CSCI 5408

Data Management, Warehousing, And Analytics

Lab 1: Introduction to MySQL

Prepared By

Bhavisha Oza (B00935827)

Summary

The lab-1 is all about getting familiar with MySQL. After refreshing the concepts of it, I did some hands on for the asked queries on given IMDB_dataset. Fetched the results by using DML (Data Manipulation Language) queries with the help of basic SQL functions, operators, sub-queries, and joins. Wanted to use both the ways of retrieving data from multiple related tables, so I've used joins as well as sub-queries.

Steps performed:

- Installed MySQL workbench [1].
- Created new connection for CSCI 5408.
- Created new schema for Lab-1
- Downloaded the IMDB dataset provided for the lab work [2].
- Imported the IMDB.SQL file in MySQL workbench.
- Refreshed the schema and tested the basic Sample select query from movies table.
- Started creating queries for the Lab Exercise [3].
- Executed them, took a screenshot of each output, and added in the report.

Lab exercise:

1. **Check how many directors are present in iMDB?**

   SELECT DISTINCT COUNT(*) AS total_directors
   FROM directors;

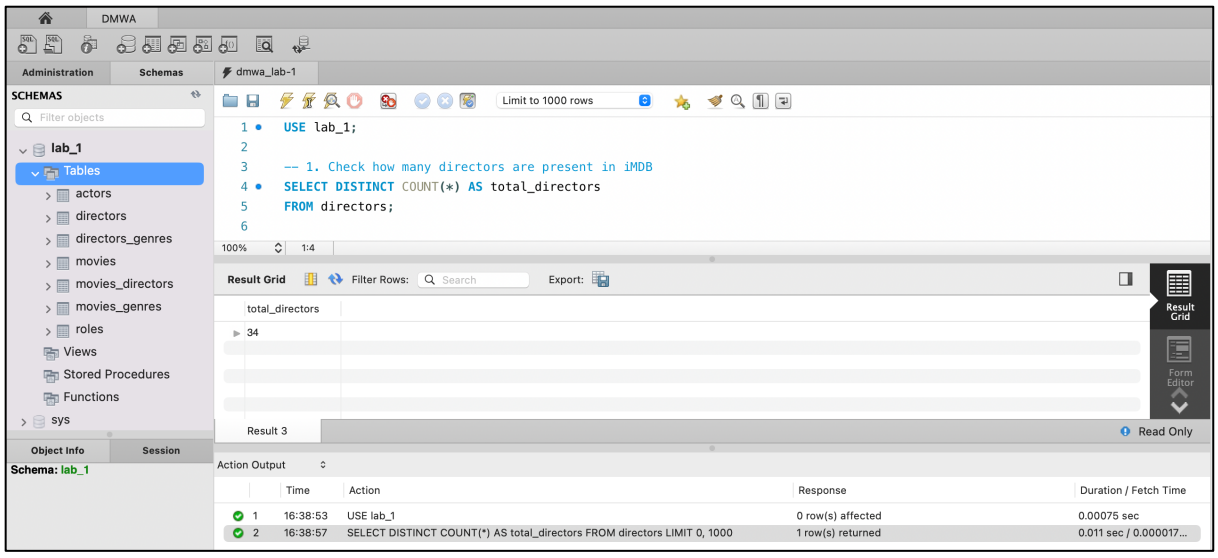

Figure 1: The result of total directors available in DB.

Explanation: This query retrieves the number of distinct directors from the "directors" table and assigns it to the alias "total_directors."

