Федеральное агентство по рыболовству
БГАРФ ФГБОУ ВО «КГТУ»
Калининградский морской рыбопромышленный колледж

Утверждаю
Врио заместителя начальника колледжа
по учебно-методической работе
М.Ю. Никишин

ДОКУМЕНТ ПОДПИСАН
ЭЛЕКТРОННОЙ ПОДПИСЬЮ

Сертификат: 00861698655D171818D836B833A86412BD
Владелец: НИКИШИН МИХАИЛ ЮРЬЕВИЧ
Действителен: с 13.02.2026 до 09.05.2027

Фонд оценочных средств

(приложение к рабочей программе профессионального модуля)

ПМ.01 РАЗРАБОТКА, АДМИНИСТРИРОВАНИЕ И ЗАЩИТА БАЗ ДАННЫХ

основной профессиональной образовательной программы среднего профессионального образования по специальности

09.02.11 Разработка и управление программным обеспечением

МО-09 02 11-ПМ.01.ФОС

РАЗРАБОТЧИК	Иванова Е.Д.
ЗАВЕДУЮЩИЙ ОТДЕЛЕНИЕМ	Судьбина Н.А.
ГОД РАЗРАБОТКИ	2026

СОДЕРЖАНИЕ

1 Паспорт фонда оценочных средств 3
1.1 Область применения фонда оценочных средств 3
1.2 Результаты освоения учебной дисциплины 3
2 Перечень оценочных средств и критерии оценивания 5
3 Оценочные средства текущего контроля и промежуточной аттестации 9
4 Сведения о фонде оценочных средств и его согласование 23

Документ управляется программными средствами 1С Колледж
Проверь актуальность версии по оригиналу, хранящемуся в 1С Колледж

1 Паспорт фонда оценочных средств

1.1 Область применения фонда оценочных средств

Фонд оценочных средств предназначен для оценки результатов освоения учебной дисциплины ПМ.01 РАЗРАБОТКА, АДМИНИСТРИРОВАНИЕ И ЗАЩИТА БАЗ ДАННЫХ.

1.2 Результаты освоения учебной дисциплины

В результате контроля и оценки по учебной дисциплине осуществляется комплексная проверка освоения следующих общих компетенций:

ПК 1.1. Проектировать базы данных

ПК 1.2. Разрабатывать и оптимизировать структуры баз данных

ПК 1.3. Администрировать базы данных

ПК 1.4. Обеспечивать резервное копирование и восстановление баз данных

ПК 1.5. Обеспечивать защиту информации и безопасность баз данных

Таблица 1- Планируемые результаты обучения

Код формируемых компетенции	**Планируемые результаты**	
	Общие	**Дисциплинарные (предметные)**
ПК1.1. Проектировать базы данных	Иметь практический опыт: - проектирования концептуальной и логической моделей баз данных с использованием специализированного ПО. Уметь: - анализировать предметную область заказчика; - разрабатывать инфологические модели данных; - применять правила нормализации отношений. Знать: - модели данных и архитектуру СУБД; - методологии и CASE-средства проектирования данных.	- Уметь выделять сущности, атрибуты и связи предметной области на основе технического задания (ТЗ); - Уметь строить ER-диаграммы (в нотациях IDEF1X, Crow's Foot или UML) без логических ошибок и дублирования данных; - Владеть навыками приведения таблиц к 1-й, 2-й и 3-й нормальным формам (1NF, 2NF, 3NF), устраняя транзитивные зависимости и аномалии; - Знать критерии выбора реляционных и нереляционных (NoSQL) подходов к хранению информации для ИС; - Владеть понятийным аппаратом реляционной алгебры и теории множеств применительно к структурам данных.
ПК 1.2. Разрабатывать и оптимизировать структуры баз данных	Иметь практический опыт: - реализации и оптимизации физической структуры баз данных на языке SQL. Уметь: - создавать таблицы, индексы, представления и схемы;	- Уметь писать синтаксически корректные SQL-скрипты для создания объектов БД с ограничениями целостности (PRIMARY KEY, FOREIGN KEY, CHECK, UNIQUE); - Владеть навыками написания сложных запросов выборки данных (SELECT) с

