MovieNet Test Queries

445 Course Project

In this document, we will provide 10 test queries for your MovieNet system. Please run all test queries correctly and efficiently. If you encounter a performance issue, please think about how to improve the performance, such as adding appropriate indexes, and revising your schema design, etc. In your extended project report, present

- your schema and your queries written in SQL,
- the performance issues that you encountered,
- the techniques you employed to fix these performance issues (if you have revised your schema design, please also provide your new schema and new queries in SQL),
- the execution time of each test query using the backend (using MySQL directly).

If you re-run a query, the result of the query may have been cached. As a result, the execution time is very short (usually 0 seconds) and does not reflect the real performance of the query. You should turn off the query cache with the following command in MySQL:

```
mysql> SET SESSION query_cache_type = OFF;
```

This command is only valid for your current session in MySQL. It means you need to run it every time when you connect to MySQL. We will also show the number of rows returned by each query in our test for your reference. The actual number of rows may differ depending on your data cleaning and loading process.

**Query 1.** Find all movies that have “Life” as a prefix in the title. (1,007 rows)

**Query 2.** Find all movies that “Pitt, Brad” acted in. (159 rows)

**Query 3.** Find the performers who acted in at least one movie directed by “Spielberg, Steven”. (2,665 row)

**Query 4.** Show the number of user ratings for each rating value. (10 rows)

**Query 5.** Find the directors of all movies that are rated 10 by “DERRICK MYERS” whose age is 36. Show the name of each director. (4 rows)

**Query 6.** For each movie that is rated more than 5,000 times, show the title, year and the average age of the users who rate the movie. (17 rows)

**Query 7.** Find the users who are 17 or under 17 and rate at least one “NC-17” movie. Show the email, name, age and location of each user. (1783 rows)
**Query 8.** Find the 5 best movies. The 5 best movies are the 5 movies with the highest average rating. Only the movies rated more than 1,000 times are considered. Show the title, year and average rating of each best movie. (5 rows)

**Query 9.** Find all the users of good taste. A user is of good taste if and only if the user rates at least 2 of the 5 best movies, and the user never rates the 5 best movies with rating less than 9. Show the email of each user. (238 rows)

**Query 10.** Now we consider only the ratings from the users of good taste. Find the 10 movies with the highest average rating. Only the movies rated more than 2 times by the users of good taste are considered. Show the title, year and average rating of each best movie. (10 rows)