

## 3.8: Performing Subqueries

### 1. The Average Amount Paid by The Top 5 Customers

- Query & answer:

```
1. -- Find the average amount paid by the top 5 customers
2.
3. SELECT AVG(total_amount_paid) as average
4. FROM (
5.     SELECT A.customer_id, A.first_name, A.last_name,
6.           D.country,
7.           C.city,
8.           SUM(E.amount) AS total_amount_paid
9.     FROM public.payment E
10.    RIGHT JOIN public.customer A ON E.customer_id = A.customer_id
11.    LEFT JOIN public.address B ON A.address_id = B.address_id
12.    LEFT JOIN public.city C ON B.city_id = C.city_id
13.    LEFT JOIN public.country D ON C.country_id = D.country_id
14.   WHERE C.city IN (
15.       SELECT C.city
16.       FROM public.city C
17.       RIGHT JOIN public.address B
18.         ON C.city_id = B.city_id
19.       LEFT JOIN public.country D
20.         ON C.country_id = D.country_id
21.       RIGHT JOIN public.customer A
22.         ON B.address_id = A.address_id
23.       WHERE D.country IN (
24.           SELECT D.country
25.           FROM public.customer A
26.           LEFT JOIN public.address B
27.             ON A.address_id = B.address_id
28.           LEFT JOIN public.city C
```

```

29.             ON B.city_id = C.city_id
30.             LEFT JOIN public.country D
31.             ON C.country_id = D.country_id
32.             GROUP BY country
33.             ORDER BY COUNT(customer_id) DESC
34.             LIMIT 10
35.         )
36.         GROUP BY C.city, D.country
37.         ORDER BY COUNT(customer_id) DESC
38.         LIMIT 10
39.     )
40. GROUP BY A.customer_id, A.first_name, A.last_name,
41.         D.country,
42.         C.city
43. ORDER BY total_amount_paid DESC
44. LIMIT 5
45. ) AS total_amount_paid

```

average
107.354

## 2. Number of The Top 5 Customers Based Within Each Country

- Query & answer:

```
1. -- Find out how many of the top 5 customers are based within each country
2.
3. SELECT D.country,
4.        COUNT( DISTINCT A.customer_id) AS all_customer_count,
5.        COUNT (DISTINCT top_5_customers.customer_id) AS top_customer_count
6. FROM public.customer A
7. LEFT JOIN public.address B ON A.address_id = B.address_id
8. LEFT JOIN public.city C ON B.city_id = C.city_id
9. LEFT JOIN public.country D ON C.country_id = D.country_id
10. LEFT JOIN (
11.        SELECT A.customer_id, A.first_name, A.last_name,
12.               D.country,
13.               C.city,
14.               SUM(E.amount) AS total_amount_paid
15.        FROM public.payment E
16.        RIGHT JOIN public.customer A ON E.customer_id = A.customer_id
17.        LEFT JOIN public.address B ON A.address_id = B.address_id
18.        LEFT JOIN public.city C ON B.city_id = C.city_id
19.        LEFT JOIN public.country D ON C.country_id = D.country_id
20.        WHERE C.city IN (
21.                SELECT C.city
22.                FROM public.city C
23.                RIGHT JOIN public.address B
24.                ON C.city_id = B.city_id
25.                LEFT JOIN public.country D
26.                ON C.country_id = D.country_id
27.                RIGHT JOIN public.customer A
28.                ON B.address_id = A.address_id
29.                WHERE D.country IN (
30.                        SELECT D.country
31.                        FROM public.customer A
32.                        LEFT JOIN public.address B
33.                        ON A.address_id = B.address_id
34.                        LEFT JOIN public.city C
```

```

35.                ON B.city_id = C.city_id
36.                LEFT JOIN public.country D
37.                ON C.country_id = D.country_id
38.                GROUP BY country
39.                ORDER BY COUNT(customer_id) DESC
40.                LIMIT 10
41.            )
42.            GROUP BY C.city, D.country
43.            ORDER BY COUNT(customer_id) DESC
44.            LIMIT 10
45.        )
46.        GROUP BY A.customer_id, A.first_name, A.last_name,
47.                D.country,
48.                C.city
49.        ORDER BY total_amount_paid DESC
50.        LIMIT 5
51.    ) AS top_5_customers
52.    ON D.country = top_5_customers.country
53. GROUP BY D.country
54. ORDER BY top_customer_count DESC
55. LIMIT 5

```

country	all_customer_count	top_customer_count
Mexico	30	2
United States	36	1
India	60	1
Turkey	15	1
American Samoa	1	0

### 3. Reflection

- **Do you think steps 1 and 2 could be done without using subqueries?**

Step 1 and 2 may be completed without subqueries if the data were already in place in new tables. In this case, the subqueries allowed for easy understanding, as the complex query were broken down into a series of logical steps.

- **When do you think subqueries are useful?**

Subqueries are useful when there is a need to utilize the results of one query to determine the results of another. They can also be used to create complex queries that would otherwise be difficult to write.