ЦЕНТРАЛЬНЫЙ УНИВЕРСИТЕТ

УТВЕРЖДЕНА

Приказом ректора
АНО ВО «Центральный университет»
от «02» июня 2026 г. № 0602.72

Рабочая программа дисциплины (модуля) «SQL и базы данных» дополнительной профессиональной программы – программы профессиональной переподготовки «Управление продуктами»

ДОКУМЕНТ ПОДПИСАН
ЭЛЕКТРОННОЙ ПОДПИСЬЮ

Сертификат: 0215B0C000D1B3FEB44A37EE631E0C27D4
Владелец: Ивашкевич Евгений Васильевич
Действителен: с 13.01.2026 до 13.04.2027

Электронный документ

Москва
2026

Содержание

1. **Краткая характеристика дисциплины (модуля)** 3
2. **Тематический план** 4
3. **Содержание дисциплины (модуля)** 5
4. **Учебно-методическое обеспечение** 6
5. **Материально-техническое обеспечение** 6
6. **Методические и оценочные материалы** 8

1. Краткая характеристика дисциплины (модуля)

Изучение дисциплины (модуля) «SQL и базы данных» является ключевым для понимания того, как эффективно управлять, хранить и извлекать данные, что критически важно в современном мире, где информация играет центральную роль в принятии решений. Освоение SQL позволяет специалистам анализировать большие объемы данных, оптимизировать процессы и разрабатывать надежные решения для бизнеса и науки.

Цель изучения дисциплины (модуля): освоение эффективного использования языка SQL и технологий баз данных для решения прикладных задач хранения, обработки и анализа данных в бизнес-среде.

Задачи изучения дисциплины (модуля):

- освоить синтаксис SQL и типы данных в современных СУБД, включая Postgres;
- научиться подключаться к базам данных и выполнять операции выборки, фильтрации и агрегации;
- освоить методы продвинутой аналитики: оконные функции, вложенные запросы, CTE;
- понять принципы реляционной модели, нормализации и проектирования схем;
- приобрести опыт создания витрин данных, оптимизации запросов и автоматизации регулярных выгрузок.

В результате освоения дисциплины (модуля) обучающийся должен:

знать:

- синтаксис SQL, типы данных, встроенные функции;
- принципы написания запросов в СУБД Postgres;
- методы продвинутой аналитики с использованием SQL;
- теория реляционных баз данных;
- основы реляционных хранилищ данных.

уметь:

- исследовать структуры данных;
- подключаться к реляционным базам данных и выполнять SQL-команды;
- использовать реляционные операции и вложенные запросы;
- вычислять метрики с помощью SQL, применять агрегацию;
- использовать оконную агрегацию и условные вычисления;
- создавать таблицы и представления;
- писать скрипты для регулярного обновления витрин данных;
- оптимизировать SQL-запросы;
- выполнять нормализацию схемы базы данных;
- читать ER-модели.

владеть:

- навыками подготовки наборов данных по заданным требованиям с помощью SQL;
- способами обработки данных в реляционных СУБД;
- практикой моделирования данных.

2. Тематический план

№ п/п	Наименование раздела дисциплины (модуля)	Трудоемкость, академические часы				ТКУ (текущий контроль успеваемости)
		Очная форма				
		Аудиторная работа		Контроль	Самостоятельная работа	
		Лекции	Семинары (практические занятия)			
1	Введение в базы данных и SQL. Простые запросы	4	4		18	Тест Тренажер Домашние задания
2	Агрегация данных	2	2		10	Тест Тренажер Домашние задания
3	Соединение и объединение данных	4	4		18	Тест Тренажер Домашние задания
4	Решение задач с помощью SQL	2	2		10	Тест Тренажер Домашние задания
5	Оконные функции	4	4	2	18	Контрольная работа
6	Оптимизация запросов	2	3		10	Тест Тренажер Домашние задания
7	SQL DDL и моделирование данных	4	4		17	Тест Тренажер Домашние задания
8	Хранилища данных и витрины	4	4		10	Тест Тренажер Домашние задания
9	Прикладные задачи аналитики в SQL	2	3		17	Тест Тренажер Домашние задания
	Экзамен			2		
	Итого:	***28***	***30***	***4***	***128***	
Объем дисциплины (модуля) (в ак. ч.)		***190***				
Объем дисциплины (модуля) (в зач. ед.)		***5***				

Электронный документ

3. Содержание дисциплины (модуля)

