

Database Querying in SQL

Step 1

The screenshot shows a database query tool interface. At the top, there's a navigation bar with tabs: Dashboard, Properties, SQL, Statistics, Dependencies, Dependents, Processes, and a file named 'Task 3.4 Database Querying in SQL.sql'. Below this is a toolbar with various icons for file operations and query execution. The main area is split into two panes: 'Query' and 'Query History'. The 'Query' pane contains five SQL queries:

- 1 `EXPLAIN SELECT * FROM film;`
- 2 `EXPLAIN SELECT film_id, title FROM film;`
- 3 `SELECT title, release_year, rental_rate FROM film ORDER BY title ASC, release_year DESC, rental_rate DESC;`
- 4 `SELECT rating, AVG(rental_rate) as avg_rental_rate FROM film GROUP BY rating;`
- 5 `SELECT rating, MIN (rental_duration) as Minimum_Rental_Duration, MAX (rental_duration) as Maximum_Rental_Duration FROM film;`

Below the queries is a 'Data Output' section with tabs for 'Data Output', 'Messages', and 'Notifications'. The 'Data Output' tab is active, showing a 'QUERY PLAN' table:

	QUERY PLAN
1	Seq Scan on film (cost=0.00..64.00 rows=1000 width=38...)

Dashboard Properties SQL Statistics Dependencies Dependents Processes Task 3.4 Database Querying in SQL.sql

Rockbuster/postgres@PostgreSQL 15

No limit

Query Query History Scratch Pad

```

1 EXPLAIN SELECT * FROM film;
2 EXPLAIN SELECT film_id, title FROM film;
3 SELECT title, release_year, rental_rate FROM film ORDER BY title ASC, release_year DESC, rental_rate DESC;
4 SELECT rating, AVG(rental_rate) as avg_rental_rate FROM film GROUP BY rating;
5 SELECT rating, MIN (rental_duration) as Minimum_Rental_Duration, MAX (rental_duration) as Maximum_Rental_Durat

```

Data Output Messages Notifications

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

Both queries appear to have the same cost.

Considering this was all that was asked for, I don't see any need to optimize the above query.

Step 2

Dashboard Properties SQL Statistics Dependencies Dependents Processes Task 3.4 Database Querying in SQL.sql

Rockbuster/postgres@PostgreSQL 15

No limit

Query Query History

```

1 EXPLAIN SELECT * FROM film;
2 EXPLAIN SELECT film_id, title FROM film;
3 SELECT title, release_year, rental_rate FROM film ORDER BY title ASC, release_year DESC, rental_rate DESC;
4 SELECT rating, AVG(rental_rate) as avg_rental_rate FROM film GROUP BY rating;
5 SELECT rating, MIN (rental_duration) as Minimum_Rental_Duration, MAX (rental_duration) as Maximum_Rental_Durat

```

Scratch Pad

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

Total rows: 1000 of 1001 Query complete 00:00:00.165 Ln 3, Col 1

Step 3

Average rental rate grouped by rating

Dashboard Properties SQL Statistics Dependencies Dependents Processes Task 3.4 Database Querying in SQL.sql

Rockbuster/postgres@PostgreSQL 15

No limit

Query Query History Scratch Pad

```

1 EXPLAIN SELECT * FROM film;
2 EXPLAIN SELECT film_id, title FROM film;
3 SELECT title, release_year, rental_rate FROM film ORDER BY title ASC, release_year DESC, rental_rate DESC;
4 SELECT rating, AVG(rental_rate) as avg_rental_rate FROM film GROUP BY rating;
5 SELECT rating, MIN (rental_duration) as Minimum_Rental_Duration, MAX (rental_duration) as Maximum_Rental_Durat

```

Data Output Messages Notifications

	rating mpaa_rating	avg_rental_rate numeric
1	PG	3.0518556701030928
2	R	2.9387179487179487
3	NC-17	2.9709523809523810
4	PG-13	3.0348430493273543
5	G	2.9006145251396648

Total rows: 5 of 5 Query complete 00:00:00.062 Ln 4, Col 1

Minimum/Maximum rental duration by rating category

DashboardPropertiesSQLStatisticsDependenciesDependentsProcessesTask 3.4 Database Querying in SQL.sql

Rockbuster/postgres@PostgreSQL 15

No limit

QueryQuery History

1

2 FROM film;

3 rental_rate FROM film ORDER BY title ASC, release_year DESC, rental_rate DESC;

4) as avg_rental_rate FROM film GROUP BY rating;

5 (rental_rate) as Minimum_Rental_Duration, MAX (rental_duration) as Maximum_Rental_Duration FROM film GROUP BY rating;

Scratch Pad

Data OutputMessagesNotifications

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

Total rows: 5 of 5

Query complete 00:00:00.069

Ln 5, Col 141

Step 4

1. Extract: release an android app to recognize client's behavior
2. Transform: generating data to another data format
3. Load: uploading the data collected to the main database

Note: Different data types will make the data harder to analysis