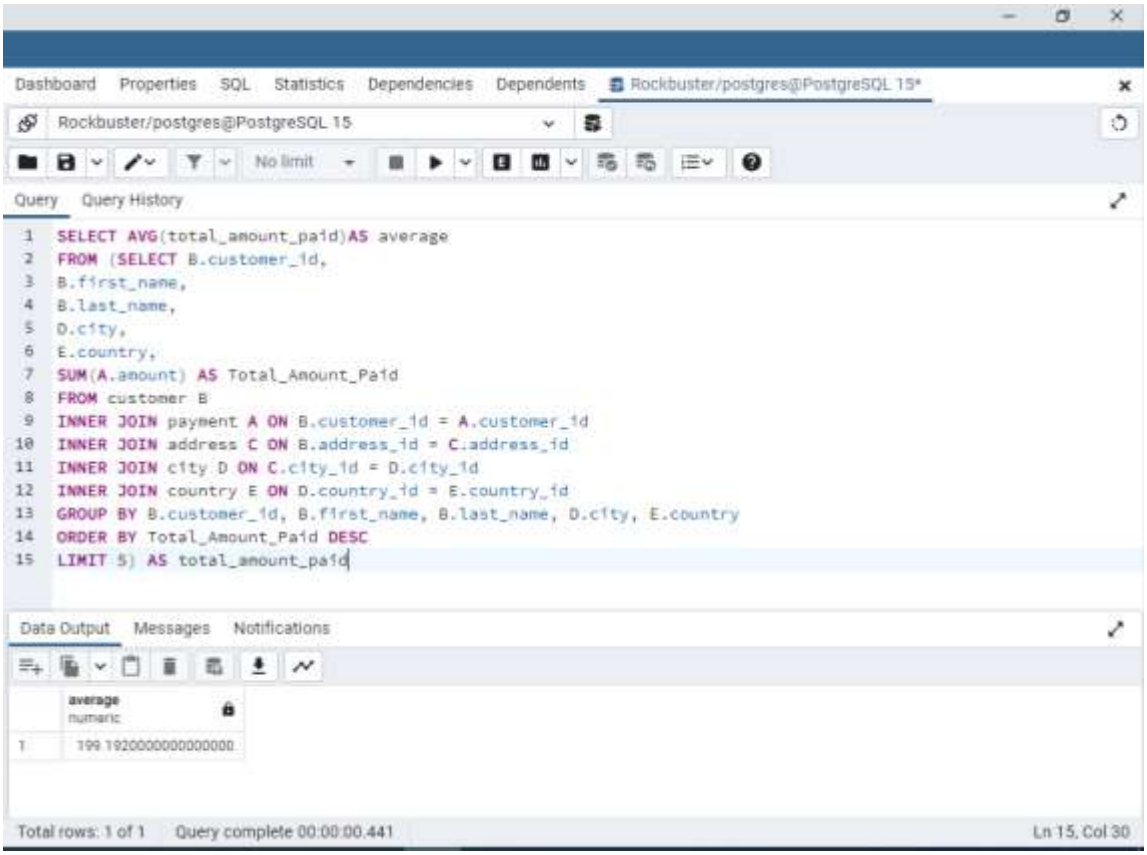
1. Copy the query you wrote in step 3 of the task from Exercise 3.7: Joining Tables of Data into the Query Tool. This will be your subquery, so give it an alias, “total_amount_paid,” and add parentheses around it.
2. Write an outer statement to calculate the average amount paid.
3. 3.Add your subquery to the outer statement. It will go in either the SELECT, WHERE, or FROM clause. (Hint: When referring to the subquery in your outer statement, make sure to use the subquery’s alias, “total_amount_paid”.)
4. 4.If you've done everything correctly, pgAdmin 4 will require you to add an alias after the subquery. Go ahead and call it “average”.

1.