2. **Check how many movies are released post-year 2000?**

```
SELECT COUNT(*) AS total_movies
FROM movies
WHERE year > 2000;
```

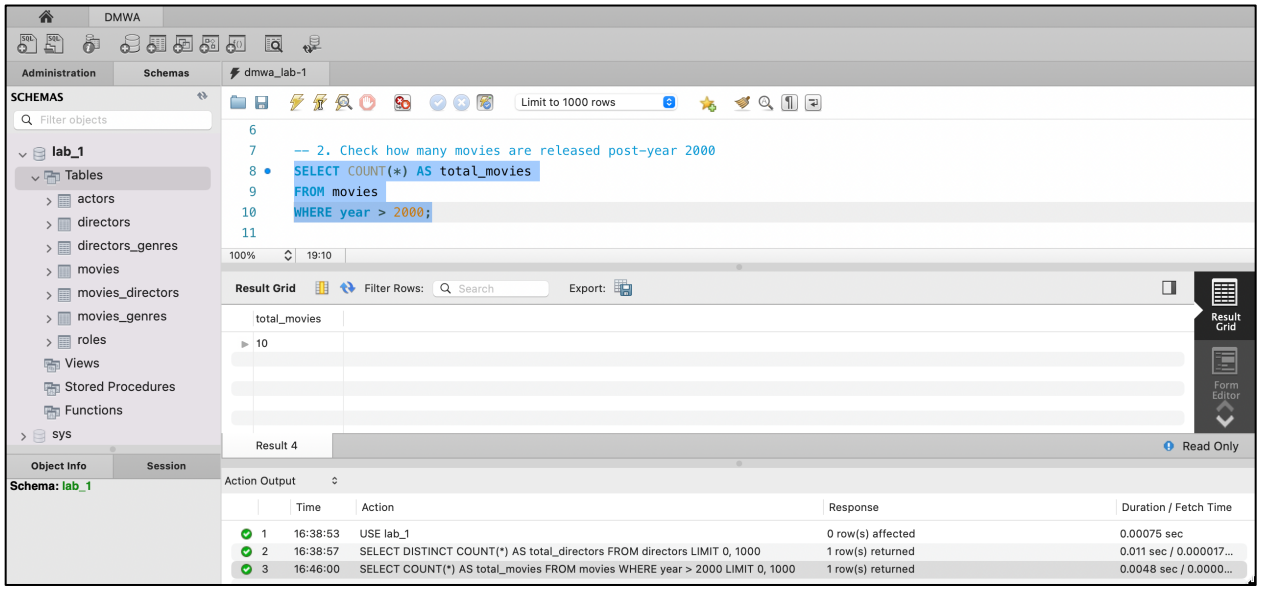

Figure 2: The result of total movies released after year 2000.

Explanation: This query retrieves the count the number of movies that were released after the year 2000 and return the result as "total_movies".

3. **Find the list of genres of movies directed by Andrew Adamson.**

```
SELECT DISTINCT mg.genre
FROM movies m
JOIN movies_directors md ON m.id = md.movie_id
JOIN directors d ON md.director_id = d.id
JOIN movies_genres mg ON m.id = mg.movie_id
WHERE d.first_name = 'Andrew' AND d.last_name = 'Adamson';
```

