**Базовый SQL для вопросов интервью с аналитиком данных**

**1. Что такое SQL и каково его назначение при анализе данных?**

[SQL (язык структурированных запросов)](https://www.geeksforgeeks.org/what-is-sql/) — это стандартный язык для управления реляционными базами данных и работы с ними. Он позволяет аналитикам эффективно запрашивать, обновлять данные и управлять ими, что крайне важно для задач по анализу данных.

**2. Как получить все столбцы из таблицы с именем sales?**

Пример: "SELECT \* FROM sales;"

**3. Напишите запрос SQL для выбора уникальных значений из столбца с именем product\_category.**

Пример: "SELECT DISTINCT product\_category FROM sales;"

**4. Как отфильтровать записи в SQL? Приведите пример.**

Используйте WHERE предложение для фильтрации записей на основе конкретных условий.

Пример: "SELECT \* FROM sales WHERE amount > 1000;"

**5. Объясните предложение GROUP BY и приведите пример.**

Предложение GROUP BY используется для объединения строк с одинаковыми значениями в указанных столбцах.

Пример: "SELECT product\_category, SUM(amount) FROM sales GROUP BY product\_category;"

**6. Какова цель HAVING предложения? Чем оно отличается от WHERE?**

HAVING используется для фильтрации групп после агрегирования, в то время как WHERE фильтрует строки до агрегирования.

Пример: "SELECT product\_category, SUM(amount) FROM sales GROUP BY product\_category HAVING SUM(amount) > 5000;"

**7. Как вы сортируете результаты запроса?**

Используйте предложение ORDER BY для сортировки результатов по возрастанию или убыванию.

Пример: "SELECT \* FROM sales ORDER BY amount DESC;"

**8. Напишите запрос для поиска общей суммы продаж по каждому товару.**

Пример: "SELECT product\_name, SUM(amount) FROM sales GROUP BY product\_name;"

**9. В чем разница между INNER JOIN и LEFT JOIN?**

INNER JOIN возвращает строки с совпадающими значениями в обеих таблицах, в то время как LEFT JOIN возвращает все строки из левой таблицы и совпадающие строки из правой таблицы, а несовпадающие строки из правой таблицы заполняются значениями NULL.

* INNER JOIN Пример: "SELECT sales.product\_name, products.price FROM sales INNER JOIN products ON sales.product\_id = products.product\_id;"
* LEFT JOIN Пример: "SELECT sales.product\_name, products.price FROM sales LEFT JOIN products ON sales.product\_id = products.product\_id;"

**10. Как найти 5 самых крупных сделок по продажам?**

Пример: "SELECT amount FROM sales ORDER BY amount DESC LIMIT 5;"

**Medium SQL Question for Data Analyst Interview**

**11. Что такое подзапрос и как его можно использовать в предложении WHERE?**

Вложенный запрос — это запрос внутри другого запроса, который используется для выполнения операций на основе результатов внешнего запроса.

Пример: "SELECT product\_name FROM sales WHERE amount > (SELECT AVG(amount) FROM sales);"

**12. Объясните концепцию CROSS JOIN.**

A CROSS JOIN возвращает декартово произведение двух таблиц, где каждая строка первой таблицы объединяется с каждой строкой второй таблицы.

Пример: "SELECT \* FROM products CROSS JOIN categories;"

**13. Как вы вычисляете среднее значение по столбцу?**

Используйте функцию AVG() для вычисления среднего значения.

Пример: "SELECT AVG(amount) FROM sales;"

**14. Какова цель COUNT() функции? Приведите пример.**

Функция COUNT() возвращает количество строк, соответствующих заданному условию.

Пример: "SELECT COUNT(\*) FROM sales WHERE product\_category = 'Electronics';"

**15. Напишите запрос для поиска минимальной и максимальной суммы продаж.**

Пример: "SELECT MIN(amount) AS MinAmount, MAX(amount) AS MaxAmount FROM sales;"

**16. Как вы обновляете записи в таблице?**

Используйте UPDATE инструкцию для изменения существующих записей.

Пример: "UPDATE sales SET amount = amount \* 1.1 WHERE product\_name = 'Laptop';"

**17. Объясните это CASE утверждение и приведите пример.**

Оператор CASE допускает условную логику в SQL-запросах.

Пример: "SELECT product\_name, CASE WHEN amount > 1000 THEN 'High' ELSE 'Low' END AS sales\_category FROM sales;"

**18. Как вы обрабатываете нулевые значения в SQL-запросах?**

Используйте такие функции, как IS NULL, IS NOT NULL, или COALESCE() для обработки значений NULL.

Пример: "SELECT COALESCE(discount, 0) FROM sales;"

**19. Для чего используется LIMIT предложение? Приведите пример.**

Предложение LIMIT ограничивает количество строк, возвращаемых запросом.

Пример: "SELECT \* FROM sales LIMIT 10;"

**20. Как использовать предложение JOIN для объединения данных из нескольких таблиц?**

Используйте JOIN для объединения строк из двух или более таблиц на основе связанных столбцов.

