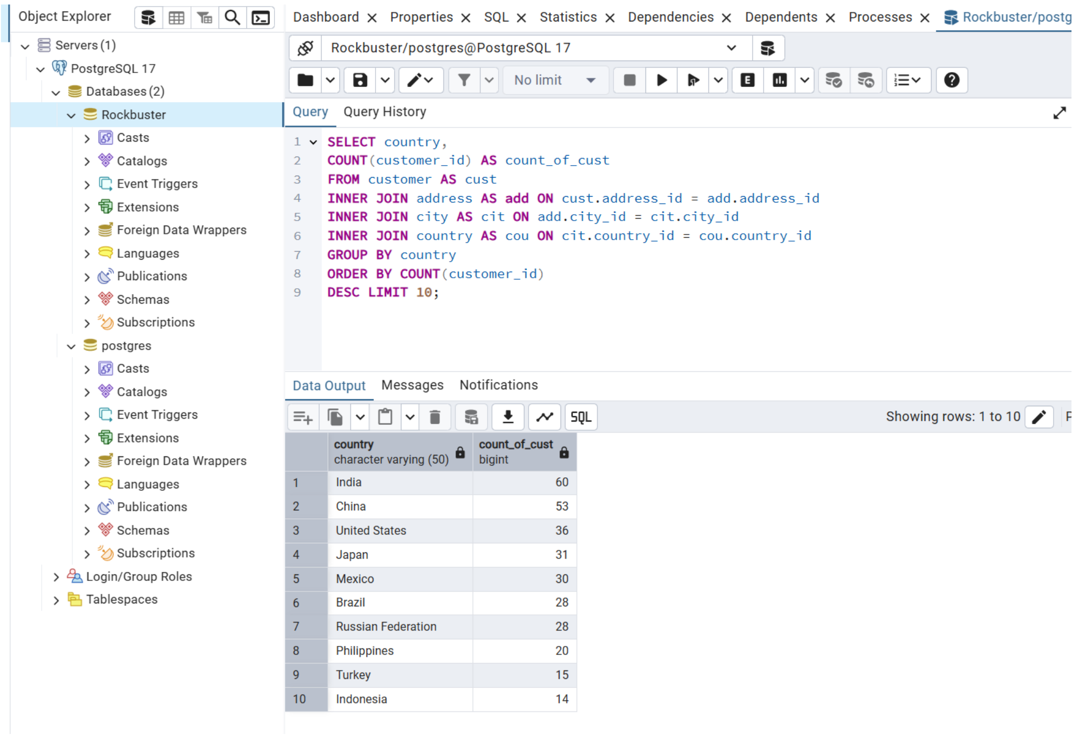
1. Write a query to find the top 10 countries for Rockbuster in terms of customer numbers. (Tip: you’ll have to use GROUP BY and ORDER BY, both of which follow the join.)

```
SELECT country,
COUNT(customer_id) AS count_of_cust
FROM customer AS cust
INNER JOIN address AS add ON cust.address_id = add.address_id
INNER JOIN city AS cit ON add.city_id = cit.city_id
INNER JOIN country AS cou ON cit.country_id = cou.country_id
GROUP BY country
ORDER BY COUNT(customer_id)
DESC LIMIT 10;
```



```
SELECT country,
COUNT(customer_id) AS count_of_cust
FROM customer AS cust
INNER JOIN address AS add ON cust.address_id = add.address_id
INNER JOIN city AS cit ON add.city_id = cit.city_id
INNER JOIN country AS cou ON cit.country_id = cou.country_id
GROUP BY country
ORDER BY COUNT(customer_id)
DESC LIMIT 10;
```

	country character varying (50)	count_of_cust bigint
1	India	60
2	China	53
3	United States	36
4	Japan	31
5	Mexico	30
6	Brazil	28
7	Russian Federation	28
8	Philippines	20
9	Turkey	15
10	Indonesia	14