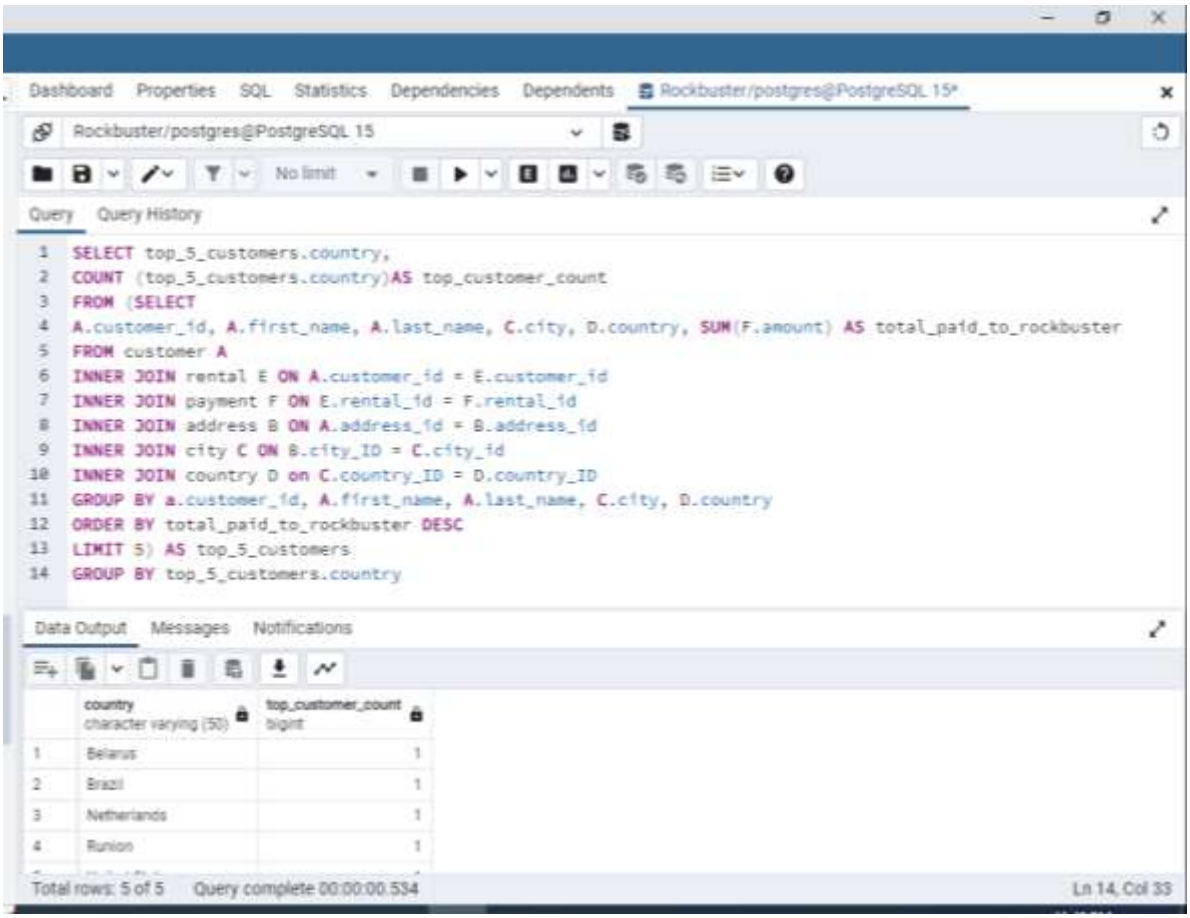
```
SELECT B.customer_id,

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

GROUP BY B.customer_id, B.first_name, B.last_name, D.city, E.country ORDER BY Total_Amount_Paid DESC LIMIT 5
```

2.



