

3.9 CTEs

1. Convert Subqueries to CTE

The screenshot shows a PostgreSQL query editor with a query using a Common Table Expression (CTE). The query is as follows:

```
1 WITH step_1_cte (customer_id, total_amount) AS
2   (SELECT B.customer_id,
3     SUM(A.amount) AS total_amount
4   FROM payment A
5   INNER JOIN customer B ON A.customer_id=B.customer_id
6   INNER JOIN address C ON B.address_id=C.address_id
7   INNER JOIN city D ON C.city_id=D.city_id
8   INNER JOIN country E ON D.country_id=E.country_id
9   WHERE D.city IN ('Aurora', 'Garden Grove', 'Salinas', 'Araatuba', 'Talavera')
10  GROUP BY B.customer_id, A.amount
11  ORDER BY A.amount DESC
12  LIMIT 5)
13 SELECT B.customer_id,
14   B.first_name,
15   B.last_name,
16   D.city,
17   E.country,
18   total_amount,
19   AVG(total_amount) AS average_payment
20 FROM step_1_cte
21 INNER JOIN customer B ON step_1_cte.customer_id=B.customer_id
22 INNER JOIN address C ON B.address_id=C.address_id
23 INNER JOIN city D ON C.city_id=D.city_id
24 INNER JOIN country E ON D.country_id=E.country_id
25 GROUP BY 1,
26   2,
27   3,
28   4,
29   5,
30   6
31 ORDER BY average_payment DESC
```

The results are displayed in the Data Output tab:

	customer_id integer	first_name character varying (45)	last_name character varying (45)	city character varying (50)	country character varying (50)	total_amount numeric	average_payment numeric
1	537	Clinton	Buford	Aurora	United States	26.97	26.970000000000000000
2	77	Jane	Bennett	Araatuba	Brazil	9.99	9.990000000000000000
3	269	Cassandra	Walters	Salinas	United States	9.99	9.990000000000000000
4	505	Rafael	Abney	Talavera	Philippines	9.99	9.990000000000000000
5	77	Jane	Bennett	Araatuba	Brazil	8.99	8.990000000000000000

Script:

```
WITH step_1_cte (customer_id, total_amount) AS
  (SELECT B.customer_id,
    SUM(A.amount) AS total_amount
  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', 'Garden Grove', 'Salinas', 'Araatuba', 'Talavera')
  GROUP BY B.customer_id, A.amount
  ORDER BY A.amount DESC
```

Alejandra Castro
30012025

```
        LIMIT 5)
SELECT B.customer_id,
       B.first_name,
       B.last_name,
       D.city,
       E.country,
       total_amount,
       AVG(total_amount) AS average_payment
FROM step_1_cte
INNER JOIN customer B ON step_1_cte.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 1,
        2,
        3,
        4,
        5,
        6
ORDER BY average_payment DESC
```

The screenshot shows a PostgreSQL query editor with a query that uses CTEs to calculate the average payment per country, limited to the top 5 countries. The query is as follows:

```
1 WITH top_consumer_cte (customer_id, total_amount, country) AS
2   (SELECT B.customer_id,
3     SUM(A.amount) AS total_amount,
4     E.country
5   FROM payment A
6   INNER JOIN customer B ON A.customer_id=B.customer_id
7   INNER JOIN address C ON B.address_id=C.address_id
8   INNER JOIN city D ON C.city_id=D.city_id
9   INNER JOIN country E ON D.country_id=E.country_id
10  WHERE D.city IN ('Aurora', 'Garden Grove', 'Salinas', 'Araatuba', 'Talavera')
11  GROUP BY B.customer_id, A.amount, E.country
12  ORDER BY A.amount DESC
13  LIMIT 5),
14 all_consumer_cte AS
15 (SELECT COUNT(DISTINCT B.customer_id) AS all_consumer_count,
16   E.country
17 FROM payment A
18 INNER JOIN customer B ON A.customer_id=B.customer_id
19 INNER JOIN address C ON B.address_id=C.address_id
20 INNER JOIN city D ON C.city_id=D.city_id
21 INNER JOIN country E ON D.country_id=E.country_id
22 WHERE D.city IN ('Aurora', 'Garden Grove', 'Salinas', 'Araatuba', 'Talavera')
23 GROUP BY B.customer_id, A.amount, E.country
24 ORDER BY A.amount DESC)
25 SELECT E.country,
26   COUNT(B.customer_id) AS all_customer_count,
27   COUNT(DISTINCT A.customer_id) AS top_consumer_count
28 FROM top_consumer_cte A
29 INNER JOIN customer B ON A.customer_id = B.customer_id
30 INNER JOIN address C ON B.address_id = C.address_id
31 INNER JOIN city D ON C.city_id = D.city_id
32 INNER JOIN country E ON D.country_id = E.country_id
33 GROUP BY 1
34 ORDER BY top_consumer_count DESC
35 LIMIT 5
```

The results of the query are shown in the Data Output tab:

	country character varying (50)	all_customer_count bigint	top_consumer_count bigint
1	United States	2	2
2	Brazil	2	1
3	Philippines	1	1

Alejandra Castro
30012025

Script:

```
WITH top_consumer_cte (customer_id, total_amount, country) AS
    (SELECT B.customer_id,
            SUM(A.amount) AS total_amount,
            E.country
    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', 'Garden Grove', 'Salinas', 'Araatuba', 'Talavera')
    GROUP BY B.customer_id, A.amount, E.country
    ORDER BY A.amount DESC
    LIMIT 5),
all_consumer_cte AS
    (SELECT COUNT(DISTINCT B.customer_id) AS all_consumer_count,
            E.country
    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', 'Garden Grove', 'Salinas', 'Araatuba', 'Talavera')
    GROUP BY B.customer_id, A.amount, E.country
    ORDER BY A.amount DESC)
SELECT E.country,
       COUNT(B.customer_id) AS all_customer_count,
       COUNT(DISTINCT A.customer_id) AS top_consumer_count
FROM top_consumer_cte 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 1
ORDER BY top_consumer_count DESC
LIMIT 5
```