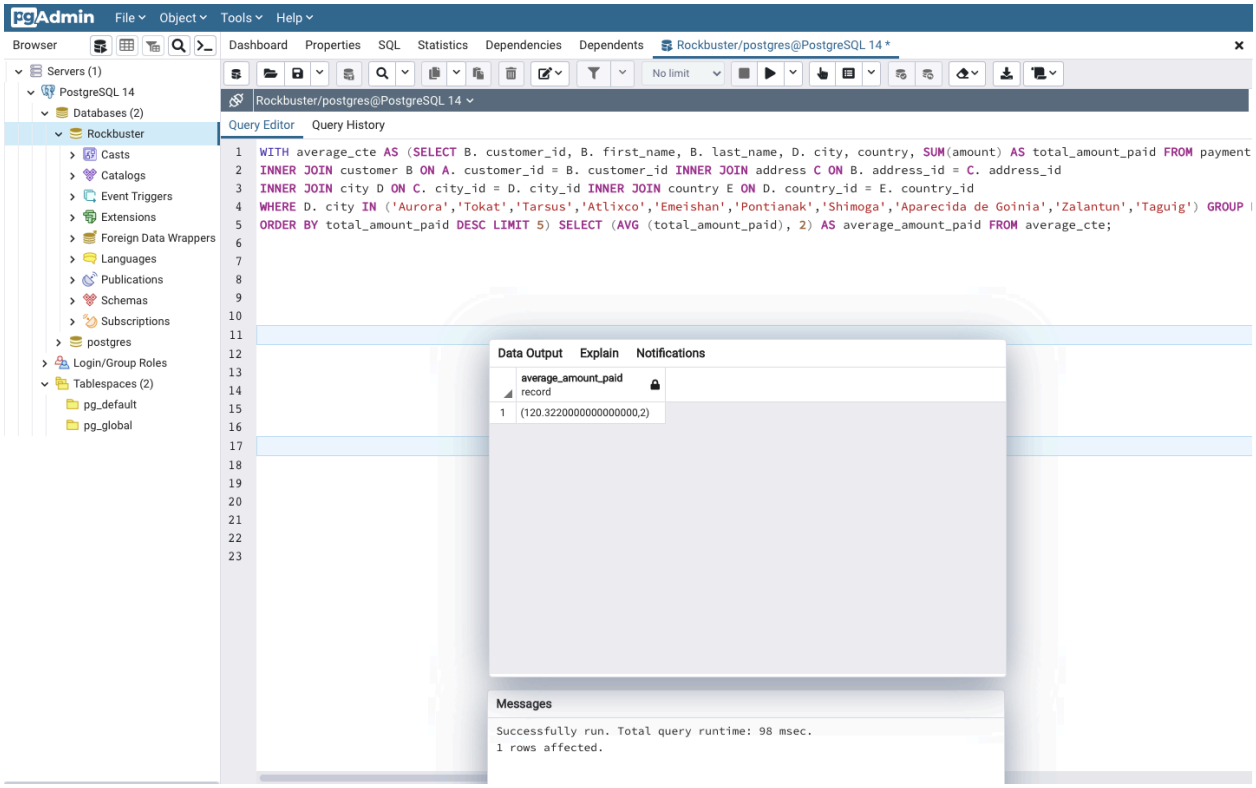
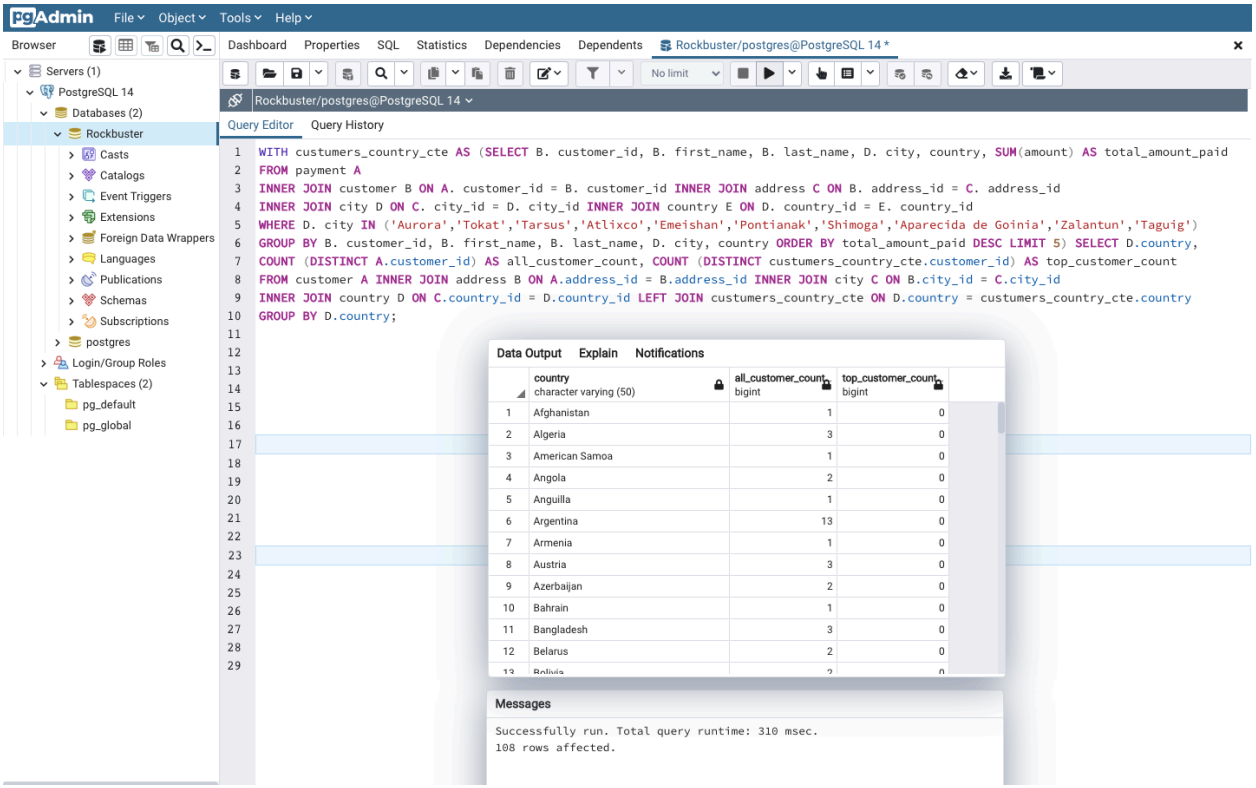
```
WITH average_cte AS (SELECT B. customer_id, B. first_name, B. last_name, D. city,
country, SUM(amount) AS total_amount_paid FROM payment A
INNER JOIN customer B ON A. customer_id = B. customer_id INNER JOIN address C ON
B. address_id = C. address_id
INNER JOIN city D ON C. city_id = D. city_id INNER JOIN country E ON D. country_id = E.
country_id
WHERE D. city IN
('Aurora','Tokat','Tarsus','Atlixco','Emeishan','Pontianak','Shimoga','Aparecida de
Goinia','Zalantun','Taguig') GROUP BY B. customer_id, B. first_name, B. last_name, D.
city, country
ORDER BY total_amount_paid DESC LIMIT 5) SELECT (AVG (total_amount_paid), 2) AS
average_amount_paid FROM average_cte;
```



