Charlie Wallin
April 7, 2017

Bowling - Project Phase 01

Objectives

- Run a SQL script to create a database
- Run a SQL script to populate tables
- Link to database with an external file
- Display contents from database in a drop-down menu

Introduction

To start with, most of this content for this part of the project is from chapter 11 and the database chapters 8 - 10. The SQL statements will be provided.

This project is designed to show you some code in a tutorial then along the way you will be asked to write additional pieces to finish the code.

Once we get the functionality working, we will add some bootstrap to format the site and give it a little shine.

Part 01 - Bowling Database

Download the bowling.sql file from Moodle. Unzip and open the file in a text editor. Read through the contents.This is the same database that we used in CTI 110. It's okay if you don't understand all of the code. The important thing is to get it to run. Here is an outline of what the code does

1. Create a database named bowling
2. Create three tables
    a. Teams
    b. Bowlers
    c. Scores
3. Populate the three tables

The relationship between the tables is as follows

- One team can have many bowlers
- One bowler can have many scores

This is usually written in the shorthand

- teams 1:M bowlers

- bowlers 1:M scores

Development environment

Use your development environment to run the blowers.sql file. This is the same process that you used in the previous exercise to run the bookorama.sql file. Re-read the instructions or watch the videos from that assignment if you are having difficulty remembering how to do this.

Production environment

Now complete the same task on your webhost (production environment). As a reminder you will need to associate a user with the bowling database. You created an admin user in the previous exercise. You can associate that user with this database. A user in MySQL can be associated with an unlimited number of databases.

Part 02 - Display Teams

Now that we have the databases configured you will work in your production environment for the remainder of this exercise until the very end when you push your code to your production environment.

We will be writing code, then copying and pasting it as we work through the exercises over the next few weeks. This breaks one of the main tenets of programming "do it once". Once we get a good base of code, we will start to refactor it until we only have one instance of most of our code.

You can see the output of the code from the site

http://charliewallin.com/web182/my-code/prj01-teams/

Code example

- Create a folder called **proj01-teams** inside your my-code folder.
- Create a file named **index.php** with the following code

```php
<?php
include 'database/db-conn.php';

// create a new database object by calling a function
$db = dbConnect();

// select all from teams
$query = "SELECT * from teams";
     $stmt = $db->prepare($query);
     $stmt->execute();
     echo '<h1>Teams</h1>';
     echo '<ul>';
     // display each returned row
     while($result = $stmt->fetch(PDO::FETCH_OBJ)) {
       echo "<li>" . $result->team_name . '</li>';
     }
     echo '</ul>';
?>
```

The book shows how to use your database configuration information by including it all in the same file. We are going to start breaking out our database information. This is known as code modularity.

- Create a new folder named **database** inside your **proj01-teams** folder.
- Create a new file in that folder named **db-conn.php**

Copy and paste the following code into the **db-conn.php** file. Note, that you may need to change some of the database connection information. I have left mine so that it will work with the default root user in XAMPP.

Side note: one of the interesting things about PHP is that you do not need the closing PHP tag (`?>`) if you are writing just PHP code.

```
<?php



function dbConnect()

{



 // set up for using PDO locally

 $user = 'root';

 $pass = '';

 $host = 'localhost';

 $dbname = 'bowling';



 $dsn = "mysql:host=$host;dbname=$dbname";



 try {
       /*
       * create a new database object
       * and return it to the call
       */

       return new PDO($dsn, $user, $pass);

     /*
     * This error trapping is from chaper 7. You can see
     * from the code that $e is an object that has
     * a method called getMessage.
     */
     } catch (PDOException $e) {
       echo "Error: ".$e->getMessage();
       exit;
     }
}
```

At this point the code should run. It will just display a list of the teams in the bowling league.

Part 03 - Display bowlers

Now use this same information to create a page that displays the bowlers.

Folder and file structure

- Create a folder called **proj01-bowlers** inside your **my-code** folder.

- Create an index.php file inside the **proj01-bowlers** folder
- Create a folder named **databases** inside the **proj01-bowlers** folder
- Create a file named **db-conn.php** inside the databases folder.
- Copy and modify the code that displayed the teams so your program displays a list of the bowlers first name and last name in alphabetical order.
- User the following query to extract the bowler data

```
$query = "SELECT * from bowlers ORDER BY last_name";
```

Test the code to see that is list out the names of the bowlers.

Here is a link to the working file

http://charliewallin.com/web182/my-code/prj01-bowlers/

Upload your files to your host

Once you have this working completely in your development environment then upload it to your webhost. Remember, you will need to change the database login credentials for your webhost. Personally, I use the editor from my FTP client and make the changes on the server. This is the only time I make edits on the server otherwise it sets up a recipe for inconsistent files between your development and production environments.

1. Zip up the two folders, **proj01-teams** and **proj01-bowlers**. Name the zipped file **proj01_yourLastName.zip** and submit it in Moodle
2. Add the URLs to your teams and bowlers links in the Moodle Notes section. For instance I would put
    a. http://charliewallin.com/web182/my-code/prj01-bowlers/
    b. http://charliewallin.com/web182/my-code/prj01-teams/
3. Update your **my-code/index.php** file so there are links to the bowlers and teams.