

php

lukabostick

January 2024



L B T M

Research and Development

0.1 Introduction

0.2 Initial Environment Setup and Composer For Windows

```
Ubuntu is already installed.  
Launching Ubuntu...  
root@LB-DallasCollege:~# |
```

1. Open an Administrator Power shell

2. Install WSL

WSL Figure 1

1. Enter code into the console if you do not already have Visual Studio

2. cd into the home and create the file linuxbrew with the following commands shown in Figure 2:
3. cd into /home/linuxbrew/

```
root@LB-DallasCollege:~# code
root@LB-DallasCollege:~# cd /home
root@LB-DallasCollege:/home# cd/linuxbrew/
-bash: cd/linuxbrew/: No such file or directory
root@LB-DallasCollege:/home# mkdir linuxbrew/
root@LB-DallasCollege:/home# cd linuxbrew/
root@LB-DallasCollege:/home/linuxbrew# |
```

1. Install Homebrew **Homebrew** and type the following

0.3 Week 2 create, read, update and delete

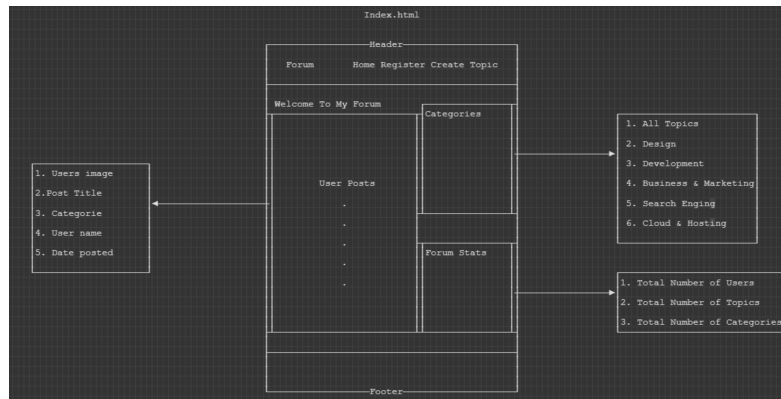
© Mohamed Hassan 2024 (the founder of Web Coding) ¹

0.3.1 Introduction

This section will teach you how to build a Complete Forum with an Admin Panel. This will use PHP, MySQL, Bootstrap, and PDO (PHP Data Objects). After this multi-week project, you will get the hang of how all of the aforementioned web technologies interconnect, and then we will deploy this project to a larval framework.

¹<https://www.udemy.com/course/php-with-mysql-2023-build-complete-forum-with-admin-panel/>