Код формируемых компетенции	Планируемые результаты	
	Общие	**Дисциплинарные (предметные)**
	- писать хранимые процедуры, функции и триггеры; - анализировать производительность SQL-запросов. Знать: - стандарты и синтаксис языка SQL (DDL, DML); - механизмы индексации и транзакций (ACID).	использованием соединений (JOIN), агрегатных функций и группировок (GROUP BY, HAVING); - Разрабатывать серверную логику СУБД: триггеры (до/после событий), хранимые процедуры с входными/выходными параметрами и пользовательские функции; - Уметь создавать кластеризованные и некластеризованные индексы, анализировать план выполнения запроса (Execution Plan) и устранять «узкие места» для ускорения работы БД; - Понимать уровни изолированности транзакций и методы предотвращения взаимоблокировок (Deadlocks)
ПК 1.3. Администрировать базы данных	Иметь практический опыт: - инсталляции, настройки параметров и текущего мониторинга СУБД. Уметь: - разворачивать СУБД в различных операционных системах; - управлять учетными записями и распределять права доступа. Знать: - конфигурационные файлы и параметры СУБД; - утилиты администрирования и мониторинга производительности.	- Уметь устанавливать и конфигурировать сервер СУБД (например, PostgreSQL, MS SQL Server) под технические требования проекта; - Уметь создавать роли, пользователей и группы пользователей средствами SQL и графических интерфейсов администрирования; - Уметь управлять привилегиями доступа к объектам БД с использованием операторов управления данными (GRANT, REVOKE, DENY); - Настраивать параметры кэширования, ограничения подключений и распределения оперативной памяти сервера СУБД; - Владеть инструментами мониторинга нагрузки (процессорное время, дисковые операции, активные сессии пользователей)
ПК 1.4. Обеспечивать резервное копирование и восстановление баз данных	Иметь практический опыт: - разработки и реализации планов резервного копирования данных и восстановления после сбоев. Уметь: - создавать полные, дифференциальные и инкрементные бэкапы; - моделировать ситуации отказов и проводить восстановление БД. Знать:	- Уметь разрабатывать автоматизированные регламенты резервного копирования в зависимости от критичности данных бизнес-процесса; - Уметь создавать резервные копии баз данных различных типов (Full Backup, Differential Backup, Transaction Log/WAL Backup) встроенными средствами СУБД и CLI-утилитами; - Уметь выполнять полное и частичное

Код формируемых компетенции	Планируемые результаты	
	Общие	**Дисциплинарные (предметные)**
	- стратегии и модели резервного копирования данных; - типы сбоев и методы обеспечения отказоустойчивости.	восстановление данных (Restore) на момент времени (Point-in-Time Recovery) в условиях критического сбоя; - Уметь проверять логическую и физическую целостность файлов базы данных, выявлять и устранять поврежденные страницы данных; - Знать методы репликации данных и создания зеркальных копий для обеспечения непрерывности работы ИС
ПК 1.5. Обеспечивать защиту информации и безопасность баз данных	Иметь практический опыт: - настройки политик безопасности СУБД и аудита событий информационной безопасности. Уметь: - выявлять уязвимости в архитектуре базы данных; - настраивать шифрование данных и каналов связи. Знать: - нормативно-правовые акты в сфере защиты данных; - методы защиты от несанкционированного доступа и инъекций кода.	- Уметь настраивать сквозной аудит действий пользователей (регистрация входов, изменений структуры данных, несанкционированных попыток доступа); - Владеть методами защиты базы данных от распространенных веб-угроз, включая внедрение SQL-кода (SQL Injection), путем контроля параметров и валидации запросов; - Уметь применять механизмы шифрования данных на уровне столбцов/таблиц (Transparent Data Encryption) и шифрования сетевого трафика (SSL/TLS); - Уметь разграничивать доступ к конфиденциальным данным (персональные данные, коммерческая тайна) на основе ролевой модели доступа (RBAC); - Знать требования законодательства РФ (в частности, ФЗ-152 «О персональных данных») к защите информационных систем и баз данных.

2 Перечень оценочных средств и критерии оценивания

2.2 К оценочным средствам текущего контроля успеваемости относятся:

- контрольные вопросы к темам практических занятий.

2.3 К оценочным средствам для промежуточной аттестации относятся:

- тестовые задания открытого и закрытого типа;
- билеты для подготовки к экзамену.

2.4 Критерии оценки результатов освоения учебной дисциплины

Критерии оценивания теоретических знаний:

«Отлично» — ставится, если обучающийся:

а) точно формулирует ответы на поставленные в билете/задании теоретические вопросы по проектированию, администрированию и защите баз данных;

б) дает технически правильные и точные формулировки профессиональных понятий и терминов (СУБД, транзакции, нормализация, репликация, SQL-инъекции, свойства ACID и др.);

в) демонстрирует глубокое понимание архитектуры данных, что выражается в умении аргументированно обосновать выбор конкретной модели данных или СУБД под задачи заказчика;

г) свободно обобщает и дифференцирует признаки и понятия (например, четко разграничивает SQL и NoSQL подходы, DDL и DML операторы, кластеризованные и некластеризованные индексы);

д) правильно и аргументированно отвечает на дополнительные вопросы преподавателя по смежным темам модуля;

е) свободно владеет профессиональной ИТ-терминологией, демонстрирует связанность, техническую грамотность и последовательность в изложении материала.

«Хорошо» — ставится, если обучающийся дает ответ, удовлетворяющий тем же требованиям, что и для отметки «отлично», но допускает единичные технические неточности или мелкие ошибки в формулировках стандартов SQL / принципов администрирования, которые сам же исправляет после наводящего замечания или вопроса преподавателя.

«Удовлетворительно» — ставится, если обучающийся демонстрирует знание и базовое понимание основных положений теории баз данных, но:

а) неточно и неуверенно воспроизводит ответы на поставленные в билете вопросы, путает этапы проектирования или администрирования СУБД;

б) дает расплывчатые или неточные формулировки ключевых понятий и терминов (например, ошибается в правилах нормализации отношений или уровнях изолированности транзакций);

в) затрудняется обосновать предложенную им архитектуру базы данных или выбранный метод резервного копирования;

г) затрудняется обобщить или дифференцировать признаки и понятия (например, путает функции триггеров и хранимых процедур, типы бэкапов);

д) испытывает серьезные затруднения или делает грубые ошибки при ответах на дополнительные вопросы преподавателя;

е) излагает материал недостаточно связанно и последовательно, с частыми заминками, перерывами и нарушением логики объяснения технических процессов.