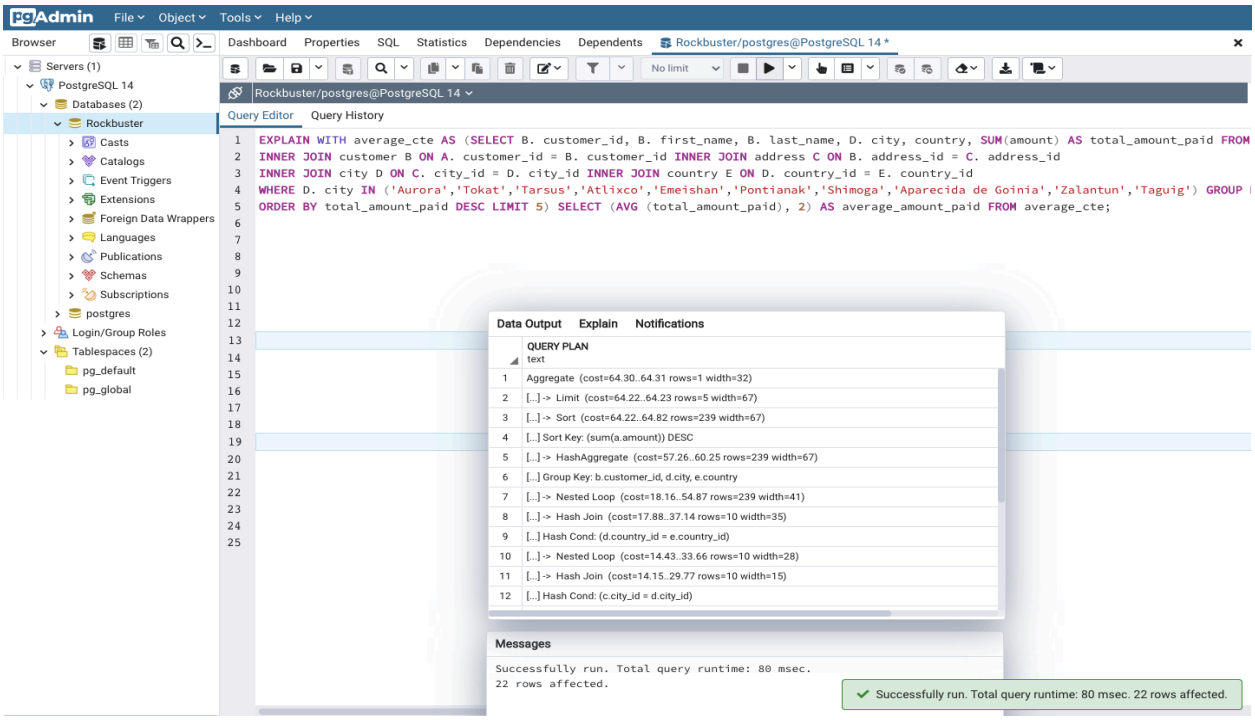
WITH custumers_country_cte AS (SELECT B. customer_id, B. first_name, B. last_name, D. city, country, SUM(amount) AS total_amount_paid FROM payment A
INNER JOIN customer B ON A. customer_id = B. customer_id INNER JOIN address C ON B. address_id = C. address_id
INNER JOIN city D ON C. city_id = D. city_id INNER JOIN country E ON D. country_id = E. country_id
WHERE D. city IN
('Aurora','Tokat','Tarsus','Atlixco','Emeishan','Pontianak','Shimoga','Aparecida de Goinia','Zalantun','Taguig') GROUP BY B. customer_id, B. first_name, B. last_name, D. city, country ORDER BY total_amount_paid DESC LIMIT 5) SELECT D.country, COUNT (DISTINCT A.customer_id) AS all_customer_count, COUNT (DISTINCT custumers_country_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 custumers_country_cte ON D.country = custumers_country_cte.country GROUP BY D.country;