№п/п	Наименование раздела дисциплины (модуля)	Содержание дисциплины (модуля) по темам
1	Введение в базы данных и SQL. Простые запросы	Введение в базы данных и SQL Разработка запросов к одной таблице
2	Агрегация данных	Агрегация в SQL
3	Соединение и объединение данных	Соединение данных с помощью JOIN Объединение данных с помощью UNION. Динамическая фильтрация
4	Решение задач с помощью SQL	Решение задач с помощью SQL
5	Оконные функции	Оконные функции - 1 Оконные функции - 2
6	Оптимизация запросов	Оптимизация запросов
7	SQL DDL и моделирование данных	Введение в SQL DDL. Использование представлений Моделирование данных
8	Хранилища данных и витрины	Введение в хранилища данных Разработка скриптов для заполнения витрин данных
9	Прикладные задачи аналитики в SQL	Прикладные задачи аналитики в SQL

4. Учебно-методическое обеспечение

Университет располагает полным набором лицензионного и свободно распространяемого программного обеспечения, включая продукты отечественного производства.

Каждый слушатель в течение всего периода обучения получает индивидуальный неограниченный доступ к электронно-библиотечной системе и электронной информационно-образовательной среде университета. Эти системы предоставляют возможность доступа к ресурсам из любой точки, где есть подключение к сети Интернет, как на территории университета, так и за его пределами.

Слушателям обеспечен удаленный доступ к современным профессиональным базам данных и информационным справочным системам.

Основная литература:

1. Фиайли, К. SQL. Руководство для использования с любыми SQL СУБД : учебное пособие / К. Фиайли ; пер. с англ. А. В. Хаванова. - 2-е изд. - Москва : ДМК Пресс, 2023. - 454 с. - ISBN 978-5-89818-323-3. - Текст : электронный. - URL: https://znanium.ru/catalog/product/2102610.

2. Короткевич, Д. SQL Server. Наладка и оптимизация для профессионалов : практическое руководство / Д. Короткевич. - Санкт-Петербург : Питер, 2023. - 512 с. - (Серия «Библиотека программиста»). - ISBN 978-5-4461-2332-2. - Текст : электронный. - URL: https://znanium.com/catalog/product/2123361.

Дополнительная литература:

1. Молинаро, Э. SQL. Сборник рецептов : практическое руководство / Э. Молинаро. - 2-е изд. - Санкт-Петербург : БХВ-Петербург, 2022. - 592 с. - ISBN 978-5-9775-6759-6. - Текст : электронный. - URL: https://znanium.com/catalog/product/2123362.

2. Изучаем и используем Presto. Быстрый и надежный SQL-движок для анализа данных : практическое руководство / А. Ло Дука, Т. Михан, В. Бхаратан, Ин Су ; пер. с англ. В. И. Комарова. – Москва : Book.kz, 2024. - 184 с. – ISBN 978-6-01810-343-8. - Текст : электронный. - URL: https://znanium.ru/catalog/product/2205126.

3. Маркин, А. В. Базы данных. PostgreSQL : учебник для вузов / А. В. Маркин. — Москва : Издательство Юрайт, 2026. — 828 с.

4. Маркин, А. В. Программирование на SQL : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2026. — 805 с.

5. Материально-техническое обеспечение

Университет располагает материально-технической базой, соответствующей действующим противопожарным правилам и нормам и обеспечивающей проведение всех видов дисциплинарной и междисциплинарной подготовки, практической и научно-исследовательской работ обучающихся, предусмотренных учебным планом.

Помещения, которые представляют собой учебные аудитории для проведения занятий лекционного типа, занятий семинарского (практического) типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, а также помещения для самостоятельной работы и помещения для хранения и профилактического обслуживания учебного оборудования. Помещения укомплектованы специализированной

мебелью и техническими средствами обучения, служащими для представления учебной информации большой аудитории.

Изучение дисциплины (модуля) обеспечивается в учебных аудиториях, оснащенных:

— столами и стульями;

— компьютерной техникой;

— механическими калькуляторами;

— специализированным оборудованием, включая демонстрационное оборудование.

Помещения для самостоятельной работы обучающихся, в том числе приспособленные для использования инвалидами и лицами с ограниченными возможностями здоровья, оснащены компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду Университета.

Обучающимся предоставляется доступ (в том числе удаленный) к ресурсам информационно-телекоммуникационной сети «Интернет», электронным ресурсам (в том числе электронным библиотечным системам, современным профессиональным базам данных и информационным справочным системам):

№	Наименование портала (издания, курса, документа)	Ссылка
1.	Научная электронная библиотека elibrary.ru библиотека	https://elibrary.ru/defaultx.asp
2.	База данных для IT-специалистов	https://habr.com
3.	База данных ScienceDirect	https://www.sciencedirect.com
4.	Официальный сайт Министерства науки и высшего образования Российской Федерации	https://minobrnauki.gov.ru/
5.	Федеральный портал «Российское образование"	https://www.edu.ru/
6.	Информационная система "Единое окно доступа к образовательным ресурсам"	http://window.edu.ru/
7.	Единая коллекция цифровых образовательных ресурсов	http://school-collection.edu.ru/
8.	Федеральный центр информационно - образовательных ресурсов	http://fcior.edu.ru/

Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), в том числе комплект лицензионного программного обеспечения, современные профессиональные базы данных и информационные справочные системы:

Наименование ПО	Производство	Лицензионное / свободно распространяемое
Операционные системы:		
Microsoft Imagine (Windows Client, Server)	зарубежное	лицензионное
Браузеры:		
Яндекс.Браузер	отечественное	свободно распространяемое
Google Chrome	зарубежное	свободно распространяемое
Офисные приложения:		
Microsoft Imagine (Visio, OneNote)	зарубежное	лицензионное

TeXstudio	зарубежное	свободно распространяемое
Adobe Acrobat Reader	зарубежное	свободно распространяемое
Программное обеспечение для планирования и учета времени:		
Toggle app	зарубежное	свободно распространяемое
Системы управления проектами:		
Microsoft Imagine (Project)	зарубежное	лицензионное
Системы управления базами данных:		
Microsoft Imagine (SQL Server)	зарубежное	лицензионное
Системы резервного копирования (backup):		
Acronis Backup Advanced for HyperV	зарубежное	лицензионное
Справочно-правовые системы:		
КонсультантПлюс: справочно-правовая система	отечественное	лицензионное
Средства антивирусной защиты:		
Kaspersky Endpoint Security для бизнеса Стандартный Russian Edition	отечественное	лицензионное
Среды разработки:		
Visual Studio Code	зарубежное	свободно распространяемое
Bash (Unix shell)	зарубежное	свободно распространяемое
Anaconda	зарубежное	свободно распространяемое
Robotic Operating System	зарубежное	свободно распространяемое
CopelliaSim	зарубежное	свободно распространяемое
Google Colaboratory	зарубежное	свободно распространяемое
Пакеты программных средств и библиотек:		
AutoPsy	зарубежное	свободно распространяемое
Interactive Disassembler (IDA)	зарубежное	свободно распространяемое
Системы управления библиографической информацией:		
Zotero	зарубежное	свободно распространяемое
Сервисы и службы:		
Bind	зарубежное	свободно распространяемое
Docker	зарубежное	свободно распространяемое

6. Методические и оценочные материалы

Методические указания для обучающихся по освоению дисциплины (модуля)

В процессе изучения дисциплины (модуля) «SQL и базы данных» в рамках текущего контроля успеваемости используются такие виды учебной работы, как лекции, семинары, тесты, тренажеры, домашние задания, контрольная работа, а также различные виды самостоятельной работы обучающихся по заданию преподавателя, направленные на развитие навыков профессиональной лексики, закрепление практических профессиональных компетенций, поощрение инициатив.

Лекция – систематическое, последовательное, монологическое изложение преподавателем учебного материала, как правило, теоретического характера.

В процессе лекций рекомендуется вести конспект лекций: кратко и схематично фиксировать основные идеи, выводы и обобщения лекции; выделять важные мысли,

ключевые слова и термины. Необходимо отметить вопросы или материалы, которые вызывают затруднения, и попытаться найти ответы в рекомендованной литературе. Если разобраться в материале не удается, следует сформулировать вопрос и задать его преподавателю на консультации или во время семинарского (практического) занятия.

Семинар – это форма учебной деятельности, проводимая в учебном заведении под руководством преподавателя, где слушатели активно участвуют в обсуждениях, практических заданиях и других формах взаимодействия.

Для успешной подготовки к семинару рекомендуется заранее ознакомиться с темой занятия и основными материалами, чтобы иметь возможность активно участвовать в обсуждении. Также полезно подготовить вопросы и идеи для обсуждения, что поможет глубже понять материал и продемонстрировать заинтересованность.

Тест – особая форма проверки знаний. Проводится после освоения одной или нескольких тем и свидетельствует о качестве понимания основных понятий изучаемого материала. Тестовые задания составлены к ключевым понятиям, основным разделам, важным терминологическим категориям изучаемой дисциплины.

Для подготовки к тесту необходимо знать терминологический аппарат дисциплины, понимать смысл научных категорий и уметь их использовать в профессиональной лексике. Владение понятийным аппаратом, включённым в тестовые задания, позволяет преподавателю быстро проверить уровень понимания слушателями важных методологических категорий.

Тренажер – домашние задания разного уровня сложности с возможностью самопроверки.

Тренажер позволяет оперативно оценивать усвоение материала и выявлять пробелы в знаниях через тесты и практические задачи. Такой формат способствует регулярной самопроверке и повышает мотивацию к изучению дисциплины (модуля).

Домашнее задание – набор задач по темам недели.