«Неудовлетворительно» — ставится, если обучающийся демонстрирует полное отсутствие знаний по теоретическим основам модуля, не знает базовых терминов (база данных, первичный ключ, таблица), не может ответить ни на один из поставленных в билете вопросов, а также на дополнительные вопросы преподавателя.

Критерии оценивания практических умений:

«Отлично» — ставится, если обучающийся при выполнении заданий на ПК:

а) самостоятельно, точно и в полном объеме выполняет все пункты практического задания в среде СУБД;

б) проектирует корректную архитектуру базы данных (ER-диаграмму), полностью устраняя аномалии и избыточность данных с соблюдением правил нормализации;

в) демонстрирует написание синтаксически верного, структурированного и оптимизированного SQL-кода (DDL/DML/автоматизация), код выполняется сервером с первого раза без ошибок;

г) безошибочно реализует задачи администрирования и защиты: настраивает ролевую модель доступа (GRANT/REVOKE), создает рабочие планы резервного копирования и механизмы защиты от угроз;

д) аргументированно обосновывает алгоритм решения практической задачи, логику работы написанных триггеров/процедур и выбор типов данных;

е) уверенно ориентируется в интерфейсе СУБД и CLI-утилитах, оперативно демонстрирует работоспособность созданных объектов и отвечает на практические микро-задачи.

«Хорошо» — ставится, если обучающийся выполняет практическое задание на ПК с соблюдением тех же требований, что и для отметки «отлично», но:

а) допускает единичные синтаксические ошибки или опечатки в SQL-коде, которые приводят к сбою при первичном запуске, но самостоятельно находит их по логам СУБД и исправляет;

б) при проектировании допускает незначительные погрешности (например, не оптимально выбирает типы данных для некоторых полей или забывает указать ограничение CHECK), не нарушающие общую целостность БД;

в) пишет работоспособные SQL-запросы, но использует неоптимальные конструкции (например, избыточные подзапросы вместо JOIN), что потенциально снижает производительность;

г) выполняет задачи по бэкапам и безопасности, но требует незначительных наводящих указаний преподавателя для проверки корректности их работы.

«Удовлетворительно» — ставится, если обучающийся демонстрирует базовые практические навыки работы в СУБД, но:

а) выполняет практическое задание не в полном объеме (менее 70% от ТЗ) или со значительными задержками по времени;

б) допускает грубые ошибки в проектировании связей между таблицами (PRIMARY KEY / FOREIGN KEY), приводящие к нарушению ссылочной целостности данных;

в) пишет SQL-скрипты с многочисленными синтаксическими и логическими ошибками, затрудняется самостоятельно провести отладку кода и запустить его на сервере;

г) испытывает серьезные затруднения при администрировании СУБД (пугает синтаксис управления правами, не может настроить параметры бэкапа или создать пользователя);

д) не может объяснить логику работы собственного программного кода или алгоритма настройки безопасности, написанного с помощью подсказок;

е) неуверенно владеет инструментарием СУБД, выполняет действия хаотично, с частыми заминками.

«Неудовлетворительно» — ставится, если обучающийся не выполнил практическое задание или выполнил менее 50% требований, база данных не создана или полностью неработоспособна, SQL-код содержит неустранимые ошибки, демонстрируется полное отсутствие навыков проектирования, администрирования и защиты данных.

Критерии текущего контроля:

Шкала перевода баллов в отметки по пятибалльной системе

Отметка	Число баллов, необходимое для получения отметки
5 (отлично)	от 81-100 %
4 (хорошо)	от 61-80 %
3 (удовлетворительно)	от 41-60%
2 (неудовлетворительно)	менее 40 %

3 Оценочные средства текущего контроля и промежуточной аттестации

ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ: ПК 1.1, ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5.

Контрольные вопросы к практическим занятиям

Практическое занятие 1. Каковы ключевые отличия между концептуальной, логической и физической моделями данных?

Концептуальная модель — это высокоуровневое бизнес-описание предметной области, не зависящее от конкретной СУБД. Она содержит только основные сущности и связи между ними (описывается понятным для заказчика языком).

Логическая модель — это детализация концептуальной модели. Здесь появляются все атрибуты, определяются первичные и внешние ключи, типы связей, а данные приводятся к нормальным формам. Она создается под определенный тип СУБД (например, реляционный), но все еще абстрагирована от конкретной программы.

Физическая модель — это точная копия будущей базы данных в конкретной СУБД (например, PostgreSQL). В ней указаны специфичные для этой СУБД типы данных (например, SERIAL, UUID), прописаны физические имена таблиц, ограничения, индексы и правила распределения памяти.

Практическое занятие 2. В чем заключается разница между связями типа «один-ко-многим» (1:M) и «многие-ко-многим» (M:N) при физической реализации таблиц?

Связь «один-ко-многим» (1:M) реализуется напрямую между двумя таблицами. Для этого первичный ключ (PRIMARY KEY) родительской таблицы («один»)

добавляется в дочернюю таблицу («многие») в качестве внешнего ключа (FOREIGN KEY). Дополнительных таблиц не требуется.