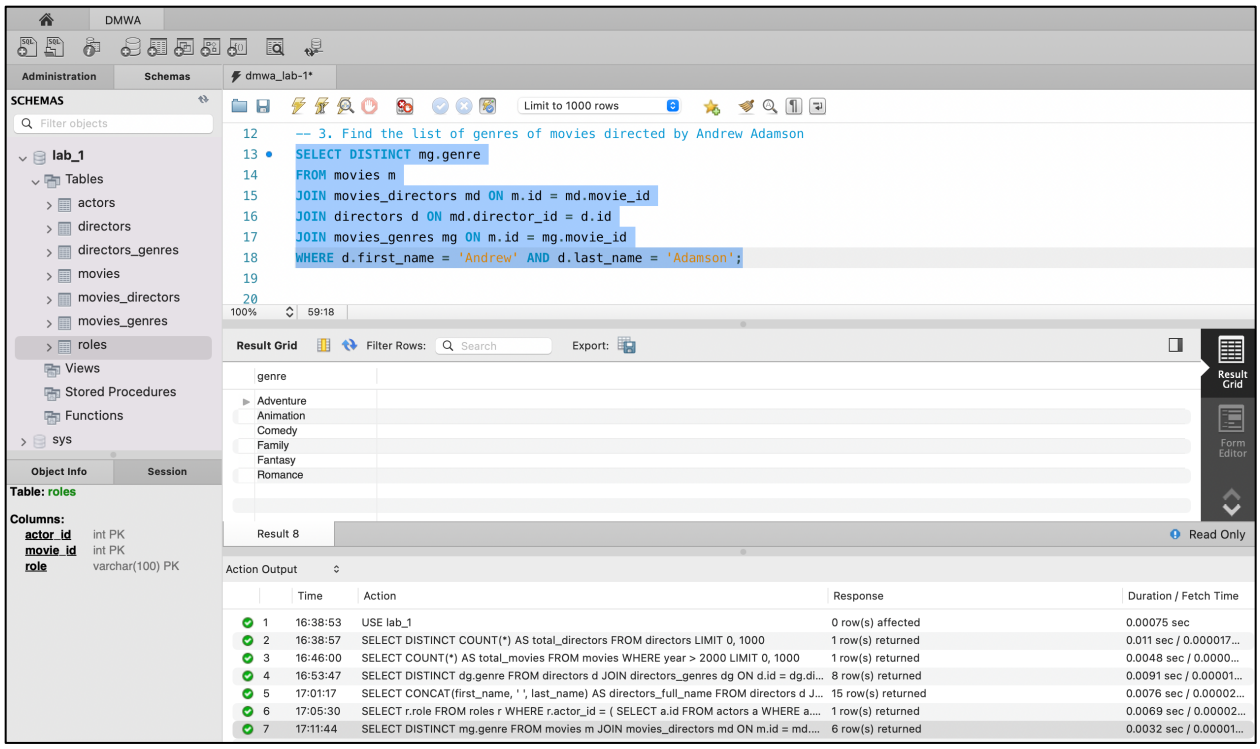

Figure 3: The list of movie genres that Andrew Adamson has directed.

Explanation: This query selects distinct movie genres from the "movies_genres" table for movies directed by Andrew Adamson. It achieves this by joining the "movies" table with the "movies_directors" table and the "directors" table on their respective IDs and then joining the resulting table with the "movies_genres" table. The query filters the results based on the first and last name of the director being "Andrew" and "Adamson," respectively. The DISTINCT keyword ensures that only unique genres are returned [3].

4. **List of directors whose movies are ranked between 7 to 8 ranking.**

```
SELECT CONCAT(first_name, ' ', last_name) AS directors_full_name
FROM directors d
JOIN movies_directors md ON d.id = md.director_id
JOIN movies m ON md.movie_id = m.id
WHERE m.rank BETWEEN 7 AND 8;
```