Пример: "SELECT orders.order\_id, customers.customer\_name FROM orders JOIN customers ON orders.customer\_id = customers.customer\_id;"

**21. Что такое самообъединение? Приведите пример.**

Самосоединение — это соединение, при котором таблица соединяется сама с собой. Оно полезно для иерархических данных.

Пример: "SELECT e1.employee\_name AS Employee, e2.employee\_name AS Manager FROM employees e1 LEFT JOIN employees e2 ON e1.manager\_id = e2.employee\_id;"

**22. Объясните операторы UNION и UNION ALL.**

UNION объединяет результаты двух запросов и удаляет дубликаты. UNION ALL объединяет результаты без удаления дубликатов.

* UNION Пример: "SELECT product\_name FROM sales UNION SELECT product\_name FROM returns;"
* UNION ALL Пример: "SELECT product\_name FROM sales UNION ALL SELECT product\_name FROM returns;"

**23. Как вы находите повторяющиеся записи в таблице?**

Используйте предложение GROUP BY с HAVING COUNT(\*) > 1, чтобы найти дубликаты.

Пример: "SELECT product\_name, COUNT(\*) FROM sales GROUP BY product\_name HAVING COUNT(\*) > 1;"

**24. Что такое оконная функция? Приведите пример.**

Оконные функции выполняют вычисления по[набору](https://www.geeksforgeeks.org/set-in-cpp-stl/) строк таблицы, связанных с текущей строкой, например, вычисляют промежуточные итоги или ранжируют.

Пример: "SELECT product\_name, amount, RANK() OVER (ORDER BY amount DESC) AS rank FROM sales;"

**25. Как вы вычисляете текущий итог в SQL?**

Используйте функцию SUM() window с соответствующим OVER условием.

Пример: "SELECT order\_date, amount, SUM(amount) OVER (ORDER BY order\_date) AS running\_total FROM orders;"

**26. Как использовать предложение EXISTS в SQL?**

Предложение EXISTS проверяет, возвращает ли подзапрос какие-либо строки.

Пример: "SELECT product\_name FROM sales WHERE EXISTS (SELECT \* FROM returns WHERE returns.product\_id = sales.product\_id);"

**27. Объясните разницу между WHERE и HAVING предложениями.**

WHEREфильтрует строки перед агрегированием, а HAVING фильтрует группы после агрегирования.

* WHERE Пример: "SELECT \* FROM sales WHERE amount > 1000;"
* HAVING Пример: "SELECT product\_name, SUM(amount) FROM sales GROUP BY product\_name HAVING SUM(amount) > 5000;"

**28. Как вы определяете количество заказов на одного клиента?**

Пример: "SELECT customer\_id, COUNT(order\_id) FROM orders GROUP BY customer\_id;"

**29. Что такое TEMPORARY таблица и как она используется?**

A TEMPORARY таблица — это таблица, которая существует временно во время сеанса и автоматически удаляется по окончании сеанса.

Пример: "CREATE TEMPORARY TABLE temp\_sales AS SELECT \* FROM sales WHERE amount > 1000;"

**30. Как вы используете оператор ALTER TABLE?**

Оператор ALTER TABLE изменяет существующую структуру таблицы, например, добавляя или удаляя столбцы.

Пример: "ALTER TABLE sales ADD COLUMN discount DECIMAL(10, 2);"

**31. Что такое нормализация и почему это важно?**

[Нормализация](https://www.geeksforgeeks.org/introduction-of-database-normalization/) — это процесс упорядочивания данных для уменьшения избыточности и повышения[целостности данных.](https://www.geeksforgeeks.org/integrity-importance-and-challenges/) Она обеспечивает эффективность базы данных и поддерживает согласованность.

**32. Как выполнить массовую вставку данных в таблицу?**

Используйте инструкцию INSERT INTO ... VALUES с несколькими значениями или утилиту массовой загрузки.

* Пример с несколькими значениями: "INSERT INTO sales (product\_name, amount) VALUES ('Laptop', 1200), ('Smartphone', 800);"
* Пример массовой загрузки (MySQL): "LOAD DATA INFILE 'file\_path.csv' INTO TABLE sales FIELDS TERMINATED BY ',';"

**33. Объясните концепцию индексации и ее преимущества.**

[Индексирование](https://www.geeksforgeeks.org/indexing-in-databases-set-1/) повышает скорость операций по извлечению данных из таблицы за счёт создания структуры данных, которая позволяет быстро выполнять поиск.

**34. Как вы создаете индекс в таблице?**

Используйте оператор CREATE INDEX для создания индекса для одного или нескольких столбцов.

