CMPE 321 Introduction to Database Systems
Spring 2025

Project 2

Due: April 14, 2025, 23:59

1 Project Description

In this project, we have provided you with a MySQL database file called MovieDB.sql. The database consists of 5 tables, whose information is provided in the next section. The primary keys are already provided for you.

Aside from the insertions/updates that are wanted in the queries, do not make any changes to the database as we will directly use the MovieDB.sql file for checking.

You need to implement 20 different SQL queries and provide the SQL files. We will also provide you with the expected results (the tables with column names and contents) in .txt files, so please do not forget to check your results.

2 Tables

- **Directors**
  - director_id INT,
  - name VARCHAR(512),
  - surname VARCHAR(512),
  - birth_date VARCHAR(512),
  - nationality VARCHAR(128),
  - PRIMARY KEY (director_id)
- **Genres**
  - genre_id INT,
  - genre_name VARCHAR(512),
  - PRIMARY KEY (genre_id)

- **Movies**
  - movie_id INT,
  - title VARCHAR(512),
  - release_date VARCHAR(512),
  - duration INT,
  - director_id INT,
  - rating DOUBLE,
  - genre_id INT,
  - budget INT,
  - PRIMARY KEY (movie_id),
  - FOREIGN KEY (director_id) REFERENCES Directors(director_id),
  - FOREIGN KEY (genre_id) REFERENCES Genres(genre_id)
- **Actors_and_Actresses**
  - actor_id INT,
  - name VARCHAR(512),
  - surname VARCHAR(512),
  - birth_date VARCHAR(512),
  - PRIMARY KEY (actor_id)
- **Cast**
  - cast_ID INT,
  - movie_id INT,
  - actor_id INT,
  - PRIMARY KEY (cast_ID)
  - FOREIGN KEY (movie_id) REFERENCES Movies(movie_id),
  - FOREIGN KEY (actor_id) REFERENCES Actors_and_Actresses(actor_id)

3 Queries

1. Find the number of movies and display this as *movie_count*. The required column name: *movie_count*.
2. List the movies released before 2024 (exclusive). Display release year in DD/MM/YYYY format. The required column names are respectively: *movie_id*, *title*, *duration*, *rating*, *director_id*, *date*. Sort the results by *date* in descending order.
3. List all the fields of the movies with the minimum *rating*. Display Date in DD.MM.YYYY format. The required column names are respectively: *movie_id*, *title*, *release_date*, *duration*, *director_id*, *rating*, *genre_id*, *budget*.
4. List the titles and budgets of the movies with the maximum *rating*. The required column names are respectively: *title*, *rating*, *budget*. Sort the results by *budget* in ascending order.
5. Find the average rating of all movies. The required column name is : *average_rating*.
6. List all actors/actresses together with the number of movies they have appeared in. The required column names are respectively: *actor_name*, *actor_surname*, *movie_count*.
7. Find the actors/actresses whose surname starts with the letter "P". The required column names are : *actor_id*, *name*, *surname*, *birth_date*.
8. Find the names and surnames of directors who have directed at least 3 movies. The required column names are respectively: *name*, *surname*. Sort by *surname* in descending order.
9. Find the names and surnames of actors/actresses who were born in the same year as "Amy Adams". The required column names are respectively: *name*, *surname*. Sort by *surname* in ascending order.
10. List all directors who directed movies between 01.01.2020 - 31.12.2025 in the Sci-Fi genre. The required column names are respectively: *name*, *surname*, *nationality*. Sort by *nationality* in ascending order.
11. List all movies that were directed by "Christopher Nolan" and whose duration is longer than 120 minutes. The required column names are respectively: *movie_id*, *title*, *release_date*, *duration*, *director_id*, *rating*, *genre_id*, *budget*. Sort by *movie_id* in descending order.
12. For each year, find and list the director who has directed the movie with the highest rating. The required column names are respectively: *name*, *surname*, *movie_name*, *year*, *rating*. Sort by *year* in ascending order.
13. For each genre, find the director who has directed the highest number of movies in that genre. Please note that there can be multiple maximums. The required column names are respectively: *genre_name*, *name*, *surname*, *directed_count*.
14. List all movies released after 2023 (meaning starting from 01.01.2010 inclusive) that were not directed by "Christopher Nolan". The required column names are

respectively: *movie_id*, *title*, *director_name*, *director_surname*. Sort by *movie_id* in ascending order.

