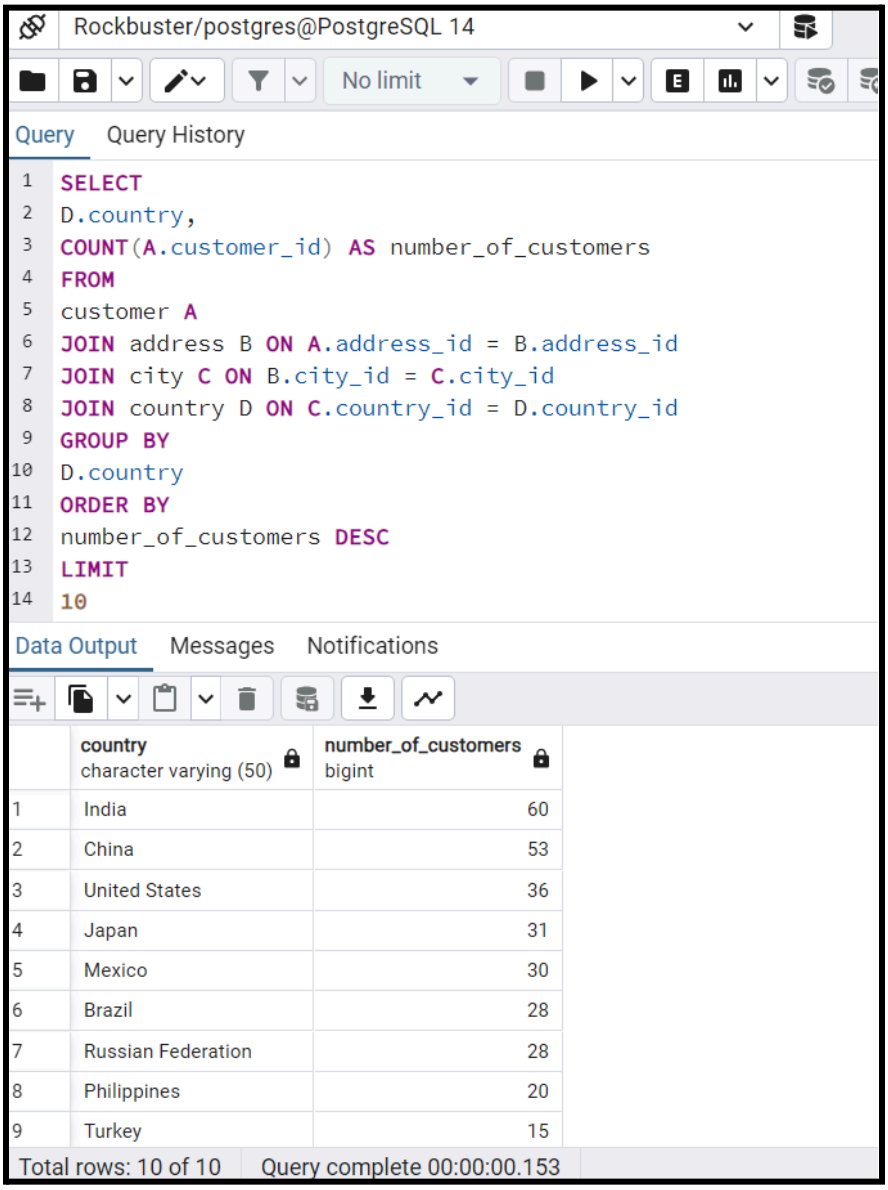
Exercise 3.7
Katherine Lecce

1. **Write a query to find the top 10 countries for Rockbuster in terms of customer numbers.**



How did you approach this query and why? You must be able to explain your thought process when writing queries, especially for future interviews.

1. We want to connect or **SELECT** the country column from the country table which is represented by 'D.'
2. Then we want to count the # of 'customer_ids' from the customer table which is represented by 'A' and will change the column name to "number of customers" on our output.
3. Then by using the **FROM** function, we use the customer table represented as 'A'
4. Using the **JOIN** function we connect address table to retrieve the 'address_id' field
5. We also want to **JOIN** the city's table 'C.' with the address table with the 'city_id' field
6. Lastly use the **JOIN** function to join the country table represented as 'D' with the city table and the country_id field
7. **GROUP BY** 'D.country': This groups the results by the country column from the country table.
8. Use **ORDER BY** function to sort the number of customer column in descending order **(DESC)**
9. **LIMIT** function (by 10) tells the Quarry how many rows of results you want - top 10 results.

