Messing With MySQL

The Coding Bootcamp

Purpose of SQL and MySQL

- SQL (often pronounced "Sequel") stands for "Structured Query Language" and is a powerful programming tool that provides ability to **create**, **populate**, **manipulate**, and **access** databases (server-side storage).

- MySQL (often pronounced "My Sequel") is a popular type of open source software that can be placed on a server so as to allow SQL commands to affect the data stored on the server.

- Data using SQL is stored in tables on the server much like those you would create in Microsoft Excel or in Google Sheets, making the data easy to visualize and search through.

MySQL Workbench…

- In order to code visually in MySQL, we will require a coding environment other than that of Visual Studio Code. We will use MySQL Workbench. MySQL Workbench enables you to visually design, model, generate, and manage databases. It provides data modeling, SQL development, and comprehensive administration tools for server configuration, user administration, backup, and much more.
https://www.mysql.com/products/workbench/

- Debugging installs is a skill you will need as as developer, so use this as a learning experience at troubleshooting installations and configuring software on your machine. If your install is easy, help someone who might be having some trouble!

Creating a Localhost Connection

- Since we do not have a defined server for you to connect to, we are going to set up something called a "localhost connection" for you to use.

- This type of connection allows us to create locally stored data on our computers as if they were an external server.

- This is a much better alternative to spending hundreds to thousands of dollars on buying a server itself for the purposes of practicing on.

Creating a Localhost Connection

- MySQL Workbench can sometimes seem a little hard to comprehend when loaded up for the first time and that confusion is perfectly normal.

- There are some errors that are likely to pop up over the course of this assignment but that we will be walking around to help them with whatever issues make themselves known.

- If you successfully created a localhost connection, help those who seem to be struggling or start reading up on how databases are created and used in MySQL.

Connections Vs. Databases

- Now that you have created a connection to the localhost does this means that you have created a database as well?
- No.
- Notice that the "SCHEMAS" section on the left side of the page is pretty much empty other than including the built-in sys, sakila, and world databases. These databases would not appear on external servers.
- The connection is a road which leads to an empty lot. Before we can populate the area, we first need to lay the foundations for houses (create a database), and build the houses (create a table).

Creating MySql Databases

- Example: CREATE DATABASE animals_db;
- This creates a MySQL database on the server you are connected to.
- The semicolon at the end of the statement is necessary.
- The lightning-bolt symbol just above the editor executes your code.
- At the bottom of the screen, there is a confirmation stating that the database has successfully been created.
- We need to reload our connection and the "animals_db" database will now appear within the navigator. We have just successfully created our first-ever database!

Creating MySql Tables

- Example: CREATE DATABASE animals_db;
- This creates a MySQL database on the server you are connected to.
- The semicolon at the end of the statement is necessary.
- The lightning-bolt symbol just above the editor executes your code.
- At the bottom of the screen, there is a confirmation stating that the database has successfully been created.
- We need to reload our connection and the "animals_db" database will now appear within the navigator. We have just successfully created our first-ever database!

The value of unique values

- We can create a column that automatically populates each new row with unique data. This allows us to select and affect that row more easily.
- A **primary key** uniquely identifies a row
- AUTO_INCREMENT generates a new value for each inserted record in the table. By default, the starting value for AUTO_INCREMENT is 1, and it will increment by 1 for each new record.
- Using this with a unique PRIMARY KEY means we automatically get unique, incrementing values for each table row.
- We do not need to insert data specifically into the id column. MySQL automatically provides a value for this column, fulfilling the uniqueness constraint by automatically incrementing the last value used as an ID.

Joins

- Joins allow those using MySQL to combine two or more individual tables together using a value that is shared between them.

- We can utilize joins to combine and display data from both tables. We use foreign keys that corresponds to an id of a row in another table. These tables have what's known as a one-to-many relationship, since a book can only have one author, but an author can have many books.

- Foreign keys are most commonly used to relate tables since primary keys are unique and won't change.

The Three Types of Joins

- INNER JOIN: Combines tables where the specified data-values within a column match one-another

- LEFT JOIN: Combines tables and shows all of the values of the first table specified while only the values which match on the second side will be shown

- RIGHT JOIN: Combines tables and shows all of the values of the second table specified while only the values which match on the first table will be shown