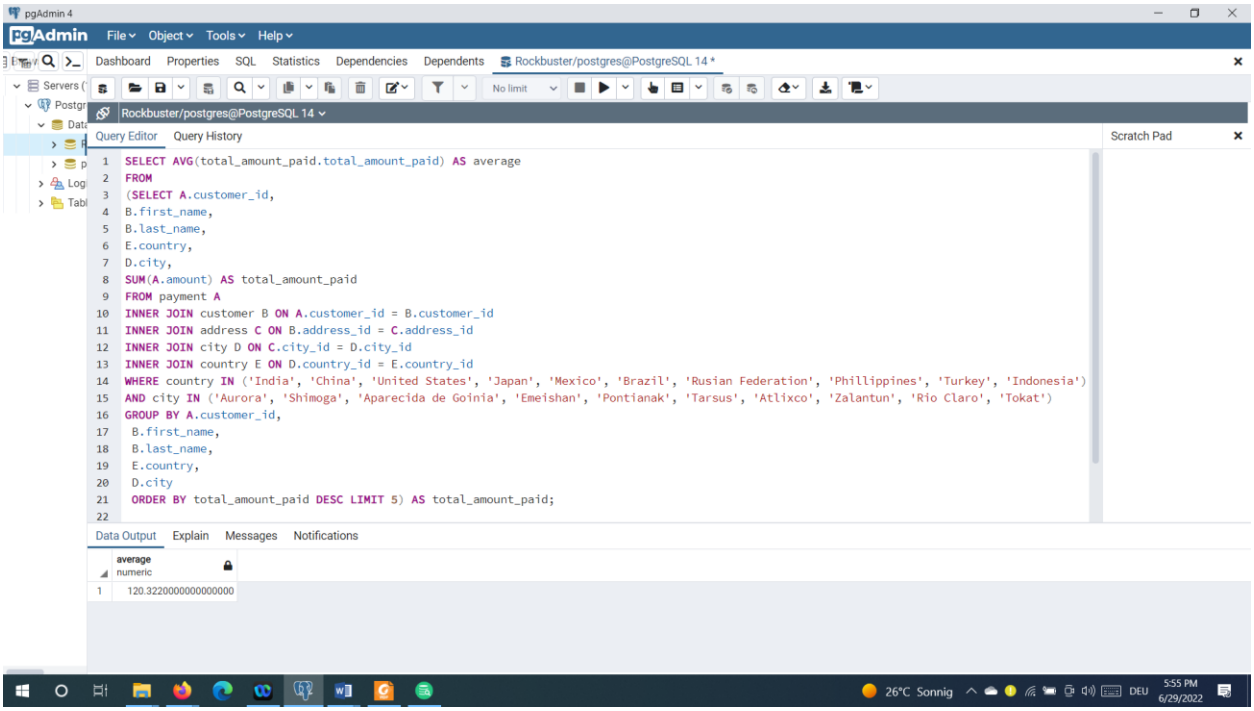
3.8: Performing Subqueries

Step 1: Find the average amount paid by the top 5 customers

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. 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. If you've done everything correctly, pgAdmin 4 will require you to add an alias after the subquery. Go ahead and call it “average”.
5. Copy-paste your queries and the final data output from pgAdmin 4 into your answers document.



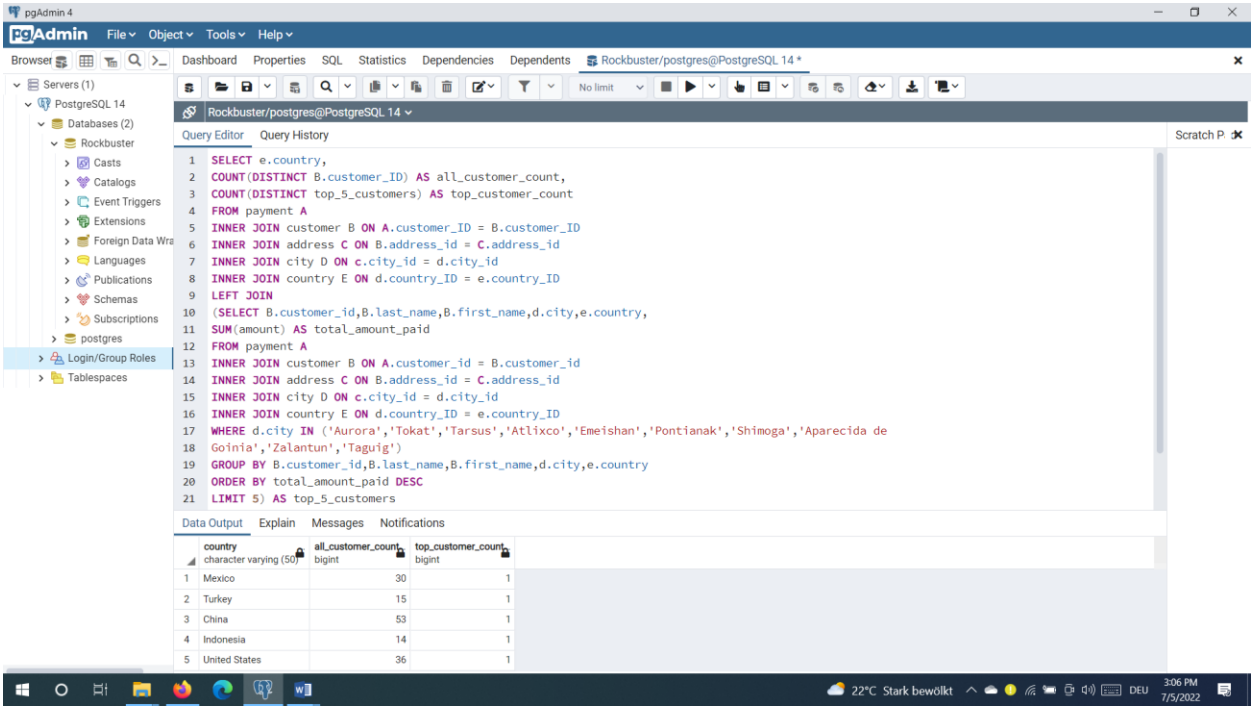
Step 2. **Find out how many of the top 5 customers are based within each country.**