2. Write a query to identify the top 10 cities that fall within the top 10 countries you identified in step 1.

←

```sql
SELECT
D.country,
C.city,
COUNT(A.customer_id) AS number_of_customers
FROM
customer A
JOIN address B ON A.address_id = B.address_id
JOIN city C ON B.city_id = C.city_id
JOIN country D ON C.country_id = D.country_id
WHERE
D.country IN (
SELECT
D.country
FROM
customer A
JOIN address B ON A.address_id = B.address_id
JOIN city C ON B.city_id = C.city_id
JOIN country D ON C.country_id = D.country_id
GROUP BY
D.country
ORDER BY
COUNT(A.customer_id) DESC
LIMIT
10
)
GROUP BY
D.country, C.city
ORDER BY
COUNT(A.customer_id) DESC
LIMIT
```

Total rows: 10 of 10 Query complete 00:00:00.135

Data Output | Messages | Notifications

	country character varying (50)	city character varying (50)	number_of_customers bigint
1	United States	Aurora	2
2	Mexico	Acua	1
3	United States	Citrus Heights	1
4	Japan	Iwaki	1
5	India	Ambattur	1
6	China	Shanwei	1
7	Brazil	So Leopoldo	1
8	Russian Federation	Teboksary	1
9	China	Tianjin	1
10	Indonesia	Cianjur	1

1. We first want to use the **SELECT** Function:
   → Select the country column from the country table that aligns with 'D.'
   → Select the city column from the city table that aligns with the 'c.'
   → **COUNT** function will then count the customer_id values from the customer table aligned with 'A.' resulting in the count as number_of_customers.
2. **FROM** clause:
   → Then using the from clause select the customer table that aligns with 'A."
   → Join the address table (B) with the address_id field.
   → Join the city table (c.) with the address table with the city_id.
   → Join the country table with (D), with the city table using the country_id (D.)

3. **WHERE** clause:
   → This allows us to filter the data to only include information where the country values are aligned with quarry D (country table)
4. **GROUP BY** clause:
   → allows the Country column from the country table and the city column from the city table.
5. **ORDER BY** clause:
   → This orders the result set by the number_of_customer column in descending order.
6. **LIMIT** clause:
   → This limits the number of rows returned by the query to 10.

Write a short explanation of how you approached this query and why

I like to start by writing out what I need to retrieve from where before I start to think about how the quarry will be set up. This allows me to organize where I need to look to retrieve the data from the set since all of the columns are marked by numerical letters from where in the set I want to retrieve them and how the Query will function when trying to combined the different parts of the query in longer queries like these. I also like to stop and make sure that I don't write the whole Query all at once and check as I go along before adding more. This allows for checking mistakes along the way so it doesn't all have to be done at the end.

3. **Now write a query to find the top 5 customers from the top 10 cities who've paid the highest total amounts to Rockbuster. The customer team would like to reward them for their loyalty!**


```sql
SELECT
A.customer_id,
A.first_name,
A.last_name,
D.country,
C.city,
SUM(B.amount) AS total_amount_paid
FROM
customer A
JOIN
payment B ON A.customer_id = B.customer_id
JOIN
address E ON A.address_id = E.address_id
JOIN
city C ON E.city_id = C.city_id
JOIN
country D ON C.country_id = D.country_id
WHERE
C.city IN (
SELECT city
FROM customer A
JOIN address E ON A.address_id = E.address_id
JOIN city C ON E.city_id = C.city_id
JOIN country D ON C.country_id = D.country_id
GROUP BY C.city
ORDER BY COUNT(A.customer_id) DESC
LIMIT 10
)
GROUP BY
A.customer_id, A.first_name, A.last_name, D.country, C.city
```

```
A.customer_id, A.first_name, A.last_name, D.country, C.city
ORDER BY
total_amount_paid DESC
LIMIT
5;
```

Data Output | Messages | Notifications

	customer_id integer	first_name character varying (45)	last_name character varying (45)	country character varying (50)	city character varying (50)	total_amount_paid numeric
1	148	Eleanor	Hunt	Runion	Saint-Denis	211.55
2	144	Clara	Shaw	Belarus	Molodetno	189.60
3	566	Casey	Mena	Turkey	Tokat	130.68
4	84	Sara	Perry	Mexico	Atlixco	128.70
5	506	Leslie	Seward	Indonesia	Pontianak	123.72