15. Find the director who has directed the highest number of movies in the database. The required column names are respectively: *name*, *surname*, *number_of_movies*.

16. Find the average rating of movies for each actor/actress and list the actors/actresses in descending order of their average ratings. The required column names are respectively: *name*, *surname*, *average_rating*.

17. Find the movies having duration longer than 150 minutes and rating greater than 8. The required column names are respectively: *movie_id*, *movie_name*, *duration*, *rating*, *director_name*. Sort by *director_name* descending.

18. Find the directors who have directed movies in multiple genres. The required column names are respectively: *name*, *surname*, *genre_count*. Sort in descending order by *genre_count*.

19. Find the genre with the highest average rating. The required column names are respectively: *genre_id*, *genre_name*, *average_rating*.

20. List all *actor_ID*'s with the column *multiple_appearance* which is either TRUE if that actor/actress appears in more than one genre or FALSE otherwise. The required column names are respectively: *name*, *surname*, *genre_count*, *multiple_appearance*. Sort in descending order by *surname*.

    The output of each query is given in the ***output_query¡index¿.txt*** files (e.g., *output1.txt, output2.txt, ... , output20.txt*.

4 Submission

- This project must be completed in teams of two students, **unless an exception is explicitly granted**. (partner dropout etc.)
- Each query must be written into a separate file named `queryi.sql` (e.g., `query1.sql`, `query2.sql`, ..., `query20.sql`).
- Each SQL file must include a comment explaining the reasoning behind the query. Files without proper comments will receive **0 points**.
- All 20 SQL files must be placed inside a folder named `queries`.
- The `queries` folder must be placed inside a parent folder named `GROUP<ID>` (e.g., `GROUP50`).
- Your individual contribution reports must be placed directly inside the `GROUP<ID>` folder, not inside `queries`.
- Then, compress the `GROUP<ID>` folder into a ZIP file and name it as `GROUP<ID>.zip` (e.g., `GROUP50.zip`).
- Submit the ZIP file via **Moodle** before the deadline.
- Each group must submit only **one ZIP file**.
- Do not include any files other than the required ones (e.g., no backups, or database files).
- **Any other submission method is not allowed.**
- **10 points will be deducted** in case of non-compliance to any of the naming and folder conventions.
- Do not inject your observations into your queries to skip some essential steps! Do not obtain values manually from the database and inject them into the queries! This kind of query will receive 0 points. Furthermore, additional penalties will be imposed as a violation of this condition will be considered dishonest behavior.
  - For example, do not find the minimum rating manually and write `WHERE rating = 5` directly.
- No points will be deducted due to floating point rounding or minor formatting issues (e.g., extra blank lines).

Final Folder Structure (Inside the ZIP File)

```
GROUP<ID>.zip
 GROUP<ID>/
    queries/
       query1.sql
       query2.sql
       ...
       query20.sql
```

`StudentID1_Contribution.pdf`
`StudentID2_Contribution.pdf`

Note: If you are working alone, only include your own contribution report inside the `GROUP<ID>` folder.

5 Individual Contribution Report Guidelines

Each student must prepare an **Individual Contribution Report** and include it in the group ZIP file. The report must be in **PDF format**, named as:

`StudentID_Contribution.pdf`
(e.g., `123456_Contribution.pdf`)

What to Include:

- **Personal Information:** Name, Student ID, Group Number.
- **Tasks & Contributions:** Describe which queries you implemented, who wrote the explanatory comments, how testing and verification were handled. (i.e summarize your work, even if you have no partner)
- **Collaboration & Challenges:** Describe how your team worked together, what problems you faced and how you overcame them. (you can skip the collaboration part if you were alone, write about the challenges)
- **Self-Assessment:** Reflect on your role in the project, what you learned, and what could be improved.

Formatting:

- **Length:** 1 page (max 2 pages)
- **Font & Size:** Times New Roman, 12pt, 1.5 spacing
- **File Format:** PDF

Submission:

- Each student must include their own individual contribution report in the submission ZIP file.
- The file must be placed directly inside the ZIP file (not in a subfolder).

Reminder: Submitting a false or misleading contribution report is considered a violation of academic honesty and will be penalized accordingly.

6 Academic Honesty

Please read carefully the academic honesty part of the syllabus as we give utmost importance to academic honesty.