COMP 3311: Database Management Systems

Task 2: Table Creation and SQL Queries

Assigned: July 14, 2022 Value: 10% of course grade
Due: 23:00 (11:00 p.m.), July 23, 2022

In this task you are to submit SQL script files that create the relational database required for the movie streaming management system and that execute some queries on this database.

> **IMPORTANT REMINDER**
>
> ***This is an individual task.*** The SQL script files that you submit should be *your own work*. While you may discuss general task issues with other students, you are not allowed to collaborate with other students (past or present), to come up with a common design, to share designs or to copy someone else's design. Copying, sharing and collaborating will be severely penalized. All those involved in a copying/sharing/collaborating incident will automatically receive a grade of 0 and may be reported for further disciplinary action.

PART 1: TABLE CREATION

Create tables with exactly the same table and attribute names as the relation schemas described in the document **Task 2 Relation Schemas**, which can be downloaded from the Project Information webpage on the course website. Moreover, the order of the attributes and their types in each table should be exactly the same as that given in the document **Task 2 Relation Schemas**. Note that the order of the relation schemas in the document **Task 2 Relation Schemas** is ***not necessarily the correct order*** for creating the tables; the relation schemas in this document merely describe the attributes, their types and some of the constraints of each relation schema.

Include for each table all attribute-level or table-level constraints that either are described in the project description, are derivable from the example E-R schema for Task 1 or are stated in the document **Task 2 Relation Schemas**.

You can check whether you have defined *some aspects* of your tables correctly with the script file Task2SchemaCheck.sql which can be downloaded from the Project Information webpage of the course website. If each tuple in this script file can be successfully inserted into its respective table, then the table has been defined correctly as to the number of the attributes and possibly the order and type of each attribute. However, the order and type of the attributes is not guaranteed to be correct even if a tuple can be inserted successfully. Furthermore, the tuples need to be inserted in the correct order and the order in which they appear in the script file **is not** necessarily the correct order in which the tables should be created. The tuples may need to be reordered according to the referential integrity constraints that you have defined so that they can be successfully inserted into the tables.

Note: ***You are required to use the relation schemas described in the document Task 2 Relation Schemas. For Task 2 you will not be provided with any sample data to populate the database. You may create your own sample data if you consider that it is necessary to do so.***

PART 2: SQL QUERIES

Construct the following SQL queries using the relation schemas in the document **Task 2 Relation Schemas**.

> **IMPORTANT NOTE**
>
> You are allowed to use only SQL constructs that have been discussed in the lectures, tutorials or labs. If SQL constructs that have not been discussed in the lectures, tutorials or labs are used in a query, then no marks will be given for that query.
>
> ***Do not use PL/SQL procedures or functions; use SQL statements only.***

1. Find the genre name and the frequency of occurrence of the five most frequently occurring genres for only those movies that members have watched. Order the result by frequency descending. **[6 marks]**
2. Find the cast member name, the role he/she played, the title and the release year of the movie in which the cast member both appeared in and directed the movie, and the movie won the Best Picture Academy award. Order the result by name ascending. **[6 marks]**
3. Find the title, release year and MPAA rating of the movies that won the Best Picture academy award and whose director won the Best Director academy award for the same movie, but the movie has not been watched by any member. Order the result by title ascending. **[10 marks]**
4. Find the title, release year, running time, MPAA rating and the number of times watched for the movies that have been watched the greatest number of times. Order the result by title ascending. **[12 marks]**
5. Find the title, MPAA rating, IMDB rating, Reelflics rating, number of times watched by different female members and number of times watched by different male members for the movies that won the Best Picture academy award and that have been watched more times by different female members than by different male members. Order the result first by the number of times watched by different female members descending, then by the number of times watched by different male members descending, then by title ascending. If a movie has not been watched, then the number of times watched should be shown as zero not as null or as a blank. **[16 marks]**

WHAT TO SUBMIT

Submit two SQL script files named as follows:

1. **task2create.sql** containing the SQL statements for creating the tables specified in Part 1. (This file should not contain any sample data; it should contain only the SQL statements for creating the tables.)
2. **task2query.sql** containing the SQL statements for the queries specified in Part 2.

>>> ***Put your name and student id as a comment on the first line of each script file.***

>>> ***In the task2query.sql script file, precede each query with a comment containing the text statement of the query as given in Part 2.***

We will not grade your submission if

- your task2create.sql script file violates any of the requirements stated in Part 1,
- you create either of the script files in other file formats (i.e., doc, docx, rtf, pdf, txt, etc.) or
- you do not put your name and student id in each script file correctly.

Note: ***Each script file will be tested directly in SQL Developer. Therefore, you are strongly advised to test your SQL solution statements in SQL Developer before submitting them. If a script file cannot be run in SQL Developer, it will not be graded, and you will get zero for that part of Task 2.***

PLEASE NOTE CAREFULLY

You may not share your sql script files for Part 1 or Part 2 with other students in the course. Sharing your sql script files will be considered collaboration and will be penalized as stated above.

HOW TO SUBMIT

1. Put both script files into a folder named “task2XXXXXXXX” where “XXXXXXXX” is your student id.
2. Compress (zip) the folder “task2XXXXXXXX”.
3. By 11:00 p.m. on Saturday, July 23, upload your “task2XXXXXXXX.zip” file to Canvas by selecting *Task 2* in the Assignments section of Canvas, and then selecting the Submit Assignment button. To check your submission, select the Submission Details button. For help, select the Help button.

It is your responsibility to ensure that your task2XXXXXXXX.zip file is correctly uploaded to Canvas. Under no circumstances will late submissions or submissions by email be accepted.

GRADING

Item	Value
Part 1: Table creation	~50%
Part 2: SQL queries	~50%

CLARIFICATION AND AMENDMENT OF PROJECT/TASK REQUIREMENTS

You can ask clarification questions regarding the requirements stated in the project description or the document **Task 2 Relation Schemas**. ***All questions should be submitted to the teaching team by email at 3311rep@cse.ust.hk.*** In addition to being answered individually, if appropriate, a submitted question and its reply will be posted on the *Project Q&A* course webpage for Task 2, which can be accessed from the *Task Q&A* section of the *Project Information* course webpage. ***You should check this webpage on a regular basis for further clarification and amendment of project requirements.*** Any requirements added or amended in a *Project Q&A* webpage will become part of the project requirements.