL с использованием нескольких таблиц. Коррелированные вложенные подзапросы.
25.Запросы SQL с использованием нескольких таблиц. Функции в подзапросе. Запросы на объединение.
26. СУБД MySQL. Общие сведения о MySQL. Основные команды. Типы данных.
27. Особенности SQL-запросов в MySQL. Создание таблиц. Простые запросы.
28. Особенности SQL-запросов в MySQL. Выборки по критериям. Использование выражений.
29. Особенности SQL-запросов в MySQL. Ограничение выборки. Работа с датами. Соответствие шаблону.
30. Особенности SQL-запросов в MySQL. Настройка и использование переменных SQL. Использование SQL-функций.
31. Особенности SQL-запросов в MySQL. Выборка из нескольких таблиц и использованием связей. Задание связей и обеспечение целостности данных в таблицах MySQL.

32. Программный интерфейс MySQL с API. Подключение к серверу. Диагностика ошибок.
33. Программный интерфейс MySQL с API. Обработка запросов. Обработка запросов, не возвращающих результатов.
34. Программный интерфейс MySQL с API. Обработка запросов, возвращающих результаты.
35. Программный интерфейс MySQL с API. Обработка запросов общего вида.
36. Программный интерфейс MySQL с API. Обработка запросов. Метаданные результирующего набора.
37. Интерфейс QMYSQL: подключение к серверу БД; классы и методы, используемые при работе с БД (отображение таблицы, получение метаданных, обработка событий и т. п.).
38.Архитектура клиент-сервер.
39. Модели архитектуры «клиент-сервер» (двухзвенные, трехзвенная, сложные схемы взаимодействия, модель монитора транзакций).
40. Основные понятия и вопросы организации систем хранения данных. Типы систем хранения данных. Логическая структура систем хранения данных.
41. Виды технологий хранения данных. Виды устройств хранения данных.
42. Логические модели хранения данных. Системы управления хранилищами.
43. Файловые системы. RAID массивы.
44. Общие идеи распределенного хранения данных Способы оценки и анализа их эффективности.
45. Управление распределенными данными: поддержка соответствия БД вносимым изменениям, доступ к общим данным, тупики.
46. Системы распределенного хранения данных.
47. Облачные системы хранения.
48. Распределенные базы данных. Основные понятия и определения. Правила Дейта для распределенных СУБД.
49. Архитектура распределенных СУБД.
50. Распределенная нереляционная базы данных ApacheCassandra.

Критерии оценивания зачета с оценкой:

Оценка	Критерии оценивания
зачтено	Обучающийся должен: уметь строить ответ в соответствии со структурой излагаемого вопроса; продемонстрировать прочное, достаточно полное усвоение знаний программного материала; продемонстрировать знание основных теоретических понятий; правильно формулировать определения; последовательно, грамотно и логически стройно изложить теоретический материал; продемонстрировать умения самостоятельной работы с литературой; уметь сделать достаточно обоснованные выводы по излагаемому материалу.
не зачтено	Обучающийся демонстрирует: незнание значительной части программного материала; не владение понятийным аппаратом дисциплины; существенные ошибки при изложении учебного материала; неумение строить ответ в соответствии со структурой излагаемого вопроса; неумение делать выводы по излагаемому материалу.

Критерии оценивания зачета с оценкой:

Оценка	Критерии оценивания
отлично	Выставляется, если обучающимся правильно и полностью раскрыто содержание материала в пределах программы, чётко и правильно даны определения и раскрыто содержание понятий, точно использованы научные и технические термины, в ответе использованы ранее приобретённые теоретические знания, сделаны необходимые выводы и обобщения
хорошо	Выставляется, если обучающимся раскрыто основное содержание материала в пределах программы, даны определения и раскрыто содержание понятий, в ответе использованы ранее приобретённые теоретические знания, сделаны необходимые выводы и обобщения, но присутствуют незначительные нарушения в последовательности изложения, имеются одна-две неточности в содержании ответа
удовлетворительно	Выставляется, если обучающимся содержание учебного материала изложено фрагментарно, не всегда последовательно, не даны определения, не раскрыто содержание понятий, или они изложены с ошибками, допускаются ошибки и неточности в использовании научной терминологии, отсутствуют выводы и обобщения из предыдущего материала, или возможны ошибки в их изложении
неудовлетворительно	Выставляется, если обучающимся основное содержание учебного материала не раскрыто, не даются ответы на основные вопросы, допускаются грубые ошибки в определении понятий, в использовании терминологии, отсутствуют выводы и обобщения

6. МАТЕРИАЛЫ ДЛЯ ДИАГНОСТИЧЕСКОЙ РАБОТЫ

Задания для диагностической работы должны обеспечивать оценку полностью или частично сформированных компетенций.Каждое задание должно быть привязано к тому или иному индикатору сформированности компетенций.