При работе над домашними заданиями важно внимательно ознакомиться с требованиями и сроками выполнения. Рекомендуется разбивать задания на этапы, чтобы избежать перегрузки и лучше усвоить материал, использовать различные источники информации, включая учебники и онлайн-ресурсы, для более глубокого понимания темы.

Контрольная работа – письменная работа с набором задач, которые нужно решить за ограниченное время.

Цель контрольной работы – получить специальные знания по одной или нескольким темам дисциплины и продемонстрировать навыки их практического применения.

Самостоятельная работа – работа слушателей, направленная на углубленное изучение отдельных тем и вопросов учебной дисциплины (модуля).

В процессе самостоятельной работы слушатели взаимодействуют с рекомендованными материалами при минимальном участии преподавателя. Задачи слушателя включают работу с конспектами лекций (обработка текста), повторное изучение учебных материалов планов и тезисов ответов, изучение дополнительных тем, выполнение учебно-исследовательских заданий и другое.

Система оценивания результатов обучения по дисциплине (модулю)

Оценивание уровня учебных достижений обучающихся по дисциплине (модулю) осуществляется в виде текущего контроля успеваемости и промежуточной аттестации.

Промежуточная аттестация по дисциплине (модулю) осуществляется в форме ***экзамена***.

Для оценивания текущего контроля успеваемости и промежуточной аттестации используется десятибалльная шкала оценивания, которая соотносится с традиционной пятибалльной шкалой следующим образом, если иное не указано в силлабусе дисциплины:

Десятибалльная оценка	**Пятибалльная оценка**	**Общая характеристика результата обучения по дисциплине (модулю)**
10	Отлично	Слушатель полностью владеет знаниями, изложенными в рабочей программе, и глубоко осмысляет дисциплину (модуль). Он самостоятельно и логически последовательно отвечает на все вопросы, акцентируя внимание на наиболее важном. Умеет анализировать, сравнивать, классифицировать, обобщать, конкретизировать и систематизировать изученный материал, выделяя ключевые моменты и устанавливая причинно-следственные связи. Четко формулирует ответы, уверенно интерпретирует результаты анализов и других исследований, а также решает сложные задачи. Слушатель хорошо знаком с методами исследования, необходимыми для практической деятельности, и умеет связывать теоретические аспекты дисциплины (модуля) с практическими задачами.
9	Отлично	
8	Отлично	
7	Хорошо	Слушатель обладает знаниями предмета почти в полном объеме рабочей программы и самостоятельно, логически последовательно и всесторонне отвечает на все вопросы, акцентируя внимание на наиболее значимых моментах. Он умеет анализировать, сравнивать, классифицировать, обобщать, конкретизировать и систематизировать изученный материал, выделяя его ключевые аспекты и устанавливая причинно-следственные связи. Формулирует свои ответы, уверенно интерпретирует результаты анализов и других исследований, а также решает сложные ситуационные задачи. Слушатель хорошо знаком с методами исследования, необходимыми для практической деятельности, и умеет связывать теоретические аспекты предмета с практическими задачами.
6	Хорошо	
5	Удовлетворительно	Слушатель обладает базовыми знаниями по дисциплине (модулю), но испытывает трудности при самостоятельных ответах и использует неточные формулировки. В ходе ответов он допускает ошибки, касающиеся сути вопросов. Слушатель способен решать только самые простые задачи и владеет лишь минимальным набором методов
4	Удовлетворительно	

Десятибалльная оценка	Пятибалльная оценка	Общая характеристика результата обучения по дисциплине (модулю)
		исследования.
3	Не сдан	Слушатель не овладел обязательным минимумом знаний по предмету и не может ответить на вопросы, даже если преподаватель задает дополнительные наводящие вопросы.
2	Не сдан	
1	Не сдан	

Дисциплина (модуль) «SQL и базы данных» оценивается следующим образом:

Активность	Вес	Количество	Описание
Тесты	10%	11	Форма проверки знаний слушателей через ответы на вопросы с выбором из нескольких вариантов ответов
Тренажеры	10%	10	Домашние задания разного уровня сложности с возможностью самопроверки
Домашние задания	30%	11	Набор задач по темам недели
Контрольная работа	20%	1	Письменная работа с набором задач, которые нужно решить за ограниченное время
Экзамен	30%	1	Письменная или устная работа над заданием, направленным на проверку полученных знаний и навыков по дисциплине (модулю)

Формула расчёта итоговой оценки по дисциплине (модулю) «SQL и базы данных»: «0,1 × среднее за тесты + 0,1 × среднее за тренажеры + 0,3 ×среднее за домашние задания + 0,2 × контрольная работа + 0,3 × экзамен».