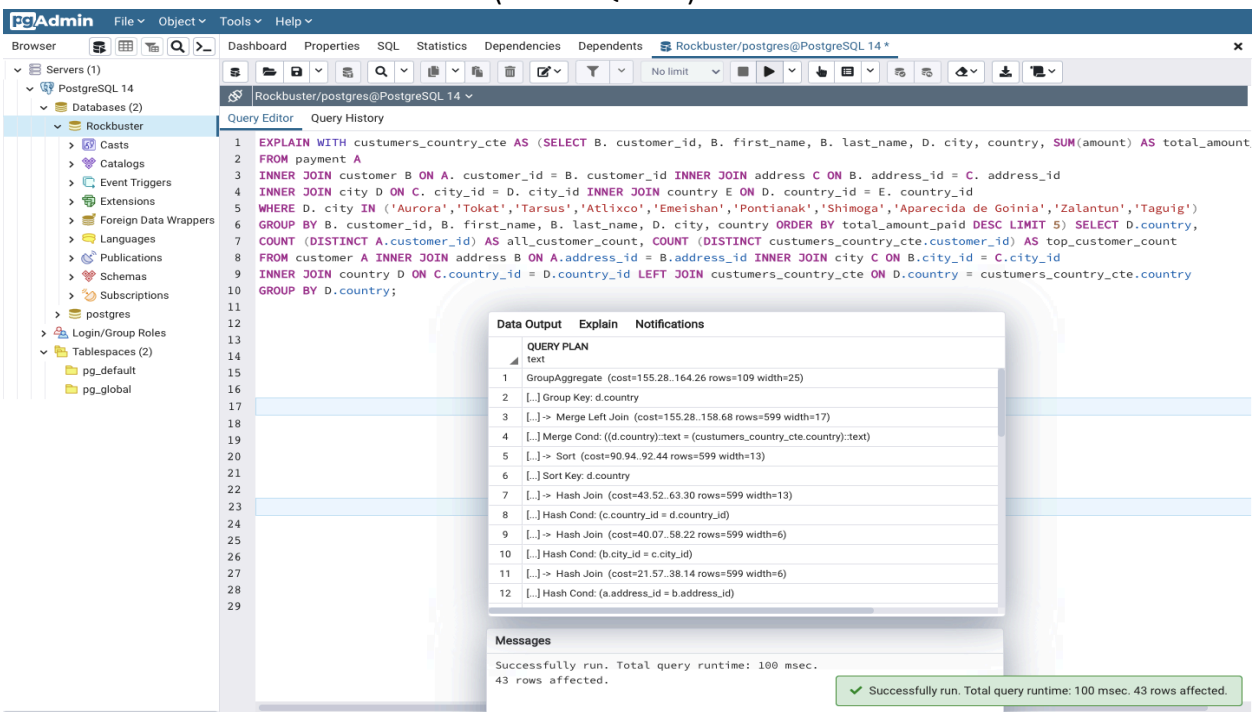
2A. Subquery may be more straightforward than CTE.