При формировании заданий для диагностической работы необходимо использовать тестовые задания следующих типов:

Тип задания 1. Задания закрытого типа на установление соответствия.Тип задания 2. Задания закрытого типа на установление последовательности.

Тип задания 3. Задания комбинированного типа, предполагающие выбор одного правильного ответа из предложенных

Тип задания 4. Задания комбинированного типа, предполагающие выбор нескольких ответов из предложенных

Тип задания 5. Задания открытого типа с развернутым ответом.

Все типы заданий должны быть представлены не менее одного раза.

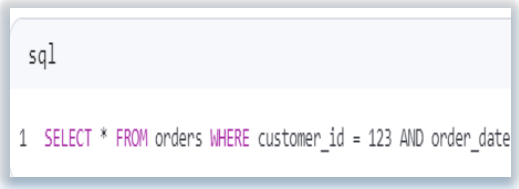
<table>
<tr><th>№ п/п</th><th>Тема занятия</th><th>Код компетенции</th><th>Индикатор</th><th>Тип задания</th><th colspan="2">Задание</th><th>Уровень освоения компетенции</th></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td>Вариант 1</td><td>Вариант 2</td><td></td></tr>
<tr><td>1</td><td>Основные понятия области баз данных и информационных систем.</td><td>ПК.15</td><td>ИПК.15.1

ИПК.15.2</td><td>1</td><td>1. Установите соответствие между уровнями архитектуры ANSI/SPARC и их описанием.
<table><tr><td>Уровень архитектуры</td><td>Описание</td></tr><tr><td>1. Внешний уровень</td><td>A. Описывает физическое хранение данных на носителе</td></tr><tr><td>2. Концептуальный уровень</td><td>B. Определяет логическую структуру всей базы данных</td></tr><tr><td>3. Внутренний уровень</td><td>C. Представляет данные с точки зрения конкретного пользователя или приложения</td></tr></table>
Правильный ответ:
1 — C, 2 — B, 3 — A
Задание 2.
Проанализируйте предложенную предметную область «Университет» (студенты, группы, преподаватели, курсы, оценки). Выделите сущности, их атрибуты и связи между ними. Постройте ER-</td><td>1. Установите соответствие между термином и его определением.
<table><tr><td>Термин</td><td>Определение</td></tr><tr><td>1. СУБД</td><td>A. Совокупность данных, организованных по определённым правилам</td></tr><tr><td>2. База данных</td><td>B. Программное обеспечение для управления базами данных</td></tr><tr><td>3. Информационная система</td><td>C. Комплекс средств, обеспечивающий сбор, обработку, хранение и выдачу информации</td></tr></table>
Правильный ответ:
1 — B, 2 — A, 3 — C

Задание 2.
Сравните файловую систему и СУБД как средства хранения данных на примере приложения «Личный архив документов». Приведите 3 преимущества и 2 недостатка каждого подхода. Сформулируйте, при</td><td>Базовый (1-3 минуты)</td></tr>
</table>