Текущий контроль успеваемости обучающихся по дисциплине (модулю)

Примерные задания для теста

Тест 1. Введение в SQL и простые запросы

1. Какой оператор используется для выборки данных из таблицы?
   A) INSERT
   B) UPDATE
   C) SELECT
   D) DELETE
   Ответ: C
2. Какой оператор позволяет отфильтровать строки по условию?
   A) GROUP BY
   B) ORDER BY
   C) WHERE
   D) HAVING
   Ответ: C
3. Какой тип данных подходит для хранения даты и времени?
   A) VARCHAR
   B) INTEGER
   C) TIMESTAMP

D) BOOLEAN
Ответ: C

4. Какой оператор сортирует результат запроса?
   A) SORT BY
   B) ORDER BY
   C) GROUP BY
   D) FILTER BY
   Ответ: B
5. Как выбрать все столбцы из таблицы **orders**?
   A) SELECT * FROM orders;
   B) GET ALL FROM orders;
   C) SELECT ALL orders;
   D) SELECT orders.*;
   Ответ: A

Тест 2. Агрегация и соединение данных

1. Какая функция вычисляет сумму значений в столбце?
   A) COUNT
   B) AVG
   C) SUM
   D) MAX
   Ответ: C
2. Какой оператор используется для группировки строк?
   A) FILTER BY
   B) GROUP BY
   C) JOIN BY
   D) ORDER BY
   Ответ: B
3. Какой тип JOIN возвращает все строки из левой таблицы и совпадающие — из правой?
   A) INNER JOIN
   B) RIGHT JOIN
   C) FULL JOIN
   D) LEFT JOIN
   Ответ: D
4. Какой оператор используется для фильтрации групп?
   A) WHERE
   B) HAVING
   C) FILTER
   D) GROUP FILTER
   Ответ: B
5. Что делает UNION?
   A) Объединяет таблицы по строкам
   B) Объединяет по столбцам
   C) Умножает таблицы
   D) Сортирует данные

Ответ: A

Тест 3. Оконные функции и оптимизация

1. Какая функция нумерует строки в окне без пропусков?
   A) RANK()
   B) ROW_NUMBER()
   C) DENSE_RANK()
   D) NTILE()
   Ответ: B
2. Что делает оконная функция AVG(amount) OVER (PARTITION BY region)?
   A) Считает среднее значение amount для каждого региона
   B) Суммирует все значения
   C) Считает общее среднее по всей таблице
   D) Группирует по датам
   Ответ: A
3. Какой индекс ускорит поиск по столбцу user_id?
   A) Кластерный индекс
   B) Хэш-индекс
   C) Полный индекс
   D) Временный индекс
   Ответ: A
4. Что делает CTE (Common Table Expression)?
   A) Создаёт временную таблицу в рамках запроса
   B) Удаляет дубликаты
   C) Обновляет данные
   D) Сохраняет результат в базе данных
   Ответ: A
5. Какой оператор используется для создания представления?
   A) CREATE TABLE
   B) CREATE VIEW
   C) CREATE INDEX
   D) CREATE QUERY
   Ответ: B

Примерные задания для тренажеров

Тренажер 1.

Задача 1.

Создай таблицу со следующей информацией о клиентах:

- customer_id — ID клиента (первичный ключ, тип данных int);
- fio — ФИО клиента (не может быть пустым, тип данных VARCHAR);
- avg_rating — средняя оценка, которую оставил данный клиент (может принимать значения от 0 до 5, тип данных float).

Таблицу назвать customer_x_ratings.

Задача 2.

Заполни созданную в первом задании таблицу данными из таблиц shop_customers и

shop_reviews.
После чего выведи топ-20 клиентов по среднему рейтингу (в случае равного среднего рейтинга отсортируй по ФИО).

Задача 3.

Добавь в созданную ранее таблицу столбец lifetime (INTERVAL), представляющий собой разницу между текущей датой и датой создания клиента в БД (поле created_at).
После чего заполни таблицу, используя данные из учебной БД. В завершении также выведи топ-20 клиентов по среднему рейтингу.

Задача 4.

1. Создай таблицу, содержащую следующую информацию:
   - category_id — ID категории, является первичным ключом, тип данных int;
   - category_name — название категории, тип данных VARCHAR;
   - product_cnt — число товаров в категории, тип данных int (и добавь ограничение, что значение больше 0);
   - avg_price — средняя цена товара в категории, тип данных float.

Таблицу назвать category_info.

2. Добавь в созданную таблицу строку:
   - category_id = 1;
   - category_name = Телефоны;
   - product_cnt = 10;
   - avg_price = 49523,12.

Задача 5.

1. Создай таблицу, содержащую следующую информацию:
   - product_id — ID продукта, является первичным ключом, тип данных int;
   - product_name — название товара, тип данных VARCHAR;
   - category_name — название категории, тип данных VARCHAR;
   - orders_cnt — число заказов, тип данных int (добавь ограничение, что поле не может быть отрицательным).