2B. CTE COST AND SPEED

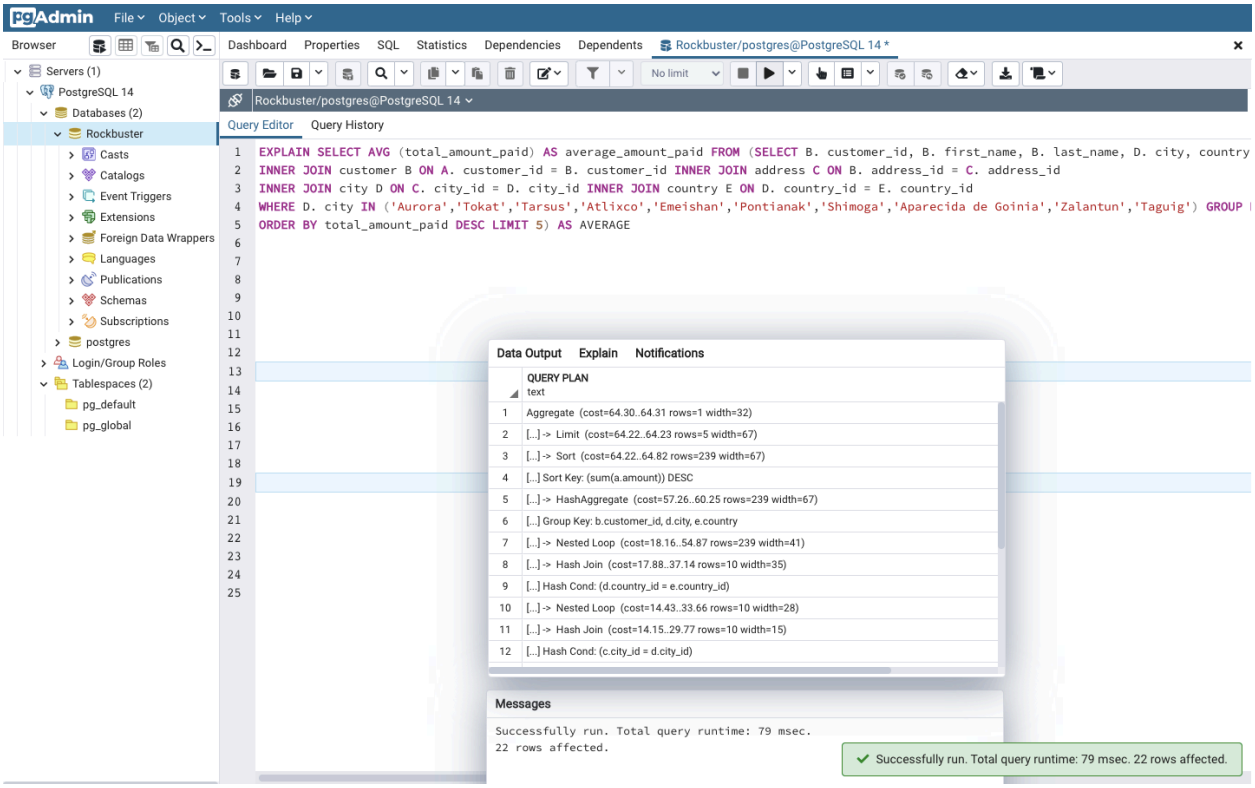
EXPLAIN AND SPEED TIME FOR CTE (STEP 1 QUERY)



