Gwyn Reniers

3.9

CTE

1. Query number 1

*I made a cte called avg\_amount\_paid that included the customer information I needed and had used subqueries to retrieve in the previous exercise. This was a little less confusing than the subquery task, at this point. It is still a hard process for my brain to walk through.*

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text

Description automatically generated

*This one required me to also look at previous submissions. I have the overall concepts, but I continue to make mistakes that don’t allow the code to run, and I don’t think I could do this without some sort of help yet, either another person or a template. I couldn’t just make this entire query up yet, even though I understand how it works.*

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

1. Which approach do you think will perform better and why?

*I think for this small data set the subquery will preform better, since the information will be retrieved and then used. I think the cte- where another table has to be created will take a little longer. With larger data sets, I think this will be the opposite, as the creation of a table will put the large amounts of information into one place.*

1. Compare the costs of all the queries by creating query plans for each one

|  |  |
| --- | --- |
| **QUERY** | **Cost** |
| Avg amount paid to top 5 customers CTE | cost=85.97..85.98 rows=1 width=32 |
| Avg amount paid to top 5 customers SUBQUERY | cost=28.99..29.01 rows=5 width=171 |
| Count of top 5 customers in country CTE | cost=3406.11..3406.12 rows=5 width=134 |
| Count of top 5 customers in country SUBQUERY | cost=1161.92..1161.93 rows=5 width=84 |
|  |  |
| **Query** | **Total query runtime** |
| Avg amount paid to top 5 customers CTE | 53 ms/ 49 rows |
| Avg amount paid to top 5 customers SUBQUERY | 39ms/22 rows |
| Count of top 5 customers in country CTE | 48ms/51 rows |
| Count of top 5 customers in country SUBQUERY | 109 ms |

Did the results surprise you? Write a few sentences to explain your answer*. The subquery was cheaper in both circumstances, and I figured it may be because of the size of the data set. A smaller set makes subqueries seem more feasible and efficient.*

**Step 3:**

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

*The challenges for the CTE’s are similar to the subqueries, although I find the subqueries more difficult to write. The CTE makes sense in that creating a new table in essence seems more efficient. I am still getting stuck with the alias piece of the writing these queries. But, when I see a template, it makes sense and I understand it, it’s trying to write it myself that is difficult.*

*I keep reminding myself that I have only 3-4 weeks of experience with SQL and that with practice and time it will make more sense. In the meantime, I feel like I felt learning Spanish and living in a foreign country, just saying (writing) things and hoping I get somewhere close enough that I can figure out my mistakes and make it work.*