Связь «многие-ко-многим» (M:N) не может быть реализована напрямую в реляционной физической модели. Она требует создания третьей — связующей (промежуточной) таблицы. В эту таблицу помещаются как минимум два поля: внешние ключи, ссылающиеся на первичные ключи обеих исходных таблиц. Вместе они чаще всего образуют составной первичный ключ связующей таблицы.

Практическое занятие 3. Каковы условия приведения таблицы к третьей нормальной форме (3NF) и какие аномалии данных при этом устраняются?

Условия приведения к 3NF:

1. Таблица должна находиться во второй нормальной форме (2NF) (все не ключевые поля зависят от полного первичного ключа, а не от его части).

2. В таблице должны отсутствовать транзитивные зависимости — ни одно не ключевое поле не должно зависеть от другого не ключевого поля (все не ключевые поля зависят только от первичного ключа).

Устраняемые аномалии:

Аномалия вставки (INSERT): невозможность добавить информацию об одном объекте, пока не создана запись о другом (например, нельзя добавить отдел ИТ, пока в него не нанят хотя бы один сотрудник).

Аномалия удаления (DELETE): потеря важных независимых данных при удалении связанной записи (например, при увольнении последнего сотрудника стирается вся информация о самом существовании отдела).

Аномалия обновления (UPDATE): избыточное дублирование данных приводит к тому, что при изменении одного значения (например, названия города) приходится менять его в тысячах строк, что порождает риск несоответствия данных.

Практическое занятие 4. В чем разница между операторами категорий DDL (Data Definition Language) и DML (Data Manipulation Language)? Приведите примеры.

DDL (Язык определения данных) используется для создания, изменения и удаления самой структуры базы данных (ее объектов: таблиц, индексов, схем, представлений). Примеры операторов: CREATE, ALTER, DROP.

DML (Язык манипулирования данными) используется для непосредственной работы со строками данных внутри уже созданных таблиц (выборка, добавление, изменение и удаление записей). Примеры операторов: SELECT, INSERT, UPDATE, DELETE.

Практическое занятие 5. Для чего используются ограничения целостности PRIMARY KEY, FOREIGN KEY, UNIQUE и CHECK при создании таблиц?

PRIMARY KEY (Первичный ключ) — однозначно идентифицирует каждую строку в таблице. Запрещает дублирование значений и пустые поля (NOT NULL), гарантируя уникальность записей.

FOREIGN KEY (Внешний ключ) — связывает таблицы между собой. Он указывает на первичный ключ другой таблицы, гарантируя ссылочную целостность (нельзя добавить дочернюю запись без родительской или удалить родительскую, если на нее кто-то ссылается).

UNIQUE (Уникальность) — запрещает дублирование значений в столбце (например, для email или ИНН), но в отличие от первичного ключа позволяет хранить значения NULL.

CHECK (Проверка) — накладывает логическое условие на значения в столбце (например, CHECK (Age >= 18) или CHECK (Price > 0)), блокируя запись данных, не подходящих под критерий.

Практическое занятие 6. Каковы отличия в работе операторов соединения таблиц INNER JOIN, LEFT JOIN и FULL OUTER JOIN?

INNER JOIN (Внутреннее соединение) — возвращает только те строки, для которых нашлись совпадения (строгие соответствия ключей) в обеих таблицах. Если в одной из таблиц данных по ключу нет, строка отбрасывается.

LEFT JOIN (Левое внешнее соединение) — возвращает все строки из левой (первой) таблицы и только совпавшие строки из правой. Если для строки из левой таблицы нет соответствия в правой, поля правой таблицы заполняются значениями NULL.

FULL OUTER JOIN (Полное внешнее соединение) — объединяет результаты левого и правого соединений. Возвращает абсолютно все строки из обеих таблиц. Если совпадения по ключу нет, пустые места с любой стороны заполняются значениями NULL.

Практическое занятие 7. Для чего используются агрегатные функции в связке с операторами GROUP BY и HAVING? В чем отличие HAVING от WHERE?

Назначение: Агрегатные функции (SUM, AVG, COUNT, MIN, MAX) используются для проведения математических расчетов над множеством строк. Оператор GROUP BY разбивает строки на группы по определенному признаку, чтобы применить к ним эти функции (например, посчитать среднюю зарплату по каждому отделу).

Отличие HAVING от WHERE:

Оператор WHERE фильтрует исходные строки до того, как они будут сгруппированы. Он не может работать с результатами агрегатных функций (запрос типа WHERE SUM(price) > 100 вызовет ошибку).

Оператор HAVING фильтрует уже готовые сгруппированные результаты после отработки GROUP BY и может использовать агрегатные функции (например: GROUP BY department HAVING AVG(salary) > 50000).

Практическое занятие 8. В чем заключаются основные архитектурные отличия хранимой процедуры от пользовательской функции в СУБД?

Возвращаемое значение: Функция обязана возвращать результат (скалярное значение, массив или таблицу). Процедура не обязана ничего возвращать (выполняется как набор команд).

Управление транзакциями: Внутри хранимой процедуры можно управлять транзакциями (вызывать команды COMMIT и ROLLBACK). Внутри функции управлять транзакциями запрещено.

Контекст вызова: Функцию можно вызывать внутри обычных SQL-запросов (например: SELECT my_func(price) FROM products). Процедуру нельзя вызвать внутри SELECT, она запускается обособленно специальной командой (CALL procedure_name() в PostgreSQL или EXEC procedure_name в T-SQL).

