

УТВЕРЖДЕНА

Решением Ученого совета
АНО ВО «Центральный университет»
«24» июня 2025 г.
Протокол № 2

**Рабочая программа дисциплины (модуля)
«SQL и базы данных»**

Направление подготовки: 02.04.01 Математика и компьютерные науки

Направленность (профиль) подготовки: Продуктовая аналитика

Квалификация (степень) выпускника: магистр

Форма обучения: очная

Срок освоения программы: 2 года

Год набора: 2025

**Москва
2025**

Содержание

1. Краткая характеристика дисциплины (модуля)	3
2. Перечень планируемых результатов обучения.....	4
3. Тематический план.....	6
4. Содержание дисциплины (модуля).....	6
5. Учебно-методическое обеспечение	8
6. Материально-техническое обеспечение	8
7. Методические и оценочные материалы	10

1. Краткая характеристика дисциплины (модуля)

Рабочая программа дисциплины (модуля) «SQL и базы данных» составлена в соответствии с федеральным государственным образовательным стандартом высшего образования – магистратура по специальности 02.04.01 Математика и компьютерные науки, профиль Продуктовая аналитика, утвержденный приказом Министерства науки и высшего образования Российской Федерации № 810 от 23.08.2017 года.

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.

Место дисциплины (модуля) в структуре образовательной программы

Настоящая дисциплина (модуль) включена в учебный план по программе подготовки магистратуры по направлению 02.04.01 Математика и компьютерные науки, профиль Продуктовая аналитика и входит в обязательную часть Блока 1.

Дисциплина (модуль) изучается на 1 курсе в 1 семестре.

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):

- изучить принципы проектирования реляционных баз данных и нормализации данных;
- освоить методы оптимизации запросов для повышения производительности работы с большими объемами данных;
- научиться реализовывать сложные запросы с использованием различных типов соединений и подзапросов;
- развить умение автоматизировать обработку данных с помощью написания и отладки SQL-скриптов;
- приобрести навыки моделирования данных для построения эффективных и масштабируемых баз данных.

В результате освоения дисциплины (модуля) обучающийся должен:

знать:

- базовый синтаксис SQL;
- типы данных SQL и способы их преобразования;
- агрегирующие функции;
- основные функции для работы с датами, числами и текстом.

уметь:

- агрегировать данные;
- соединять таблицы;
- работать со множествами;
- использовать подзапросы;
- писать оконные функции;
- создавать таблицы, представления, индексы;
- писать скрипты для регулярного обновления витрин данных.

владеть:

- навыками решения бизнес-задач с помощью SQL;
- навыками оптимизации SQL запросов;
- навыками моделирования данных.

2. Перечень планируемых результатов обучения

Компетенции, формируемые в результате освоения дисциплины (модуля) при проведении учебных занятий в форме контактной работы обучающихся с педагогическими работниками Университета и в форме самостоятельной работы обучающихся:

Компетенция	Содержание компетенции	Индикатор компетенции	Перечень планируемых результатов обучения по дисциплине (модулю)
ОПК-3.	Способен самостоятельно создавать прикладные программные средства на основе современных информационных технологий и сетевых ресурсов, в том числе отечественного производства	ОПК-3.1.	Знает основные принципы программирования, архитектуры программного обеспечения и современные языки программирования, а также особенности отечественных информационных технологий и сетевых ресурсов
		ОПК-3.2.	Умеет разрабатывать прикладные программные средства, используя современные инструменты и технологии, а также интегрировать их с сетевыми ресурсами для решения конкретных задач
		ОПК-3.3.	Имеет практический опыт разработки программных средств, используемых при построении математических моделей в естественных науках
ПК-3.	Способен решать задачи профессиональной деятельности в области продуктовой аналитики, формулировать результаты анализа и выявлять последствия полученных данных для принятия обоснованных решений и оптимизации продуктов	ПК-3.1.	Знает методы и инструменты продуктовой аналитики
		ПК-3.2.	Умеет применять аналитические инструменты и программное обеспечение для обработки и визуализации данных, а также формулировать выводы на основе проведенного анализа
		ПК-3.3.	Имеет опыт работы над реальными проектами в области продуктовой аналитики, включая анализ пользовательского поведения и оптимизацию продуктов на основе полученных данных
ПК-6.	Способен разрабатывать программное обеспечение для решения прикладных задач в сфере продуктовой аналитики	ПК-6.1.	Знает основные языки программирования, методы разработки программного обеспечения, а также принципы проектирования и архитектуры программных систем, применяемых в продуктовой аналитике

