

3.4: Database Querying in SQL

1. **Refining Your Query:** You need to get some data from the “film” table and decide to use the query `SELECT * FROM film`.
 - You realize that only the “film_id” and “title” columns are needed. Write a new query that selects only those 2 columns.
 - Compare the cost of the original query and the revised query, and write a few sentences explaining the comparison. Can you suggest any ways to optimize this query?

The screenshot shows a PostgreSQL query editor interface. At the top, the connection is 'Rockbuster/postgres@posgre'. Below the connection bar is a toolbar with various icons for file operations, query execution, and settings. The main query editor area contains the following SQL code:

```
1 EXPLAIN
2 SELECT *
3 FROM film
```

Below the query editor is a panel with tabs for 'Data output', 'Messages', and 'Notifications'. The 'Data output' tab is active, showing a table with the query plan results:

	QUERY PLAN
1	Seq Scan on film (cost=0.00..64.00 rows=1000 width=384)

The screenshot shows the pgAdmin Query Tool interface. At the top, the connection is 'Rockbuster/postgres@posgre'. Below the connection bar is a toolbar with various icons. The 'Query' tab is active, showing a SQL query:

```
1 EXPLAIN
2 SELECT film_id, title
3 FROM film
```

Below the query editor is the 'Data output' tab, which displays the 'QUERY PLAN' for the executed query. The plan shows a single step:

	QUERY PLAN
1	Seq Scan on film (cost=0.00..64.00 rows=1000 width=19)

From the data output message the cost of the original query and the revised query are the same (cost=0.00..64.00). Although they have different runtimes as shown in each of their output. The best way to optimise and safe cost is to create a script.

Ordering the Data:

- In the pgAdmin Query Tool, run a query that selects every film from the “film” table, with the movies sorted by title from A to Z, then by most recent release year, and then by highest to lowest rental rate.
- Extract the data output of your query into a csv file for the film collection department to analyze in Excel. To do this, click the button “Save results to file”:

https://1drv.ms/u/s!AhfveARMqALDgn93TILT1Q_lyhLB?e=84WAih

Rockbuster/postgres@posgre

Query Query History

```

1
2 SELECT title, release_year, rental_rate
3 FROM film
4 ORDER BY title, release_year, rental_rate Desc

```

Data output Messages Notifications

	title character varying (255)	release_year integer	rental_rate numeric (4,2)
1	Academy Dinosaur	2006	0.99
2	Ace Goldfinger	2006	4.99
3	Adaptation Holes	2006	2.99
4	Affair Prejudice	2006	2.99
5	African Egg	2006	2.99
6	Agent Truman	2006	2.99
7	Airplane Sierra	2006	4.99
8	Airport Pollock	2006	4.99
9	Alabama Devil	2006	2.99
10	Aladdin Calendar	2006	4.99
11	Alamo Videotape	2006	0.99

Total rows: 1000 of 1000 Query complete 00:00:00.793 Ln 4, Col 47

1. **Grouping Data:** The strategy department has asked you the questions below. Write a SQL query to retrieve the correct answers, then extract your results as a csv file.
 - What is the average rental rate for each rating category?
 - What are the minimum and maximum rental durations for each rating category?

<https://1drv.ms/u/s!AhfveARMqALDgnxJOGA6lk1UE9pr?e=gyda4m>

Dashboard Properties SQL Statistics Dependencies Dependents



Rockbuster/postgres@posgre



No limit



Query Query History

```
1 SELECT rating,  
2 AVG (rental_rate) AS avg_rental_rate  
3 FROM film  
4 GROUP BY rating
```