Практическое занятие 9. Для каких целей создаются триггеры и чем отличаются триггеры типа BEFORE (или INSTEAD OF) от триггеров типа AFTER?

Триггеры нужны для автоматического выполнения заданного кода при наступлении событий изменения данных (INSERT, UPDATE, DELETE). Они обеспечивают сложную бизнес-логику, аудит и расширенную валидацию.

Отличия по типам:

BEFORE — срабатывает до физической записи изменений в таблицу. Позволяет модифицировать или проверить вводимые значения прямо «на лету» до того, как они попадут на диск.

AFTER — срабатывает после успешной записи данных в таблицу. Используется для каскадных обновлений в других таблицах, логирования действий или отправки уведомлений, когда факт вставки/изменения уже подтвержден.

INSTEAD OF — срабатывает вместо запланированного действия. Чаще всего применяется для обновления сложных не редактируемых представлений (Views).

Практическое занятие 10. Что такое индекс в СУБД, каков принцип его работы и в каких случаях создание индекса может замедлить работу с базой данных?

Индекс — это отдельная служебная структура (обычно в виде B-Дерева), привязанная к столбцу таблицы, которая работает как алфавитный указатель в книге. Вместо полного последовательного сканирования всей таблицы (Sequential Scan), СУБД мгновенно находит адрес нужной строки по индексу (Index Scan), резко ускоряя операции чтения данных (SELECT).

Индексы замедляют операции модификации данных — INSERT, UPDATE, DELETE. При каждом изменении, добавлении или удалении строки СУБД вынуждена одновременно перестраивать и обновлять структуру самого индекса на диске. Также избыток индексов тратит оперативную память и дисковое пространство.

Практическое занятие 11. Что означает аббревиатура ACID в контексте транзакций СУБД? Опишите кратко каждое свойство.

Атомарность: Транзакция выполняется по принципу «всё или ничего». Если хоть одна команда внутри транзакции завершилась сбоем, вся транзакция отменяется (ROLLBACK), возвращая базу в исходное состояние.

Согласованность: Транзакция переводит базу данных из одного корректного (согласованного) состояния в другое. Все правила, ограничения (CHECK, внешние ключи) должны строго соблюдаться по завершении транзакции.

Изолированность: Параллельно выполняющиеся транзакции не должны влиять друг на друга. Результаты транзакции невидимы для других пользователей до тех пор, пока она не будет зафиксирована (COMMIT).

Долговечность: Если транзакция завершена успешно (COMMIT), внесенные изменения записываются на диск гарантированно. Никакие последующие сбои питания или крушение сервера не могут уничтожить эти данные.

Практическое занятие 12. Каковы различия между ролевой моделью доступа и индивидуальными правами пользователей? С помощью каких SQL-операторов они настраиваются?

При индивидуальных правах привилегии (чтение, запись) выдаются вручную конкретному пользователю. Это неудобно при расширении штата, так как администрировать каждого сотрудника приходится отдельно.

При ролевой модели (RBAC) права привязываются к созданной абстрактной роли (например, роль Operator, роль Manager). Затем пользователи просто назначаются членами этой роли и мгновенно наследуют все ее права. Это стандарт автоматизации администрирования.

• SQL-операторы: GRANT (выдать права/роль), REVOKE (отозвать права), DENY (явный запрет доступа, имеет наивысший приоритет в некоторых СУБД, например MS SQL Server).

Практическое занятие 13. Чем отличаются друг от друга полный (Full), дифференциальный (Differential) бэкапы и бэкап журнала транзакций (Transaction Log / WAL)?

Полный бэкап (Full): Копирует абсолютно все данные и структуру базы данных. Является основой любой стратегии, но занимает много места и требует много времени на создание.

Дифференциальный бэкап (Differential): Копирует только те данные, которые изменились с момента создания последнего полного бэкапа. Занимает меньше места и создается быстрее, чем полный.

Бэкап журнала транзакций (Transaction Log / WAL в PostgreSQL): Копирует только историю выполненных сервером команд (логов) с момента последнего бэкапа любого типа. Занимает минимум места, создается мгновенно и позволяет восстановить базу с точностью до секунды.

Практическое занятие 14. Что такое восстановление базы данных на определенный момент времени (Point-in-Time Recovery) и какие файлы для этого необходимы?

Это процесс восстановления базы данных до точного хронологического момента в прошлом (например, на состояние вчера в 14:05:12). Применяется, если пользователь случайно удалил критическую таблицу или произошел сбой.

Необходимые файлы:

1. Последний полный бэкап, сделанный до указанного момента времени.

2. Все последовательные файлы журналов транзакций (Transaction Logs / WAL), накопленные в период времени от этого полного бэкапа до целевой секунды сбоя.

Практическое занятие 15. Что представляет собой угроза типа «SQL-инъекция» (SQL Injection) и какие методы защиты от нее применяются на уровне СУБД и кода?

Атака, при которой злоумышленник внедряет вредоносный SQL-код в текстовые поля ввода веб-приложения. Из-за отсутствия проверки этот код напрямую передается СУБД и выполняется, позволяя хакеру украсть данные, обойти авторизацию или удалить базу.

Методы защиты:

Использование параметризованных (подготовленных) запросов (Prepared Statements): СУБД жестко разделяет тело запроса и данные. Вводимый пользователем текст воспринимается строго как строка, а не как исполняемая команда.

