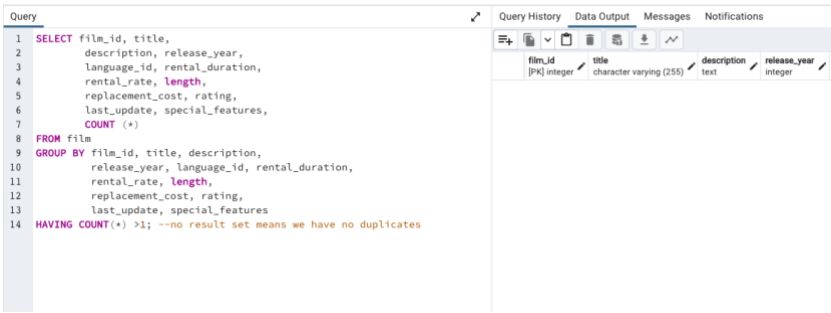
# 3.6: Summarizing and Cleaning Data

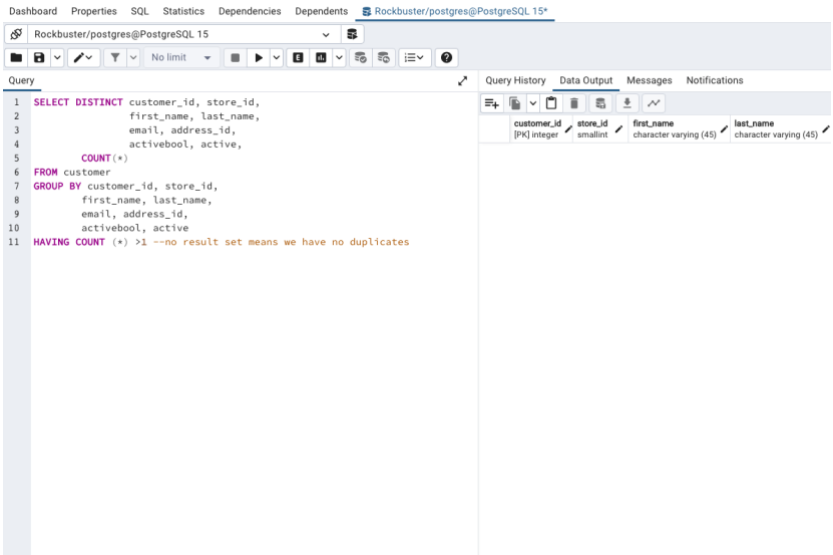
## Answers 3.6

1. **Check for and clean dirty data:** Find out if the film table and the customer table contain any dirty data, specifically non-uniform or duplicate data, or missing values. Create a new “Answers 3.6” document and copy-paste your queries. Next to each query, write 2 to 3 sentences explaining how you would clean the data (even if the data is not dirty).

-Duplicate Data for film table:




-Duplicate Data for customer table:



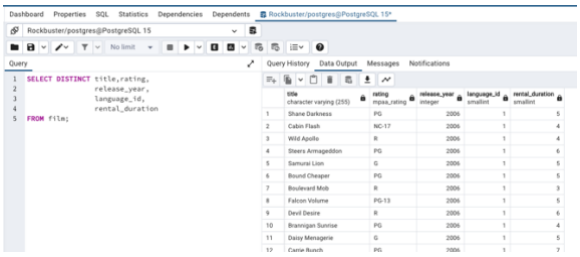
**There is no returned duplicate value.**
The are no returned duplicate values in both tables.

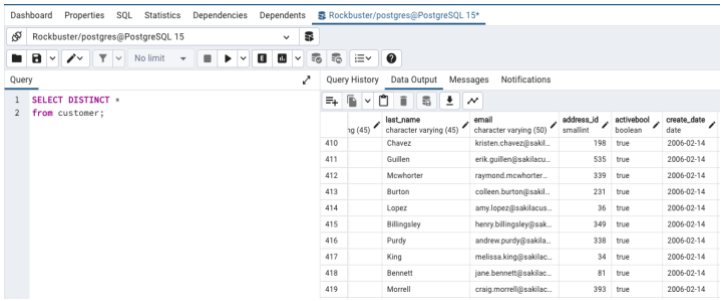
**Non-uniform for costumer and film tables:**
I didn't see any non-uniform columns on these two tables

There are a few ways of dealing with **duplicate values**. If we have permission to alter the database, we could create a view and select only the unique records or delete the identical form from the table or view. If we cannot alter the database, you can use GROUP BY or DISTINCT to select unique records.

