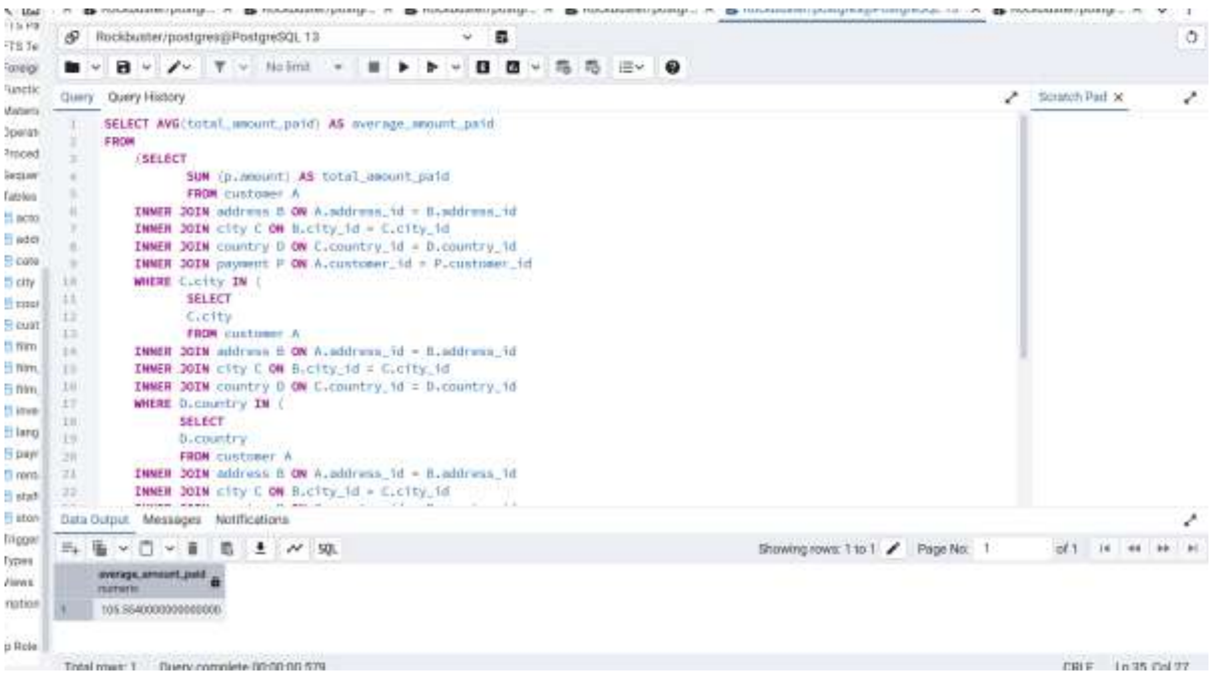
Ex 3.8

Prepared By: Nikita Kasrekar

1. **Step 1: Find the average amount paid by the top 5 customers.**

```
SELECT AVG(total_amount_paid) AS average_amount_paid
FROM
  (SELECT
            SUM (p.amount) AS total_amount_paid
                 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
       INNER JOIN payment P ON A.customer_id = P.customer_id
  WHERE C.city IN (
      SELECT
                 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 A.customer_id, A.first_name, A.last_name, D.country, C.city
       ORDER BY total_amount_paid DESC
       LIMIT 5
       ) AS average_amount_paid;
```