0.3.2 Design

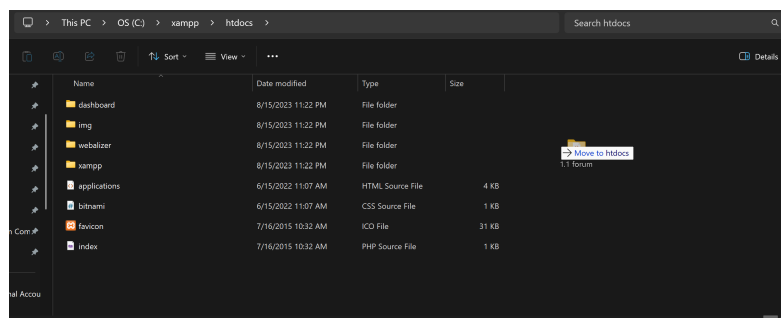


0.3.3 Creating the Config File

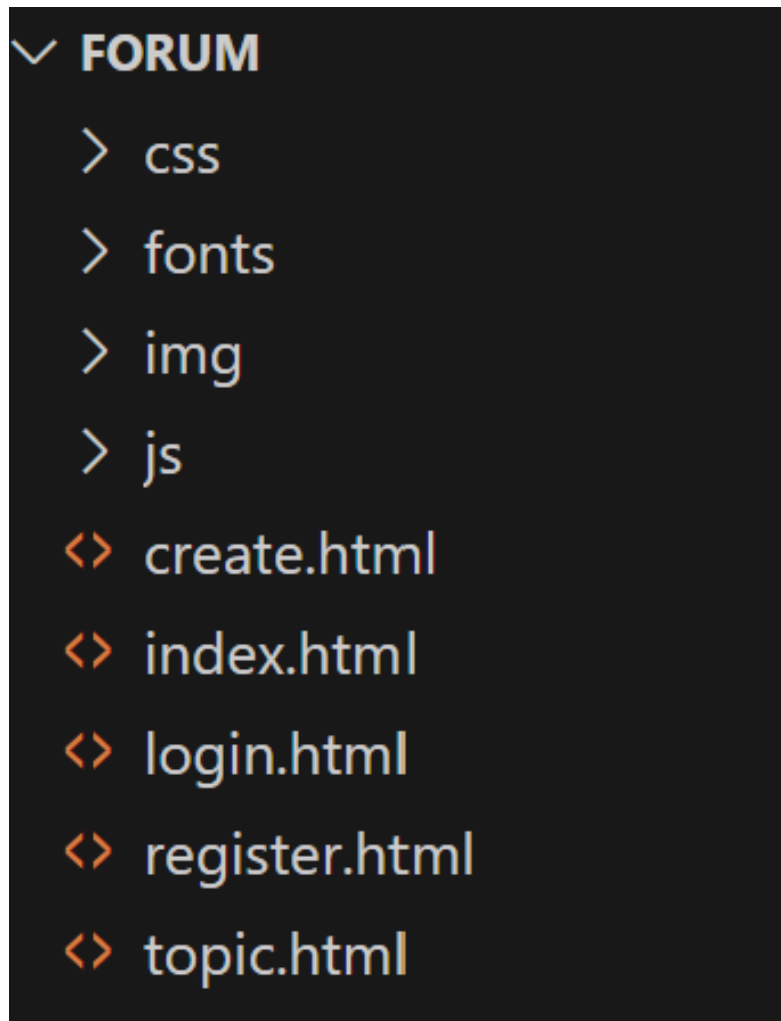
1. Copy the 1.1 forum file into the following File directory

C:/ xampp/ htdocs

then open it in vscode



1. You should see the following



1. Next, we will create our config file. The config file allows us to manage our MySQL queries and houses the code that connects our website to the database. This is one of the most important files when creating a full-stack application. No matter what language or framework (c sharp maintains and automates the config files in Visual Studio), if your project has a backend, it has a configuration management system.
2. Create a folder called config.
3. Create a file in our newly created folder and name it config.php
4. Now, we need four things to create a connection to our database.
 - Host
 - Database name
 - user
 - password

```
1 <?php
2
3 //host
4 //      name      value
5 define("HOST",  "localhost"); // define initializes a constant
6      variable
```

```
7 //dbname
8 define("DBNAME","forum"); // database name
9
10 //user
11 define("USER", "root"); // user credentials
12
13 //password
14 define("PASS", ""); // leaving password blank for the demo
```

- Connect Database with PDO

```
1
2 //          specify what database      concat dynamic host val      end
3 $conn = new PDO(      "mysql:host="      .HOST.      ";dbname="
      ".DBNAME." . "",USER,PASS);
```