Таблицу необходимо назвать product_info.

2. Заполни таблицу данными из таблиц shop_products, shop_categories и shop_orders.
3. Удали информацию по категории «Ноутбуки».

Тренажер 2.

Задача 1.

Условие задачи

Найти всех покупателей, которые возвращали заказы в 2024 году, и для каждого клиента подсчитать количество и сумму возвращённых заказов. Ограничьтесь только теми покупателями, которые произвели строго больше одного возврата (т.е. было больше 1 order_id со статусом 'Refunded').
Выведите топ-3 покупателя, у которых сумма заказов была больше всех.

Результирующие столбцы

1. *customer_id* - идентификатор пользователя.
2. *cnt_orders* - количество заказов (в штуках).
3. *total_amount* - сумма заказов (в рублях).

Задача 2.

Условие задачи

Для каждого месяца из таблицы *databases_and_sql.shop_orders* посчитай среднюю сумму заказа.
Округли результаты до двух знаков после запятой и отсортируй по возрастанию даты.
Ограничься только первыми шестью месяцами 2023 года.

Результирующие столбцы

1. *dt* - месяц совершения заказа (в формате даты).
2. *avg_amount* - средняя сумма заказа, округлённая до двух знаков после запятой.

Задача 3.

Условие задачи

При работе с данными важно понимать, какие поля имеют пропущенные значения.
Для каждого столбца в таблице *databases_and_sql.shop_orders* посчитай, сколько в нём **заполненных** значений, и выведи результаты в одну строку.
Используй свойства оператора *COUNT*.

Результирующие столбцы

1. *order_id_not_nulls* - кол-во не нулловых значений поля order_id.
2. *customer_id_not_nulls* - кол-во не нулловых значений поля customer_id.
3. *order_date_not_nulls* - кол-во не нулловых значений поля order_date.
4. *order_status_not_nulls* - кол-во не нулловых значений поля order_status.
5. *delivery_date_not_nulls* - кол-во не нулловых значений поля delivery_date.
6. *total_amount_not_nulls* - кол-во не нулловых значений поля total_amount.
7. *shipping_address_id_not_nulls* - кол-во не нулловых значений поля shipping_address_id.

(В результирующей таблице будет только одна строка)

Задача 4.

Условие задачи

По таблице *databases_and_sql.shop_orders* для каждого года заказа найди общее количество заказов и количество доставленных заказов (т.е. заказов с с указанной датой доставки), а также конверсию из заказа в доставку (т.е. отношение вышеуказанных полей).
Не забудь привести конверсию к формату *float*.
Используй свойства оператора *COUNT*.
Отсортируй результат по возрастанию даты.

Результирующие столбцы

1. *dt* - год заказа (в формате даты).
2. *cnt_orders* - всего заказов.
3. *cnt_delivered_orders* - заказов с датой доставки.
4. *share_delivered* - доля доставленных заказов.

Задача 5.

Условие задачи

Для каждого клиента рассчитай количество заказов, которые он сделал за всё время, а также долю успешных заказов (т.е. со статусом Completed).
Ограничься только теми клиентами, которые совершили строго больше 3 заказов за всё время, а также хотя бы один заказ без статуса Completed.
Выведи топ-3 клиента по доле заказов со статусом Completed среди всех заказов клиента.

Результирующие столбцы

1. *customer_id* - идентификатор пользователя

2. *cnt_orders* - количество заказов пользователя.
3. *share_compl* - доля заказов со статусом Completed.

Задача 6.

Условие задачи

По таблице *databases_and_sql.shop_orders* найди два значения:

- среднее количество дней между датой доставки и датой заказа;
- среднее количество дней между датой доставки и датой заказа при условии, что в случае NULL-значения даты доставки его нужно заменить на {дата заказа + 3 дня}.

Округли значения до трёх знаков после запятой.

Результирующие столбцы

1. *general_avg* - среднее количество дней.
2. *coalesce_avg* - скорректированное среднее количество дней.

Задача 7.

Условие задачи

Для каждого месяца заказа рассчитай общее количество заказов, количество заказов с доставкой больше (строго) трёх дней, а также долю заказов с доставкой дольше 3 дней среди всех заказов.

Ограничься только вторым полугодием 2024 года.

Долю необходимо округлить до 3 знаков после запятой.

Результирующие столбцы

1. *dt* - месяц заказа (в формате даты).
2. *cnt_orders* - количество заказов.
3. *cnt_long_orders* - количество заказов с доставкой больше 3 дней.
4. *share_long* - доля заказов с доставкой больше 3 дней из всех заказов.

Задача 8.