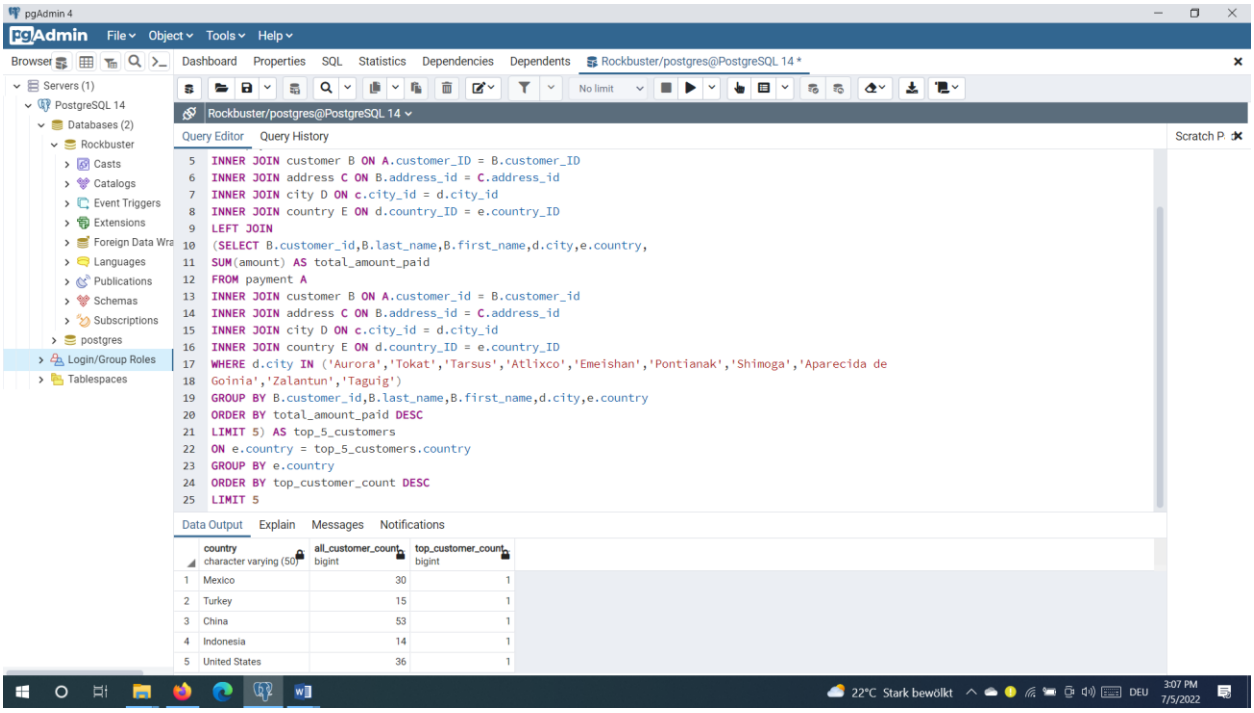
Your final output should include 3 columns:

- “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

You'll notice that this step is quite difficult. We've broken down each part and provided you with some helpful hints below:

1. Copy the query from step 3 of task 3.7 into the Query Tool and add parentheses around it. This will be your inner query.
2. Write an outer statement that counts the number of customers living in each country. You'll need to refer to your entity relationship diagram or data dictionary in order to do this. The information you need is in different tables, so you'll have to use a join. To get the count for each country, use COUNT(DISTINCT) and GROUP BY. Give your second column the alias "all_customer_count" for readability.
3. Place your inner query in the outer query. Since you want to merge the entire output of the outer query with the information from your inner query, use a left join to connect the two queries on the "country" column.
4. Add a left join after your outer query, followed by the subquery in parentheses.
5. Give your subquery an alias so you can refer to it in your outer query, for example, "top_5_customers".
6. Remember to specify which columns to join the two tables on using ON. Both ON and the column names should follow the alias.
7. Count the top 5 customers for the third column using GROUP BY and COUNT (DISTINCT). Give this column the alias "top_customer_count".
8. Copy-paste your query and the data output into your "Answers 3.8" document.





Step 3:

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

    Step 1 could be performed without subquery using aggregate function. In Step 2 we need subquery because we need result from different table.

    - When do you think subqueries are useful?

Subqueries are useful when we need to summarize results from different tables.