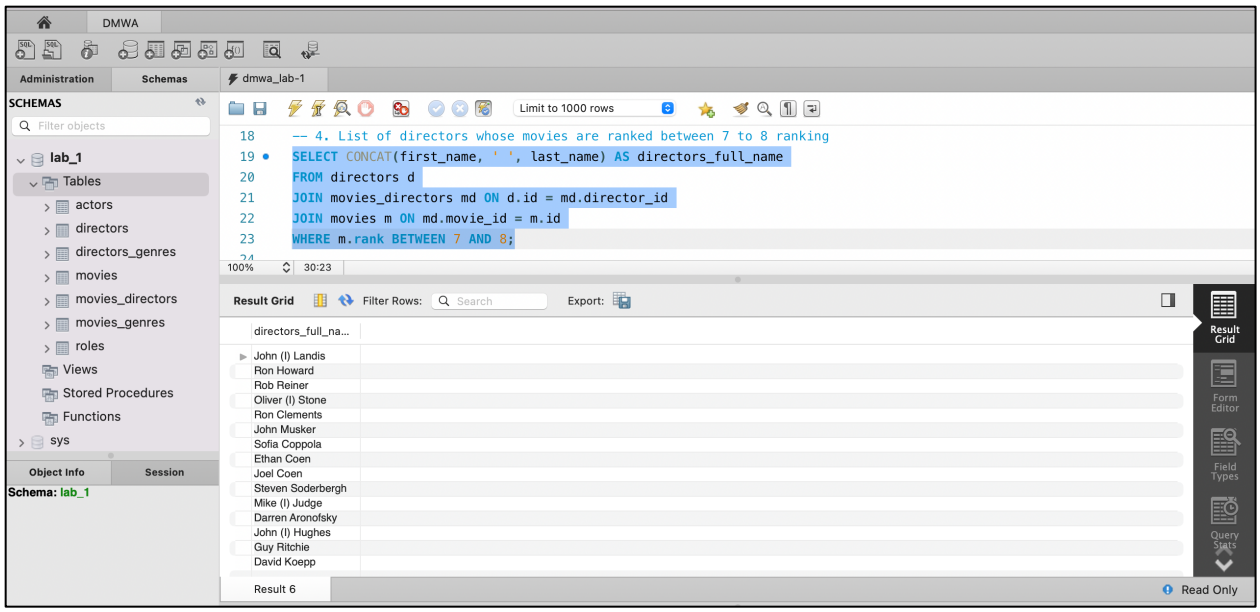

Figure 4: The list of directors whose movies are in the seventh to eighth ranks.

Explanation: This query retrieves the full names of directors who have directed movies with a rank between 7 and 8. It combines data from the 'directors' table, 'movies_directors' table, and 'movies' table using joins, and concatenates the first name and last name of the directors into a single column called 'directors_full_name' [3].

5. **Find the role of Julliet Akinyi in Lost in Translation movie.**

```
SELECT r.role
FROM roles r
WHERE r.actor_id = (
    SELECT a.id
    FROM actors a
    WHERE a.first_name = 'Julliet' AND a.last_name = 'Akinyi'
    )
AND r.movie_id = (
    SELECT m.id
    FROM movies m
    WHERE m.name = 'Lost in Translation'
    );
```

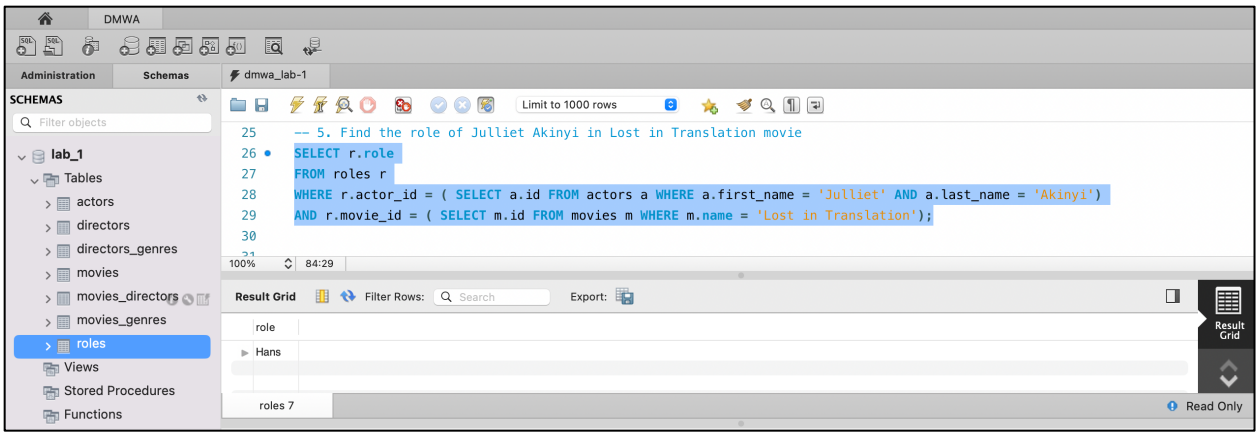

Figure 5: The result of Julliet Akinyi's role in the movie Lost in Translation.