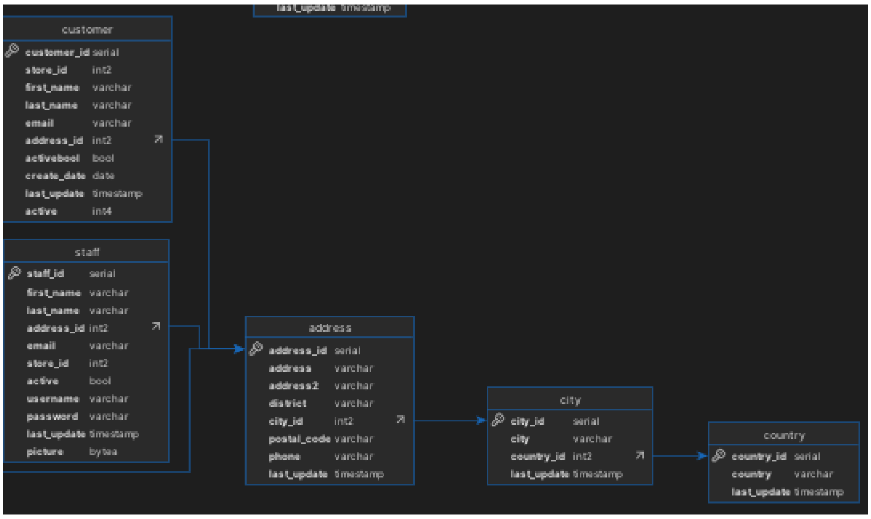
To find the **top 10 countries with the most customers**, I broke the problem down into the following steps:

1. **Understand the Table Relationships**

   The customer table does not directly include country information. So, I traced the path:

   - customer → address → city → country

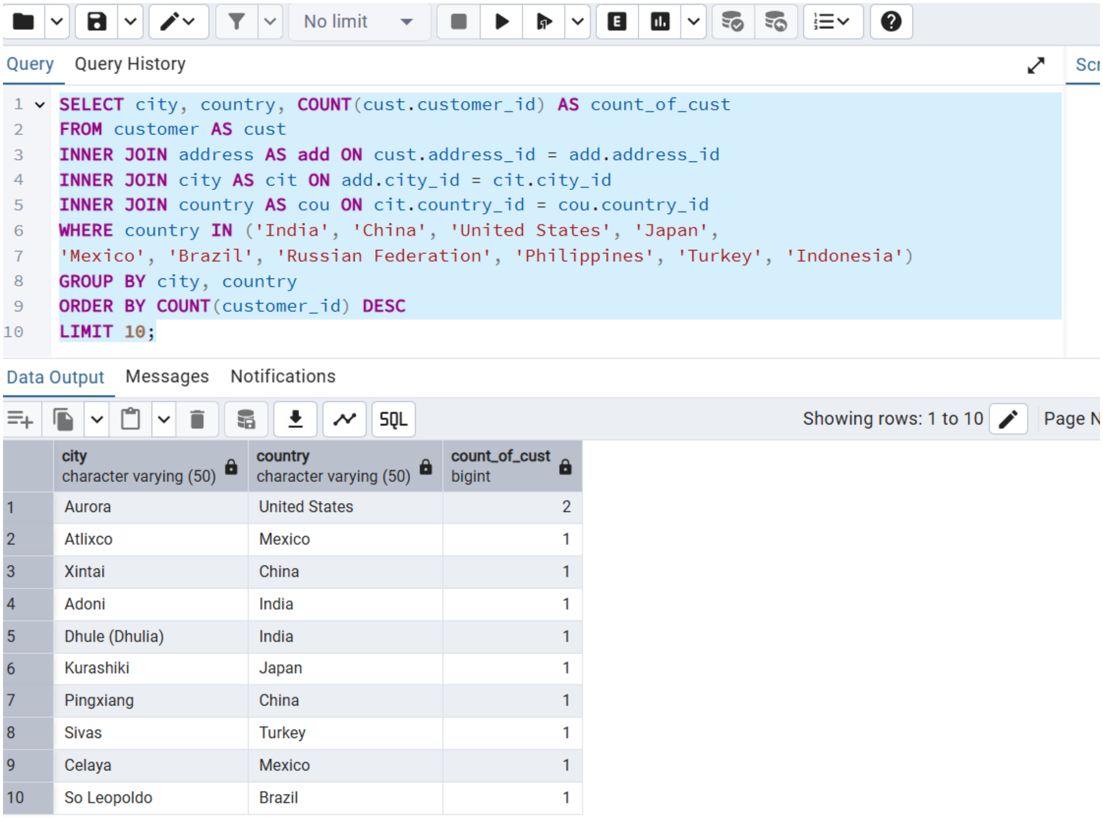
     Using this hierarchy, I knew I had to use three INNER JOINs to get to the country name.



2. **Join the Tables**
   I joined the customer, address, city, and country tables using appropriate foreign key relationships:
   - cust.address_id = add.address_id
   - add.city_id = cit.city_id
   - cit.country_id = cou.country_id
3. **Count the Customers**
   I used COUNT(customer_id) to calculate the number of customers in each country.
4. **Group and Sort**
   - Grouped by country to get country-level data.
   - Ordered the results in **descending order** to bring the countries with the highest customer count to the top.
5. **Limit to Top 10**
   Finally, I applied LIMIT 10 to get only the top 10 countries.

