



UNIVERSITY OF SOUTHERN MAINE

Database System, Fall 2023, Assignment 4

Instructor: Behrooz Mansouri (behrooz.mansouri@maine.edu)

Due: November 13, 2023

Notes for submission: Read the deliverables carefully and provide the necessary files.

In our class for “Database Programming Techniques”, Session 16, we have loaded IMDB database into MySQL. You might go back to your codes (and those available for you on BrightSpace) to apply fixes (e.g., for the series_director table the director names are missing).

In this assignment, we will write SQL commands through Python. Your final deliverable will be one Python file named db_assignment4.py (please make sure it is .py) that has methods, plus a PDF file. Please, do not submit any other formats. Note that the tables should be similar to what we had in class, as your code will be run on MySQL backend, with the same tables.

Here are the methods:

1. `releaseyearCount()`: this method has no input and returns a list of tuples. The first element of each tuple represents the Release Year, and the second element indicates the number of series released in that year. (e.g., [(1892, 1), (1894, 14), ...])
2. `getSeries(starName)`: this method takes in the name of a star, and returns a list of series (only titles) in which the star has played.
3. `getSeries(starName, genre)`: this method takes in the name of a star, and a genre and prints the titles of series titles of that genre in which the star has played in it, along with their ratings.
4. `getSeriesCostar(list_starNames)`: this method takes in a list of stars' names, and returns the list of series (only titles) that all the input stars have played in.
5. `getPopularSeries(starName)`: this method returns a list of series (only titles) that the input star has played in and have a rating above the average rating.
6. `getRatingPerGenre()`: this method prints the average rating of each genre.
7. `getSeriesDirectorStarGenre(director, star, genre)`: this method returns a list of series (only titles) directed by the input director, feature the input star played, and belong to the specified genre.
8. The main method: for each of the methods above, provide running examples. For those returning values, provide printing statements to print the returned values.

**** Note that for all the star and director names, the full name will be used. (You do not need to consider like operator for strings)**

Functions and Procedures:

1. Write the method `createAvgFunction()` in Python that creates an SQL function, returning the average rating per genre. Then write another method in Python, `pythonAvgFunction`, that reads the data from MySQL (no other functionality of SQL), and within your code calculate the average rating for each genre. In your main method, call both methods and measure the performance (how much time did it take to execute each method). In your PDF file, discuss these timings and explain which one was faster.
2. Write the method `createProcedure()` in Python to create an SQL procedure for updating series ratings. This procedure will take in the name of the series, and update its rating. Write another Python method, `updateRating (seriesName, newRating)` that will take in the name of the series and the new rating that updates the rating. In your procedure, you should check the new rating to be in the IMDB range [0,10]. In your main method, call the second method.

Deliverables:

A .PDF file, a .py file:

1. All the Python methods should follow the names specified in the assignment. Make sure your code is documented well.
2. A .PDF file answering the question about functions in SQL and Python.
3. Provide the results of your main method at the end of the PDF file. This can be either a screenshot or copy-paste results from the console.

To address any ambiguities in the instructions, please reach out to the instructor, using email, Slack (preferred so that other students can use it), or office hours.