Database Setup

Database: Create and Populate Database Tables, SQL Select Statements

CONTENTS

1. OVERVIEW ........................................................................................................................................... 1
2. BEFORE YOU START .......................................................................................................................... 2
3. CHANGING YOUR MIND ABOUT WEB APPLICATION FUNCTIONALITY ............................................. 2
4. DATABASE REQUIREMENTS ............................................................................................................... 3
   a) Create Tables .............................................................................................................................. 3
   b) Add Data ..................................................................................................................................... 4
   c) Run SQL Select Statements ........................................................................................................ 4
5. BLOG ................................................................................................................................................. 5
6. SUBMISSION REQUIREMENTS .......................................................................................................... 5

1. Overview

In this homework, using MySQL Workbench, you will:

1. Create database tables:
   - web_user and user_role (exactly as prescribed) and
   - an “other” table named and designed as you propose (meeting the requirements of this document).

2. Populate all three tables with test data.

3. Generate several SQL select statements to extract data from the database.

4. Create a word document that shows your database work.

5. When you have a blog page created in your web site, be sure to create a blog entry that describes your database
   experience and links to the word document that you just created.
2. Before You Start

Before starting on this Homework, make sure that you have installed MySQL Workbench and tested that you can connect to your database (if you added this course late, you may have to email your instructor to request a database be created for you ASAP). From the course web page (for this assignment) you’ll find links to the following documents:

- How to install MySQLWorkbench
- How to connect to your Temple database (shows you how to find your database credentials, etc)
- MySQL Workbench Tutorial: How to create database tables and relationships (primary keys, foreign keys), enter data, and write SELECT statements

Create a word document to hold this week’s homework submission. Into this document put the following:

1. Your name and NetAccess user name
2. Your Web Site title
3. The (possibly updated) content of your home page (copy/pasted from your first Homework), where you specify the functionality of your web site and entice users to visit your site.
4. In the next sections, you will be asked to copy/paste screens (into this word document) from MySQL Workbench – to show that you have done the prescribed database work.
   
   - To get a screen capture, click on Alt-PrtSc (copies active window into the clipboard), then paste into an image editor like MSPaint, then copy out just the part you want to show and paste that into the word document. If you paste from Alt-PrtSc directly into the word document, the screen capture shows too much, making things too small to read.

3. Changing Your Mind about Web Application Functionality

It is OK to change your mind (from last week’s homework) about the functionality that your web application will provide – as long as your database table meets the design requirements of this homework AND the web functionality you describe on your home page matches up with your database design. Basically, think of your web application as a place where users can contribute information (whatever info you designed in your “other” table) and browse information contributed by others.
4. Database Requirements

a) Create Tables

*Design tables web_user and user_role exactly as shown below* (this is so that sample code will work against your database).

![Table Designs](image)

Then add a foreign key from web_user.user_id to user_role.user_id. The MySQL Workbench tutorial (linked from this week’s homework) tells you how to do this, but you open up a new query tab (File – New Query Tab) then enter and execute the following SQL:

```
ALTER TABLE web_user ADD FOREIGN KEY (user_role_id) REFERENCES user_role (user_role_id);
```

Then **create your own unique table** (which I will often refer to as your “other” table). You’ll name this table and design its fields according to what you proposed in your last homework. Make sure that what you propose for your database table aligns with what you said your website will allow users to do. Also be sure to meet these additional requirements:

- Id: primary key, auto-increment, name this field with table name followed by "-_id"
- Name or some kind of character identifier (require it to be unique, click on “UQ”)
- image URL – a long varchar since fully qualified URLs can be pretty long
- long character (description)
- at least two nullable (user optional) non-character fields, e.g., integer, decimal (for money), date
- “web_user_id”, a foreign key that points to the user who contributed this information.

You can add more fields if you like, but don’t add too many extra fields because it will be more work for you with every homework assignment. **Don’t use any SQL keywords** as table names nor field names or else your web app will have trouble trying to run various SQL commands against your table. (Google to see list of SQL keywords, but you cannot use these for sure: user, role, password, state, date.)

**Paste a screen capture of the following table designs** into your word document:

- table design of user_role (the “columns” tab of what you see when you right click a table and select “alter table”).
- table design of web_user. Then also capture the “foreign keys” tab to demonstrate that you added the foreign key from web_user to user_role.
- table design of your “other” table then also capture the “foreign keys” tab to demonstrate that you added the foreign key from your other table to web_user.
b) Add Data

Add realistic looking data into your three database tables.

- Add 3-4 records into your user_role table.
  - Since your web_user table references data from user_role, you’ll need to enter records into user_role first.
  - Try to add a record with a duplicate primary key and notice that the database management system will not let that record be inserted.

- Add 5-7 records into your web_user table.
  - Since your web_user table references data from user_role, you’ll need to enter records into user_role first.
  - Try to add a record with a duplicate primary key and notice that the database management system will not let that record be inserted.
  - Then try to delete a user_role record that has been referenced by a web_user record. Notice the DBMS will not allow you to do that either. It is the “job” of the DBMS to maintain the integrity of the DB at all times, allowing no invalid foreign key references.

- Add about 15 records into your “other” table.
  - At least one of these records shall have null for all nullable non character fields. At least one of these records shall have all of its fields populated.


c) Run SQL Select Statements

Execute each of the following SELECT statements (and paste screen captures into word doc). Each screen capture should show the Select Statement and the result set like this:

a. Run a SELECT statement that lists all the columns of all the records of your “other” table, selecting out each column and presenting them in an order that you think users would like to view the data (don’t use SELECT * ). Sort the data by whichever column you decided to show first.

b. The SELECT statement above but adding a WHERE clause so that you see some of the records but not all.

c. A SELECT statement that shows all the records from the web_user table joined with the user_role table. Show the role name first, then email address then all the other columns (from both tables) in an order you think users would like to see the data – except don’t list the role id twice. Order the records of the result set by the first columns (primary sort role name, secondary sort email address). Note: There should be as many rows in your result set as you have records in your user table. If you have a lot more (and see duplication), you have forgotten the WHERE clause that joins the two tables together.

If you don’t know how to do the above, go back and read the MySQL tutorials mentioned at the start of this document.
5. **Blog**

Once you get a blog page to your web site (or if you already have one), add a blog entry describing your database experience, telling what you found easy/hard about setting up your database, and linking to your database document.

6. **Submission Requirements**

Submit your word document into canvas. It should contain the following:

- Your name and NetAccess user name
- Your Web Site title
- The content of your home page (copy/pasted from your first Homework), where you specify the functionality of your web site and entice users to visit your site.
  - It’s OK if you changed your mind (about your web site functionality) between now and last homework, but your web site functionality must match what you propose for your “other” database table and your “other” database table design must meet requirements of this document.
- Table designs of three tables: user_role, web_user, and “other”. For web_user and “other”, also include screen captures of the foreign key tabs.
- Screen captures of the SELECT statements mentioned above (after you filled the tables with data).