Mastering Subqueries!

@Vaishnavi Dauale

◆ What are Subqueries?

- Subqueries, also known as inner queries or nested queries, are queries within queries. They allow you to perform more complex operations by embedding a query inside another query.

- This makes it possible to dynamically retrieve data based on results from other queries.

2

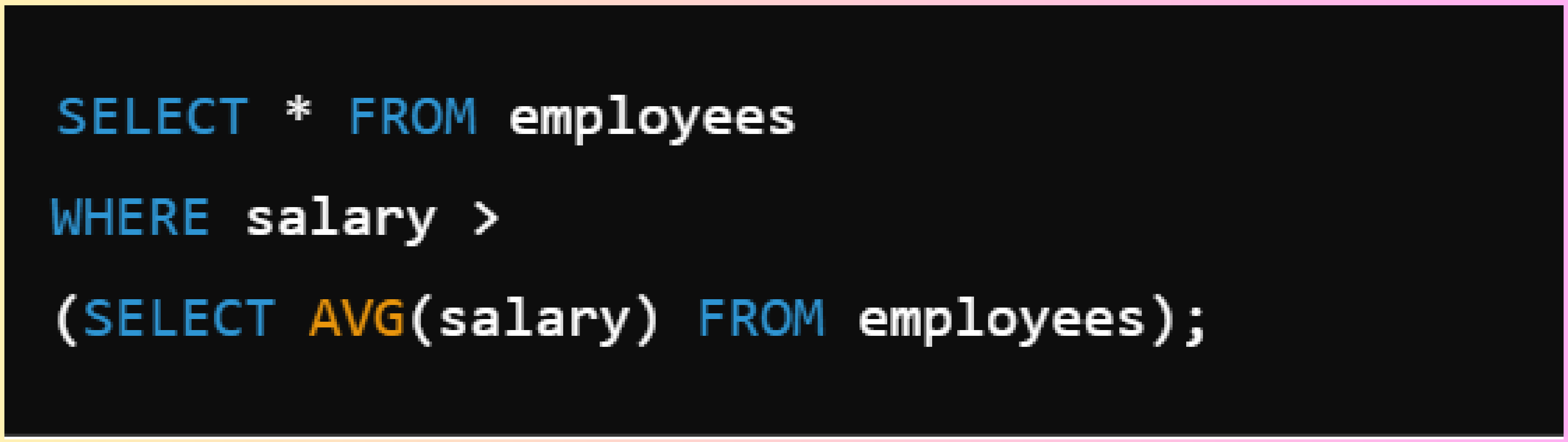
Why Use Subqueries in SQL?

- Subqueries allow you to break down complex problems into manageable parts, filter data dynamically based on nested conditions, and enhance your queries' readability and efficiency. They’re a powerful tool for simplifying intricate data retrieval tasks

Types of Subqueries :

1️⃣ Single-Row Subqueries :

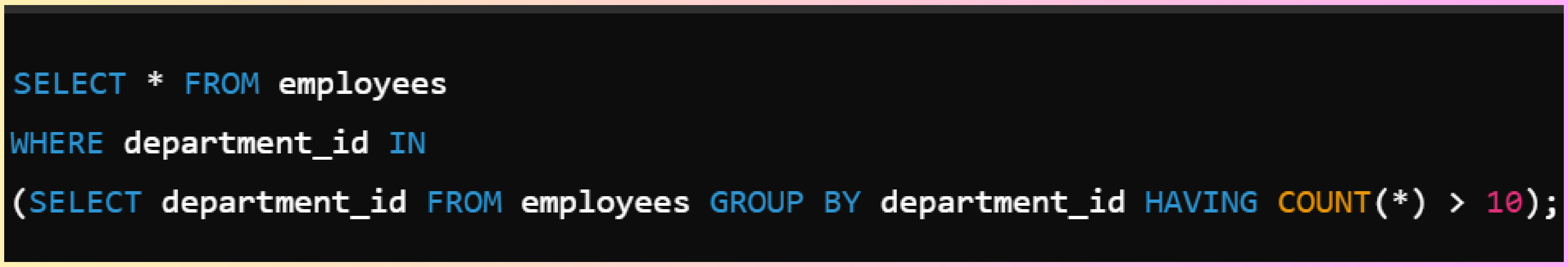
- These return just one value.



```
SELECT * FROM employees
WHERE salary >
(SELECT AVG(salary) FROM employees);
```

2️⃣ Multiple-Row Subqueries :

- These return multiple values, usually a list.



```
SELECT * FROM employees
WHERE department_id IN
(SELECT department_id FROM employees GROUP BY department_id HAVING COUNT(*) > 10);
```

💠Advantages :

- Makes complex queries easier to understand by breaking them into smaller parts.
- Allows for complex data retrieval and filtering.
- Modular Design: Simplifies queries by allowing you to build them in steps.
- Subqueries promote modular query design by allowing reusable and structured components within SQL statements.

💠Disadvantages :

- Can slow down your query, especially with large datasets.
- Nested subqueries can become difficult to manage and debug.
- Not always the best option for large-scale data processing.
- Subqueries may introduce scope issues or unexpected results if not properly understood or used within the context of the main query.

Interview Questions

- Explain the concept of a subquery in SQL and give an example of how you would use it in a query.
- What are the different types of subqueries in SQL? Can you briefly describe each type?
- What are the advantages of using subqueries in SQL? Provide examples of scenarios where subqueries are particularly useful.
- How would you optimize a query that includes nested subqueries for better performance?

8

- Have you encountered challenges when using subqueries in SQL?
- How did you overcome them, and what did you learn from the experience?
- Can you explain the difference between a correlated subquery and a non-correlated subquery? Provide examples of each.
- When would you choose to use a subquery versus a join in SQL queries? Can you provide specific scenarios for each approach?

Thank You

@Vaishnavi Dauale