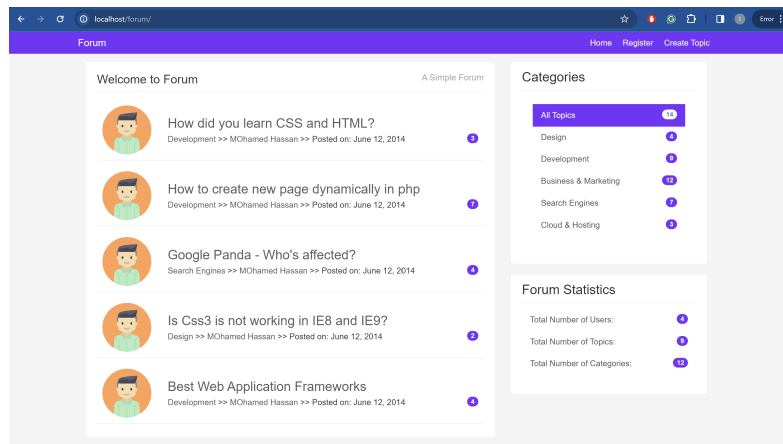
- now every time we connect to our db, grab data, update data, and so on, we will use our PDO connection object
- Lastly, test if it is connected to our database. The config.php file should appear as follows:

```
1 <?php
2
3 //host
4 //      name      value
5 define("HOST", "localhost"); // define initializes a constant
      variable
6
7 //dbname
8 define("DBNAME","forum"); // database name
9
10 //user
11 define("USER", "root"); // user credentials
12
13 //password
14 define("PASS", ""); // leaving password blank for the demo
15
16 // we instantiate a PDO connection object when dealing with php
      databases
17 // allows us to validate and manage the connection more securely
18
19 //          specify what database      concat dynamic host val      end
20 $conn = new PDO(      "mysql:host="      .HOST.      ";dbname="
      ".DBNAME." . "",USER,PASS);
21
22 if($conn == true){
23     echo "db connection is a success";
24 }
25 else
26     echo "error";
```

- Next, in our web browser, navigate to the following address

<http://localhost/forum/>

0.3. WEEK 2 CREATE, READ, UPDATE AND DELETE



- now if we navigate to the address

`http://localhost/forum/config/config.php`

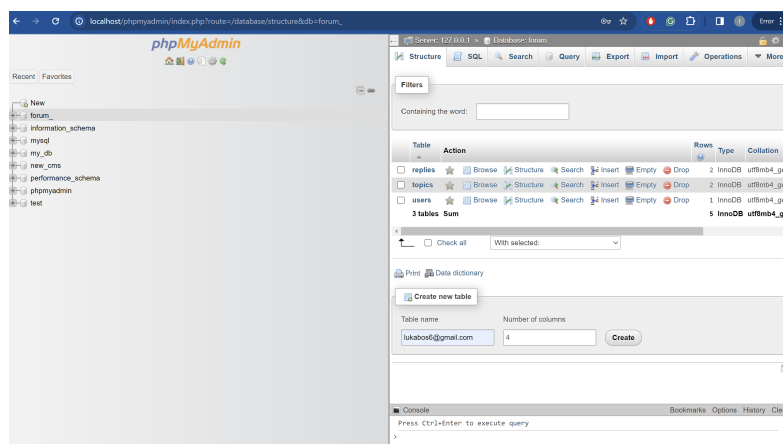
- , we get the following error

`Fatal error: Uncaught PDOException: SQLSTATE[HY000][1049]Unknown database 'forum'...`

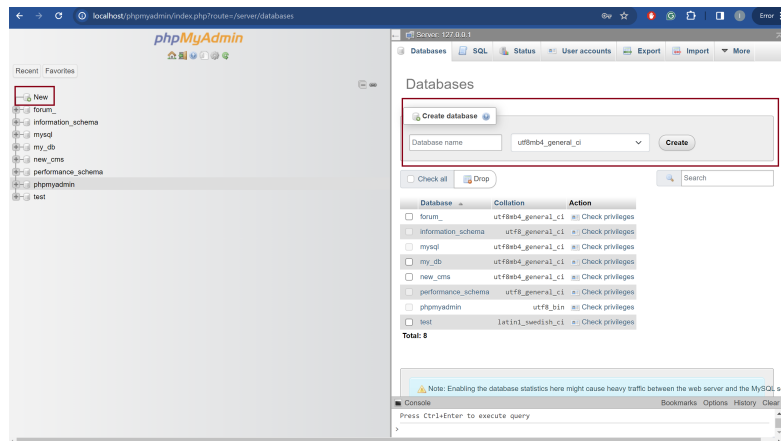
- As you can see, this error tells us we have not created a database named forum, so let's do that now! In the web browser, navigate to