2. write a query to identify the top 10 cities that fall within the top 10 countries you identified in step 1. (Hint: the top 10 cities can be in any of the countries identified you don't need to create a separate list for each country.)

```
SELECT city, country, COUNT(cust.customer_id) AS count_of_cust
FROM customer AS cust
INNER JOIN address AS add ON cust.address_id = add.address_id
INNER JOIN city AS cit ON add.city_id = cit.city_id
INNER JOIN country AS cou ON cit.country_id = cou.country_id
WHERE country IN ('India', 'China', 'United States', 'Japan',
'Mexico', 'Brazil', 'Russian Federation', 'Philippines', 'Turkey', 'Indonesia')
GROUP BY city, country
ORDER BY COUNT(customer_id) DESC
LIMIT 10;
```



```sql
SELECT city, country, COUNT(cust.customer_id) AS count_of_cust
FROM customer AS cust
INNER JOIN address AS add ON cust.address_id = add.address_id
INNER JOIN city AS cit ON add.city_id = cit.city_id
INNER JOIN country AS cou ON cit.country_id = cou.country_id
WHERE country IN ('India', 'China', 'United States', 'Japan',
'Mexico', 'Brazil', 'Russian Federation', 'Philippines', 'Turkey', 'Indonesia')
GROUP BY city, country
ORDER BY COUNT(customer_id) DESC
LIMIT 10;
```

	city character varying (50)	country character varying (50)	count_of_cust bigint
1	Aurora	United States	2
2	Atlixco	Mexico	1
3	Xintai	China	1
4	Adoni	India	1
5	Dhule (Dhulia)	India	1
6	Kurashiki	Japan	1
7	Pingxiang	China	1
8	Sivas	Turkey	1
9	Celaya	Mexico	1
10	So Leopoldo	Brazil	1

For this query, my goal was to find the **top 10 cities** with the highest number of customers, but **only within the top 10 countries** identified earlier. Since I already had the list of those countries, I decided to use a WHERE IN (...) clause to filter the dataset directly.

I joined the relevant tables (customer, address, city, and country) to access both the city and country for each customer. Then, I grouped the data by both city and country and used COUNT(customer_id) to get the number of customers in each city. Finally, I sorted the results in descending order by customer count and used LIMIT 10 to return the top-performing cities.

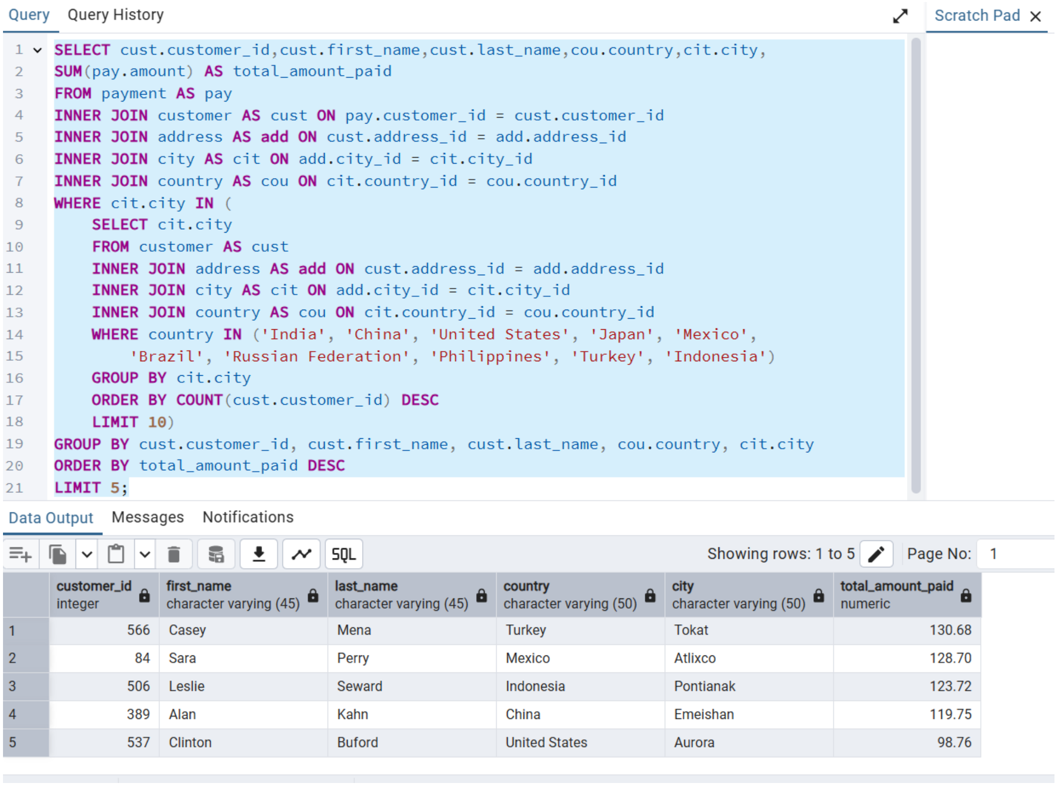
This approach is straightforward and efficient when the list of countries is known in advance. It avoids the need for a subquery and clearly reflects the business logic behind the analysis.

