Subqueries

In My SQL

Subqueries in SQL

- Subqueries, also known as inner queries or nested queries, are queries embedded within the WHERE, SELECT, or FROM clauses of another query (referred to as the outer query).
- Subqueries can be used to perform operations that require multiple steps, such as filtering based on aggregated values or using a result set as a dynamic value in another query.

Types of Subqueries:

Single-row Subqueries:

- Return a single row and can be used with comparison operators like =, <, >, etc.

Multiple-row Subqueries:

- Return more than one row and are typically used with operators like `IN`, `ANY`, `ALL`.

Correlated Subqueries:

- Reference columns from the outer query, meaning the subquery is executed once for each row processed by the outer query.

Basic Subqueries

- Subqueries can be placed in several parts of an SQL statement:

1. Subquery in WHERE Clause

  - Used to filter rows in the outer query based on values calculated from the inner query.

- Example:
- **Scenario:** Finding employees who earn more than the average salary.

	A	B	C	D
1	**Employees Table**			
2	**Employee_ID**	**Name**	**Department**	**Salary**
3	1	John Doe	IT	50000
4	2	Jane Smith	HR	60000
5	3	Bob Johnson	IT	55000
6	4	Alice Davis	Finance	70000
7	5	Carol King	IT	65000
8				

Ex: Query

Name	Salary
Jane Smith	60000
Alice Davis	70000
Carol King	65000

- **SELECT Name, Salary**
- **FROM Employees**
- **WHERE Salary > (SELECT AVG(Salary) FROM Employees);**

Here, the subquery `(SELECT AVG(Salary) FROM Employees)` calculates the average salary, and the outer query returns employees whose salaries are above this average.

2. Subquery in SELECT Clause

- Used to include additional calculated data in the result set.
- **Scenario:** Display each employee's name and their department's average salary.

- **Query:**
- **SELECT Name,**
- **(SELECT AVG(Salary) FROM Employees e2 WHERE e1.Department = e2.Department) AS Dept_Avg_Salary**
- **FROM Employees e1;**

Output:

Name	Dept_Avg_Salary
John Doe	56666.67
Jane Smith	60000
Bob Johnson	56666.67
Alice Davis	70000
Carol King	56666.67

3. Subquery in FROM Clause

- Also known as an inline view, it allows for the creation of a temporary table that the outer query can use.

- **Scenario:** Find the total sales amount per department, including only those departments with sales over $1000.

Sale_ID	Department	Amount
1	IT	1200
2	HR	800
3	Finance	1500
4	IT	600
5	HR	300
6	Finance	200

Query:

Department	Total_Sales
IT	1800
Finance	1700

- SELECT Department, Total_Sales
- FROM
- (SELECT Department, SUM(Amount) AS Total_Sales
- FROM Sales
- GROUP BY Department)
- AS DeptSales
- WHERE Total_Sales > 1000;

Correlated Subqueries

- A correlated subquery references columns from the outer query, thus being dependent on the outer query for each row it processes.

- **Scenario:** Find all employees who earn more than the average salary in their department.

Query

- SELECT Name, Salary
- FROM Employees e1
- WHERE Salary > (SELECT AVG(Salary)
- FROM Employees e2
- WHERE e1.Department = e2.Department);

Name	Salary
John Doe	50000
Bob Johnson	55000
Alice Davis	70000
Carol King	65000