					диаграмму в нотации Чена или UML.. 2. Для предметной области «Медицинская клиника» (пациенты, врачи, приёмы, диагнозы, анализы) выделите сущности, атрибуты и связи. Спроектируйте логическую структуру данных в виде списка сущностей с их атрибутами и типами (например: Пациент(ID: int, ФИО: string, Дата_рождения: date)).	каких условиях целесообразно использовать СУБД. 2. Сравните реляционную и иерархическую структуры данных на примере хранения информации о подразделениях и сотрудниках компании. Постройте обе структуры и укажите, какая лучше подходит для частых запросов вида «Показать всех сотрудников отдела и его подотделов».	
2	Проектирование баз данных. Элементы математической теории баз данных.	ПК.15	ИПК.15.1 ИПК.15.2	2	3. Укажите правильную последовательность этапов нормализации отношений при логическом проектировании БД. A. Приведение к третьей нормальной форме (3НФ) B. Приведение ко второй нормальной форме (2НФ) C. Приведение к первой нормальной форме (1НФ) D. Приведение к нормальной форме Бойса–Кодда (НФБК) Правильный ответ: C → B → A → D 4. Дана таблица Заказы(Номер_заказа, Дата, Клиент, Товар, Цена, Категория_товара, Менеджер). – Приведите её к первой нормальной форме (1НФ). – Выявите функциональные зависимости. – Приведите к третьей нормальной форме	3. Установите правильную последовательность действий при построении ER-диаграммы в рамках концептуального проектирования. A. Определение связей между сущностями B. Выявление сущностей предметной области C. Уточнение кардинальности связей D. Определение атрибутов сущностей Правильный ответ: B → D → A → C 4. Для отношения R(A, B, C, D) заданы функциональные зависимости: A → B, B → C, C → D. Определите ключ отношения, проверьте, находится ли оно в НФБК, и если нет — выполните декомпозицию на отношения, удовлетворяющие НФБК.	**Базовый (1-3 минуты)**

					(3НФ). 5. Дана неструктурированная таблица: Сотрудники(Табельный_номер, ФИО, Отдел, Руководитель_отдела, Проект, Роль_в_проекте, Ставка) – Выявите функциональные зависимости. – Приведите структуру к третьей нормальной форме (3НФ). – Нарисуйте схему полученных отношений и укажите внешние ключи.	5. Для отношения R(A, B, C, D, E) заданы ФЗ: $A \rightarrow B$, $B \rightarrow C$, $C \rightarrow D$, $D \rightarrow E$. – Найдите замыкание $\{A\}^+$. – Определите все возможные ключи. – Проверьте, находится ли отношение в НФБК, и если нет — выполните декомпозицию на отношения, удовлетворяющие НФБК, сохранив зависимости.	
3	СУБД MySQL.	ПК.15	ИПК.15.1 ИПК.15.2 ИПК.15.3	3	6. Какой движок хранения в MySQL по умолчанию поддерживает транзакции и внешние ключи? A. MyISAM B. MEMORY C. InnoDB D. CSV Правильный ответ: C Объяснение: InnoDB — это стандартный движок хранения в современных версиях MySQL, поддерживающий ACID-транзакции, внешние ключи и восстановление после сбоев. MyISAM, напротив, не поддерживает транзакции и внешние ключи. 7. Создайте в MySQL базу данных library со следующими таблицами: – authors(id, name) – books(id, title, author_id, year, isbn)	6. Что произойдёт, если выполнить команду DELETE FROM users; без предложения WHERE? A. Команда вызовет синтаксическую ошибку B. Будут удалены все записи из таблицы users, но сама таблица останется C. Таблица users будет полностью удалена из базы данных D. Система запросит подтверждение перед удалением Правильный ответ: B Объяснение: Команда DELETE FROM без условия WHERE удаляет все строки из указанной таблицы, но структура таблицы (схема) сохраняется. Для полного удаления таблицы используется DROP TABLE. 7. Напишите SQL-запросы для базы library: 1. Вывести список всех книг с именами авторов.	**Повышенный (3-5 минут)**

					– readers(id, full_name, email) – loans(book_id, reader_id, issue_date, return_date) Настройте первичные и внешние ключи, используя движок InnoDB. Напишите SQL-скрипт, включающий создание БД, таблиц и ограничений. 8. Спроектируйте структуру базы данных online_store для интернет-магазина. Определите таблицы: – users, products, categories, orders, order_items. Для каждой таблицы укажите: – поля, – типы данных – первичные и внешние ключи. Реализуйте структуру в MySQL.	2. Найти читателей, у которых есть невозвращённые книги (return_date IS NULL). 3. Создать индекс по полю isbn в таблице books. 8. Проанализируйте запрос:  – Предложите оптимальную структуру индекса для ускорения этого запроса.	
4	Разработка клиентских приложений MySQL. Интерфейс MySQL C API.	ПК.15	ИПК.15.1 ИПК.15.2	4	9. Какие из перечисленных действий являются обязательными при корректной работе с MySQL через C API? A. Проверка возвращаемых значений функций на NULL или ошибки B. Явное освобождение памяти результата с помощью mysql_free_result() C. Использование только глобальных переменных для хранения соединения D. Закрытие соединения с помощью mysql_close() Правильные ответы: A, B, D Объяснение: Проверка ошибок, освобождение ресурсов (mysql_free_result) и закрытие соединения	9. Какие особенности характерны для клиентских приложений, использующих MySQL C API? A. Требуют линковки с библиотекой libmysqlclient B. Могут выполняться без установки MySQL Server на том же компьютере C. Автоматически генерируют графический интерфейс пользователя D. Позволяют напрямую управлять транзакциями через SQL-команды Правильные ответы: A, B, D Объяснение: Клиентское приложение на C с MySQL C API должно быть скомпоновано с	**Повышенный (3-5 минут)** **Высокий (5-10 минут)**