`http://localhost/phpmyadmin/`

- You should see the following



- To create a new database, click on new and enter the name of our database (forum)



- Now, if we navigate back to the URL

http://localhost/forum/config/config.php

- We now see that our if statement outputs that the connection to the server is successful!
- What happens if we remove the logic that detects a successful connection? Let's see if we comment out the selection statement and check the config URL. We see the following:



- Next, we want to validate that the connection to our database is legitimate. To do this, we will use PDO's error-handling feature: specifically (place below obj init)

```
1 $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
```

2

- The line above us activates error mode and will throw an exception if there is something wrong with our database connection
- Next, we will add a try-catch block to receive our exceptions. The config.php file should now look like the following: ³

```
1 <?php
2 try{
3 //host
```

²<https://www.php.net/manual/en/pdo.error-handling.php>

³<https://www.php.net/manual/en/class.pdoexception.php>

```

4 //      name      value
5 define("HOST",  "localhost"); // define initializes a constant
    variable
6
7 //dbname
8 define("DBNAME","forum"); // database name
9
10 //user
11 define("USER", "root"); // user credentials
12
13 //password
14 define("PASS", ""); // leaving password blank for the demo
15
16 // we instantiate a PDO connection object when dealing with php
    databases
17 // allows us to validate and manage the connection more securely
18
19 //      specify what database      concat dynamic host val      end
20 $conn = new PDO(      "mysql:host="      .HOST.      ";dbname="
    ".DBNAME.",USER,PASS);
21 $dbh->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
22 // if($conn == true){
23 //      echo "db connection is a success";
24 // }
25 // else
26 //      echo "error";
27 }catch( PDOException $Exception ) {
28     echo $Exception->getMessage();
29 }

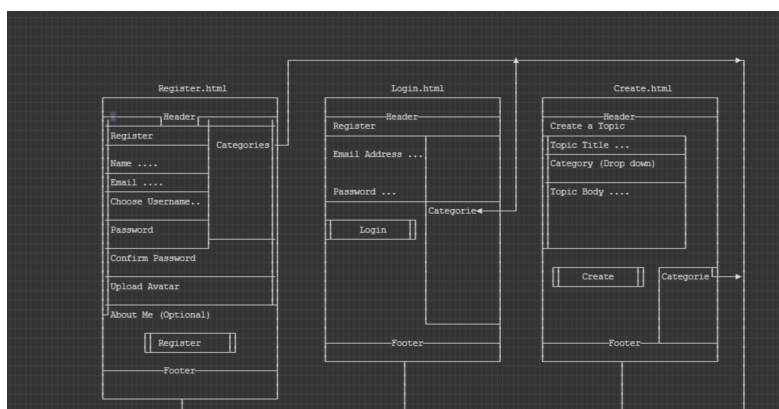
```

- To sum up the above, the try-catch block works by first executing the commands in the try block; then, if any exception is thrown (if we have any errors connecting to our server), the catch block executes, and the error is echoed.
- For example, if we change the DBname to a misspelling of the word forum (forume), we get the following error.

SQLSTATE[HY000][1049]Unknown database 'forume'

0.3.4 Creating Include Files

- Next, we want to create a header and a footer file. You might wonder why we must create separate files for the footer and header.



- If we look at our design doc, we see that we have common items on every form. We create separate files for them to reduce the amount of code reuse.

- Create a new directory named

includes

Now, create a file named

header.php

in the folder.

