ITSE 2302 – Program 09

All of the assignments in this course must be uploaded to Bb **and** *published to your student website*. Use the numbered list below **and** the Common Requirements as a confirmation checklist.

- Complete the steps described at the Website Setup & Publishing link in the "Assignments" section in Bb.
- Use the course material located at
  - HTML Tutorial , CSS Tutorial , JavaScript Tutorial , PHP Tutorial
- Program 09 can be completed using content from these tutorial sections:
  - HTML HOME – MySQL Insert Multiple
- Use only MySQLi Object Oriented or PDO. Do not use the older procedural syntax of MySQLi.
- Develop the page/site for your user – Communicate with your user.

Requirements (remember to identify the requirements by #*number*):

Note: the requirements below refer to the index.php within the program-## folder unless otherwise stated.

Note: a database has been created for you. The name of the database is the same as the database username and is of the format: itse230200#0## which is like your HTTP and SFTP usernames. The details are on your Website Credentials in Bb.

1. The Program 09 link on your top-level page should open an index.php page in the program-09 folder. Include a button on the page that connects to your database and displays the connection status.
2. Include another button that creates a connection, closes the connection, and then displays the connection status.
3. Button - Creates a table named teams with the following fields: id, teamname, city, bestplayer, yearformed, website. Display the success/fail status of the table creation. Note: the id field should be UNSIGNED, AUTO_INCREMENT, and a PRIMARY KEY.

4. Ask user to enter data for 4 teams. Use the "insert multiple" technique to insert all 4 rows at the same time. Ensure all data is valid. Ensure the website URL is a valid URL. Insert data into the table. Display the success/fail status of the row insertion operation **and** the id of the last row inserted.

```php
// When using multi_query, need to cycle through result set
// to get last autoincrement number with insert_id.
do {
  $lastId = $DBconnect->insert_id;
} while ($DBconnect->next_result());
```

NOTE: This may be helpful when using submitted variables in SQL statements:

```php
// Obtaining and using submitted variables.

// Suppose a string was submitted.
$c = $_POST['c'];

// Include quotes for string data
$sql = "SELECT * FROM teams WHERE city = '$c'";

// Sometimes written like this:
$sql = "SELECT * FROM teams WHERE city = '".$c."'";

//////////////////

// Converting q to an integer (intval not
// required if sure q is an integer).
$q = intval($_POST['q']);

// This version works if $q is numeric
$sql = "SELECT * FROM teams WHERE id = $q";
```

Database Access – DB Name, Server Name, and Credentials

Locally: (this is your local machine)

Database Name: itse230200#0## (you need to create this database)

Server: localhost (or 127.0.0.1)

Username: a local account of your choosing

Password: an appropriate local password of your choosing

Remote: (this is the lineofcode.com server)

Database Name: itse230200#0## (this database has been created for you)

Server: lineofcode.com (or localhost if your code is published)

Username: itse230200#00# (per Website Credentials)

Password: ******* (per Website Credentials)

Server Name in the Connection String

This is a typical string used to connect to a MySQL database via PHP code:

$DBConnect = @new mysqli($dbhost, $dbuser, $dbpassword, $dbname);

The variable $dbhost will contain the value of the desired server name. Here are a few examples:

$dbhost = localhost; (Use this when the PHP code AND the MySQL database are on your *local* PC. The *local* PHP code will point to the *local* database.)

$dbhost = 127.0.0.1; (same as localhost but less common; requires setting in a host file)

$dbhost = lineofcode.com; (Use this when you want PHP code running on your *local* PC to connect to the *remote* lineofcode.com server.)

$dbhost = localhost; (Use this when you publish PHP code to the lineofcode.com server and the code is running '*locally*' on the same server or the same network context as the database server.)

TEST – TEST – TEST your **website** to ensure the requirements are met.

1. Use the list above **and** the Common Requirements as a confirmation checklist.
2. Not meeting all requirements = 0 points for the assignment.
3. Complete the appropriate Program ## confirmation in Bb by the due date.
4. Upload your .zip file to Bb under the appropriate Program ## link by the due date.
5. Ensure your assignment functions correctly on your student website and that all requirements are identified by "#XX".
6. Remember to study for the exam. Distributing learning over time is far superior than panic-study in close proximity to the exam date.