

SQL Subqueries (Nested Queries)

A **subquery** (also called a **nested query**) is a query inside another SQL query. It is used to **retrieve data** that will be used by the main query.

There are **two main types** of subqueries:

- 1 **Non-Correlated Subqueries** (Independent Execution)
- 2 **Correlated Subqueries** (Depends on the Outer Query)

1 Non-Correlated Subquery (Independent Execution)

Runs **once before** the main query.

The subquery result is **passed** to the outer query.

Example: Find Customers Who Have Made Payments

```
SELECT customerNumber, customerName
FROM Customers
WHERE customerNumber IN (SELECT DISTINCT customerNumber
FROM Payments);
```

How It Works:

- The **subquery** retrieves all customers who have made payments.
- The **main query** selects customers that exist in that list.

2 Correlated Subquery (Dependent Execution)

Runs **once per row** of the outer query.

The subquery **depends** on the outer query for its execution.

Example: Find Payments Above Customer's Own Average Payment

```
SELECT customerNumber, checkNumber, amount
FROM Payments p1
```

**WHERE amount > (SELECT AVG(amount) FROM Payments p2
WHERE p1.customerNumber = p2.customerNumber);**

How It Works:

- The **subquery (p2)** calculates the **average payment amount per customer**.
 - The **outer query (p1)** filters payments that are **higher than their respective customer's average**.
-

Other Types of Subqueries

Subquery in SELECT Clause

**SELECT customerNumber,
(SELECT COUNT(*) FROM Payments WHERE
Payments.customerNumber = Customers.customerNumber) AS
payment_count
FROM Customers;**

Finds the number of payments each customer has made.

Subquery in FROM Clause (Derived Table)

**SELECT customerNumber, avg_payment
FROM (SELECT customerNumber, AVG(amount) AS avg_payment
FROM Payments GROUP BY customerNumber) AS sub;**

Uses a subquery as a temporary table to get the average payment per customer.

Subquery in UPDATE Statement

UPDATE Customers

SET creditLimit = (SELECT AVG(amount) FROM Payments)

WHERE country = 'India';

Updates the credit limit for customers based on the average payment.

Summary Table

Subquery Type	Execution	Use Case
Non-Correlated	Runs once before the main query	Finding records based on independent results
Correlated	Runs for each row in the outer query	Comparing each row with a computed value
In SELECT Clause	Used to compute values	Getting a count or total per row
In FROM Clause	Acts as a temporary table	Using aggregated data in queries
In UPDATE Statement	Updates based on a computed value	Modifying records dynamically