					(mysql_close) — обязательные шаги для предотвращения утечек памяти и неопределённого поведения. Использование глобальных переменных не требуется — соединение может быть локальным или передаваться как параметр. 10. Напишите программу на языке C, которая подключается к локальной базе library (из Темы 3), выполняет запрос SELECT title FROM books WHERE year > 2020 и выводит результат в консоль. Обработайте возможные ошибки подключения и выполнения запроса с помощью mysql_error(). 11. Спроектируйте структуру данных на языке C для хранения информации о книге (ID, название, год, ISBN). Напишите функцию, которая принимает MYSQL_RES* и заполняет массив таких структур данными из таблицы books. Обеспечьте корректное управление памятью.	libmysqlclient. Оно может подключаться к удалённому серверу MySQL, поэтому локальная установка СУБД не обязательна. Транзакции управляются через SQL (START TRANSACTION, COMMIT, ROLLBACK). Графический интерфейс не создаётся автоматически — его нужно реализовывать отдельно. 10. Модифицируйте программу которая подключается к локальной базе library так, чтобы она принимала параметр года из командной строки и безопасно вставляла его в SQL-запрос с использованием экранирования (mysql_real_escape_string()). Протестируйте на попытке SQL-инъекции. 11. Разработайте структуру данных для кэширования часто запрашиваемых записей из БД (например, справочник категорий). Реализуйте простой кэш в виде связного списка на C, который загружается один раз при старте программы и используется вместо повторных SQL-запросов.	
5	Разработка клиентских приложений MySQL. Интерфейс	ПК.15	ИПК.15.1 ИПК.15.2	5	12. Как использование интерфейса QMYSQL в рамках фреймворка Qt способствует эффективному проектированию клиентских приложений, взаимодействующих с MySQL. Какие принципы разработки ПО реализуются при этом? Ожидаемые элементы ответа:	12. Сравните подходы к разработке клиентских приложений для MySQL с использованием «голого» C API и с использованием QMYSQL в Qt. В каких ситуациях предпочтителен каждый из них? Ожидаемые элементы ответа: • C API:	**Высокий (5-10 минут)** **Высокий (5-10 минут)**

	QMYSQL для GUI-приложений MySQL.				• QMYSQL интегрирован в архитектуру Qt, что позволяет использовать единый стиль работы с данными (MVC-подход). • Классы QSqlDatabase, QSqlQuery, QSqlTableModel обеспечивают абстракцию над низкоуровневыми операциями, повышая читаемость и сопровождаемость кода. • Поддержка сигналов и слотов упрощает реакцию на изменения данных. • Реализуются принципы: ✓ Инкапсуляции — детали подключения скрыты; ✓ Разделения ответственности — логика БД отделена от GUI; ✓ Повторного использования — модели можно применять в разных частях приложения.	Плюсы: максимальная производительность, минимальные зависимости, полный контроль. Минусы: ручное управление памятью, отсутствие встроенной поддержки GUI, сложность обработки ошибок. Применение: системные утилиты, embedded-приложения без GUI, high-performance backend-компоненты. • QMYSQL (Qt): Плюсы: тесная интеграция с GUI, автоматическое управление ресурсами, поддержка MVC, кроссплатформенность. Минусы: накладные расходы фреймворка, зависимость от Qt и плагинов. Применение: десктопные GUI-приложения (админки, аналитические панели, CRM-клиенты). • С точки зрения проектирования GUI-приложений: Использование QMYSQL соответствует современным практикам — ускоряет разработку, снижает количество ошибок, обеспечивает согласованность архитектуры. Для прикладной информатики, где важна связь между интерфейсом и данными, Qt + QMYSQL — оптимальный выбор. Задание 14. Добавьте в GUI-приложение на Qt (C++), которое подключается к базе library через драйвер QMYSQL форму для добавления новой книги: поля title, year, выпадающий список авторов (загружается из таблицы authors). После сохранения данные	

<table>
<tr><td></td><td></td><td></td><td></td><td></td><td>13. Создайте пользовательскую модель данных на основе QAbstractTableModel для отображения списка студентов с дополнительными вычисляемыми полями (например, «Средний балл»). Спроектируйте внутреннюю структуру данных (например, QList<Student>), где Student — ваш класс с полями.</td><td>должны появиться в таблице без перезапуска приложения.