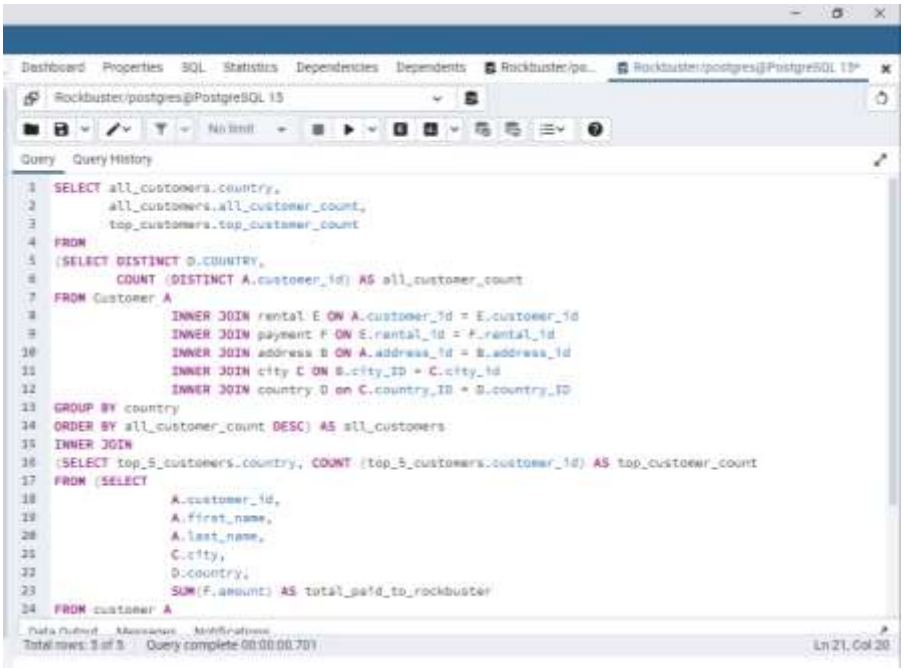
Step 2: Find out how many of the top 5 customers are based within each country.
Your final output should include 3 columns:

- "country"
- "all_customer_count" with the total number of customers in each country
- "top_customer_count" showing how many of the top 5 customers live in each country

