CT230 DATABASE SYSTEMS

PROBLEM SHEET 2

WEEK 4: 26TH, 27TH AND 29TH SEPTEMBER 2022

Learning Outcomes:

- Fully create movie database (DDL stubs provided).
- Insert data into movie database (data provided).
- Try Sample test 1 in preparation for the real test on Friday.

SCHEMA:

Information is held on movies, actors and directors in six tables.

movie(mID, title, relYear, category, runTime, studioName, description, rating**)**

actor(aID, fName, surname, nationality, gender**)**

director(dID, dfName, dSurname**)**

genres(movieID, genre**)**

directs(movieID, directorID**)**

stars(movieID, actorID**)**

The table **movie** holds details on each movie (with mID the primary key): the title, the year of release of the film, the category (e.g. 12A, G, etc.), the run time of the film (in minutes), studio which released film (e.g. MGM, Warner, etc.), a description of the main plot and an average rating (real number) given by reviewers (in the range 1.0 to 10.0).

The **actor** table holds details on actors (with primary key aID) and includes actor name, nationality and gender. The **director** table holds details on directors (with primary key dID) and includes first name and surname of the director. The **genres** table holds details on the film genre with movieID and genre as the primary key (e.g. drama, comedy, etc.). Note that one film can have a number of genres (e.g., drama, thriller). movieID is a foreign key to mID in the table movie.

The **stars** table holds details on the actors in each movie with movieID and actorID as the primary key. movieID is a foreign key to mID in the table movie. actorID is a foreign key to aID in the table actor.

The **directs** table holds details on the directors of each movie with movieID and directorID (the director ID) as the primary key. movieID is a foreign key to mID in the table movie. directorID is a foreign key to dID in the table director.

TASKS:

1. You can use your existing mySQL database or request a new database account on the server for the movie database.

2. Partial DDL code is available for each of the 6 tables. You should download this code and complete it (and save locally). Use the given code and sample data to choose appropriate data types if needed and ensure all Primary keys and Foreign keys are correctly specified.

3. Comments in the DDL code indicate which table needs additional code.

4. Once you are happy with your code you can create the tables in one of two ways:

   > Paste the code, one query at a time or all together, into your SQL editor and choose Go/Execute.
   > Choose the "Import" option and import your file. If using this option you might also want to add in the sample data code to this file (next step).

5. Write the commands needed to **enter the sample data** given for each table using the INSERT INTO command and the sample data file on Blackboard (6 files). Note that the tuples are separated by commas but you will need to add the correct INSERT INTO commands and paste the command and data in your SQL editor and choose Go/Execute or add it to your sql file and import.

6. Please obtain help if needed from your lab tutor.

7. Try Sample Test 1.