Data output Messages Notifications



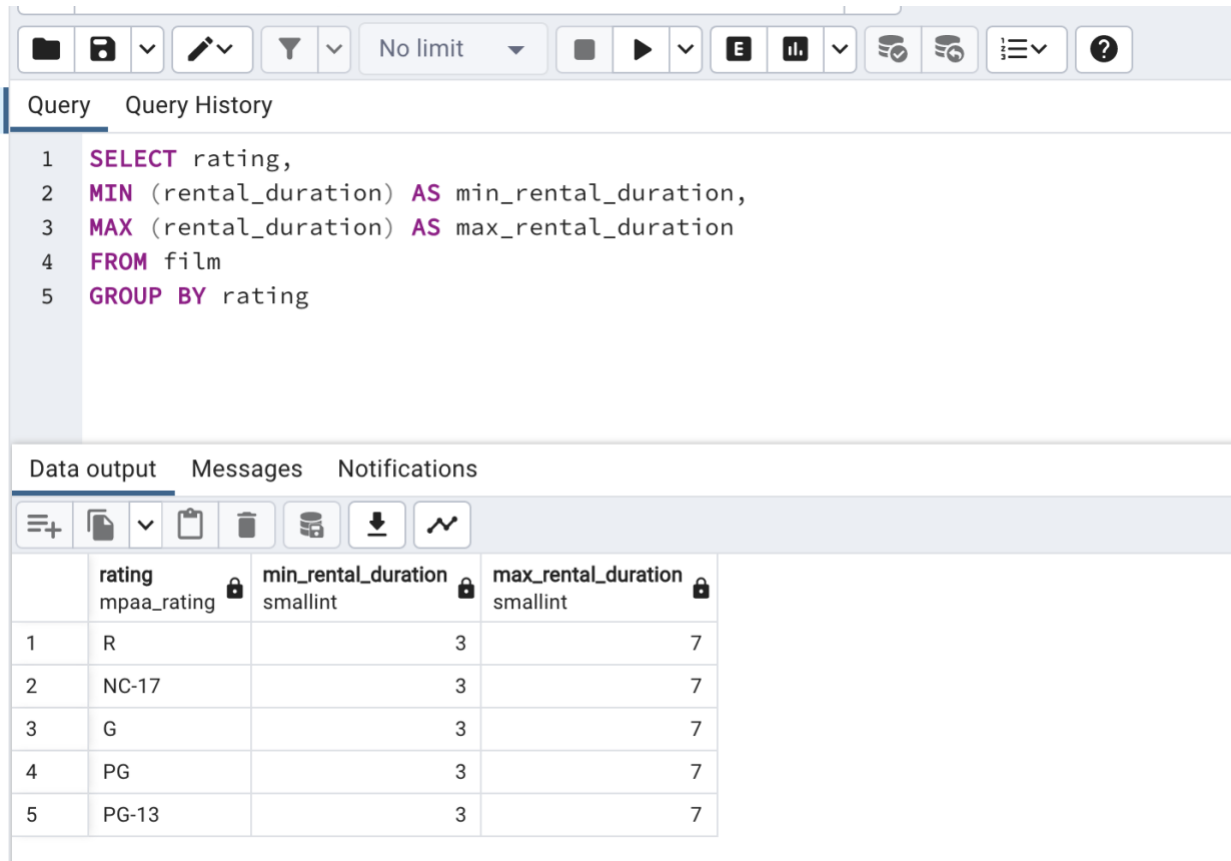
	rating mpaa_rating	avg_rental_rate numeric
1	R	2.9387179487179
2	NC-17	2.9709523809523
3	G	2.8888764044943
4	PG	3.0518556701030
5	PG-13	3.0348430493273

Total rows: 5 of 5

Query complete 00:00:00.100

minimum rental durations
maximum rental durations

<https://1drv.ms/u/s!AhfveARMqALDgn7R159ftUXnIH0n?e=Xlc9Rp>



Query Query History

```
1 SELECT rating,
2 MIN (rental_duration) AS min_rental_duration,
3 MAX (rental_duration) AS max_rental_duration
4 FROM film
5 GROUP BY rating
```

Data output Messages Notifications

	rating mpaa_rating	min_rental_duration smallint	max_rental_duration smallint
1	R	3	7
2	NC-17	3	7
3	G	3	7
4	PG	3	7
5	PG-13	3	7

1. **Database Migration:** Your team has decided to use an external tool to collect data on user behavior in the new Rockbuster Android app. Data collected from this new source will need to be loaded into the data warehouse before you can analyze it.
 - Can you outline the procedure for migrating the data and who will be responsible for it?
 - What problems do you foresee if you start analyzing the data before it's been loaded into the data warehouse?

The procedure for migrating data to the data warehouse is called the ETL (Extract, Transform and Load). **First step: Extract:** Involves collecting data from different sources, **second step:**

Transform: This is a situation where the extracted data is converted into another format and **Last step: Load:** is the point the transformed data is transferred or loaded into the data warehouse. The ETL is a data engineer's job. However, an awareness of the basic concepts is important for data analysis as it helps the Analyst in situation of atabase migration process.

Bonus task

You've not yet covered custom sorting; however, let's imagine you've found the two resources below that explain it. Read each one, then try to write a query to answer the following question: What are the minimum and the maximum replacement costs for each rating category ordered by rating as follows: G, PG, PG-13, R, NC-17?

Rockbuster/postgres@posgre

No limit

Query Query History

```

1 SELECT rating,
2 MIN (replacement_cost) AS min_rental_duration,
3 MAX (replacement_cost) AS max_rental_duration
4 FROM film
5 GROUP BY rating
6 ORDER BY rating
    
```

Data output Messages Notifications

	rating mpaa_rating 🔒	min_rental_duration numeric 🔒	max_rental_duration numeric 🔒
1	G	9.99	29.99
2	PG	9.99	29.99
3	PG-13	9.99	29.99
4	R	9.99	29.99
5	NC-17	9.99	29.99