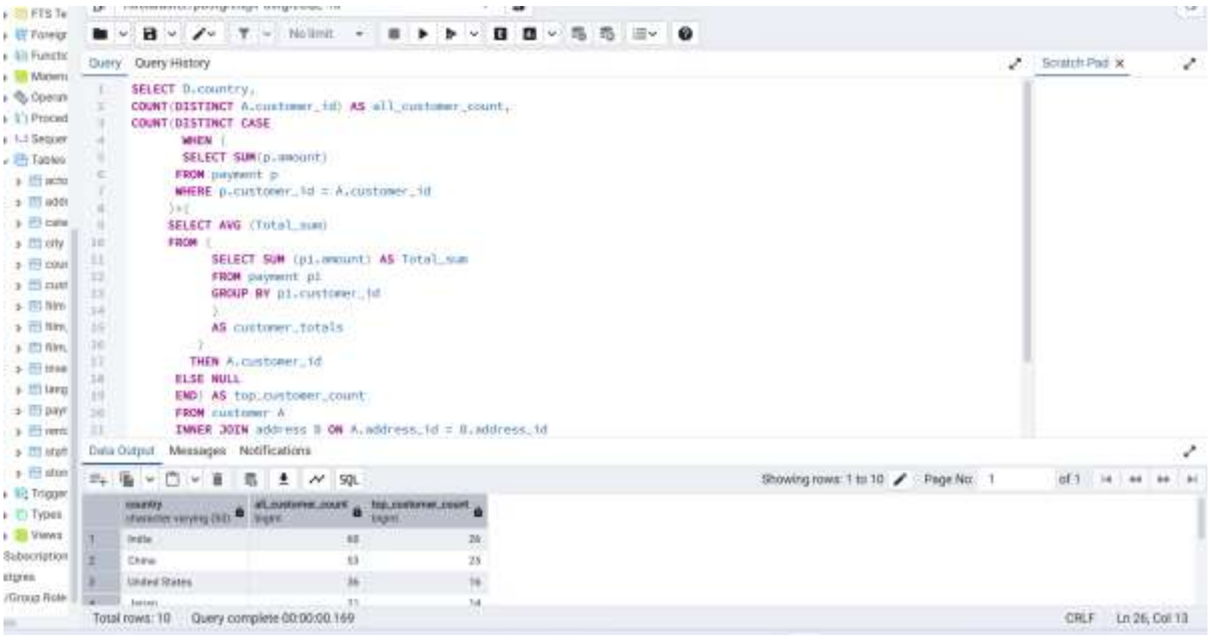
```
SELECT AVG(total_amount_paid) AS average_amount_paid
FROM
    (SELECT
            SUM (p.amount) AS total_amount_paid
            FROM customer A
    INNER JOIN address B ON A.address_id = B.address_id
    INNER JOIN city C ON B.city_id = C.city_id
    INNER JOIN country D ON C.country_id = D.country_id
    INNER JOIN payment P ON A.customer_id = P.customer_id
    WHERE C.city IN (
            SELECT
            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
```

```
     average_amount_paid
     numeric
1    105.554000000000000
```

Total rows: 1   Query complete 00:00:00.579

Step 2: Find out how many of the top 5 customers you identified in step 1 are based within each country.

```
SELECT D.country,
COUNT(DISTINCT A.customer_id) AS all_customer_count,
COUNT(DISTINCT CASE
    WHEN (
          SELECT SUM(p.amount)
FROM payment p
         WHERE p.customer_id = A.customer_id
        )>(
        SELECT AVG (Total_sum)
        FROM (
             SELECT SUM (p1.amount) AS Total_sum
                      FROM payment p1
                      GROUP BY p1.customer_id
                      )
                      AS customer_totals
                    )
                   THEN A.customer_id
ELSE NULL
         END) AS top_customer_count
         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 top_customer_count DESC
         LIMIT 10;
```



```
SELECT B.country,
COUNT(DISTINCT A.customer_id) AS all_customer_count,
COUNT(DISTINCT CASE
      WHEN (
      SELECT SUM(p.amount)
     FROM payment p
     WHERE p.customer_id = A.customer_id
    )>(
    SELECT AVG (Total_sum)
    FROM (
        SELECT SUM (p1.amount) AS Total_sum
        FROM payment p1
        GROUP BY p1.customer_id
        )
        AS customer_totals
      )
     THEN A.customer_id
    ELSE NULL
    END) AS top_customer_count
    FROM customer A
    INNER JOIN address B ON A.address_id = B.address_id
```

	country character varying (50)	all_customer_count bigint	top_customer_count bigint
1	India	60	26
2	China	53	25
3	United States	36	16
4	Japan	31	14

Total rows: 10 Query complete 00:00:00.169

Step 3: Write 1 to 2 short paragraphs on the following:

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

Yes, steps 1 and 2 would be very hard to do without subqueries because we first needed to isolate the top 5 customers before calculating averages or joining that list with all customers by country. Without subqueries, the query would become too long and complicated.

- When do you think subqueries are useful?

Subqueries are useful when we need to work with a filtered or aggregated result before using it in another calculation or join. They make complex problems easier by breaking them into smaller, clearer steps.