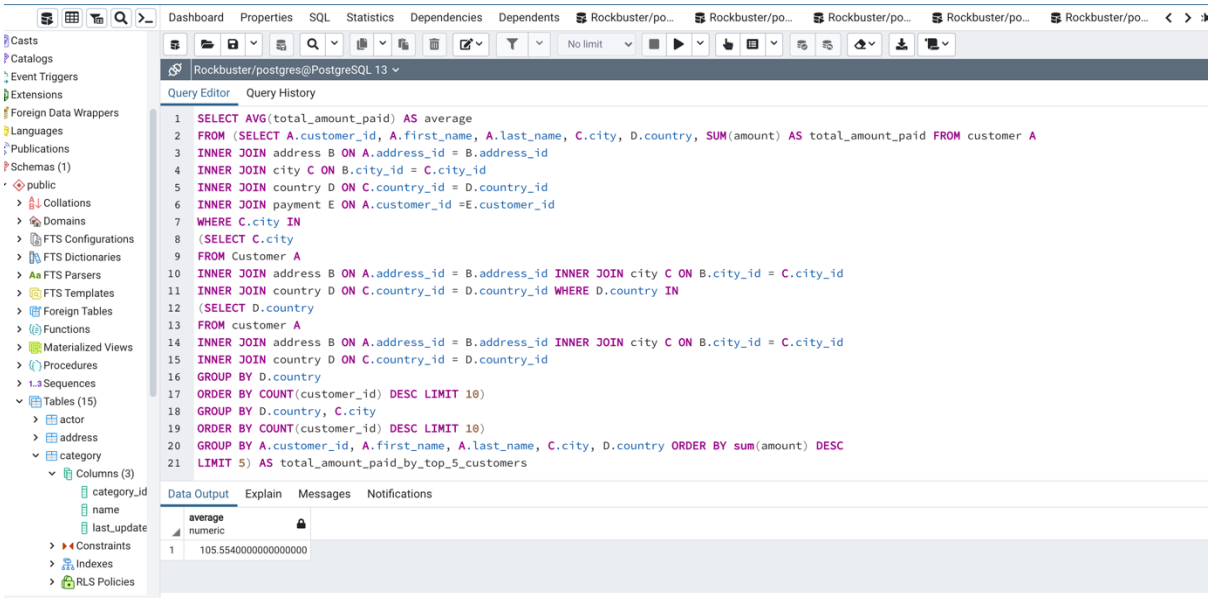
TASK_8:

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



QUERY:

SELECT AVG(total_amount_paid) AS average

FROM (SELECT A.customer_id, A.first_name, A.last_name, C.city, D.country, SUM(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 E ON A.customer_id =E.customer_id

WHERE C.city IN

(SELECT C.city

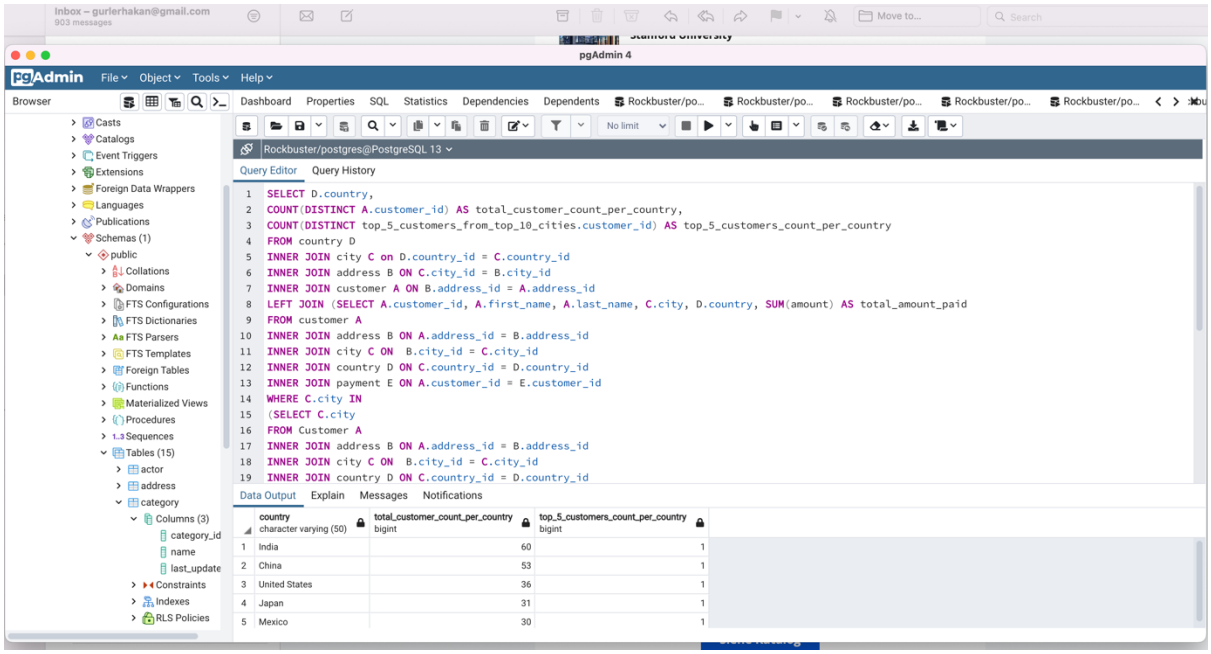
FROM Customer A

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(customer_id) DESC

LIMIT 10)

GROUP BY D.country, C.city

ORDER BY COUNT(customer_id) DESC

LIMIT 10)

GROUP BY A.customer_id, A.first_name, A.last_name, C.city, D.country

ORDER BY sum(amount) DESC

LIMIT 5) AS total_amount_paid_by_top_5_customers
```

Step 2: Find out how many of the top 5 customers are based within each country.



QUERY:

```
SELECT D.country,
COUNT(DISTINCT A.customer_id) AS total_customer_count_per_country,
COUNT(DISTINCT top_5_customers_from_top_10_cities.customer_id) AS
top_5_customers_count_per_country
FROM country D
INNER JOIN city C on D.country_id = C.country_id
INNER JOIN address B ON C.city_id = B.city_id
INNER JOIN customer A ON B.address_id = A.address_id
LEFT JOIN (SELECT A.customer_id, A.first_name, A.last_name, C.city, D.country,
SUM(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 E ON A.customer_id = E.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(customer_id) DESC
LIMIT 10)
GROUP BY D.country, C.city
ORDER BY COUNT(customer_id) DESC
LIMIT 10)
GROUP BY A.customer_id, A.first_name, A.last_name, C.city, D.country
ORDER BY sum(amount) DESC
LIMIT 5) AS top_5_customers_from_top_10_cities
ON D.country = top_5_customers_from_top_10_cities.country
GROUP BY D.country, top_5_customers_from_top_10_cities
ORDER BY total_customer_count_per_country DESC
LIMIT 5
```

Step 3:

1. Write 1 to 2 short paragraphs on the following:
    - Do you think steps 1 and 2 could be done without using subqueries?
    - When do you think subqueries are useful?

Probably step 1 and 2 can be done without using subqueries but it would be more time consuming, and complicated. Even using subqueries are little bit complicated (my opinion), writing multiple independent queries will be more fragile to mistakes. After having more practice, I feel that I will not think that the subqueries are complicated to create.

Subqueries divide the complex query into parts so that complex query can be written into a series of logical steps. It is easy to understand and query mainteanance also at ease, and also the subqueries allows you to use the results of another query in outer query.