

Tutorial 11

Editor – Brackets

Goals

Creating a blog with PHP and MySQL.

- Set up and configuration of Xampp
- Learning Ajax

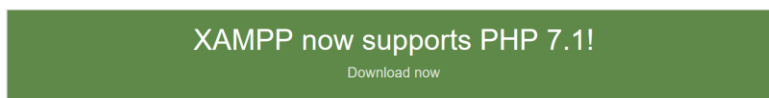
Things to note: Each week Xampp will need to be installed. Xampp is Windows software, similar software is available for Mac, called Mamp.

Installing and configuring Xampp

Go to the website: <https://www.apachefriends.org/index.html>



Download the Xampp with PHP 7.1, click on the following link



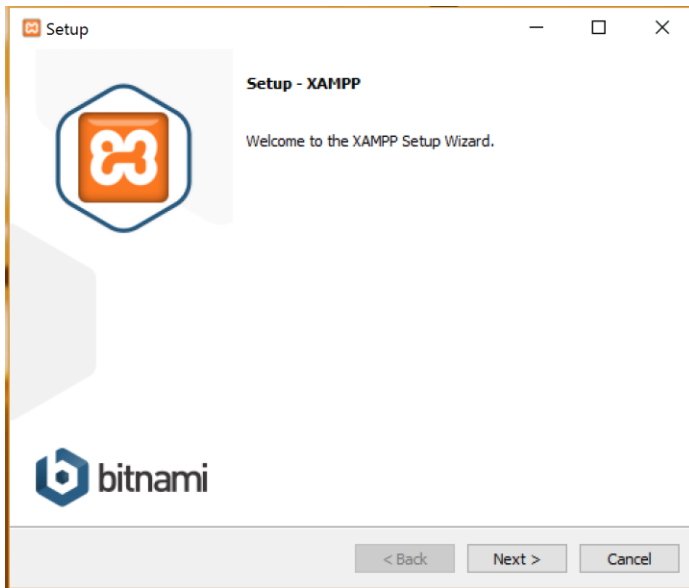
Then click on



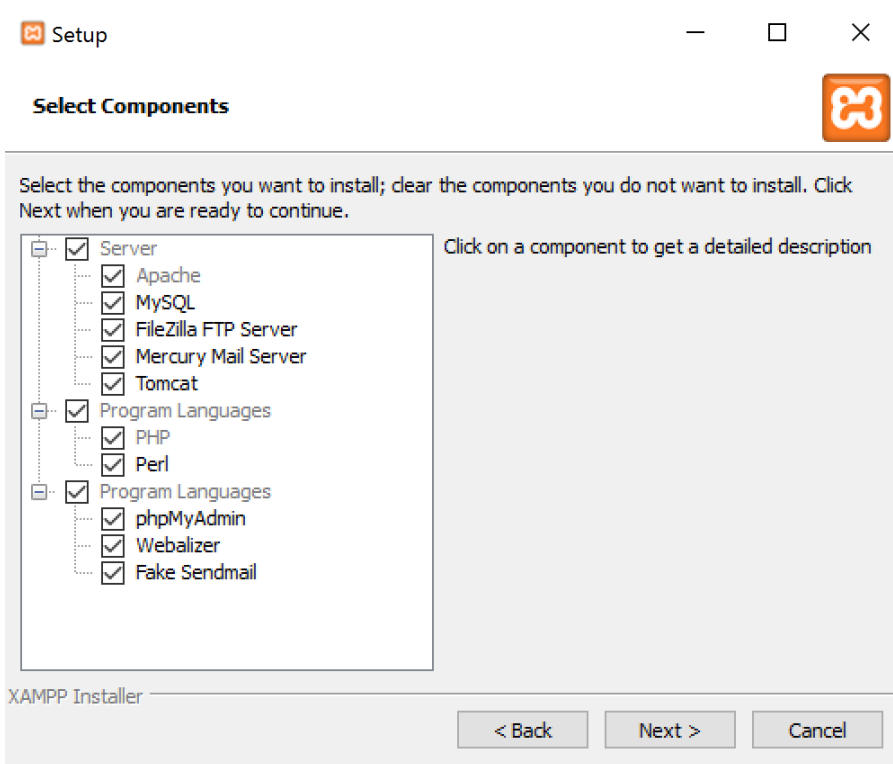
Download to the desktop, Once there, double click on the file and following the prompts



