1. Overview

In this Lab Activity, you will:

- Using MySQL Workbench,
  - Design & populate data into a database table named “web_user” (all students will have the exact same database design, but with their own data inside).
  - Design & populate data into a second (which I usually refer to as your “other”) database table named and designed to hold data relevant whatever topic you chose for your web application (and which also meet the requirements described below).
  - Write some SQL SELECT Statements.
- Paste screen captures from MySQL Workbench into a word document (and demo and upload to Canvas before end of lab period).

For the related homework, the only additional things you need to do (beyond the Lab Activity) is:

- Update your web application so that your blog has an entry about your database work, including a link to the word document mentioned above.
- Make sure that the “marketing material” in your home page (explaining what your site has to offer and enticing users to visit your site,) aligns with the design of your “other” database table. Basically, your web site will allow users to add records to your “other” table and it will allow them to see “other” records contributed by other users.

2. Before You Start

In the 3344 Web Page (where you found this document), there are three documents to help you if needed:

1. How to install MySQL Workbench on your home PC/Mac (this will be helpful for you to be able to easily see and modify the data in your database).
2. How to connect to your database (there is a special file created in your home folder under cis-linux2 that holds your auto-generated database password that is different from your NetAccess password).
3. A MySQL Workbench/Database tutorial that tells you how to create tables and modify data.
3. Lab Activity Requirements

- After connecting to MySQL through MySQL Workbench, **set your schema to be the default schema** (and then you should see your database schema shown in bold). To do this, double click on your database schema or right click your database schema and select “set as default schema”. So that you don’t have to scroll through zillions of databases, you can type in all or part of your database name (e.g., SP19_3344_tua12345) into the filter box under “SCHEMAS”.

To create a new table, open up your database schema (click on triangle) then right click Tables and select “Create Table”.

**B. In your database, create a table named “web_user”** designed EXACTLY as shown in the diagram below. (If you don’t have it exactly the same, sample code will not work for you.) Please note that the relevant column attribute headings are as follows (or you can hover over the column headings in MySQL Workbench to see what the abbreviations stand for).

- PK means primary key of the table
- NN means “not null” which means the field is required to be provided whenever a record is inserted/updated.
- UQ means unique (cannot have more than one record with same value for this column in this table)
- AI means Auto-increment – the database will assign the next available number when you insert a record (and do not provide a value for this field).
C. Insert web_user Records.

- Right click your web_user table (from the database navigator area) and “Select Rows” from your (empty) user table.
- Type in one row of data filling out user_email and user_password but not web_user_id (remember this is an auto-increment type field that the database will supply for you).
- Click the Apply button and check out the INSERT SQL command that MySQL Workbench created for you (to do the insert). It should look something like this (but with your own database name and your own data values):

```
INSERT INTO 'FA19_3344_tua12345'.web_user (user_email, user_password) VALUES ('sallyk', 'pw');
```

- Click through the process to insert the record and hopefully the record will be successfully inserted (as shown at the bottom pane). The message “1 row(s) affected” means that the insert was successful.

- Type in a second web_user record, this time filling in all the columns. MySQL wants you to type in dates in this format: YYYY-MM-DD. Note: put in any values you want – you don’t have to use the data values I show here:

```
<table>
<thead>
<tr>
<th>web_user_id</th>
<th>user_email</th>
<th>user_password</th>
<th>birthday</th>
<th>membership_fee</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><a href="mailto:joe.blow@temple.edu">joe.blow@temple.edu</a></td>
<td>dw</td>
<td>NULL</td>
<td>NULL</td>
</tr>
<tr>
<td>2</td>
<td>test</td>
<td>test</td>
<td>1998-02-03</td>
<td>123.08</td>
</tr>
</tbody>
</table>
```

- Type in more records until you have 5-7 realistic looking records in your web_user table.
D. In your database, create a second table for data related to the topic you chose for your web application. Although I often refer to this table as your “other” table, give your table a self-documenting name (don’t call it “other” or “myTable”). Include at least the following fields in your table:

- xxx_id (primary key, auto-increment), where “xxx” is the table’s name.
- A character descriptor or other identifier (specify this as a unique type field)
- A long varchar field to hold a fully qualified URL to an image.
- If you wish, you can also have a field that will hold a URL to a website.
- At least 2 other fields which are non-character (either date, integer, or decimal which is good to hold monetary values) and nullable.

As you design this table:

- Do not any SQL keywords in your table design (table name, field names) because it will cause unwanted database exceptions later in the semester accessing your database from java code. Some SQL keywords to avoid: role, user, password, state (google "SQL Keywords" for complete list)
- Table names should be singular, not plural, e.g. "student" not "students".
- Table names should not include space – you can use underscore if you have a multi-word name, or you can use camel case since the SQL in MySQL is case sensitive.