		ПК-6.2.	Умеет анализировать прикладные задачи, разрабатывать алгоритмы и реализовывать их в виде программного обеспечения, используя современные инструменты и технологии, а также проводить тестирование и отладку созданных решений
		ПК-6.3.	Имеет практический опыт разработки программного обеспечения в рамках реальных проектов, включая участие в командах, где были успешно реализованы решения для конкретных прикладных задач

3. 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	Тренажер, Подготовка к семинару
2	Агрегация данных	2	2		10	Тренажер, Подготовка к семинару
3	Соединение и объединение данных	4	4		18	Тренажер, Подготовка к семинару
4	Решение задач с помощью SQL	2	2		10	Тренажер, Подготовка к семинару
5	Оконные функции	4	4		18	Тренажер, Подготовка к семинару, Контрольная работа
6	Оптимизация запросов	2	2		10	Тренажер, Подготовка к семинару
7	SQL DDL и моделирование данных	4	4		18	Тренажер, Подготовка к семинару
8	Хранилища данных и витрины	4	4		10	Тренажер, Подготовка к семинару
9	Прикладные задачи аналитики в SQL	2	2		18	Тренажер, Подготовка к семинару
	<i>Зачет с оценкой</i>			4		
	Итого:	28	28	4	130	
	Объем дисциплины (модуля) (в ак. ч.)	190				
	Объем дисциплины (модуля) (в зач. ед.)	5				

4. 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

5. Учебно-методическое обеспечение

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

Каждый студент 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.

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

Основная литература:

1. Файли, К. SQL. Руководство для использования с любыми SQL СУБД : учебное пособие / К. Файли ; 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. Настройка 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. Молиаро, Э. SQL. Сборник рецептов : практическое руководство / Э. Молиаро. - 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>.

6. 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

7. 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, аудиторная работа, тренажеры, 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 удастся, 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 студенты 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.

Аудиторная работа – активная работа студента на семинаре/практическом занятии, его ответы на вопросы преподавателя и участие в дискуссии.

Для успешного участия в семинаре/практическом занятии студентам рекомендуется заранее ознакомиться с темой обсуждения, прочитать необходимые материалы и подготовить вопросы. Важно активно слушать и вовлекаться в дискуссию, высказывая свои мнения и аргументируя их. При ответах на вопросы преподавателя стоит быть уверенным, четким и логичным, опираясь на изученный материал. Также полезно поддерживать диалог

с однокурсниками, чтобы обогатить обсуждение и расширить свои знания.

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).

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.

Бонусные баллы – это оценки, которые студенты могут получить за выполнение дополнительных заданий.

Формат бонусных баллов позволяет студентам улучшить общую оценку по дисциплине (модулю) и стимулирует углубленное изучение материала.

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

В процессе самостоятельной работы студенты взаимодействуют с рекомендованными материалами при минимальном участии преподавателя. Задачи студента 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)

Критерии получения уровня и оценивания сформированности компетенций по дисциплине (модулю) «SQL и базы данных»

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 *зачета с оценкой*, при этом проводится оценка компетенций, сформированных по дисциплине.

Для оценивания текущего контроля успеваемости и промежуточной аттестации используется десятибалльная шкала оценивания, которая соотносится с традиционной пятибалльной шкалой следующим образом:

Десятибалльная оценка	Пятибалльная оценка	Оценка за зачет	Общая характеристика результата обучения по дисциплине (модулю)
10	Отлично	Зачтено	Студент 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
9	Отлично	Зачтено	
8	Отлично	Зачтено	

Десятибалльная оценка	Пятибалльная оценка	Оценка за зачет	Общая характеристика результата обучения по дисциплине (модулю)
			моменты и устанавливая причинно-следственные связи. Четко формулирует ответы, уверенно интерпретирует результаты анализов и других исследований, а также решает сложные задачи. Студент хорошо знаком с методами исследования, необходимыми для практической деятельности, и умеет связывать теоретические аспекты дисциплины (модуля) с практическими задачами.
7	Хорошо	Зачтено	Студент 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. Студент хорошо знаком с методами исследования, необходимыми для практической деятельности, и умеет связывать теоретические аспекты предмета с практическими задачами.
6	Хорошо	Зачтено	
5	Удовлетворительно	Зачтено	Студент 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. Студент способен решать только самые простые задачи и владеет лишь минимальным набором методов исследования.
4	Удовлетворительно	Зачтено	
3	Не сдан	Не зачтено	Студент не овладел обязательным минимумом знаний по предмету и не может ответить на вопросы, даже если преподаватель задает дополнительные наводящие вопросы.
2	Не сдан	Не зачтено	
1	Не сдан	Не зачтено	