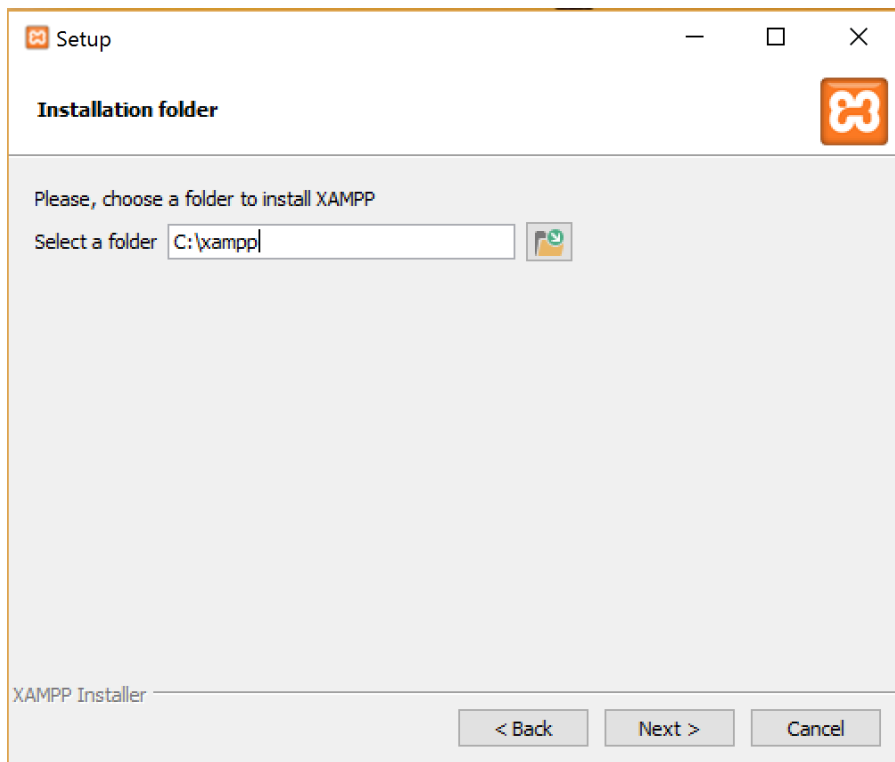
Click Next



Click Next

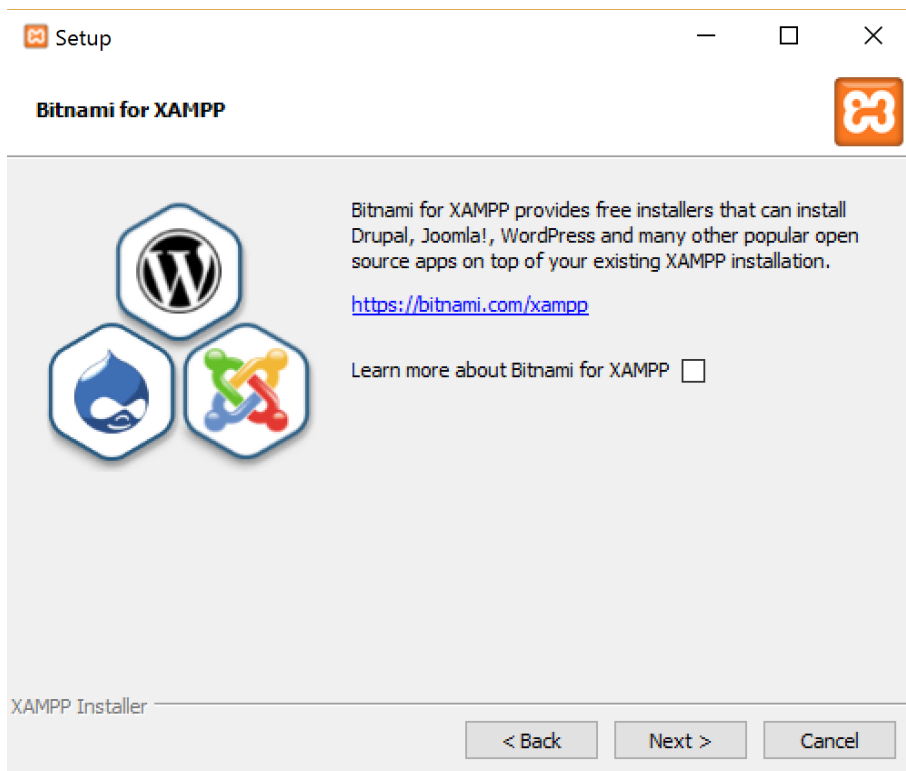


Click Next