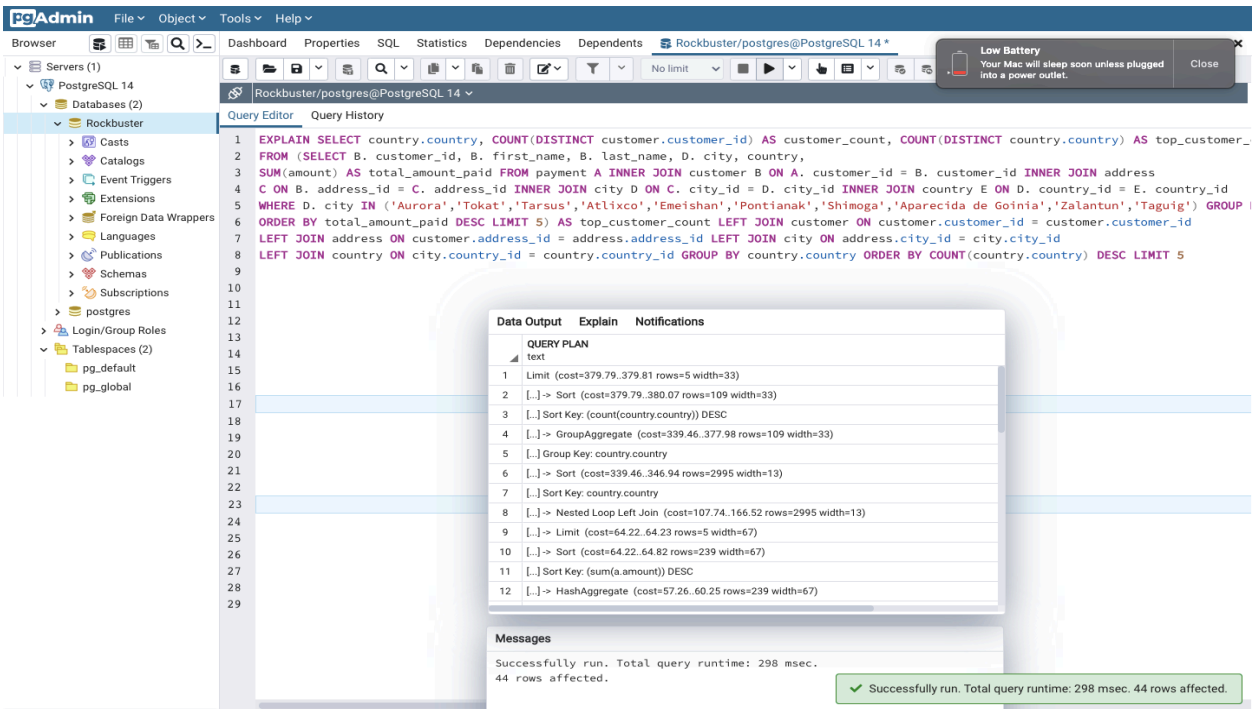
ii. EXPLAIN AND SPEED TIME FOR CTE (STEP 2 QUERY)



SUBQUERY COST AND SPEED (QUERY 1)



QUERY 2



2D. Quite surprised as the CTE appears faster to run than the subquery but I presume the subquery is more straightforward than the CTE.

3. There are complexities around writing the two approaches due to the technicalities. The main challenge is the development of the subqueries (coming up with the INNER JOINs of different keys/tables) that were nested in the CTEs. I believe being able to overcome the challenges around writing the subquery itself may get the job done as I presume the subqueries are more straightforward than the CTEs.