Отвечу на вопросы по SQL максимально просто и понятно:

**1. Что такое подзапрос?**

Подзапрос — это SQL-запрос, который находится внутри другого (основного) запроса. Он заключается в скобки и выполняется первым, чтобы передать результат в основной запрос. Используется, например, для фильтрации, вычислений или получения промежуточных данных.

**Пример:**

sql

SELECT name FROM employees WHERE salary > (SELECT AVG(salary) FROM employees);

Здесь подзапрос (SELECT AVG(salary) FROM employees) вычисляет среднюю зарплату, а основной запрос использует этот результат.

**2. Виды подзапросов и их особенности**

Подзапросы делятся на несколько видов:

* **Однострочный подзапрос** Возвращает **одну строку и один столбец**. Используется с операторами сравнения (=, >, <, и т.д.). **Пример:** Найти сотрудников с зарплатой выше средней:

sql

SELECT name FROM employees WHERE salary > (SELECT AVG(salary) FROM employees);

* **Многострочный подзапрос** Возвращает **несколько строк и один столбец**. Используется с операторами IN, ANY, ALL. **Пример:** Найти сотрудников из отделов 1 и 2:

sql

SELECT name FROM employees WHERE dept\_id IN (SELECT dept\_id FROM departments WHERE dept\_id IN (1, 2));

* **Многоколоночный подзапрос** Возвращает **несколько строк и несколько столбцов**. Используется для сравнения парного соответствия. **Пример:** Найти сотрудников с определённой зарплатой и отделом:

sql

SELECT name FROM employees WHERE (salary, dept\_id) IN (SELECT salary, dept\_id FROM salary\_history);

* **Коррелированный подзапрос** Зависит от данных основного запроса, выполняется для каждой строки основного запроса (см. пункт 5).

**3. Порядок выполнения запросов с подзапросами**

SQL выполняет запросы по следующему порядку:

1. Сначала выполняется **подзапрос** (в скобках), так как его результат нужен основному запросу.
2. Результат подзапроса передаётся в **основной запрос**.
3. Основной запрос выполняется с учётом результата подзапроса.
4. Если подзапрос коррелированный, он выполняется многократно — для каждой строки основного запроса.

**Пример порядка выполнения:**

sql

SELECT name FROM employees WHERE salary > (SELECT AVG(salary) FROM employees);

* Сначала вычисляется (SELECT AVG(salary) FROM employees) → результат, например, 5000.
* Затем основной запрос фильтрует сотрудников с salary > 5000.

**4. Правила использования подзапросов в разных частях основного запроса**

Подзапросы можно использовать в разных частях основного запроса, но есть правила:

* **В SELECT:** Подзапрос возвращает **одну строку и один столбец**, чтобы подставить значение в выборку. **Пример:** Вывести имя сотрудника и среднюю зарплату по компании:

sql

SELECT name, (SELECT AVG(salary) FROM employees) AS avg\_salary FROM employees;

* **В FROM:** Подзапрос создаёт временную таблицу, которая используется в основном запросе. **Пример:** Выбрать данные из подзапроса:

sql

SELECT avg\_salary FROM (SELECT AVG(salary) AS avg\_salary FROM employees) AS temp;

* **В WHERE:** Подзапрос используется для фильтрации. Может быть однострочным (с =, >), многострочным (с IN, ANY, ALL) или коррелированным. **Пример:** Найти сотрудников с зарплатой выше средней:

sql

SELECT name FROM employees WHERE salary > (SELECT AVG(salary) FROM employees);

* **В HAVING:** Подзапрос используется для фильтрации групп. Обычно возвращает одно значение. **Пример:** Найти отделы с суммарной зарплатой выше средней по компании:

sql

SELECT dept\_id FROM employees GROUP BY dept\_id HAVING SUM(salary) > (SELECT AVG(salary) \* COUNT(\*) FROM employees);

**Общие правила:**

* Подзапрос должен быть в скобках.
* Подзапрос должен возвращать данные, совместимые с местом использования (например, одна строка для =).
* Коррелированные подзапросы чаще используются в WHERE.

**5. Коррелированные (соотнесённые) подзапросы. Отличительные особенности**

* **Что это?** Коррелированный подзапрос зависит от данных основного запроса. Он выполняется для каждой строки основного запроса, используя значения из этой строки.
* **Особенности:**
  + Выполняется многократно (медленнее, чем обычные подзапросы).
  + Часто используется с EXISTS или для сравнения с текущей строкой.
  + Ссылается на таблицу основного запроса через алиасы.
* **Пример:** Найти сотрудников, у которых зарплата выше средней по их отделу:

sql

SELECT name FROM employees e1 WHERE salary > (SELECT AVG(salary) FROM employees e2 WHERE e2.dept\_id = e1.dept\_id);

Здесь подзапрос вычисляет среднюю зарплату для отдела каждого сотрудника (e1.dept\_id).

* **Отличие от обычного подзапроса:** Обычный подзапрос выполняется один раз и возвращает независимый результат. Коррелированный подзапрос выполняется для каждой строки основного запроса и использует её данные.

**6. Кванторы EXISTS, ALL, ANY, SOME**

Эти операторы используются с подзапросами для сравнения значений.

* **EXISTS** Проверяет, возвращает ли подзапрос **хотя бы одну строку**. Возвращает TRUE или FALSE. **Пример:** Найти отделы, в которых есть сотрудники с зарплатой > 5000:

sql

SELECT dept\_name FROM departments d WHERE EXISTS (SELECT 1 FROM employees e WHERE e.dept\_id = d.dept\_id AND e.salary > 5000);

**Особенность:** Подзапрос с EXISTS останавливается, как только находит первую подходящую строку.

* **ALL** Сравнивает значение с **каждым значением** из подзапроса. Условие истинно, если сравнение выполняется для всех строк подзапроса. **Пример:** Найти сотрудников с зарплатой выше всех в отделе 1:

sql

SELECT name FROM employees WHERE salary > ALL (SELECT salary FROM employees WHERE dept\_id = 1);

* **ANY / SOME** Сравнивает значение с **хотя бы одним значением** из подзапроса. Условие истинно, если сравнение выполняется хотя бы для одной строки. (SOME — синоним ANY). **Пример:** Найти сотрудников с зарплатой выше хотя бы одной в отделе 1:

sql

SELECT name FROM employees WHERE salary > ANY (SELECT salary FROM employees WHERE dept\_id = 1);

**Ключевые моменты:**

* EXISTS проверяет наличие строк, а не их значения.
* ALL требует соответствия всем значениям подзапроса.
* ANY/SOME требует соответствия хотя бы одному значению.