Explanation: This query retrieves the role of an actor named Julliet Akinyi in the movie 'Lost in Translation'. It does so by selecting the role from the 'roles' table, filtering for rows where the actor's ID matches the ID of Julliet Akinyi (obtained from the 'actors' table) and the movie's ID matches the ID of the movie 'Lost in Translation' (obtained from the 'movies' table). The result will be the role played by Julliet Akinyi in that particular movie [3].

6. List of the movies that contain the letter ‘o' in any position.

```
SELECT name
FROM movies
WHERE name LIKE '%o%';
```

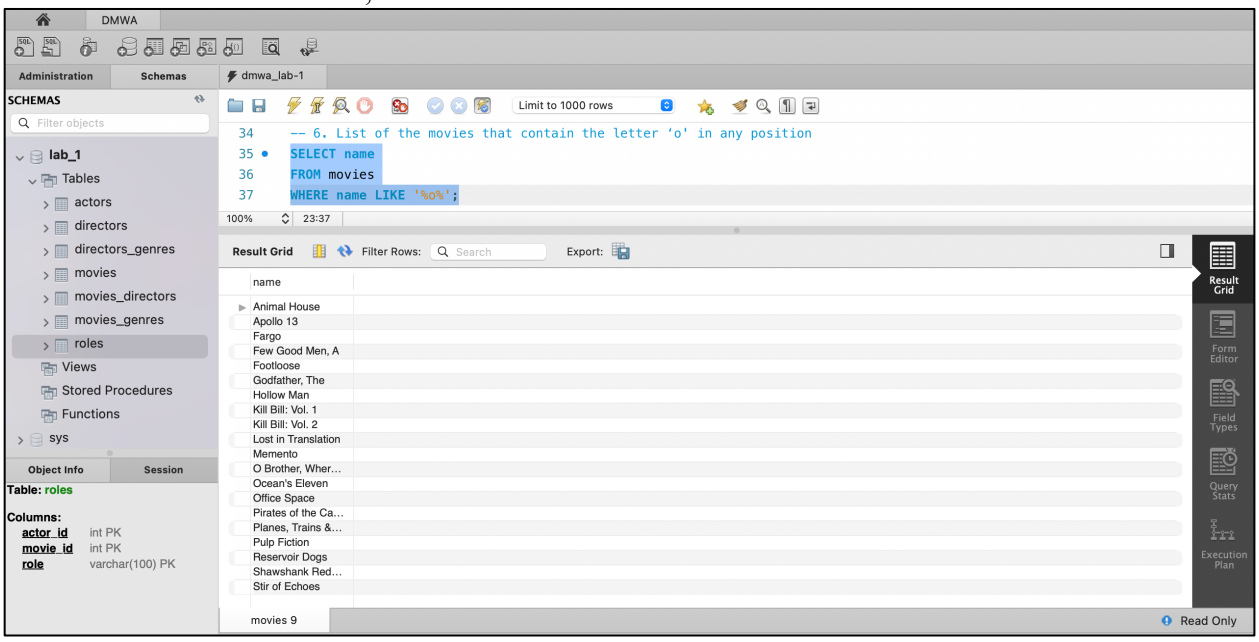

Figure 6: The result of movies with the letter o in any place.

Explanation: This query selects the names of movies from a table called "movies" where the name contains the letter "o" anywhere in it.

References:

[1] "MySQL Community Downloads," *MySQL* [Online]. Available: https://dev.mysql.com/downloads/workbench/ [Accessed: May 10, 2023].

[2] "IMDB_dataset," *Brightspace Dalhousie University* [Online]. Available: https://dal.brightspace.com/d2l/le/content/271677/viewContent/3623394/View [Accessed: May 10, 2023].

[3] "Lab-1," *Brightspace Dalhousie University* [Online]. Available: https://dal.brightspace.com/d2l/le/content/271677/viewContent/3623368/View [Accessed: May 10, 2023].