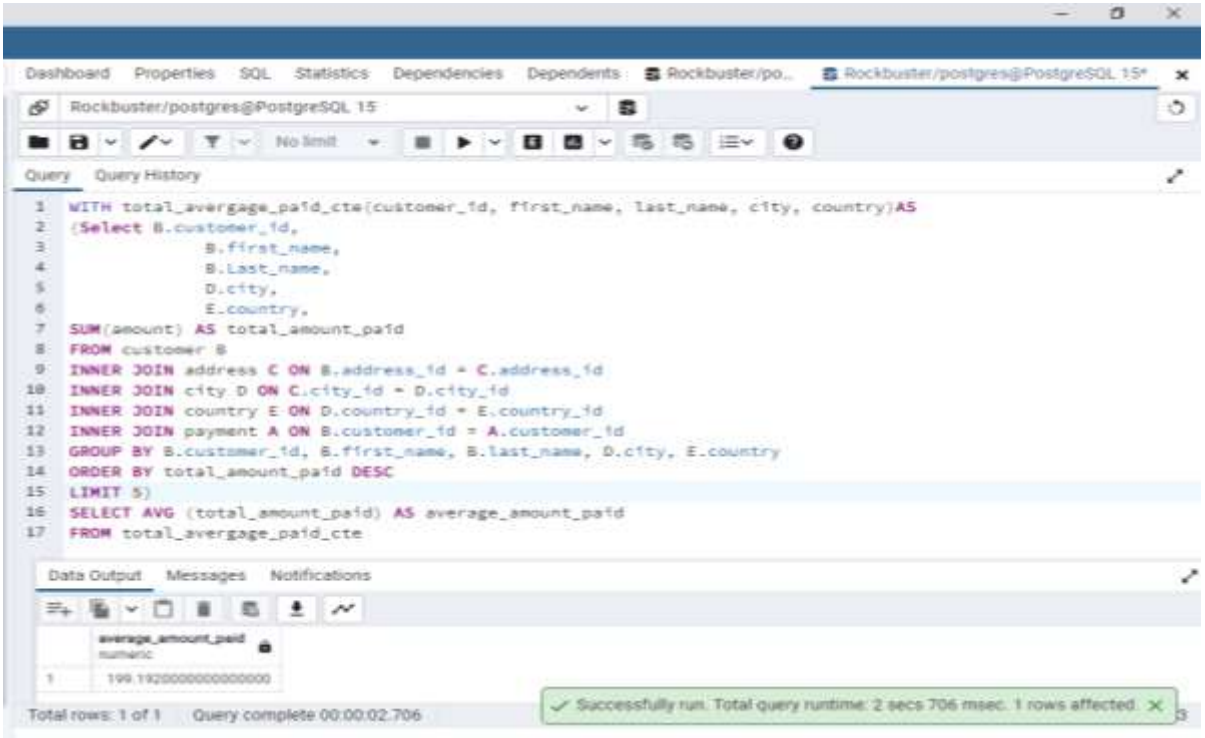
Step 1: Answer the business questions from step 1 and 2 of task 3.8 using CTEs

1. Rewrite your queries from steps 1 and 2 of task 3.8 as CTEs.
2. Copy-paste your CTEs and their outputs into your answers document.
3. Write 2 to 3 sentences explaining how you approached this step, for example, what you did first, second, and so on.

1.



2.

WITH top_customer_count_cte(amount,customer_id,first_name,last_name,city,country, total_amount_paid) AS