Использование хранимых процедур: Скрытие прямой структуры таблиц и работа через защищенный программный интерфейс СУБД.

Принцип наименьших привилегий: Ограничение прав учетной записи, из-под которой веб-приложение подключается к СУБД (например, запрет веб-пользователю выполнять команду DROP TABLE).

Ответы полностью готовы к интеграции в ФОС. Что м

Тестовые задания для текущего контроля

Задания открытого типа

Тема 1.1. Язык структурированных запросов

1. Какой результат вернет оператор SELECT COUNT(*) при выборке из таблицы, в которой часть строк содержит значения NULL в ключевых столбцах, и почему?

2. Опишите пошаговый порядок логического выполнения СУБД основных предложений в стандартном запросе выборки (от FROM до LIMIT/OFFSET).

3. Напишите, в каких случаях для объединения запросов следует использовать оператор UNION, а в каких — UNION ALL, указав разницу в их производительности.

Тема 1.2. NoSQL базы данных

4. Опишите основные архитектурные отличия документоориентированных баз данных (например, MongoDB) от классических реляционных СУБД.

5. Для решения каких прикладных задач наиболее эффективно применять графовые и K-V (ключ-значение) NoSQL-хранилища?

6. Сформулируйте суть теоремы CAP (теоремы Брюера) применительно к распределенным NoSQL-системам.

Тема 1.3. Программирование на стороне сервера (SQL+)

7. Опишите алгоритм работы и область применения курсоров в серверном программировании (PL/pgSQL или T-SQL).

8. Объясните назначение служебных виртуальных таблиц NEW и OLD (или INSERTED и DELETED) в теле триггера при обработке транзакций.

9. В каких случаях разработчику базы данных необходимо принудительно генерировать пользовательские исключения (RAISE EXCEPTION / THROW) внутри серверного кода?

Тема 1.4. Администрирование и управление доступом

10. Каким образом с помощью языка SQL реализовать наследование привилегий одной группы пользователей другой группой?

11. В чем заключается различие между системными привилегиями (на уровне сервера) и объектными привилегиями (на уровне конкретных таблиц или схем)?

Тема 1.5. Оптимизация, масштабирование и безопасность

12. Каким образом использование покрывающих индексов (Covering Indexes) позволяет СУБД избегать фазы обращения к страницам данных таблицы (Heap Fetch)?

13. Опишите суть проблемы «грязного чтения» (Dirty Read) и укажите минимальный уровень изолированности транзакций, который её предотвращает.

Тема 2.1. Установка и настройка сервера системы управления базами данных

14. Какие ключевые параметры конфигурационного файла (например, postgresql.conf или my.cnf) отвечают за объем выделяемой оперативной памяти под кэш данных и рабочие буферы?

15. Опишите порядок действий системного администратора при переносе директории хранения файлов базы данных (data directory) на другой физический диск сервера без потери информации.

Тема 2.2. Управление доступом к базам данных

16. Сформулируйте назначение файла конфигурации клиентской аутентификации (например, pg_hba.conf) и опишите принцип валидации подключений по IP-адресам.

17. Каким образом реализуется политика безопасности на уровне отдельных строк таблицы (Row-Level Security — RLS)?

Тема 2.1. Резервное копирование и восстановление данных в штатном режиме

18. Сформулируйте техническое различие между «горячим» (Hot Backup) и «холодным» (Cold Backup) резервным копированием промышленной СУБД.

19. Каким образом механизм упреждающей записи в журнал (Write-Ahead Logging — WAL) гарантирует восстановление согласованного состояния БД при внезапном отключении электропитания?

Тема 2.3. Мониторинг и журналирование событий, возникающих в процессе функционирования баз данных

20. Какие метрики производительности сервера СУБД необходимо отслеживать в первую очередь для обнаружения ситуации нехватки дискового ввода-вывода (I/O Bottleneck)?

21. Для чего в СУБД настраивается журналирование медленных запросов (Slow Query Log) и какая информация извлекается из этих логов для анализа?

Тема 2.4. Обеспечение безопасной работы сервера системы управления базами данных

22. Опишите технический процесс настройки шифрования трафика между клиентским приложением и сервером СУБД с использованием протокола SSL/TLS.

23. Какие меры безопасности необходимо предпринять на уровне операционной системы сервера для защиты файлов баз данных от прямого копирования злоумышленником?

Задания закрытого типа

Тема 1.1. Язык структурированных запросов (SQL)

1. Какое значение вернет встроенная функция COUNT(имя_столбца), если в таблице из 10 строк этот столбец содержит 3 значения NULL, 5 уникальных значений и 2 дубликата?

а) 5

б) 7

в) 10

г) Ошибку синтаксиса

2. Какое предложение в операторе SELECT выполняется СУБД самым первым по порядку логической обработки запроса?
а) FROM
б) WHERE
в) SELECT
г) ORDER BY

Тема 1.2. NoSQL базы данных
3. Какая модель представления данных является базовой для СУБД MongoDB?
а) Графовая
б) Ключ-значение (Key-Value)
в) Документоориентированная (BSON/JSON)
г) Табличная (широкие столбцы)

.