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

Активность	Вес	Количество	Описание
Аудиторная работа	30%	14	Активное участие в семинарах: отвечать на вопросы преподавателя и задавать свои
Тренажер	20%	14	Домашние задания разного уровня сложности с возможностью самопроверки
Контрольная работа	20%	1	Письменная работа с набором задач, которые нужно решить за ограниченное время
Зачет с оценкой	20%	1	Письменная или устная работа над заданием, направленным на проверку полученных знаний и навыков по дисциплине (модулю)

В рамках изучения дисциплины (модуля) возможно получение бонусных баллов.

Формула расчёта итоговой оценки по дисциплине (модулю) «SQL и базы данных»: « $0,3 \times$ аудиторная работа + $0,2 \times$ среднее за тренажеры + $0,2 \times$ контрольная работа + $0,3 \times$ зачет с оценкой».

Текущий контроль успеваемости обучающихся по дисциплине (модулю)

Примерные задания для подготовки к семинарам

Подготовка к семинару 1.

Задача 1.

Вывести из таблицы продуктов (`shop_products`) все уникальные пары «Название продукта» — «Цена».

Задача 2.

Тебе необходимо подготовить выборку для обзвона клиентов. В выборке нужно вывести из таблицы клиентов (`shop_customers`) следующую информацию:

- ID клиента (`customer_id`);
- имя клиента (`first_name`);
- номер телефона (`phone_number`).

Задача 3.

Вывести все столбцы из таблицы платежей (`shop_payments`).

Задача 4.

Вывести все уникальные статусы платежей из таблицы `shop_payments`. Задать в операторе FROM alias для таблицы.

Подготовка к семинару 2.

Задача 1.

По таблице `databases_and_sql.shop_orders` посчитать общее количество заказов, сумму заказов и количество уникальных клиентов при условии, что заказ совершён в ноябре 2024 года или позже.

Что требуется в ответе

Ответ должен содержать колонки:

- 1) `cnt_orders` = количество заказов;
- 2) `total_amount` = сумма заказов;
- 3) `cntd_customers` = количество уникальных клиентов.

Задача 2.

По таблице `databases_and_sql.shop_products` найти общее количество товаров, их суммарную стоимость, а также количество уникальных категорий товаров при условии, что товар добавлен в 2022 году.

Что требуется в ответе

Ответ должен содержать колонки:

- 1) `cnt_products` = количество товаров;
- 2) `total_price` = суммарная стоимость;
- 3) `cntd_categories` = количество уникальных категорий.

Задача 4.

По таблице `databases_and_sql.shop_payments` посчитать минимальную, среднюю и максимальную сумму платежа при условии, что платёж был совершён с помощью карты не ранее 1 июня 2024 года.

Что требуется в ответе

Ответ должен содержать колонки:

- 1) `min_amount` = минимальная сумма платежа;
- 2) `avg_amount` = средняя сумма платежа;
- 3) `max_amount` = максимальная сумма платежа.

Задача 5.

По таблице `databases_and_sql.shop_orders` посчитать среднюю и медианную сумму заказа при условии, что заказ не был отменён. Округлить значения до целых.

Что требуется в ответе

Ответ должен содержать колонки:

- 1) `avg_amount` = средняя сумма заказа;
- 2) `median_amount` = медианная сумма заказа.

Задача 6.

По таблице `databases_and_sql.shop_payments` посчитать среднюю сумму платежа со статусом 'Success', а также 10-й и 90-й перцентиль суммы платежа со статусом 'Success'. Округлить значения до целых.

Что требуется в ответе

Ответ должен содержать колонки:

- 1) `avg_amount` = средняя сумма платежа;
- 2) `perc10` = 10-й перцентиль суммы платежа;
- 3) `perc90` = 90-й перцентиль суммы платежа.

Подготовка к семинару 3.

Задача 1.

По таблице `databases_and_sql.shop_customers` определить клиентов, которые совершали заказ за последние 3 месяца.

Что требуется в ответе

Ответ должен содержать колонки:

- ID клиента,
- имя,
- фамилия.

Задача 2.