Условие задачи

Для каждого календарного месяца (т.е. месяца в числовом формате) выведи количество заказов, количество уникальных клиентов, а также рассчитай среднее количество заказов на одного клиента.

Выведи три самых "занятых" месяца, т.е. когда среднее количество заказов на клиента наибольшее.

Результирующие столбцы

1. *dt* - месяц заказа (в числовом формате).
2. *cnt_orders* - количествло заказов.
3. *cntd_customers* - количество уникальных клиентов.
4. *cnt_per_cust* - среднее количество заказов на одного клиента.

Задача 9.

Условие задачи

Таблица *databases_and_sql.shop_orderitems* отражает связь заказа (*order_id*) и товаров в заказе (*product_id*).

Найди такие заказы, в которых было три или более одинаковых товаров, и выведи идентификатор заказа, товара и количество товаров.

Отранжируйте результат по возрастанию идентификатора заказа.

Результирующие столбцы

1. *order_id* - идентификатор заказа.

2. *product_id* - идентификатор продукта.
3. *cnt_products* - количество продуктов в заказе.

Задача 10.

Условие задачи

Для каждого месяца заказа за 2024 год из
таблицы *databases_and_sql.shop_orders* определи:

- минимальное количество дней между датой доставки и датой создания заказа;
- максимальное количество дней между датой доставки и датой создания заказа;
- среднее количество дней между датой доставки и датой создания заказа;
- медианное количество дней между датой доставки и датой создания заказа;
- разницу между средним и медианным значением.

Округли все значения до целых и отсортируй по возрастанию даты.

Результирующие столбцы

1. *dt* - месяц заказа.
2. *min_diff* - минимальное количество дней.
3. *max_diff* - максимальное количество дней.
4. *avg_diff* - среднее количество дней.
5. *median_diff* - медианное количество дней.
6. *avg_median_diff* - разница между средним и медианным значением.

Тренажер 3.

Задача 1.

Клиенты магазина совершают покупки, в результате чего формируются заказы. У каждого заказа есть уникальный номер order_id, время покупки order_date, общая сумма заказа order_total.

К заказам можно писать отзывы. Отзыв тоже имеет уникальный идентификатор review_id. Вместе с текстом review_text сохраняется автор user_login, время записи review_ts и номер заказа.

Таблица, в которой менеджеры регистрируют заказы, имеет следующую структуру:

SALES (order_id, order_date, order_total, review_id, user_login, review_text, review_ts)

Что требуется в ответе:

Напиши DDL-скрипт (create table …), с помощью которого можно будет создать нормализованную базу данных (3NF). Обязательно указывай первичный ключ.

Используй только названия полей из задания. Список полей приводится ниже:

- order_id,
- order_date,
- order_total,
- review_id,
- user_login,
- review_text,
- review_ts.

Задача 2.

Изучи описание ниже и попробуй составить список таблиц, которые можно использовать в базе данных интернет-магазина.

Описание:

Как только появляется новый заказ, ему назначается новый уникальный номер (order_id), определяется итоговая сумма к оплате (order_total), время создания (order_date). В заказе может быть несколько товаров. Если при этом в заказ включается несколько одинаковых товаров, то соответствующее количество записывается в поле qty.

У каждого товара есть цена за единицу (unit_price), название (product_desc).

Можно оставлять отзывы к товарам. При этом у каждого отзыва есть уникальный номер (review_id), текст (review_text), автор (user_login), а также время отправки отзыва (review_ts).

Сейчас данные хранятся в таблицах со следующей структурой:

Orders (order_id, order_date, order_total, product_id, product_desc, unit_price, qty)

ProductReviews (product_id, review_id, user_login, review_text, review_ts)

Тебя попросили исправить структуру таблиц.

Что требуется в ответе:

Напиши DDL-скрипт, с помощью которого можно будет создать нормализованную базу данных (3NF). Используй только названия полей из задания.

Задача 3.

Задача на проектирование базы данных для приложения Random Coffee, которое в случайном порядке формирует пары или тройки людей для проведения неформальных встреч.

Краткое описание работы сервиса Каждую неделю запускается новый раунд встреч, каждый из которых имеет две даты: дату открытия приглашения пользователей отметить согласие на участие и дату публикации пар или троек пользователей.

Пользователи регистрируются в приложении Random Coffee. В личном кабинете они могут включить или выключить своё участие. Каждый пользователь во время регистрации указывает своё имя и email. Информация о созданных парах (или тройках) в новом раунде встреч доступна в интерфейсе.

Перед каждым раундом сервис открывает запись и отправляет приглашение на участие во встречах. Пользователи отмечают своё согласие или отказ участвовать в раунде.

Затем, когда наступает дата публикации групп для встреч, программа формирует набор предложений. В одном раунде они нумеруются, начиная с 1. Также записывается время создания предложения и предполагаемые участники встречи. Обычно участников два, но иногда может быть и больше. Эта информация доступна пользователям в интерфейсе сервиса.