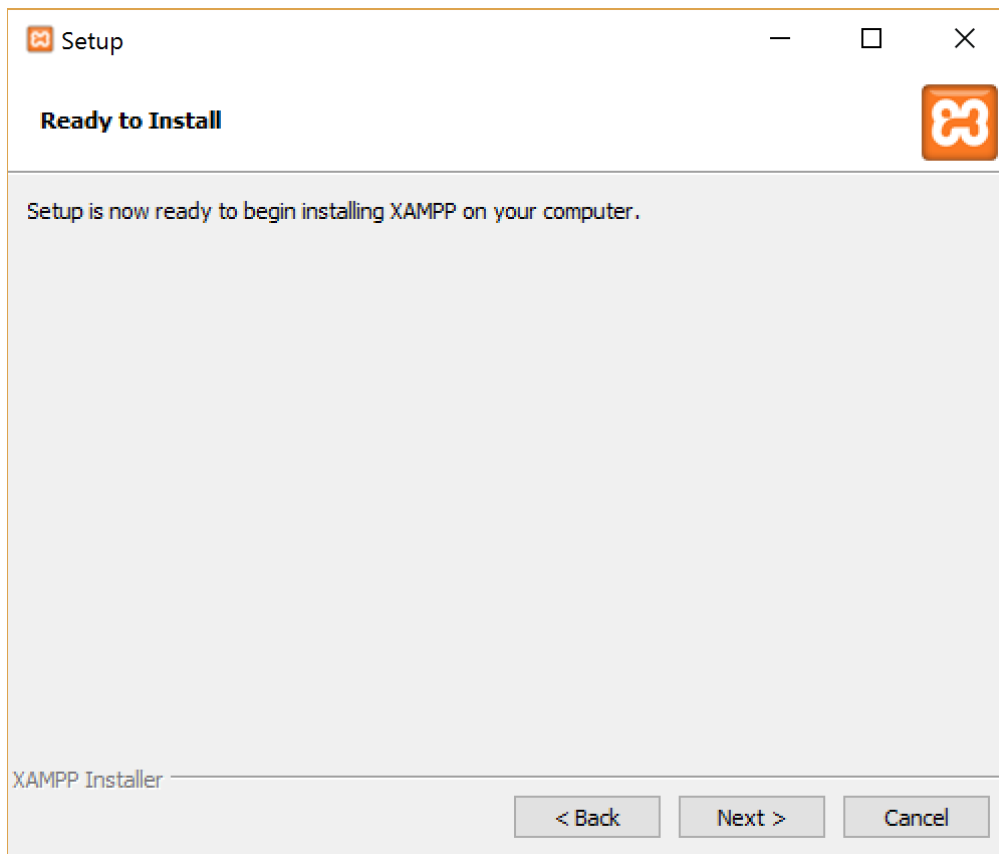


NB: This is the installation folder, all of your web pages will belong in c:\xampp\htdocs

