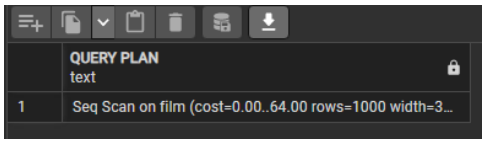
3.4 Database Querying in SQL

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?

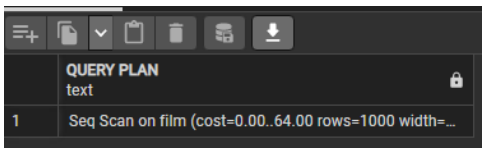
   Original query



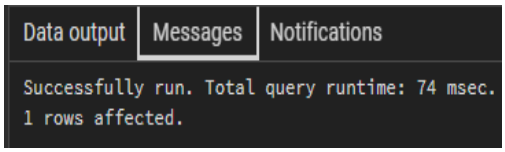
	QUERY PLAN text
1	Seq Scan on film (cost=0.00..64.00 rows=1000 width=3...

   Revised query

```
SELECT film_id, title
FROM film
```



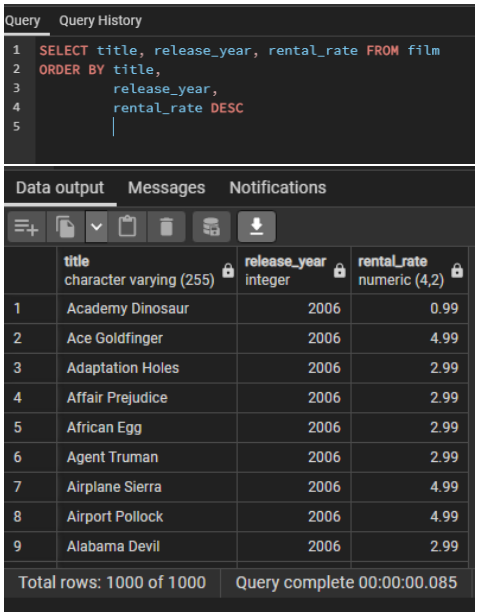
	QUERY PLAN text
1	Seq Scan on film (cost=0.00..64.00 rows=1000 width=...



   The cost (startup cost and total cost) from both queries are the same 0.00-64.00. Since costs are in an arbitrary unit, all this tells us is that a query with the same cost will take equally long. However, the actual query runtime differs that the revised query takes faster than the original one because the second query only returns two columns instead of all columns from film table.

2. **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.



	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

- 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”:PRIMARY KEY: category_id is assigned as primary key and given a unique ID which can’t contain any null or duplicate values.

  Film Query

3. **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?

```
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

AVG Rental Rate

- What are the minimum and maximum rental durations for each rating category?

```
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

MIN MAX Rental Duration

4. **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?
  Data engineers are responsible for the ETL process: user behaviour data need to be extracted and converted into desired format, then it is ready to be loaded into data warehouse.

- What problems do you foresee if you start analyzing the data before it's been loaded into the data warehouse?
  The raw data might be not correctly structured and appear as standalone tables (not connected to each other) that makes querying more difficult which in turn makes analysing difficult as well.

Bonus Task

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?

Query | Query History

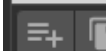
```sql
SELECT rating,
       MIN(replacement_cost) AS min_rental_duration,
       MAX(replacement_cost) AS max_rental_duration
FROM film
GROUP BY rating
ORDER BY rating
```

OR with CASE

Query | Query History

```sql
SELECT rating,
       MIN(replacement_cost) AS min_rental_duration,
       MAX(replacement_cost) AS max_rental_duration
FROM film
GROUP BY rating
ORDER BY CASE WHEN rating = 'G' THEN 1
              WHEN rating = 'PG' THEN 2
              WHEN rating = 'PG-13' THEN 3
              WHEN rating = 'R' THEN 4
              ELSE 5
          END
```

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