1) Определить для клиентов из таблицы `databases_and_sql.shop_customers` их адрес из таблицы `databases_and_sql.shop_addresses`. Если у клиента нет адреса, то информацию о нём выводить не нужно.

Что требуется в ответе

Ответ должен содержать колонки:

- ID клиента,
- имя,
- фамилия,
- адрес клиента.

2) Посчитать число уникальных адресов на клиента.

Что требуется в ответе

Ответ должен содержать колонки:

- id клиента,

- число его адресов.

Задача 3.

Определить число заказов для клиентов из таблицы *databases_and_sql.shop_customers*, которые за последние 3 месяца совершили более одного заказа.

Что требуется в ответе

Ответ должен содержать колонки:

- ID клиента,
- имя,
- фамилия,
- число заказов клиента.

Задача 4.

Определить клиентов из таблицы *databases_and_sql.shop_customers*, которые оставляли отзывы за последние полгода (таблица *databases_and_sql.shop_reviews*). Также рассчитай число отзывов, оставленных клиентами, и среднюю оценку, которую они ставили.

Что требуется в ответе

Ответ должен содержать колонки:

- ID клиента,
- имя
- фамилия,
- число отзывов,
- средняя оценка.

Подготовка к семинару 4.

Задача 1.

Из таблицы *databases_and_sql.shop_products* выведи все ID продуктов, их названия, цену, а также среднюю цену по всем продуктам и разницу между ценой товара и средней ценой. Оставь только продукты, цена которых выше среднего значения.

Что требуется в ответе

Ответ должен содержать колонки:

- *product_id*;
- *name*;
- *price*;
- *avg_price* = средняя цена по всем продуктам;
- *diff* = разница между ценой продукта и средней ценой всех продуктов.

Задача 2.

По таблице *databases_and_sql.shop_payments* для каждого метода оплаты посчитай количество платежей, общее количество платежей по всем методам, а также долю этого метода платежа от всех платежей.

Что требуется в ответе

Ответ должен содержать колонки:

- *payment_method*;
- *cnt_payments* = количество платежей для метода оплаты;
- *cnt_all* = общее количество платежей по всем методам оплаты;
- *pct* = доля количества платежей для метода оплаты от общего количества платежей по всем методам.

Задача 3.

Выведи названия категорий и среднюю цену товаров в них при условии, что средняя цена выше 100 000.

Что требуется в ответе

Ответ должен содержать колонки:

- name;
- avg_price = средняя цена товаров категории.

Задача 4.

Выведи фамилии и имена клиентов, а также общую сумму их заказов при условии, что сумма превышает 1 000 000. Отсортируй по убыванию суммы.

Что требуется в ответе

Ответ должен содержать колонки:

- first_name;
- last_name;
- sum_amount = общая сумма заказов клиента.

Подготовка к семинару 5.

Задание 1.

Выделите топ-5 продуктов, которые имеют наибольший относительный вклад по суммарной выручке внутри своей категории.

Что требуется в ответе

Ответ должен содержать колонки:

1. name - название продукта
2. share_in_cat - процентный вклад данного продукта в выручку по его категории

В ответе должно быть 5 строк.

Задание 2.

Пронумеруй все заказы в рамках каждого клиента по времени, однако нумерация должна начинаться только после того, как пользователь совершил две покупки со статусом *Success*.

Что требуется в ответе

Выведи строки результирующей таблицы, соответствующие клиенту с *customer_id = 116*.

Ответ должен содержать колонки:

- order_id;
- customer_id;
- order_date;
- total_amount;
- rn — номер заказа в рамках клиента.

Задание 3.

Постройте динамику суммарных выплат по дням.

Постройте скользящее среднее 3 видов:

- MA(3): За 3 последних наблюдения
- MA(10): За последние 5 и следующие 5 наблюдений
- Месячное скользящее среднее: За последние 15 и следующие 15 наблюдений.

Дайте колонкам названия в вашем SQL-запросе:

- payment_date - дата платежа
- sum_amt - суммарные выплаты
- avg_3 - MA(3)
- avg_55 - MA(10)
- avg_month - MA(30)

Исполните ячейку ниже для визуализации своего результата.

Задание 4.

Необходимо исследовать динамику того, как пользователи уходят и возвращаются.

Будем считать, что если промежуток между покупками клиента составлял больше 90 дней, то он «впадал в спячку», после чего вернулся к нам, если совершил покупку.