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!**

- Tip: After the join syntax, you'll need to use the WHERE clause with an operator, followed by GROUP BY and ORDER BY. Your output should include the following columns: Customer ID, Customer First Name and Last Name, Country, City, and Total Amount Paid.

SELECT cust.customer_id,cust.first_name,cust.last_name,cou.country,cit.city,

```
SUM(pay.amount) AS total_amount_paid
FROM payment AS pay
INNER JOIN customer AS cust ON pay.customer_id = cust.customer_id
INNER JOIN address AS add ON cust.address_id = add.address_id
INNER JOIN city AS cit ON add.city_id = cit.city_id
INNER JOIN country AS cou ON cit.country_id = cou.country_id
WHERE cit.city IN (
          SELECT cit.city
    FROM customer AS cust
    INNER JOIN address AS add ON cust.address_id = add.address_id
    INNER JOIN city AS cit ON add.city_id = cit.city_id
    INNER JOIN country AS cou ON cit.country_id = cou.country_id
    WHERE country IN ('India', 'China', 'United States', 'Japan', 'Mexico',
        'Brazil', 'Russian Federation', 'Philippines', 'Turkey', 'Indonesia')
    GROUP BY cit.city
    ORDER BY COUNT(cust.customer_id) DESC
    LIMIT 10)
GROUP BY cust.customer_id, cust.first_name, cust.last_name, cou.country,
cit.city
ORDER BY total_amount_paid DESC
LIMIT 5;
```



```
SELECT cust.customer_id,cust.first_name,cust.last_name,cou.country,cit.city,
SUM(pay.amount) AS total_amount_paid
FROM payment AS pay
INNER JOIN customer AS cust ON pay.customer_id = cust.customer_id
INNER JOIN address AS add ON cust.address_id = add.address_id
INNER JOIN city AS cit ON add.city_id = cit.city_id
INNER JOIN country AS cou ON cit.country_id = cou.country_id
WHERE cit.city IN (
    SELECT cit.city
    FROM customer AS cust
    INNER JOIN address AS add ON cust.address_id = add.address_id
    INNER JOIN city AS cit ON add.city_id = cit.city_id
    INNER JOIN country AS cou ON cit.country_id = cou.country_id
    WHERE country IN ('India', 'China', 'United States', 'Japan', 'Mexico',
        'Brazil', 'Russian Federation', 'Philippines', 'Turkey', 'Indonesia')
    GROUP BY cit.city
    ORDER BY COUNT(cust.customer_id) DESC
    LIMIT 10)
GROUP BY cust.customer_id, cust.first_name, cust.last_name, cou.country, cit.city
ORDER BY total_amount_paid DESC
LIMIT 5;
```

	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	566	Casey	Mena	Turkey	Tokat	130.68
2	84	Sara	Perry	Mexico	Atlixco	128.70
3	506	Leslie	Seward	Indonesia	Pontianak	123.72
4	389	Alan	Kahn	China	Emeishan	119.75
5	537	Clinton	Buford	United States	Aurora	98.76

The objective was to identify the top 5 customers who have paid the highest total amounts, but only among those living in the top 10 cities from the top 10 countries (as identified earlier).

Here’s how I approached the query:

1. **Joined Relevant Tables**
   I joined payment, customer, address, city, and country to get all the required columns and link each payment to a customer, their city, and country.
2. **Filtered for Top 10 Cities**
   Using a subquery in the WHERE clause, I selected only those customers who live in one of the top 10 cities (from top 10 countries). These cities were identified by counting customer volume per city.
3. **Aggregation**
   I used SUM(amount) to calculate the **total amount paid by each customer**.
4. **Grouped and Sorted**
   I grouped by customer_id, name, city, and country to get correct aggregation and then ordered the results by total amount paid in descending order.
5. **Limit to Top 5**
   Finally, I used LIMIT 5 to return the most loyal (highest-paying) customers from the top 10 cities.
   This approach allows Rockbuster to reward its most valuable customers in its most active cities—a great strategy for customer retention and engagement.