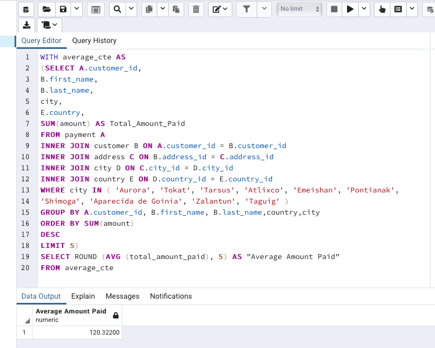
**ANSWERS 3.9**

1. **Answer the business questions from step 1 and 2 of task 3.8 using CTEs.**

* **1 and 2**

****

1. **WITH top\_customer\_cte AS**
2. **(SELECT D.country, COUNT(DISTINCT A.customer\_id) AS all\_customer\_count,**
3. **COUNT(DISTINCT top\_5\_customers.customer\_id) AS top\_customer\_count**
4. **FROM customer A**
5. **INNER JOIN address B ON A.address\_id = B.address\_id**
6. **INNER JOIN city C ON B.city\_id = C.city\_id**
7. **INNER JOIN country D ON C.country\_id = D.country\_id**
8. **LEFT JOIN**
9. **--INNER QUERY top 5 customers:**
10. **(SELECT country,**
11. **city,**
12. **A.customer\_id,**
13. **first\_name,**
14. **last\_name,**
15. **SUM(E.amount) AS total\_paid**
16. **FROM customer A**
17. **INNER JOIN address B ON A.address\_id = B.address\_id**
18. **INNER JOIN city C ON B.city\_id = C.city\_id**
19. **INNER JOIN country D ON C.country\_id = D.country\_id**
20. **INNER JOIN payment E ON A.customer\_id = E.customer\_id**
21. **WHERE city in ('Aurora', 'Tokat', 'Tarsus', 'Atlixco', 'Emeishan', 'Pontianak', 'Shimoga', 'Aparecida de**
22. **Goinia', 'Zalantun', 'Taguig')**
23. **GROUP BY city, country, A.customer\_id, first\_name, last\_name**
24. **ORDER BY SUM(E.amount) DESC**
25. **LIMIT 5) AS top\_5\_customers**
26. **ON D.country = top\_5\_customers.country**
27. **GROUP BY D.country**
28. **ORDER BY top\_customer\_count DESC**
29. **LIMIT 5)**
30. **SELECT \***
31. **FROM top\_customer\_cte**

* **3**

In both cases I used the same method. I followed what the theory from this unit explains. Since I did not need to look into the ERD and I already had the queries from the previous task I moved on to step 3.

I wrote the inner query into the CTE form by starting with WITH. I named both CTEs: “*average\_cte*” and “*customers\_per\_country\_cte.*” Later I inserted the inner query with parenthesis and preceded by the AS keyword.

Finally, I wrote the CTE names on both queries where appropriate to refer to them in the main query.

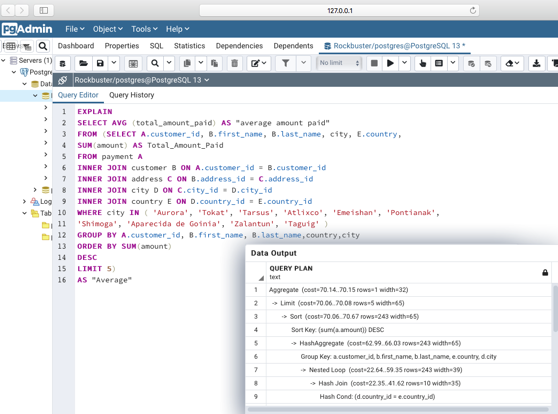
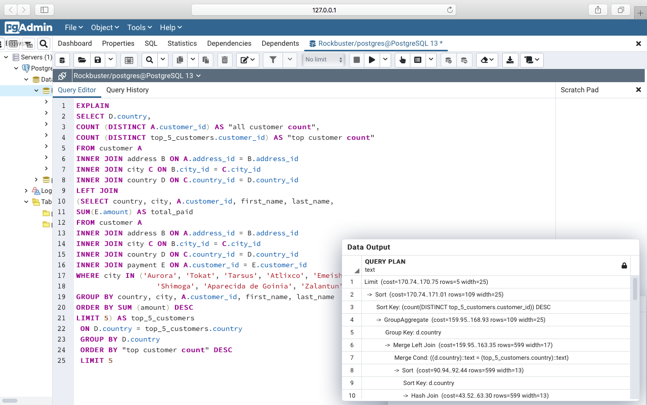
1. **Compare the performance of your CTEs and subqueries.**