(SELECT A.amount,B.customer_id,B.first_name,B.last_name,D.city,E.country,

SUM(amount)AS total_amount_paid

FROM payment A

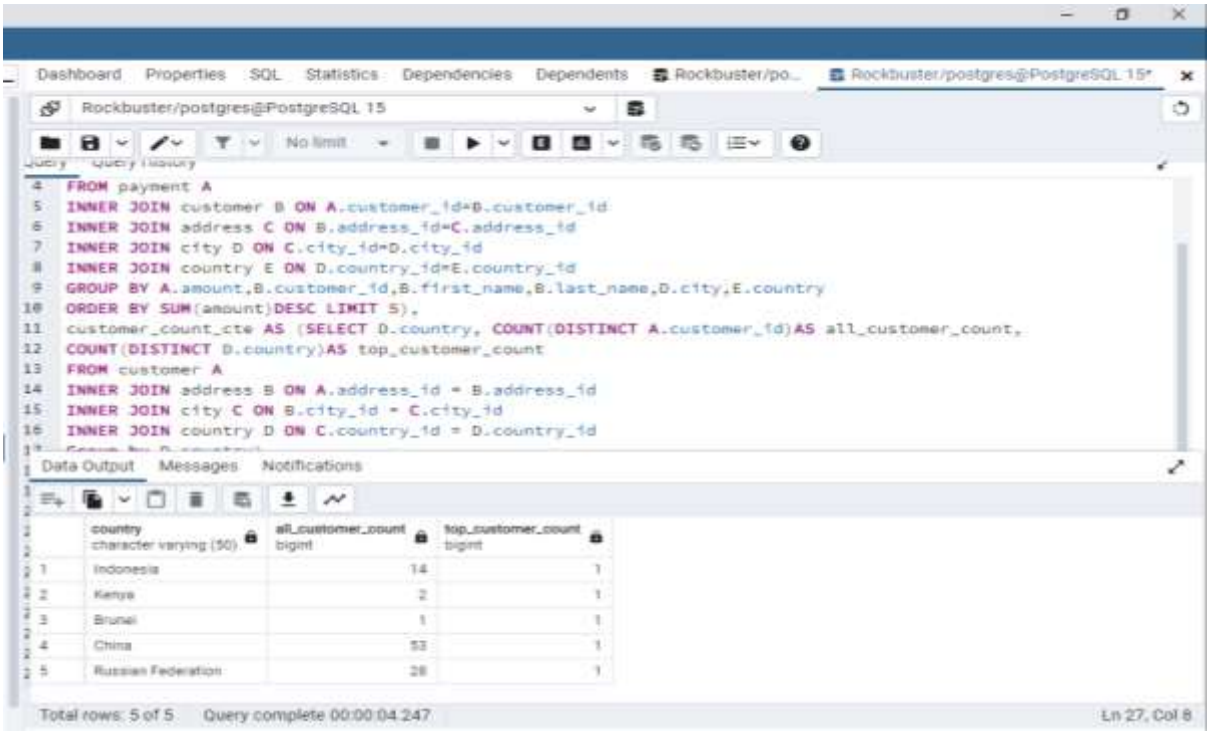
INNER JOIN customer B ON A.customer_id=B.customer_id

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

```
GROUP BY A.amount,B.customer_id,B.first_name,B.last_name,D.city,E.country
ORDER BY SUM(amount)DESC LIMIT 5),
customer_count_cte AS (SELECT D.country, COUNT(DISTINCT A.customer_id)AS all_customer_count,
COUNT(DISTINCT D.country)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)
SELECT D.country,COUNT(DISTINCT A.customer_id)AS all_customer_count,
COUNT(DISTINCT Top_customer_count_cte.customer_id)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
LEFT JOIN Top_customer_count_cte ON D.country=Top_customer_count_cte.country
GROUP BY D.country
ORDER BY top_customer_count DESC
LIMIT 5
```



The first thing I did was copy the subquery I had in exercise 3.8. Then I took out the outer query from the subquery and replaced it with CTE syntax and left the inner query as it is for the step 1 task but in the step 2 task, created 2 CTEs names for the two inner queries(one was to get the total amount paid from top 5 customers in top 10 cities within the top 10 countries, and the second's query focus on the customer counts). I finally wrote the main statement to query the information required from the CTE table created.

Step 2: Compare the performance of your CTEs and subqueries.

1. Which approach do you think will perform better and why?
2. Compare the costs of all the queries by creating query plans for each one.
3. The EXPLAIN command gives you an *estimated* cost. To find out the actual speed of your queries, run them in pgAdmin 4. After each query has been run, a pop-up window will display its speed in milliseconds.
4. Did the results surprise you? Write a few sentences to explain your answer.

   Using CTE: cost = 3406.11 runtime 142 ms

   Using subqueries cost = 101.53-101.54 runtime: 69ms

   CTE:

✓ Successfully run. Total query runtime: 2 secs 650 msec. 5 rows affected. ✕

Subquery

- Yes, I would have expected in step 2 that the subquery would be more runtime and more costly than using the CTE, but this was not the case. Maybe due to the increased number of clauses and multiple inner statements could be the reason.

3.Write 1 to 2 paragraphs on the challenges you faced when replacing your subqueries with CTEs.

The task1 was straightforward as I just have one inner query and followed the example given in the study note. But on the other hand, task two gave me a tough time as I did not realize I had to rename the second inner query with another cte name only that I won't start with the "WITH" statement. Combining the two CTEs was difficult which makes time-consuming to get the output I wanted. I just kept playing around with it until the answer appeared.