INF115 Databases and Modelling
Obligatory Assignment 1

Deadline: 18:00, Friday, 25.02.2022

Submission

The submission for this assignment should contain four .sql files corresponding to the code needed to answer each of the four tasks. These should be submitted in a .zip file. The deadline for submission is 18:00 on Friday the 25th of February. Late submissions will be penalized with a 10% point deduction for each day it is late. The percentage of points contributed to the total assignment grade for each completed task is indicated in the header for each task. Successful submission of the assignment (i.e. 4 .sql files within a .zip file), accounts for 10% of the assignment grade.

Introduction

You are recruited as an IT volunteer for the university film festival (UiBIFF). In this position, your duties revolve around working on the information infrastructure of the festival. Among other things, this enables increased ease in planning film screenings and logistics at the venue.

To make this happen, you need to enable the storage and retrieval of information pertaining to films, other volunteer workers, and the screenings of the films themselves. Luckily, you're not the first IT volunteer employed by the festival, and someone has already started your information storage work for you. Useful code (and their corresponding task) for creating and populating tables can be found in the file compulsory1_code_handout.txt. The administration would like to see a demonstration of the currently existing tables, as well as an expansion upon on this for further information storage. The following tables contain information about films and volunteers:

- Film: (film_id, title, director, genre, country, duration)
- Volunteer: (volunteer_id, name, position, age)
- Screening: (screening_id, film_id*, showtime, room)
- Ticket: (ticket_id, full_name, screening_id*, ticket_price)
- Volunteer_at_screening: (screening_id*, volunteer_id*)

Film: This table describes the films that have been admitted to be shown at the festival.

- film_id: A number uniquely identifying each film. Primary key.
- title: The name of the film.
- director: The name of the director of the film.
- genre: The genre of the film.
- country: The country in which the film was made.

- duration: The duration of the film.

Volunteer: This table describes the volunteers that are registered to work for the festival.

- volunteer_id: A number uniquely identifying the volunteer. Primary key.
- name: The name of the volunteer.
- position: The position occupied by the volunteer.
- age: The age of the volunteer.

Screening: This table describes when and where the films are to be shown, as well as who are the volunteers responsible for the screening.

- screening_id: A number uniquely identifying the screening. Primary key.
- film_id: A number identifying the film to be shown at the screening. Foreign key.
- showtime: The star time of the screening.
- room: The room in which the screening is set to take place.

Ticket: This table describes tickets bought for the various screenings. The name of the attendee, the screening they plan to go to, and the price of their ticket.

- ticket_id: A number identifying the ticket. Primary key.
- full_name: The name of the ticket holder.
- screening_id: A number uniquely identifying the screening. Foreign key.
- ticket_price: The price paid for the ticket.

Volunteer_at_screening: This table describes the various volunteers and which screenings they will be working.

- screening_id: A number identifying the screening the volunteer is working. Primary key. Foreign key.
- volunteer_id: A number identifying the volunteer working a certain screening. Primary key. Foreign key.

Figure 1: Logo for your companies: BIFF and UiB

Task 1: Single table queries (20%)

Demonstrate the working state of the pre-made Film table by formulating queries to answer the following questions:

(a) Write a query to get a list of countries that have submitted films to the festival without duplicates.

(b) Write a query to display information about all drama films shorter than 2 hours long.

(c) Write a query to display information about films starting with the letter L.

(d) Write a query to get a list of countries that have submitted more than 2 films and the number of films they submitted. The column with the number of films should be named NumFilms.

(e) Write a query to get the total duration of all Italian films put together in a column named TotalTime. Hint: use sec_to_time and time_to_sec functions to calculate with time datatype.

Task 2: Creating and modifying tables (20%)

Your next task is to expand upon the pre-made tables

(a) Create the Screening table. Choose datatype for each column based on the values in Table 1 (1), and define the primary and foreign key as indicated respectively by the underline and the asterisk.

(b) Insert the values from the table below into the Screening table you just created (1).

(c) Formulate the following constraints and apply them to the Screening table.

  (i) The venue opens at 08:30 and closes at 23:00. Write a constraint that ensures no screenings begin before the opening time or after closing time.

  (ii) Five rooms at the venue are being used for the festival screenings. Write a constraint to ensure that no screenings are entered at any other room that MB01, MB02, MB03, MB04, and MB05.

(d) In the case that two films have the same title, their titles need to be uniquely identifiable. Formulate a constraint that ensures that no two films in the film table have the same title.

Screening_id	film_id	showtime	room
1	20	09:00:00	MB04
2	18	09:30:00	MB03
3	23	10:30:00	MB02
4	12	10:30:00	MB05
5	19	11:00:00	MB04
6	19	12:00:00	MB03
7	24	12:00:00	MB05
8	6	13:30:00	MB04
9	25	14:00:00	MB03
10	11	16:00:00	MB01
11	13	16:00:00	MB03
12	21	16:00:00	MB04
13	12	16:45:00	MB05
14	10	18:30:00	MB01
15	13	17:00:00	MB01
16	23	17:00:00	MB02
17	7	18:00:00	MB04
18	25	19:15:00	MB05
19	22	20:00:00	MB03
20	12	20:00:00	MB04
21	4	22:00:00	MB03
22	14	21:30:00	MB02
23	21	22:00:00	MB04
24	13	22:00:00	MB05
25	3	22:30:00	MB01

Table 1: This is the data that should be inserted into the Screening table.

Task 3: Multiple Table Queries (25%)

There are also two more tables available. Create and populate these, then extract information from multiple tables by constructing queries to address each of the prompts below.

(a) Write a query that shows the title and showtime of all horror movies.

(b) Write a query that shows the title, director, and number of screenings for every movie that is screening more than once. The results should be shown in order of descending number of screenings.

(c) Write a query that shows the average ticket price each visitor pays for movies in each genre.

(d) Write a query to show the name and the average ticket price for the director whose films have the highest average ticket price.

(e) You have been given the task of creating a poster with all the screenings of the film festival. Write a query that displays all the screenings, when they are, where they are and what film they are showing.

Task 4: Advanced Queries (25%)

Demonstrate that the infrastructure can also answer more advanced questions by writing queries to address each of the following prompts.

(a) Write a query to calculate the average age of all volunteers that are working thriller movie-screenings in a column called AverageAge. The result should be shown as an integer.

(b) Write a query to calculate which volunteer spends the most time at American movie-screenings. The result should show the total time the volunteer spends at these screenings, as well as their volunteer ID, name, and age.

(c) Write a query to find all visitor-volunteer pairs that will attend at least two different French movies together. The result should show the name of the visitor, the volunteer, and the number of French movies they are watching together.

(d) Write a query to display the screening-schedule of the volunteer with ID 4. The result should display the showtime, the room, and the title of movie at each of the screenings for this volunteer.