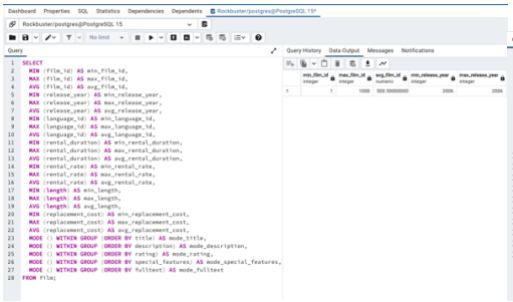
There were also no **non-uniform values** when I scanned through the tables with the DISTINCT statement for both tables. If we encounter one, let's say in the rating column, as we saw in the example in the exercise, we can create an UPDATE query that would make all of the results uniform.

UPDATE film SET rating = 'G' WHERE rating IN ('gen','g', 'General')




**2. Summarize your data:** Use SQL to calculate descriptive statistics for the film and customer table. This means finding the minimum, maximum, and average values for numerical columns. For non-numerical columns, calculate the mode value. Copy-paste your SQL queries and their outputs into your answers document.



**** I will put Queries at the end of the last page****

2. **Reflect on your work:** Back in Achievement 1, you learned about data profiling in Excel. Based on your previous experience, which tool (Excel or SQL) do you think is more effective for data profiling, and why? Consider their respective functions, ease of use, and speed. Write a short paragraph in the running document that you have started.

As I already mentioned in the previous Task. It depends on the amount that we deal with. When it comes to small amounts of data or a few columns, excel will quickly and efficiently categorize and analyze any data set. However, using SQL would be more efficient when working with large amounts of data. Once you learn how to write queries efficiently, you can quickly analyze and categorize data.

## Queries

| | |
|---|---|
| SELECT<br>MIN (customer_id) AS min_customer_id,<br>MAX (customer_id) AS max_customer_id,<br>AVG (customer_id) AS avg_customer_id,<br>MIN (store_id) AS min_store_id,<br>MAX (store_id) AS max_store_id,<br>AVG (store_id) AS avg_store_id,<br>MIN (address_id) AS min_address_id,<br>MAX (address_id) AS max_address_id,<br>AVG (address_id) AS avg_address_id,<br>MIN (active) AS min_active,<br>MAX (active) AS max_active,<br>AVG (active) AS avg_active,<br>MODE () WITHIN GROUP (ORDER BY first_name) AS mode_first_name,<br>MODE () WITHIN GROUP (ORDER BY last_name) AS mode_last_name,<br>MODE () WITHIN GROUP (ORDER BY email) AS mode_email,<br>MODE () WITHIN GROUP (ORDER BY active) AS mode_active,<br>MODE () WITHIN GROUP (ORDER BY activebool) AS mode_activebool<br>FROM customer; | SELECT MIN(rental_rate) AS min_renatl_rate,<br>MAX(rental_rate) AS max_rental_rate,<br>AVG(rental_rate) AS avg_renatal_rate,<br>MIN(rental_duration) AS min_rental_duration,<br>MAX(rental_duration) AS max_rental_duration,<br>AVG(rental_duration) AS avg_rental_duration,<br>MIN(film_id) AS min_film,<br>MAX(film_id) AS max_film,<br>AVG(film_id) AS avg_film,<br>MIN(language_id) AS min_language,<br>MAX(language_id) AS max_language,<br>AVG(language_id) AS avg_language,<br>MIN(length) AS min_length,<br>MAX(length) AS max_length,<br>AVG(length) AS avg_length,<br>MIN(replacement_cost) AS min_replacement_cost,<br>MAX(replacement_cost) AS max_replacement_cost,<br>AVG(replacement_cost) AS avg_replacement_cost,<br>MODE() WITHIN GROUP (ORDER BY rating) AS rating_value,<br>MODE() WITHIN GROUP (ORDER BY special_features) AS feature_value,<br>MODE() WITHIN GROUP (ORDER BY release_year) AS release_year,<br>MODE() WITHIN GROUP (ORDER BY title) AS title_value,<br>MODE() WITHIN GROUP (ORDER BY fulltext) AS fulltext<br>FROM film |