Тема 1.3. Программирование на стороне сервера (SQL+)
4. В каких триггерах доступны для изменения и чтения служебные переменные NEW и OLD (или таблицы inserted и deleted)?
а) Только в триггерах на оператор SELECT
б) В триггерах на операции INSERT, UPDATE, DELETE
в) Только при удалении таблиц (DROP)
г) В любых хранимых процедурах

Тема 1.4. Администрирование и управление доступом
5. Какой SQL-оператор используется для безвозвратного отзыва ранее выданных прав у пользователя или роли?
а) DENY
б) DROP USER
в) REVOKE
г) ALTER ROLE

Тема 1.5. Оптимизация, масштабирование и безопасность
6. Какой уровень изолированности транзакций полностью защищает от аномалии «фантомного чтения» (Phantom Read)?
а) Read Uncommitted
б) Read Committed
в) Repeatable Read
г) Serializable

Тема 2.1. Установка и настройка сервера системы управления базами данных
7. Какой параметр в конфигурационном файле postgresql.conf отвечает за объем памяти, используемой сервером для кэширования данных таблиц и индексов?
а) work_mem
б) max_connections
в) shared_buffers
г) maintenance_work_mem

Тема 2.2. Управление доступом к базам данных
8. Какая запись в файле pg_hba.conf разрешает локальное подключение к серверу PostgreSQL через доменные сокеты без пароля?

а) local all all trust
б) host all all 127.0.0.1/32 md5
в) local all all md5
г) host all all 0.0.0.0/0 reject

Тема 2.3. Резервное копирование и восстановление данных в штатном режиме
9. Какое резервное копирование копирует исключительно изменения, произошедшие с момента создания последнего ПОЛНОГО бэкапа?
а) Инкрементное (Incremental)
б) Дифференциальное (Differential)
в) Зеркальное (Mirror)
г) Логический дамп (Logical dump)

Тема 2.4. Мониторинг и журналирование событий, возникающих в процессе функционирования баз данных
10. Для чего в промышленной СУБД включается логирование slow queries (медленных запросов)?
а) Для архивации старых данных
б) Для автоматического удаления зависших сессий
в) Для выявления неоптимальных запросов, требующих создания индексов
г) Для шифрования пользовательских паролей

Тема 2.5. Обеспечение безопасной работы сервера системы управления базами данных
11. Какая мера является наиболее эффективной защитой базы данных от атак типа SQL-инъекция (SQL Injection) на уровне кода?
а) Регулярное создание полных бэкапов
б) Перенос сервера СУБД на другой порт
в) Использование параметризованных запросов (Prepared Statements)
г) Установка сложного пароля суперпользователя

12. Выберите ВСЕ операторы, относящиеся к категории DDL (Data Definition Language) языка SQL (выберите несколько вариантов):
а) CREATE
б) UPDATE
в) ALTER
г) DROP
д) INSERT

13. Какие типы ограничений (Constraints) гарантируют уникальность значений в столбцах реляционной таблицы? (выберите несколько вариантов):
а) PRIMARY KEY
б) FOREIGN KEY
в) UNIQUE
г) CHECK

14. Какие данные необходимы для реализации технологии восстановления Point-in-Time Recovery (PITR)? (выберите несколько вариантов):
а) Базовый полный бэкап (Base backup)
б) Структура NoSQL каталогов

в) Непрерывный архив журналов транзакций (WAL / Transaction logs)
г) Текстовый файл slow-логов

15. Какие действия НЕЛЬЗЯ выполнять внутри пользовательских функций (Functions) в большинстве современных реляционных СУБД? (выберите несколько вариантов):
а) Вызывать оператор SELECT
б) Выполнять команду фиксации транзакции COMMIT
в) Выполнять команду отката транзакции ROLLBACK
г) Использовать циклы и ветвления

Перечень практических заданий к промежуточной аттестации

1.Напишите SQL-скрипт (DDL) для создания таблиц Товары (id, название, цена, остаток_на_складе) и Заказы (id, товар_id, количество, дата_заказа). Наложите ограничения: первичные и внешние ключи, а также CHECK на то, что цена товара и количество в заказе должны быть строго больше нуля.

2.Напишите триггер и триггерную функцию, которая при добавлении новой записи в таблицу Заказы автоматически уменьшает остаток на складе соответствующего товара. Если товара на складе недостаточно, триггер должен вызывать исключение (RAISE EXCEPTION / THROW) и отменять транзакцию.

3.Создайте роль Sales_Manager. Выдайте ей права на чтение обеих таблиц и добавление записей в таблицу Заказы, но запретите удаление (DELETE) данных.

4. Создайте таблицы Отделы (id, название_отдела) и Сотрудники (id, фио, оклад, отдел_id). Заполните их тестовыми данными (3 отдела, 5 сотрудников). Напишите SQL-запрос, который выводит название отдела и средний оклад его сотрудников, но только для тех отделов, где средний оклад превышает 50 000 рублей.

5.Напишите запрос для создания некластеризованного индекса для оптимизации поиска сотрудников по их ФИО. Опишите, с помощью какой команды (в выбранной вами СУБД) вы можете проверить, задействует ли сервер данный индекс при выборке.

Пример задания для дифференцированного зачета

Теоретическая часть

Вопрос 1. В каком регистре (DML или DDL) выполняется команда DROP TABLE clients;?

Вопрос 2. Какое значение вернет запрос SELECT COUNT(phone) FROM users;, если в таблице 10 строк, но у 3 пользователей поле phone содержит NULL?

