Co-Related Subquery

By Dhandapani Yedappalli Krishnamurthi Sep 30, 2025

🔹 What is a Correlated Subquery?

- A **correlated subquery** is a subquery that **depends on a value from the outer query**.
- Unlike a normal subquery (which runs once), a correlated subquery **runs once for each row** of the outer query.
- That’s why it can be slower, but also very powerful.

🔹 General Syntax

```
SELECT column1, column2
FROM tableA a
WHERE column3 = (
   SELECT columnX
   FROM tableB b
   WHERE b.colY = a.colY   -- 👈 correlated: uses value from outer query
);
```

🔹 Example in AdventureWorks

🎯 Business Question:

Find all **orders** where the **OrderQty** is greater than the **average quantity ordered for that product**.

Query:

```
SELECT sod.salesorderid,
      sod.productid,
      sod.orderqty
FROM sales.salesorderdetail sod
WHERE sod.orderqty >
     (SELECT AVG(sod2.orderqty)
      FROM sales.salesorderdetail sod2
```

```
    WHERE sod2.productid = sod.productid); -- correlated with outer query
```

🔍 Explanation:

- Outer query → gets order details (sod).
- Subquery → calculates **average quantity per product** (sod2).
- The subquery **depends on the productid** of the outer query row.
- For each order line, it checks if its quantity is higher than the product's average.

🔹 Another Example: Customer Orders

🎯 Business Question:

List customers who have placed an order **greater than their own average order amount**.

Query:

```
SELECT soh.salesorderid,
    soh.customerid,
    soh.totaldue
FROM sales.salesorderheader soh
WHERE soh.totaldue >
   (SELECT AVG(soh2.totaldue)
    FROM sales.salesorderheader soh2
    WHERE soh2.customerid = soh.customerid);
```

🔍 Explanation:

- Outer query → gets customer's orders.
- Subquery → finds average total amount per customer.
- Correlated because soh2.customerid = soh.customerid.

🔹 Illustration (Step-by-Step)

Imagine 3 orders:

CustomerID	OrderID	TotalDue
101	1	500
101	2	1000
102	3	200

For Customer 101: Avg = (500+1000)/2 = 750 → Order 2 (1000) qualifies.

For Customer 102: Avg = 200 → none qualify.

👉 Output = only Customer 101's Order 2.

🔹 Key Notes

✅ **Normal Subquery** → executes once, independent of outer query
✅ **Correlated Subquery** → executes once **per row** of outer query
✅ Often used for: filtering by averages, max/min per group, existence checks

🔹 Practice Questions (AdventureWorks)

1. Find products where the **list price** is greater than the **average list price of products in the same subcategory**.

2. List all employees whose **sick leave hours** are greater than the **average sick leave hours in their department**.
3. Find orders where the **order total** is greater than the **average order total for the same customer**.
4. Retrieve all sales orders where the **order date** is the **latest order date for that customer**.