Необходимо для каждой покупки каждого клиента разметить, какой она была по счёту

Электронный документ

у данного клиента, а также в данном цикле активности (то есть «спячка» обнуляет цикл активности).

Например, если пользователь X совершал покупки в даты [01.01.2023,01.01.2024,02.01.2024], то:

- общая нумерация будет [1,2,3];
- нумерация по циклам будет [1,1,2], так как между первым и вторым заказом пользователя прошло больше дней.

Что требуется в ответе

Выведи строки результирующей таблицы, соответствующие клиенту с *customer_id = 91*.

Ответ должен содержать колонки:

- **order_id**;
- **customer_id**;
- **order_date**;
- **rn_loc** — «локальный» номер/ранг в рамках покупательского цикла клиента;
- **rn_glob** — «глобальный» номер/ранг в рамках клиента.

Задание 5.

Для каждого пользователя рассчитайте, сколько раз он переходил с одного типа оплаты на другой за всё время.

Выделите топ-5 пользователей, у которых наибольшая доля "смен" среди всех покупок.

Возьмите только тех пользователей, которые совершили строго больше 5 покупок, а также имели строго больше 1 смены типа оплаты.

Что требуется в ответе

Ответ должен содержать колонки:

- **customer_id**
- **share_change** - доля "смен" среди всех покупок.

В ответе должно быть 5 строк.

Примерные задания для тренажеров

Тренажер 1.

Задача 1.

Создай таблицу со следующей информацией о клиентах:

- **customer_id** — ID клиента (первичный ключ, тип данных int);
- **fiо** — 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 нулевых значений поля `order_id`.
2. `customer_id_not_nulls` - кол-во не нулевых значений поля `customer_id`.
3. `order_date_not_nulls` - кол-во не нулевых значений поля `order_date`.
4. `order_status_not_nulls` - кол-во не нулевых значений поля `order_status`.
5. `delivery_date_not_nulls` - кол-во не нулевых 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 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* - количество 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 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.

Задания для промежуточной аттестации по дисциплине (модулю)

№ п/п	Задание	Ответ	Компетенция
1.	Какой оператор используется в SQL для создания таблицы?	CREATE/CREATE TABLE	ПК-6
2.	Как расшифровывается технология хранения данных «БД»?	База данных / база данных / Базы данных / базы данных	ОПК-3
3.	Какой язык используется для работы с реляционными базами данных?	SQL	ОПК-3

4.	Выбери SQL запрос, который выведет все данные из таблицы users: 1. GET * FROM users 2. SELECT * FROM users 3. FIND * IN users	2 / два / второй	ПК-3
5.	Установите порядок написания SQL запроса. В ответ запишите последовательность цифр без знаков препинания и пробелов, пример: 3241 1. FROM users u 2. JOIN orders o ON u.id = o.user_id 3. WHERE o.date > '2023-01-01' 4. SELECT u.name, o.amount	4123/4,1,2,3/4, 1, 2, 3	ПК-3
6.	В каком виде хранятся данные в реляционных базах данных?	В виде таблиц/В табличном виде/Таблицами/Таблиц/Таблицы / таблица / в виде таблицы	ПК-3
7.	Как называется тип баз данных, где данные хранятся не в табличном виде?	NoSQL / nosql/ NOSQL / нереляционные/ нереляционные базы данных	ОПК-3
8.	Какое ключевое слово используется с оператором ORDER BY для указания сортировки в порядке убывания?	DESC	ПК-3
9.	Какой тип данных используется в SQL для хранения целых чисел?	INT/INTEGER	ПК-6
10.	Какой оператор SQL отвечает за удаление таблицы вместе с ее содержимым?	DROP/DROP TABLE	ПК-3
11.	После какой команды транзакция считается успешно завершённой?	COMMIT	ОПК-3
12.	Как расшифровывается термин «СУБД»?	Система управления базами данных/Системы управления базой данных/Системы управления базами данных/Система управления базой данных / система управления базами данных/системы управления базой данных/системы управления базами данных/система управления базой данных	ОПК-3
13.	Какое ключевое слово в SQL используется для фильтрации строк, полученных из оператора FROM?	WHERE	ПК-6
14.	Какая агрегирующая функция используется в SQL для нахождения минимального значения?	MIN/MIN()	ПК-3
15.	Какой тип соединения таблиц в SQL подразумевается под оператором JOIN?	INNER/INNER JOIN/ inner/inner join/Inner Join/Inner join /Внутреннее/Внутреннее соединение/внутреннее/внутреннее соединение	ПК-6