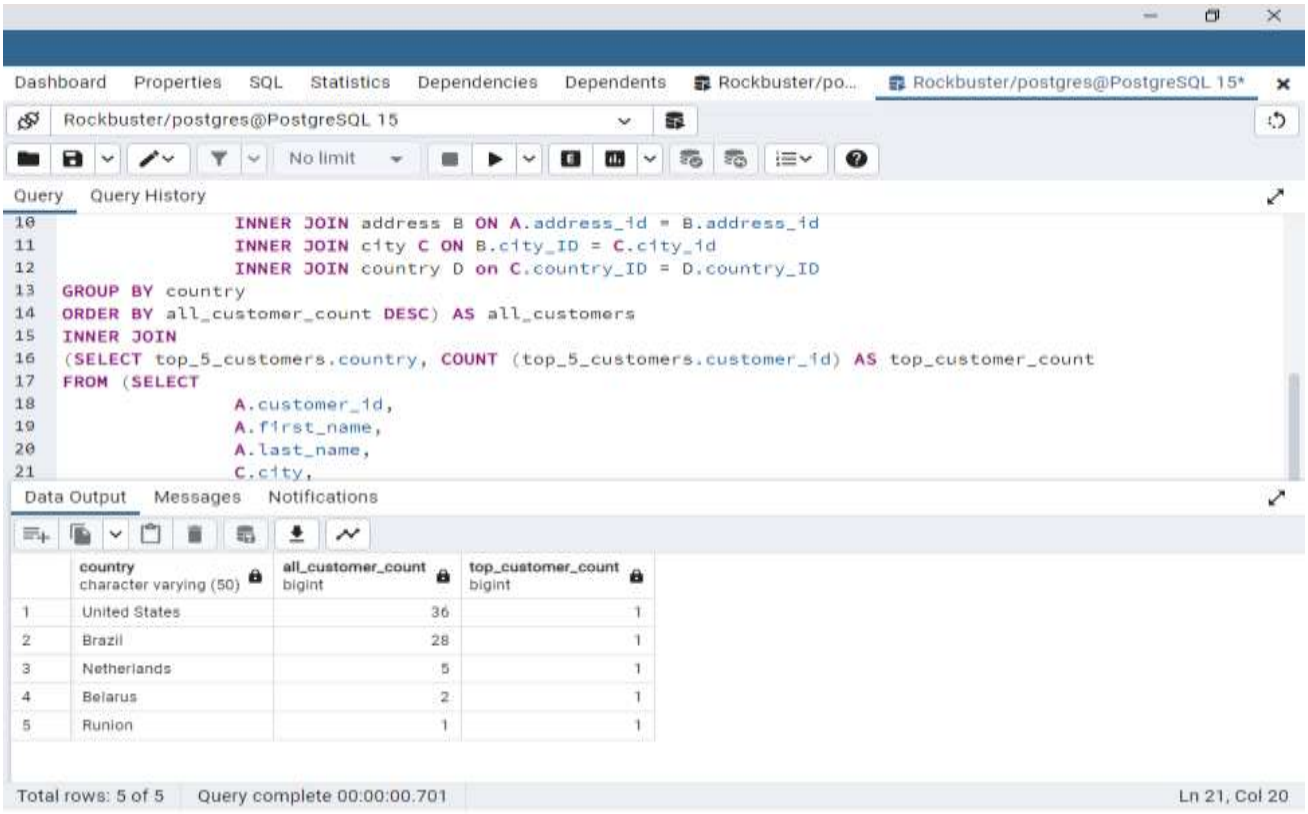



```
SELECT all_customers.country,
        all_customers.all_customer_count,
                                top_customers.top_customer_count
FROM
(SELECT DISTINCT D.COUNTRY,
                                 COUNT (DISTINCT A.customer_id) AS all_customer_count
FROM Customer A
                                        INNER JOIN rental E ON A.customer_id = E.customer_id
    INNER JOIN payment F ON E.rental_id = F.rental_id
                                            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
```

```
GROUP BY country
ORDER BY all_customer_count DESC) AS all_customers
INNER JOIN
(SELECT top_5_customers.country, COUNT (top_5_customers.customer_id) AS top_customer_count
FROM (SELECT
        A.customer_id,
                A.first_name,
                A.last_name,
                C.city,
                D.country,
                SUM(F.amount) AS total_paid_to_rockbuster
FROM customer A
        INNER JOIN rental E ON A.customer_id = E.customer_id
        INNER JOIN payment F ON E.rental_id = F.rental_id
        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
GROUP BY a.customer_id, A.first_name, A.last_name, C.city, D.country
ORDER BY total_paid_to_rockbuster DESC
LIMIT 5)AS top_5_customers
GROUP BY top_5_customers.country) AS top_customers
ON all_customers.country = top_customers.country
```



```
SELECT all_customers.country,
       all_customers.all_customer_count,
       top_customers.top_customer_count
FROM
(SELECT DISTINCT D.COUNTRY,
        COUNT (DISTINCT A.customer_id) AS all_customer_count
FROM Customer A
                INNER JOIN rental E ON A.customer_id = E.customer_id
                INNER JOIN payment F ON E.rental_id = F.rental_id
                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
GROUP BY country
ORDER BY all_customer_count DESC) AS all_customers
INNER JOIN
(SELECT top_5_customers.country, COUNT (top_5_customers.customer_id) AS top_customer_count
FROM (SELECT
                A.customer_id,
                A.first_name,
                A.last_name,
                C.city,
                D.country,
                SUM(F.amount) AS total_paid_to_rockbuster
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
GROUP BY country
ORDER BY all_customer_count DESC) AS all_customers
INNER JOIN
(SELECT top_5_customers.country, COUNT (top_5_customers.customer_id) AS top_customer_count
FROM (SELECT
                A.customer_id,
                A.first_name,
                A.last_name,
                C.city,
```

	country character varying (50)	all_customer_count bigint	top_customer_count bigint
1	United States	36	1
2	Brazil	28	1
3	Netherlands	5	1
4	Belarus	2	1
5	Runion	1	1

1. Write 1 to 2 short paragraphs on the following:
    - Do you think steps 1 and 2 could be done without using subqueries?
    - When do you think subqueries are useful?

- There's no way I could figure out how to do them without using subqueries (at least not while keeping them dynamic so that the results update if the database updates). Maybe with more practice I would eventually figure it out. Step 1 - average of top 5 customers should be the easier of the two to figure out how to do it without subqueries… but I would think that ordering and limiting for top 5 would already be the subquery, and I'm not sure how to do an "AVG" of limited results without having those limits be the sub-query itself.
- I think subqueries are useful and necessary any time you want to know aggregated or limited details about anything within a database that is changing/updated on a regular basis.