

## 3.9- Common Table Expressions

### Answers 3.9

- Step 1:

```
Subquerie----SELECT AVG(Total_Amount_Paid) AS average_by_top5_customers
FROM
```

```
(SELECT B.customer_id, B.first_name,
      B.last_name, D.city, E.country,
SUM(A.amount) AS Total_Amount_Paid
FROM customer B
INNER JOIN payment A ON B.customer_id = A.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', 'Atlixco',
                'Xintai', 'Adoni', 'Dhule (Dhulia)',
                'Kurashiki', 'Pingxiang',
                'Sivas', 'Celaya', 'So Leopoldo')
GROUP BY B.customer_id, B.first_name, B.last_name, D.city, E.country
ORDER BY Total_Amount_Paid DESC
LIMIT 5) AS total_amount_paid;
```

```
CTE----WITH Average_total_amount_paid_cte(amount,customer_id, first_name,
                                           last_name, city, total_amount_paid)AS
```

```
(SELECT B.customer_id, B.first_name,
      B.last_name, D.city, E.country,
SUM(A.amount) AS Total_Amount_Paid
FROM customer B
INNER JOIN payment A ON B.customer_id = A.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', 'Atlixco',
                'Xintai', 'Adoni', 'Dhule (Dhulia)',
                'Kurashiki', 'Pingxiang',
                'Sivas', 'Celaya', 'So Leopoldo')
GROUP BY B.customer_id, B.first_name, B.last_name, D.city, E.country
ORDER BY Total_Amount_Paid DESC
LIMIT 5)
SELECT AVG(total_amount_paid) AS Averga_total_amount_paid
FROM Average_total_amount_paid_cte
```

Dashboard
Properties
SQL
Statistics
Dependencies
Dependents
Rockbuster/postgres@PostgreSQL 15\*

Rockbuster/postgres@PostgreSQL 15

No limit

Query

```

1 WITH Average_total_amount_paid_cte(amount,customer_id, first_name,
2   last_name, city, total_amount_paid) AS
3   (SELECT B.customer_id, B.first_name,
4    B.last_name, D.city, E.country,
5    SUM(A.amount) AS Total_Amount_Paid
6   FROM customer B
7   INNER JOIN payment A ON B.customer_id = A.customer_id
8   INNER JOIN address C ON B.address_id = C.address_id
9   INNER JOIN city D ON C.city_id = D.city_id
10  INNER JOIN country E ON D.country_id = E.country_id
11  WHERE D.city IN ('Aurora', 'Atlixco',
12   'Xintai', 'Adoni', 'Dhule (Dhulia)',
13   'Kurashiki', 'Pingxiang',
14   'Sivas', 'Celaya', 'So Leopoldo')
15  GROUP BY B.customer_id, B.first_name, B.last_name, D.city, E.country
16  ORDER BY Total_Amount_Paid DESC
17  LIMIT 5)
18  SELECT AVG(total_amount_paid) AS Averga_total_amount_paid
19  FROM Average_total_amount_paid_cte

```

Query History
Data Output
Messages

Averga_total_amount_paid	
numeric	
1	107.354000000000000000

After copying the subquery from the previous exercise, I define the CTE using the WITH clause. I gave it a CTE name; "Average\_total\_amount\_paid\_cte" I provided the AS keyword, and I added the query from step 2 in parentheses where it generates the average total amount paid.

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

## SUBQUERY

The screenshot shows a PostgreSQL query editor with a subquery used to calculate the average total amount paid for the top 5 customers. The query is as follows:

```
1 EXPLAIN
2 SELECT AVG(Total_Amount_Paid) AS average_by_top5_customers
3 FROM
4
5 (SELECT B.customer_id, B.first_name,
6      B.last_name, D.city, E.country,
7      SUM(A.amount) AS Total_Amount_Paid
8   FROM customer B
9   INNER JOIN payment A ON B.customer_id = A.customer_id
10  INNER JOIN address C ON B.address_id = C.address_id
11  INNER JOIN city D ON C.city_id = D.city_id
12  INNER JOIN country E ON D.country_id = E.country_id
13  WHERE D.city IN ('Aurora', 'Atlixco',
14                  'Xintai', 'Adoni', 'Dhule (Dhulia)',
15                  'Kurashiki', 'Pingxiang',
16                  'Sivas', 'Celaya', 'So Leopoldo'))
17 GROUP BY B.customer_id, B.first_name, B.last_name, D.city, E.country
18 ORDER BY Total_Amount_Paid DESC
19 LIMIT 5) AS total_amount_paid;
20
```

The query plan on the right shows the execution steps, including a subquery (lines 5-16) that calculates the total amount paid for each customer, followed by an aggregate operation (line 2) to calculate the average.

## CTE

The screenshot shows a PostgreSQL query editor with a Common Table Expression (CTE) used to calculate the average total amount paid for the top 5 customers. The query is as follows:

```
1 EXPLAIN
2 WITH Average_total_amount_paid_cte(amount, customer_id, first_name,
3      last_name, city, total_amount_paid) AS
4 (SELECT B.customer_id, B.first_name,
5      B.last_name, D.city, E.country,
6      SUM(A.amount) AS Total_Amount_Paid
7   FROM customer B
8   INNER JOIN payment A ON B.customer_id = A.customer_id
9   INNER JOIN address C ON B.address_id = C.address_id
10  INNER JOIN city D ON C.city_id = D.city_id
11  INNER JOIN country E ON D.country_id = E.country_id
12  WHERE D.city IN ('Aurora', 'Atlixco',
13                  'Xintai', 'Adoni', 'Dhule (Dhulia)',
14                  'Kurashiki', 'Pingxiang',
15                  'Sivas', 'Celaya', 'So Leopoldo'))
16 GROUP BY B.customer_id, B.first_name, B.last_name, D.city, E.country
17 ORDER BY Total_Amount_Paid DESC
18 LIMIT 5)
19 SELECT AVG(total_amount_paid) AS Average_total_amount_paid
20 FROM Average_total_amount_paid_cte
21
```

The query plan on the right shows the execution steps, including the CTE (lines 4-16) that calculates the total amount paid for each customer, followed by an aggregate operation (line 19) to calculate the average.

I don't see any difference in the cost between CTE and subquery.

### **Step 3:**

**Replacing the subquery with a CTE wasn't as straightforward as in the example in the exercise. I had to dive deeper into what to do. I need to practice more to become more confident in querying complex queries with confidence.**