Lab Exercise 7

Dr. Sarvar Abdullaev
`s.abdullaev@inha.uz`

March 13, 2019

The purpose of this lab is to practice MySQL, and implement simple registration and login forms using MySQL, Cookies and Session.

1 SQL Queries

In this exercise, you are supposed to practice SQL queries. Find `imdb_small.sql` file uploaded with this lab exercise in `sql` folder and set it up on your MySQL instance using phpMyAdmin.

Figure 1 shows an example snapshot from IMDB database:

actors

id	first_name	last_name	gender
433259	William	Shatner	M
797926	Britney	Spears	F
831289	Sigourney	Weaver	F
...			

movies

id	name	year	rank
112290	Fight Club	1999	8.5
209658	Meet the Parents	2000	7
210511	Memento	2000	8.7
...			

roles

actor_id	movie_id	role
433259	313398	Capt. James T. Kirk
433259	407323	Sgt. T.J. Hooker
797926	342189	Herself
...		

movies_genres

movie_id	genre
209658	Comedy
313398	Action
313398	Sci-Fi
...	

directors

id	first_name	last_name
24758	David	Fincher
66965	Jay	Roach
72723	William	Shatner
...		

movies_directors

director_id	movie_id
24758	112290
66965	209658
72723	313398
...	

Figure 1: IMDB Database Tables

You can view database scheme given in Figure 2

Write SQL queries for IMDB database that answer following questions. Save your SQL queries in `imdb_queries.sql` file.

1. What are the names of all movies released in 1995?

2. How many people played a part in the movie "Lost in Translation"?

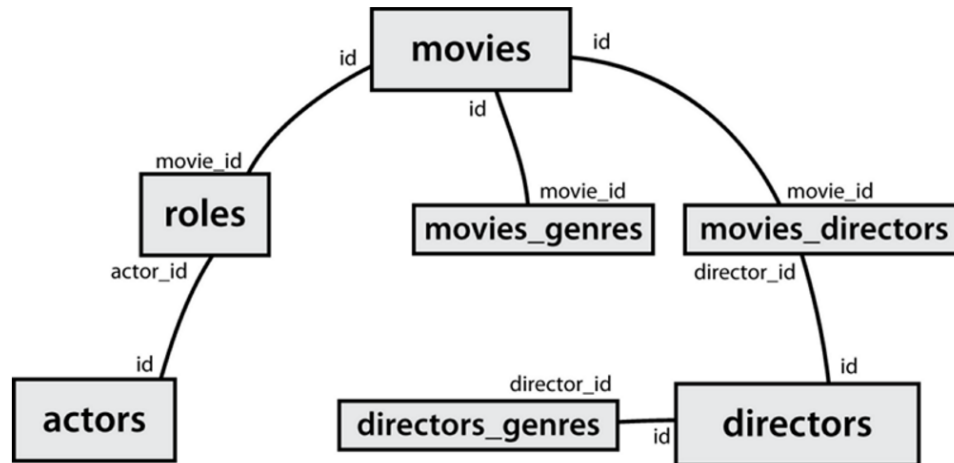3. What are the names of all the people who played a part in the movie "Lost in Translation"?

Figure 2: IMDB Database Schema

4. Who directed the movie "Fight Club"?

5. How many movies has Clint Eastwood directed?

6. What are the names of all movies Clint Eastwood has directed?

7. What are the names of all directors who have directed at least one horror film?

8. What are the names of every actor who has appeared in a movie directed by Christopher Nolan?

2 My Blog

In this section, you will create a personal blog which registers users and authenticates them. It will also allow them to make blog posts. Below are the tasks that should be accomplished:

1. Open phpMyAdmin and create a new database named `blog`. In this database, create 2 tables as specified below:

Table 1: Users

Field Name	Data Type
id	int (PK^a, AI^b)
username	varchar(250)
email	varchar(250)
password	varchar(250)
fullname	varchar(250)
dob	date

^{a}Primary Key
^{b}Autoincrement

Table 2: Posts

Field Name	Data Type
id	int (PK, AI)
title	varchar(250)
body	text
publishDate	date
userId	int (FK^a from users.id)

^{a}Forein Key

2. Review `register.php` file inside `webpage` folder. It provides a simple form for user registration. Modify this form to create a new user in `users` table in `blog` database when user presses *Submit* button. After creating the new user, it should redirect user to `index.php` page. You can use `header('Location: index.php');` function to redirect user to `index.php` page.

3. Review `index.php` file inside `webpage` folder. It should display *login form, add new post form* and *list of all posts*. Modify this form to authenticate user (i.e. find corresponding user from database

and check password) and store his/her details in `$_SESSION` variable. For example, you can assign retrieved user row object to `$_SESSION` like this `$_SESSION["user"]=$row`. If user does not exist in database, show corresponding message about that. Once the user successfully logs in, hide *login form* and show *add new post form* which had to be hidden before. You can check whether user is logged in or not by `isset($_SESSION["user"])`. Also after successful login, a link *Log Out* referring to `index.php?logout=1` should appear next to *add new post form*. When this link is pressed, `index.php` should capture `logout` parameter passed by `GET` and then call `session_destroy()` to clear all values stored in `$_SESSION`. This should automatically show *login form* again. Login form is shown in Figure 3.

Figure 3: Login form

4. In *login form* of `index.php` implement *Remember Me* function using cookies. If *Remember Me* checkbox is checked, then in *index.php* you should create a cookie that stores `username` of the user (i.e. call `setcookie("username", "name", time()+60*60*24*365)`), so next time user opens this page, it should populate `username` field in *login form* from `$_COOKIE["username"]`. If *Remember Me* is unchecked, then `username` cookie should be removed (i.e. call `setcookie("username", "name", time()-1)`).

5. In `index.php`, you should retrieve all blog posts from database and modify static HTML to display every field dynamically for each post.

6. In `index.php`, you should implement functionality of adding new posts for logged in user. It means that you should create new post in `posts` table of `blog` database. Make sure that `userId` is equal to the `id` of the user that is logged in. Figure 4 shows how it should look.

7. In `index.php` add a link "My Profile" to `register.php` which shows only if the user is logged in. If user is already logged in and he/she opens *register.php* it should populate the registration form with user's details. All of these details can be taken from `$_SESSION["user"]` variable. After pressing *Submit* button, all changes to user's data should be updated in the database, and user should be redirected to `index.php` page again.

My Profile | Log Out

New Post

Title

Body

Post

TITLE HEADING

Author, Sep 2, 2017

Some text..

Sunt in culpa qui officia deserunt mollit anim id est laborum consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco.

Figure 4: Add New Post form

My Blog

In this web site you can leave any post you want.

Registration Form

Please, fill below fields correctly

Username

Full Name

Email

Password

Confirm Password

Submit

Already registered? Login

Figure 5: Registration/My Profile Form