E. Insert “other” records.

- Populate your “other” table with around 15 records of realistic looking data. Make sure that at least one record has all fields filled out and at least one record has null in all nullable fields.
- Note: So that your web application can actually show images, make sure that the image URLs are valid and point to images out in the internet (use fully qualified URLs like http://...)

F. Write and execute the following three **SQL SELECT statements**, then get a screen capture of each and paste this screen capture into your document. Each screen capture shall include: SQL Query area, Result Set area, plus Output area, as shown below. To get a good (legible) screen capture, size the areas of MySQL WorkBench as small as possible showing only the data you need to show, then Alt-PrtSc as mentioned before, and select out just the part you need to show.

- **First select statement**: All the fields of all records in your `web_user` table, ordered by email address. Select the column names individually (don't use `SELECT *`) beginning with the email address which is the “ORDER BY” column.

- **Second select statement**: All the fields of records in your Other table, ordered by the unique descriptor field. Select the column names individually (don't use `SELECT *`) beginning with the descriptor field which is the ORDER BY column and followed by the other columns in the order you think your users would want to see the data on a web page.
4. Lab Activity Summary: Contents of Word Document (to Demo and Upload)

The document with your screen captures shall be named with your last name in it and it shall include the following:

- Table Designs (from MySQL WorkBench):
  - Screen capture your web_user table design (right click on table and select “Alter Table”)
  - Screen capture of your “other” table design.
- SQL select statements: Screen capture of each of the 3 SQL select statements listed on the previous page.

Note: to get a screen capture, press Alt-PrtSc (hold down the Alt key while pressing the PrtSc key) to copy the active window into the clipboard. Then paste into a program such as MSPaint. From MSPaint, select / copy just the part you need (e.g., the table design) then paste into your homework document. If you try to paste the whole window directly into the HW document, usually the part that we need to check is too small to see.

Lab Activity Grading Milestones:

- 2/8: web_user table designed as specified → screen capture of table design in Word document.
- 4/8: web_user table populated with 5-7 records (one record having null for all nullable non character fields, and one record with all fields populated). SQL select statement (starting with email address, ordered by email address, all fields specified, all records listed) in Word document.
- 6/8: “other” table designed as specified → screen capture of table design put in Word document.
- 8/8: “other” table populated with about 5 records (one with null for all nullable non character fields and one record with all fields populated). SQL select statement (starting with your descriptor field, ordered by descriptor field, all fields specified, all records listed) in Word Document.

5. Homework Requirements

For your homework, we will more carefully scrutinize what you did in lab activity – see example deductions. If you didn’t have time to complete everything perfectly in Lab Activity, fix things up before submitting/publishing your homework. Make sure your data is complete and realistic, complete with good image fully qualified URLs (fully qualified meaning http://completeLink). Select images that are not super huge in file size so that eventually your web pages are not slow to load (super huge defined as 750K or more). Add more records (about 15) to your “other” database table.

Make sure that the content you see when you first visit your web application (home content) provides an explanation of what your web site offers (your site will basically allow users to insert records into your “other” table and see records uploaded by others).

To your blog page, add a blog that 1) describes what you did in this homework (what was important, what was easy, what was hard or confusing) and 2) links to the Word document described just above. You link to the document just like you would like to a html page, for example:

Click <a href="smithDataBase.doc">here</a> to see my database work.

NOTE: This is a VERY EASY Homework – spend extra time working on your Tutorial...
6. Submission

As always, publish your work and test that your blog is there and links correctly to the word doc. Then submit a zip file of your whole project into blackboard. Start getting used to submitting the WHOLE project (not just the “web pages” folder) since beginning next week, the whole project becomes part of the necessary source code.

7. Homework Grading: Example Deductions

- Up to -2: web_user table not designed exactly as specified in the requirements.
- Up to -2: web_user table not populated with realistic data, as required (5-7 records, one record with null values for all nullable non character fields, one record with all fields populated).
- Up to -2: web_user SELECT statement not as specified.
- Up to -2: “Other” table not designed as specified in the requirements.
- Up to -2: “Other” table not populated with realistic data, as required (about 15 records, one record with null values for all nullable non character fields, one record with all fields populated).
- Up to -2: “Other” table SELECT statement not as specified.
- Up to -2: Marketing Material (initially in content area of index page) missing or not professional/compelling.
- Up to -2: Mismatch between Marketing material and design of “other” database table.