Пример: "CREATE INDEX idx\_product\_name ON sales(product\_name);"

**Hard SQL Question For Data Analyst Interview Questions**

**35. Каковы некоторые распространенные методы оптимизации производительности в SQL?**

К распространённым методам относятся использование индексов, оптимизация запросов, сокращение использования подзапросов и обеспечение эффективности[объединений.](https://www.geeksforgeeks.org/sql-join-set-1-inner-left-right-and-full-joins/)

**36. Как вы используете оператор EXPLAIN для анализа производительности запросов?**

Оператор EXPLAIN предоставляет информацию о том, как выполняется запрос, включая порядок операций и использование индексов.

Пример: "EXPLAIN SELECT \* FROM sales WHERE amount > 1000;"

**37. Какова цель COALESCE() функции?**

Функция COALESCE() возвращает первое ненулевое значение из списка аргументов.

Пример: "SELECT COALESCE(discount, 0) FROM sales;"

**38. Как вы рассчитываете процент от общего объёма продаж для каждого товара?**

Используйте комбинацию SUM() и оконную функцию для вычисления [процента.](https://www.geeksforgeeks.org/percentage/)

Пример: "SELECT product\_name, SUM(amount) AS total\_sales, (SUM(amount) / SUM(SUM(amount)) OVER ()) \* 100 AS percentage FROM sales GROUP BY product\_name;"

**39. Что такое VIEW и как его создать?**

A VIEW — это виртуальная таблица, основанная на результатах запроса. Она упрощает сложные запросы и повышает безопасность.

Пример: "CREATE VIEW high\_value\_sales AS SELECT \* FROM sales WHERE amount > 1000;"

**40. Как вы используете оператор DROP TABLE?**

Оператор DROP TABLE удаляет из базы данных всю таблицу и её данные.

Пример: "DROP TABLE old\_sales;"

**41. В чём разница между DELETE и TRUNCATE?**

DELETE удаляет строки из таблицы на основе условия и может быть отменено, в то время как TRUNCATE удаляет все строки из таблицы и не может быть отменено.

* DELETE Пример: "DELETE FROM sales WHERE amount < 500;"
* TRUNCATE Пример: "TRUNCATE TABLE sales;"

**42. Как найти среднее значение столбца в SQL?**

Для нахождения медианы требуется[сортировка](https://www.geeksforgeeks.org/sorting-algorithms/) и использование оконных функций.

Пример: "WITH OrderedSales AS (SELECT amount, ROW\_NUMBER() OVER (ORDER BY amount) AS rn, COUNT(\*) OVER () AS total\_count FROM sales) SELECT AVG(amount) AS median FROM OrderedSales WHERE rn IN ((total\_count + 1) / 2, (total\_count + 2) / 2);"

**43. Что такое RANK() функция и как она используется?**

Функция RANK() присваивает ранг каждой строке в разделе набора результатов с пропусками в значениях ранга при наличии совпадений.

Пример: "SELECT product\_name, amount, RANK() OVER (ORDER BY amount DESC) AS rank FROM sales;"

**44. Как выполнить JOIN с несколькими таблицами?**

Вы можете объединить более двух таблиц с помощью[цепочки](https://www.geeksforgeeks.org/hashing-set-2-separate-chaining/) JOIN операций.

Пример: "SELECT orders.order\_id, customers.customer\_name, products.product\_name FROM orders JOIN customers ON orders.customer\_id = customers.customer\_id JOIN products ON orders.product\_id = products.product\_id;"

**45. Что такое TEMPORARY таблица и когда вы её используете?**

Таблица TEMPORARY используется для временного хранения промежуточных результатов во время сеанса, что полезно для сложных запросов или[пакетной обработки.](https://www.geeksforgeeks.org/batch-processing-operating-system/)

Пример: "CREATE TEMPORARY TABLE temp\_sales AS SELECT \* FROM sales WHERE amount > 1000;"

**46. Как получить последние N записей из таблицы?**

Используйте ORDER BY предложение с LIMIT для получения последних N записей.

Пример: "SELECT \* FROM sales ORDER BY sale\_date DESC LIMIT 10;"

**47. Объясните функцию DATEPART() на примере.**

DATEPART() извлекает определённую часть (например, год, месяц) из даты.

Пример: "SELECT DATEPART(year, sale\_date) AS sale\_year FROM sales;"

**48. Как использовать JOIN для включения строк, не имеющих совпадений в одной из таблиц?**

Используйте LEFT JOIN или RIGHT JOIN для включения строк из одной таблицы, даже если в другой таблице нет совпадающих строк.

Пример: "SELECT employees.name, departments.department\_name FROM employees LEFT JOIN departments ON employees.department\_id = departments.department\_id;"

**49. Как вычислить разницу между двумя датами в SQL?**

Используйте функции даты для вычисления разницы между двумя датами.

Пример: "SELECT DATEDIFF(day, start\_date, end\_date) AS date\_difference FROM projects;"

**50. В чём разница между CHAR и VARCHAR типами данных?**

CHAR — это строка фиксированной длины, а VARCHAR — это строка переменной длины. CHAR использует заданную длину для всех значений, при необходимости дополняя пробелы, в то время как VARCHAR использует только пространство, необходимое для каждого значения.