13. Спроектируйте структуру данных для хранения дерева подразделений компании (иерархия). Реализуйте отображение в QTreeView с использованием QStandardItemModel. Обеспечьте возможность добавления/удаления узлов и сохранения структуры в БД через самоссылающееся отношение (id, parent_id, name).</td><td></td></tr>
<tr><td>6</td><td>Некоторые вопросы общей теории баз данных. Использование баз данных.</td><td>ПК.15</td><td>ИПК.15.1

ИПК.15.2

ИПК.15.3</td><td>1</td><td>14. Установите соответствие между свойством базы данных и его описанием.<table><tr><td>Свойство</td><td>Описание</td></tr><tr><td>1. Независимость данных</td><td>A. Возможность одновременной работы нескольких пользователей без конфликтов</td></tr><tr><td>2. Целостность данных</td><td>B. Логическая и физическая структура данных не влияет на приложения</td></tr><tr><td>3. Безопасность</td><td>C. Данные удовлетворяют заданным бизнес-правилам и ограничениям</td></tr></table></td><td>14. Установите соответствие между типом использования БД и примером применения.<table><tr><td>Тип использования</td><td>Пример</td></tr><tr><td>1. OLTP (оперативная обработка транзакций)</td><td>A. Анализ продаж за последние 5 лет в BI-системе</td></tr><tr><td>2. OLAP (аналитическая обработка)</td><td>B. Регистрация нового заказа в интернет-магазине</td></tr><tr><td>3. Хранилище данных (Data Warehouse)</td><td>C. Единое хранилище исторических данных из разных</td></tr></table></td><td>Базовый (1-3 минуты)</td></tr>
</table>

4. Многопользовательский доступ	D. Защита данных от несанкционированного доступа и изменений

Правильный ответ:
1 — B, 2 — C, 3 — D, 4 — A

15. Разработайте сценарий использования базы данных в мобильном приложении «Заметки». Определите:
– Тип СУБД (встраиваемая/серверная),
– Необходимость транзакций,
– Требования к целостности и безопасности.

16. Для мобильного приложения «Личный бюджет» спроектируйте структуру данных, поддерживающую:
– категории расходов,
– транзакции (доход/расход),
– привязку к счёту.
Учтите необходимость локального хранения (SQLite) и возможную синхронизацию с облаком. Предложите формат хранения временных меток и валют.

	источников
4. Встраиваемая база данных	D. Использование SQLite в мобильном приложении для хранения заметок

Правильный ответ:
1 — B, 2 — A, 3 — C, 4 — D

15. Смоделируйте ситуацию: при одновременном редактировании одной записи двумя пользователями в веб-приложении произошла потеря изменений. Предложите два технических решения (например, optimistic/pessimistic locking) и опишите их реализацию на уровне приложения и СУБД.

16. Спроектируйте структуру данных для журналирования операций в системе (audit log): кто, что, когда изменил. Сделайте выбор между:
– хранением в отдельной таблице,
– использованием триггеров,
– ведением лога на уровне приложения.
Укажите, какие поля должны быть включены для восстановления состояния.