* **1**

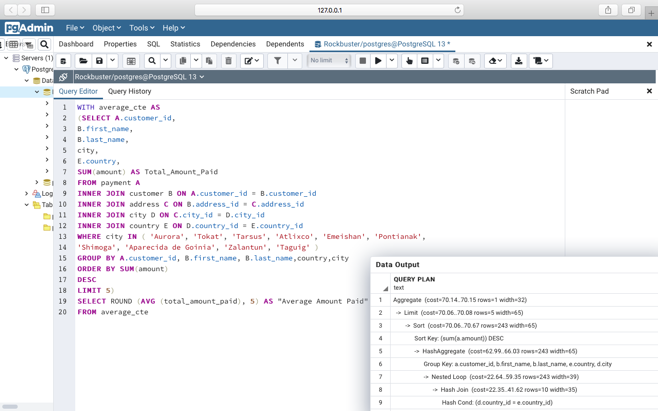
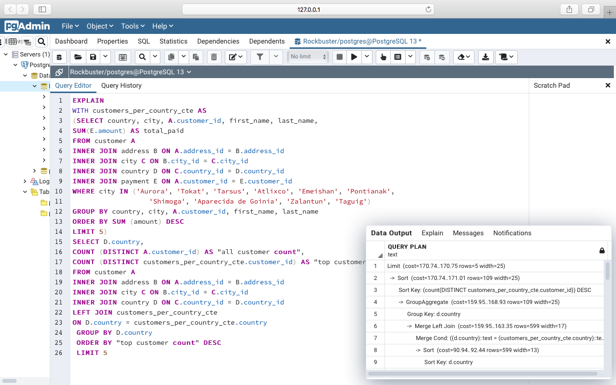
In this case I am not sure which approach will perform better. I assume CTE could be a bit faster since it took a bit less time to create that a subquery.

* **2**

The query plan for both subqueries

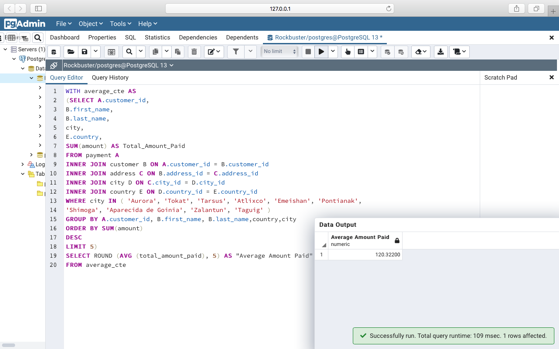
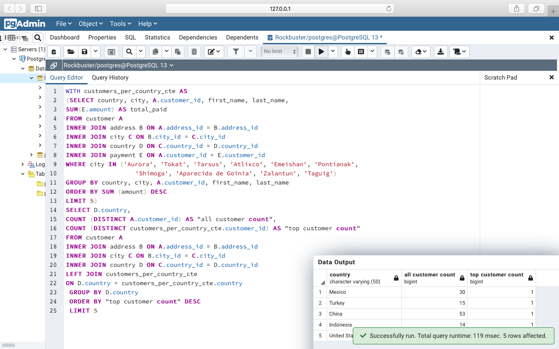
 

The query plan for both CTEs:

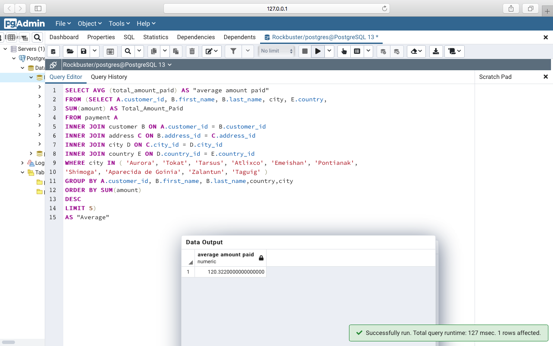
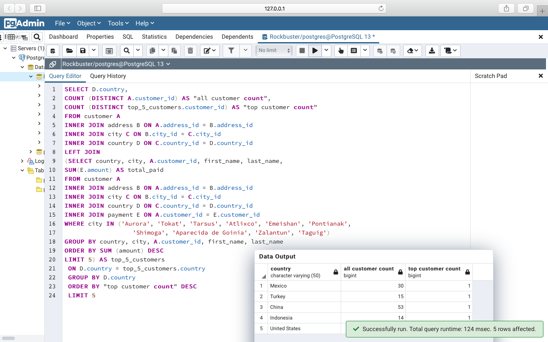
 

* **3 Actual running time (cost)**

For both CTEs

** **

For both subqueries

** **

* **4**

When running the query plan for both queries (CTEs and Subqueries) the time (cost) estimated was the same for both types. However, when actually running the query the actual time was shorter for the CTEs than the subqueries. That did come as a surprise to me because I expected the query plans to give more accurate information.

1. **Write 1 to 2 paragraphs on the challenges…**

The challenges I faced when replacing my subqueries with CTEs came when I had to understand how to write a CTE. After that another challenge was decide where to update the main query to make sure it referred to the CTE since the order is different than a subquery.