Раунд может быть помечен как закрытый или открытый. Это нужно для корректного отображения текущего раунда встреч.

Что требуется в ответе:

Разработай скрипт DDL для создания базы данных, которую можно будет использовать в описанном выше сервисе. Ты можешь воспользоваться ERDplus для проектирования ER-модели с последующим переходом к SQL DDL.

Можно выбирать только поля из списка ниже:

- user_id — номер пользователя;
- user_name — имя пользователя;
- email — адрес электронной почты;
- active_flg — признак активности или доступности;
- round_id — номер раунда;
- open_date — дата открытия сбора заявок на участие;

- publish_date — дата публикации предложений;
- meeting_no — номер встречи (предложения);
- created_at — время создания;
- confirm_flg — признак подтверждения.

Вставь скрипт для проверки своего решения и запусти ячейку.

Примерные домашние задания

Домашнее задание 1. Анализ продаж с агрегацией

1. Дана таблица sales с полями: order_id, product_id, user_id, amount, order_date, region.
2. Найдите общую сумму продаж.
3. Посчитайте количество уникальных пользователей, совершивших покупки.
4. Выведите сумму продаж по каждому региону.
5. Рассчитайте средний чек (средняя сумма заказа).
6. Определите день с максимальной выручкой.
7. Добавьте комментарии к каждому запросу.
8. Предоставьте результаты в виде таблиц и краткий вывод.

Домашнее задание 2. Работа с JOIN и оконными функциями

1. Даны таблицы:
   - users (user_id, name, signup_date)
   - orders (order_id, user_id, amount, order_date)
2. Выберите всех пользователей и количество их заказов (включая тех, у кого заказов нет).
3. Для каждого заказа выведите имя пользователя и накопленную сумму его покупок на момент заказа.
4. Пронумеруйте заказы каждого пользователя по дате (с начала и далее).
5. Найдите пользователей, сделавших более 3 заказов.
6. Используйте CTE для улучшения структуры запроса.
7. Добавьте пояснения к каждому шагу.

Домашнее задание 3. Построение витрины и оптимизация

1. Создайте витрину sales_summary_monthly со следующими полями:
   - month, region, total_sales, avg_check, new_users_count
2. Напишите SQL-скрипт, который обновляет витрину за предыдущий месяц.
3. Добавьте индекс на поле order_date в таблице orders для ускорения запросов.
4. Используйте команду EXPLAIN для анализа плана выполнения и предложите 2 способа оптимизации.
5. Выделите справочник products из таблицы sales и выполните нормализацию до 3НФ.
6. Оформите всё в виде SQL-скрипта с комментариями.

Примерные задания для контрольной работы

1. Опишите основные концепции реляционных баз данных и объясните назначение языка SQL в контексте работы с данными. Приведите пример простого SELECT-запроса к одной таблице для извлечения всех записей о клиентах с указанием их имен и email.
2. Напишите SQL-запрос к таблице "Продукты", который извлекает названия и цены

товаров, отсортированные по цене в убывающем порядке, с использованием операторов WHERE и ORDER BY.

3. Используя агрегирующие функции, составьте запрос к таблице "Заказы", который вычисляет общее количество заказов и среднюю сумму заказа по каждому клиенту.

4. Напишите запрос с использованием INNER JOIN для соединения таблиц "Клиенты" и "Заказы" с целью получения списка клиентов и их заказов, включая имена клиентов и даты заказов.

5. Создайте запрос с помощью UNION для объединения результатов двух таблиц: "Продукты_2023" и "Продукты_2024", чтобы получить уникальный список названий продуктов, и добавьте динамическую фильтрацию по цене выше 1000.

6. Решите задачу: на основе таблиц "Сотрудники" и "Департаменты" определите департаменты, где средняя зарплата сотрудников превышает 50000, используя подзапросы и агрегацию.

7. Напишите запрос с оконной функцией ROW_NUMBER() для ранжирования сотрудников по зарплате в рамках каждого департамента из таблицы "Сотрудники".

8. Проанализируйте и предложите оптимизацию для медленного запроса, который объединяет три таблицы и использует подзапросы; объясните, как использование индексов и EXPLAIN может улучшить производительность.

9. Опишите процесс создания таблицы с помощью DDL-команд (CREATE TABLE) для хранения информации о пользователях, включая первичный ключ и внешние ключи. Приведите пример создания представления (VIEW) на основе этой таблицы.

10. Решите прикладную задачу: используя оконные функции и агрегацию, проанализируйте продажи по месяцам в таблице "Продажи", вычислив кумулятивную сумму продаж и процентное изменение по сравнению с предыдущим месяцем.