Performing Subqueries

**To find the average amount paid by the top 5 customers**

**Query :**

SELECT AVG("Total Amount Paid") AS average

FROM(

SELECT B.customer\_id,

B.first\_name AS "First Name",

B.last\_name AS "Last Name",

E.country AS "Country",

D.city AS "City",

SUM(A.amount) AS "Total Amount Paid"

FROM payment A

INNER JOIN customer B ON A.customer\_id = B.customer\_id

INNER JOIN address C ON B.address\_id = C.address\_id

INNER JOIN city D ON C.city\_id = D.city\_id

INNER JOIN country E ON D.country\_id = E.country\_id

WHERE(E.country, D.city) IN (

SELECT D.country, C.city

FROM customer A

INNER JOIN address B ON A.address\_id = B.address\_id

INNER JOIN city C ON B.city\_id = C.city\_id

INNER JOIN country D ON C.country\_ID = D.country\_ID

WHERE D.country IN(

SELECT D.country

FROM customer A

INNER JOIN address B ON A.address\_id = B.address\_id

INNER JOIN city C ON B.city\_id = C.city\_id

INNER JOIN country D ON C.country\_ID = D.country\_ID

GROUP BY D.country

ORDER BY COUNT(A.customer\_id) DESC

LIMIT 10)

GROUP BY D.country,C.city

ORDER BY COUNT(A.customer\_id) DESC

LIMIT 10)

GROUP BY B.customer\_id, B.first\_name, B.last\_name, E.country, D.city

ORDER BY "Total Amount Paid" DESC

LIMIT 5)

AS total\_amount\_paid;

A screenshot of a computer

Description automatically generated

**Working with subqueries has been an interesting but challenging experience, especially when handling complex queries. While I see their value in filtering and calculations, managing long and nested subqueries can sometimes be tricky. Similarly, techniques like CTEs, views and temporary tables offer great advantages, though I am still refining my skills in using them effectively. I look forward to strengthening my SQL expertise in the coming months.**