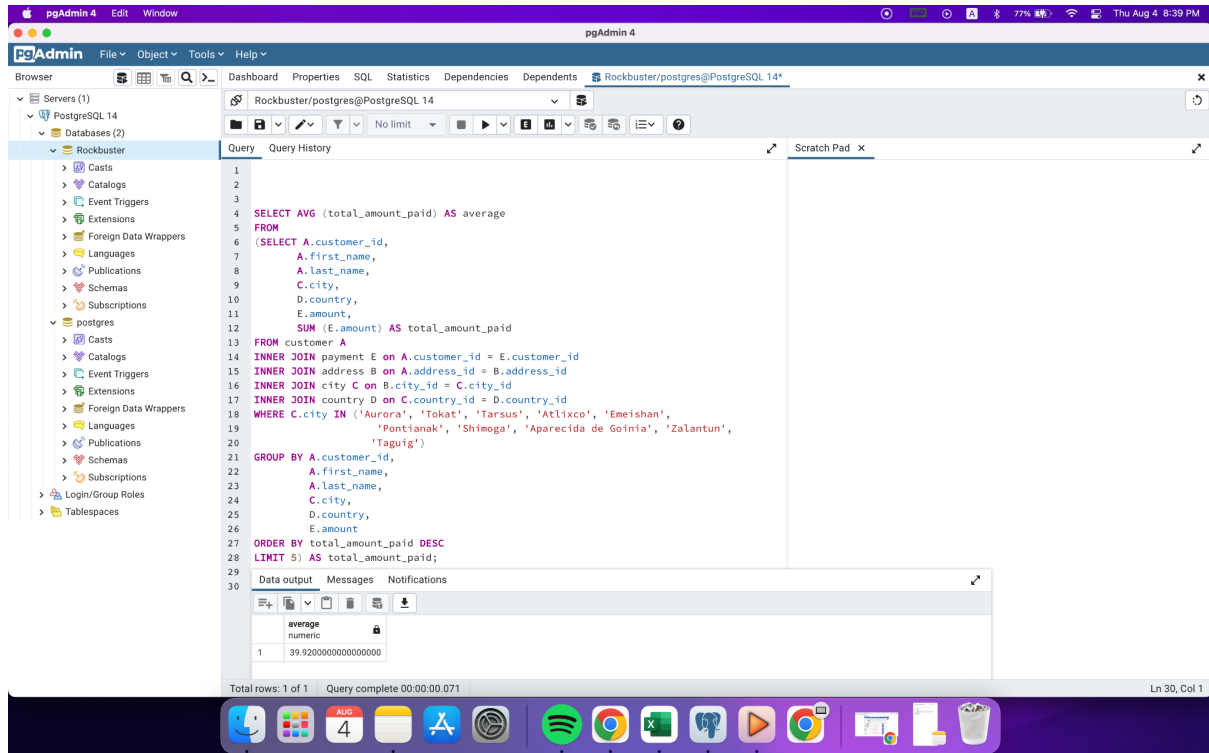


Answers 3.8

1.



```
SELECT AVG (total_amount_paid) AS average
FROM
(SELECT A.customer_id,
A.first_name,
A.last_name,
C.city,
D.country,
E.amount,
SUM (E.amount) AS total_amount_paid
FROM customer A
INNER JOIN payment E on A.customer_id = E.customer_id
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 C.city IN ('Aurora', 'Tokat', 'Tarsus', 'Atlixco', 'Emeishan',
'Pontianak', 'Shimoga', 'Aparecida de Goinia', 'Zalantun',
'Taguig'))
GROUP BY A.customer_id,
A.first_name,
A.last_name,
C.city,
D.country,
```

E.amount
ORDER BY total_amount_paid DESC
LIMIT 5) AS total_amount_paid;

2.

The screenshot shows the pgAdmin 4 interface with a SQL query editor and a results pane. The query is a complex JOIN statement that calculates the total amount paid by customers, grouped by country, and then ranks the top 5 countries by the number of customers. The results pane shows the output of the query, which is a table with 5 rows and 4 columns: country, all_customer_count, top_5_customers, and total_amount_paid.

country	all_customer_count	top_5_customers	total_amount_paid
India	60	1	
China	53	1	
United States	36	1	
Japan	31	1	
Mexico	30	1	

```

SELECT DISTINCT (D.country),
COUNT (DISTINCT A.customer_id) AS all_customer_count,
COUNT (DISTINCT D.country) AS top_5_customers
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
LEFT JOIN (SELECT A.customer_id,
A.first_name,
A.last_name,
C.city,
D.country,
E.amount,
SUM (E.amount) AS total_amount_paid
FROM customer A
INNER JOIN payment E on A.customer_id = E.customer_id
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 C.city IN ('Aurora', 'Tokat', 'Tarsus', 'Atlixco', 'Emeishan',
'Pontianak', 'Shimoga', 'Aparecida de Goinia', 'Zalantun',
'Taguig')
GROUP BY A.customer_id,
A.first_name,
A.last_name,
C.city,
D.country,
E.amount
ORDER BY total_amount_paid DESC
LIMIT 5) AS top_5_customers
ON D.country = top_5_customers.country
GROUP BY D.country
ORDER BY all_customer_count DESC
LIMIT 5;
```

```

        'Pontianak', 'Shimoga', 'Aparecida de Goinia', 'Zalantun',
        'Taguig')
GROUP BY A.customer_id,
        A.first_name,
        A.last_name,
        C.city,
        D.country,
        E.amount
ORDER BY total_amount_paid DESC
LIMIT 5) AS top_5_customers
ON D.country = top_5_customers.country
GROUP BY D.country
ORDER BY all_customer_count DESC
LIMIT 5;

```

3.

The subquery is not needed for step 1. It can be done by using the aggregate function. However, for step 2, the subquery is needed because it requires 2 different tables for the output.

When I am working with data that is constantly changing, subqueries are needed.