7	Информационные системы в сетях.	ПК.15	ИПК.15.1 ИПК.15.2	2	17. Укажите правильную последовательность этапов взаимодействия пользователя с веб-приложением, обращающимся к базе данных. A. Веб-сервер передаёт запрос серверу приложений B. Пользователь вводит данные в браузере и отправляет форму C. Сервер приложений формирует SQL-запрос и отправляет его СУБД D. Результат отображается в браузере пользователя E. СУБД выполняет запрос и возвращает данные Правильный ответ: B → A → C → E → D 18. Спроектируйте архитектуру трёхзвенной информационной системы «Электронный журнал школы»: – клиент (веб-браузер), – сервер приложений (на Python/Node.js), – СУБД (MySQL). Опишите взаимодействие компонентов при операции «Выставить оценку ученику», включая форматы передачи данных (например, JSON) и протоколы (HTTP, SQL). 19. Спроектируйте структуру данных для RESTful API сервиса «Управление задачами» (task manager). Определите: – JSON-схемы для сущностей Task, User, Project,	17. Установите правильную последовательность действий при обеспечении отказоустойчивости сетевой информационной системы. A. Настройка репликации базы данных на резервный сервер B. Размещение системы в нескольких географических зонах C. Внедрение механизма автоматического переключения (failover) D. Мониторинг доступности основных компонентов Правильный ответ: D → A → C → B 18. Реализуйте RESTful API-эндпоинт (на любом языке) для получения списка студентов группы по её ID. Эндпоинт должен: – принимать GET-запрос, – обращаться к MySQL, – возвращать JSON с данными или ошибку 404. Продемонстрируйте вызов через curl или Postman. 19. Спроектируйте структуру кэша на стороне сервера для хранения часто запрашиваемых профилей пользователей. Определите: – ключ кэша (например, user:123),	**Базовый (1-3 минуты)**

					– связи между ними, – поля для пагинации и фильтрации. Обеспечьте согласованность имён и типов (например, task_id: integer, created_at: ISO 8601 string).	– формат значения (JSON или сериализованный объект), – политику инвалидации при обновлении данных в БД.	
8	Распределенные базы данных.	ПК.15	ИПК.15.1 ИПК.15.2 ИПК.15.3	3	20. Какой из перечисленных принципов лежит в основе CAP-теоремы применительно к распределённым базам данных? A. Невозможно одновременно обеспечить согласованность, доступность и устойчивость к сетевым разделениям B. Все распределённые системы должны использовать реляционную модель данных C. Репликация всегда гарантирует полную согласованность данных D. Производительность распределённой БД всегда выше централизованной Правильный ответ: A Объяснение: CAP-теорема (теорема Брюера) утверждает, что в условиях сетевого разделения (Partition tolerance) система может обеспечить либо согласованность (Consistency), либо доступность (Availability), но не оба свойства одновременно. Это фундаментальное ограничение при проектировании распределённых систем. 21. Рассмотрите систему «Онлайн-магазин с филиалами в разных странах». – Предложите схему горизонтального шардинга данных по регионам. – Опишите, как будет обрабатываться	20. Что такое «прозрачность» в распределённых базах данных? A. Данные хранятся в открытом виде без шифрования B. Пользователь не замечает распределённой природы системы и работает с ней как с локальной C. Все узлы сети имеют равный доступ ко всем данным D. Система автоматически удаляет устаревшие реплики Правильный ответ: B Объяснение: Прозрачность — ключевой принцип распределённых БД: пользователь или приложение взаимодействует с системой так, будто она централизована. Это включает прозрачность расположения, репликации, фрагментации и сбоев, что упрощает разработку прикладного ПО. 21. Смоделируйте отказ одного узла в распределённой системе с репликацией (мастер + 2 реплики). – Опишите последовательность действий механизма failover.	**Повышенный (3-5 минут) Высокий (5-10 минут)**

					запрос «Показать все заказы клиента», если клиент делал покупки в разных регионах. – Укажите, какой тип согласованности (strong/eventual) вы выберете и почему. 22. Для системы «Глобальная социальная сеть» спроектируйте структуру данных с учётом географического шардинга: – как будут распределены пользователи по регионам, – как будет храниться информация о друзьях из разных регионов, – предложите схему глобального ID (например, UUID или составной ключ).	– Как будет восстановлен узел после возврата в строй? – Какие данные могут быть потеряны при асинхронной репликации? 22. Спроектируйте структуру данных для хранения событий в системе с eventual consistency (например, лайки, комментарии). Учтите: – необходимость идемпотентности, – временные метки, – идентификаторы источника события. Предложите формат события, позволяющий безопасно применять его в любом порядке.	

Критерии оценивания диагностической работы:

Оценка	Критерии оценивания
отлично	от 90 до 100 % правильно выполненных заданий
хорошо	от 70 до 89 % правильно выполненных заданий
удовлетворительно	от 50 до 69 % правильно выполненных заданий
неудовлетворительно	менее 50 % правильно выполненных заданий