Вопрос 3. К какому типу NoSQL СУБД (ключ-значение, документно-ориентированные, графовые, колоночные) относится Redis?

Вопрос 4. Внутри триггерной функции PostgreSQL какая служебная переменная содержит строку данных в том виде, в каком она была до выполнения операции UPDATE?

Вопрос 5.Допишите пропущенное ключевое слово в SQL-скрипте, отзывающем права: _____ SELECT ON orders FROM manager_role;

Вопрос 6.Какая аномалия конкурентных транзакций происходит, когда одна транзакция читает данные, измененные другой, еще не зафиксированной (COMMIT) транзакцией?

Вопрос 7. В каком конфигурационном файле PostgreSQL настраивается максимальное количество одновременных подключений к серверу (max_connections)?

Вопрос 8. Какой тип резервного копирования сохраняет только те изменения данных, которые произошли с момента последнего дифференциального или полного бэкапа?

Вопрос 9. Как называется встроенный механизм СУБД, который фиксирует SQL-запросы, время выполнения которых превысило заданный лимит времени (например, log_min_duration_statement)?

Вопрос 10. Как называется атака на информационную систему, при которой вредоносный код внедряется в тело SQL-запроса через незащищенные поля ввода веб-приложения?

Практическая часть

Задание 1. Напишите SQL-код и запустите его на ПК.
Напишите SQL-запрос, выводящий название отдела (departments.name) и количество работающих в нем сотрудников из таблицы employees. В выборку должны попасть даже те отделы, в которых нет сотрудников (использовать LEFT JOIN и GROUP BY).

Задание 2. Напишите SQL-скрипт создания таблицы products (id, title, price), где для поля price задано ограничение CHECK, запрещающее устанавливать цену менее или равную 0.

Задание 3. Создайте роль app_developer и предоставьте ей права на добавление (INSERT) и обновление (UPDATE) записей только в таблице products.

Задание 4. Напишите синтаксис терминальной команды операционной системы для восстановления базы данных shop_db из ранее созданного файла резервной копии backup.sql из-под пользователя postgres.

Задание 5. Создайте триггер и триггерную функцию tg_check_stock(). При попытке вставки новой строки в таблицу sales (id, product_id, quantity) триггер должен проверять, достаточно ли товара в таблице warehouse (product_id, stock_quantity). Если товара на складе меньше, чем указано в sales.quantity, триггер должен выдать

ошибку RAISE EXCEPTION с текстом 'Недостаточно товара на складе!' и заблокировать транзакцию.

Пример задания для экзамена

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 1

ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

Вопрос 1. Реляционная модель данных и теория нормализации. Понятие избыточности данных. Первая, вторая и третья нормальные формы (1NF, 2NF, 3NF). Аномалии обновления, удаления и вставки данных, методы их ликвидации.

Вопрос 2. Транзакции в СУБД. Понятие транзакции и её свойства (концепция ACID). Уровни изолированности транзакций по стандарту ANSI SQL (Read Uncommitted, Read Committed, Repeatable Read, Serializable). Проблемы конкурентного доступа к данным.

Вопрос 3. Информационная безопасность СУБД. Понятие и механизм реализации атаки типа SQL-инъекция (SQL Injection). Принципы защиты баз данных: параметризация запросов, ролевая модель доступа (RBAC), шифрование данных и каналов связи (SSL/TLS).

ПРАКТИЧЕСКАЯ ЧАСТЬ

Задание 1. На основе предметной области «Аренда недвижимости» напишите SQL-скрипт (DDL) для создания двух связанных таблиц: `Квартиры` (`id`, `адрес`, `стоимость_сутки`) и `Аренда` (`id`, `квартира_id`, `дата_начала`, `количество_дней`).
Наложите ограничения:
- Первичные ключи для обеих таблиц.
- Внешний ключ в таблице `Аренда` с каскадным удалением при удалении квартиры (`ON DELETE CASCADE`).
- Ограничение `CHECK`, контролирующее, что стоимость аренды за сутки строго больше нуля.
Напишите SQL-запрос, который выводит адреса квартир, общая сумма дохода от аренды которых за все время превысила 100 000 рублей.

Задание 2. Выполните комплексную настройку безопасности и отказоустойчивости на сервере PostgreSQL:
1. Создайте роль доступа `auditor_role` с правами только на чтение (`SELECT`) таблицы `Аренда`.
2. Создайте пользователя `secure_user` с паролем 'P@ssw0rd2026' и назначьте ему роль `auditor_role`.
3. Напишите в консоли операционной системы команду с использованием утилиты `pg_dump`
для создания резервной копии схемы созданной вами базы данных в файл `security_export.sql`.

4 Сведения о фонде оценочных средств и его согласование

Фонд оценочных средств для аттестации по учебному модулю ПМ.01 «Разработка, администрирование защита баз данных» представляет собой компонент основной образовательной программы среднего профессионального образования по 09.02.11 «Разработка и управление программным обеспечением».

Фонд оценочных средств рассмотрен и одобрен на заседании методической комиссии «Информационных систем и программирования»

Протокол № 9 от «20» мая 2026 г.

Председатель методической комиссии _____/Т.Н.Богатырева /

Документ управляется программными средствами 1С Колледж
Проверь актуальность версии по оригиналу, хранящемуся в 1С Колледж