- Next navigate to

index.php

and cut out the following lines of code

```

1      <!DOCTYPE html>
2      <html lang="en">
3      <head>
4          <meta charset="utf-8">
5          <meta http-equiv="X-UA-Compatible" content="IE=edge">
6          <meta name="viewport" content="width=device-width, initial-scale=1">
7          <title>Welcome To Forum</title>
8
9          <!-- Bootstrap core CSS -->
10         <link href="css/bootstrap.css" rel="stylesheet">
11         <!-- Custom styles for this template -->
12         <link href="css/custom.css" rel="stylesheet">
13     </head>
14
15     <body>
16
17         <div class="navbar navbar-inverse navbar-fixed-top" role="navigation">
18             <div class="container">
19                 <div class="navbar-header">
20                     <button type="button" class="navbar-toggle" data-toggle="collapse"
21 " data-target=".navbar-collapse">
22                         <span class="sr-only">Toggle navigation</span>
23                         <span class="icon-bar"></span>
24                         <span class="icon-bar"></span>
25                         <span class="icon-bar"></span>
26                     </button>
27                     <a class="navbar-brand" href="index.html">Forum</a>
28                 </div>
29                 <div class="collapse navbar-collapse">
30                     <ul class="nav navbar-nav navbar-right">
31                         <li class="active"><a href="index.html">Home</a></li>
32                         <li><a href="register.html">Register</a></li>
33                         <li><a href="create.html">Create Topic</a></li>
34                     </ul>
35                 </div><!--/.nav-collapse -->
36             </div>
37         </div>

```

- Next, navigate back to index.html and enter the following include at the very top.

```

1      <?php require "includes/header.php"?>
2

```

- **Rename**

index.html → *index.php*

- navigate to the includes directory and create a new file named:

footer.php

- Back in

index.php

- Cut the following lines of code and paste it into footer.php

```
1      <div class="col-md-4">
2          <div class="sidebar">
3
4
5              <div class="block">
6                  <h3>Categories</h3>
7                  <div class="list-group block ">
8                      <a href="#" class="list-group-item active">All Topics <span
9                          class="badge pull-right">14</span></a>
10                     <a href="#" class="list-group-item">Design<span class="color
11                         badge pull-right">4</span></a>
12                     <a href="#" class="list-group-item">Development<span class="
13                         color badge pull-right">9</span></a>
14                     <a href="#" class="list-group-item">Business & Marketing <span
15                         class="color badge pull-right">12</span></a>
16                     <a href="#" class="list-group-item">Search Engines<span class="
17                         color badge pull-right">7</span></a>
18                     <a href="#" class="list-group-item">Cloud & Hosting <span class
19                         ="color badge pull-right">3</span></a>
20                 </div>
21             </div>
22
23             <div class="block" style="margin-top: 20px;">
24                 <h3 class="margin-top: 40px">Forum Statistics</h3>
25                 <div class="list-group">
26                     <a href="#" class="list-group-item">Total Number of Users:<
27                         span class="color badge pull-right">4</span></a>
28                     <a href="#" class="list-group-item">Total Number of Topics:<
29                         span class="color badge pull-right">9</span></a>
30                     <a href="#" class="list-group-item">Total Number of
31                         Categories: <span class="color badge pull-right">12</span></a>
32                 </div>
33             </div>
34         </div>
35     </div>
36 </div><!-- /.container -->
37
38 <!-- Bootstrap core JavaScript
39 ==/===== -->
40 <!-- Placed at the end of the document so the pages load faster -->
41 <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.1/jquery
42     .min.js"></script>
43 <script src="js/bootstrap.js"></script>
44 </body>
45 </html>
```

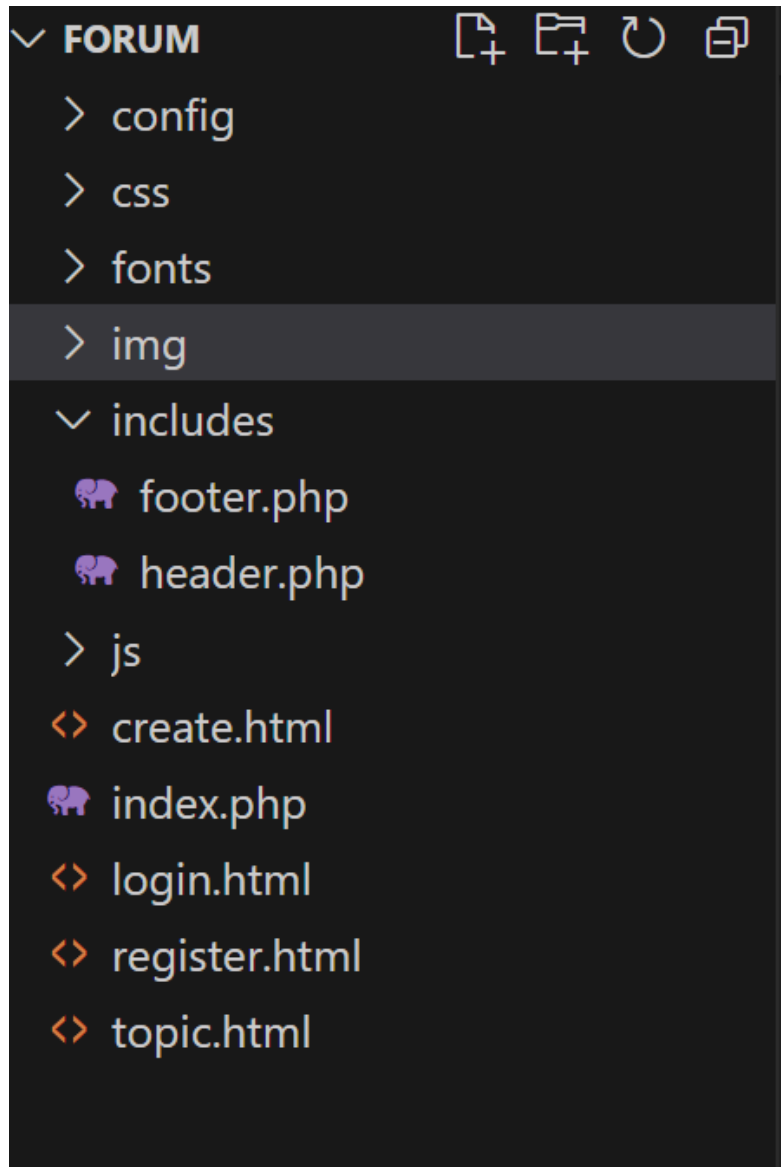
- lastly at the very bottom of

index.php

- enter the following

```
1      <?php require "includes/footer.php"; ?>
2
```

- The workspace directory should look as follows, (check to make sure that the pages are rendering correctly. Note that nothing should be different)



0.4 Creating Authentication System

0.4.1 Registering Users

0.4.2 Logging Users in

0.4.3 Starting Sessions and Validating

***0.4.4 Creating Topics**

0.4.5 Showing Topics

0.4.6 Displaying a Single Topic

0.4.7 Displaying Replies for every Topic

0.4.8 Deleting Topics

0.4.9 Updating Topics

0.4.10 Creating Replies

0.4.11 Deleting Replies

0.4.12 Updating Replies

0.4.13 Counting Topics

0.4.14 Showing Categories

0.4.15 Showing Topics Inside every Category

0.4.16 Creating 404 Page

0.4.17 Showing Statistics for our Forum

0.4.18 Updating Users Data

0.4.19 Fixing Bugs

0.4.20 Creating Public Profiles

0.4.21 Design for Admin Panel

0.4.22 Creating Login Page

0.4.23 Creating Sessions

0.4.24 Finishing Login Page and Validating

0.4.25 Showing Admins

0.4.26 Creating Admins

0.4.27 Displaying Stats

0.4.28 Showing Categories

0.4.29 Creating Categories

0.4.30 Updating Categories

0.4.31 Deleting Categories

0.4.32 Showing Topics

0.4.33 Deleting Topics

0.4.34 Showing Replies

0.4.35 Deleting Replies