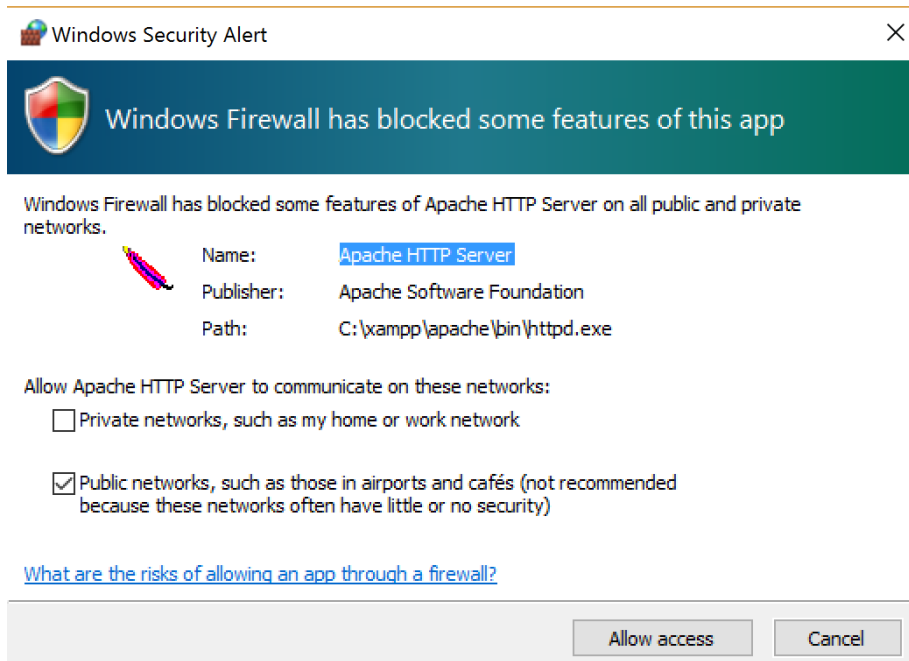
Untick Bitnami and click Next



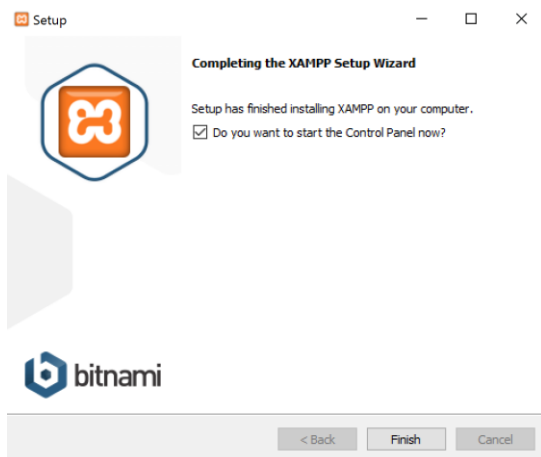
Click next to install



Allow any pop ups such as Apache or MySQL

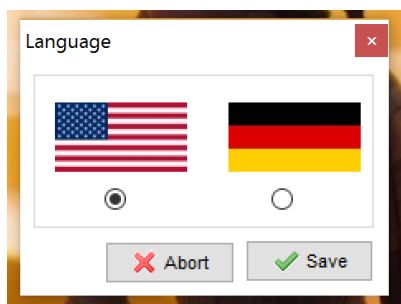


Then click finish



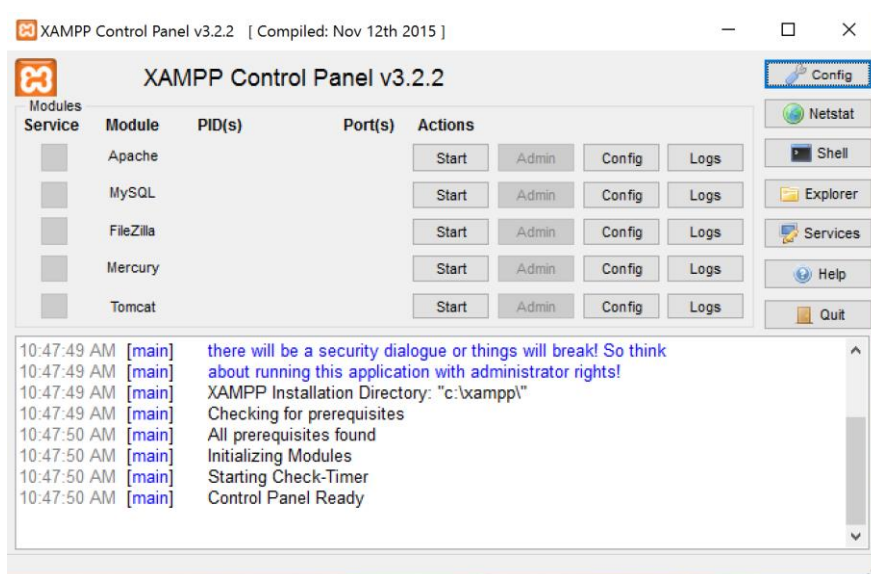
Now we configure Xampp.

Assuming you left the "Do you want to start the Control Panel" ticked, you should see the following on your screen.



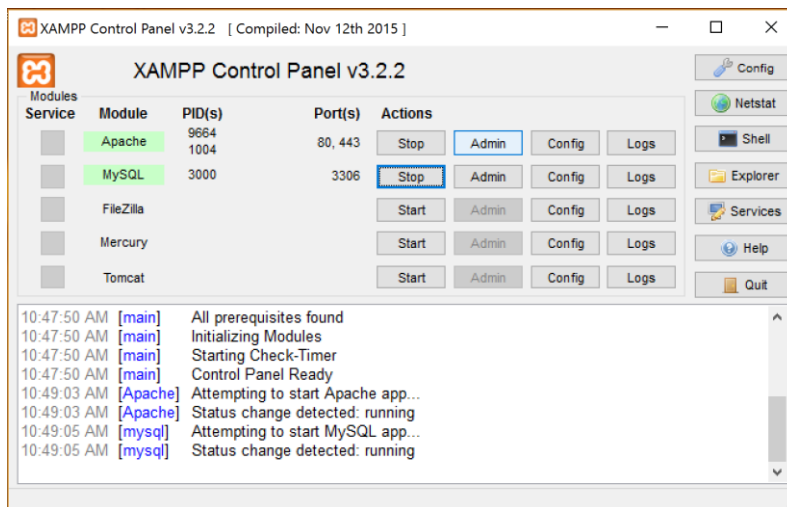
Click on Save

The following should appear:



From here we need to click Start on Apache and MySQL.

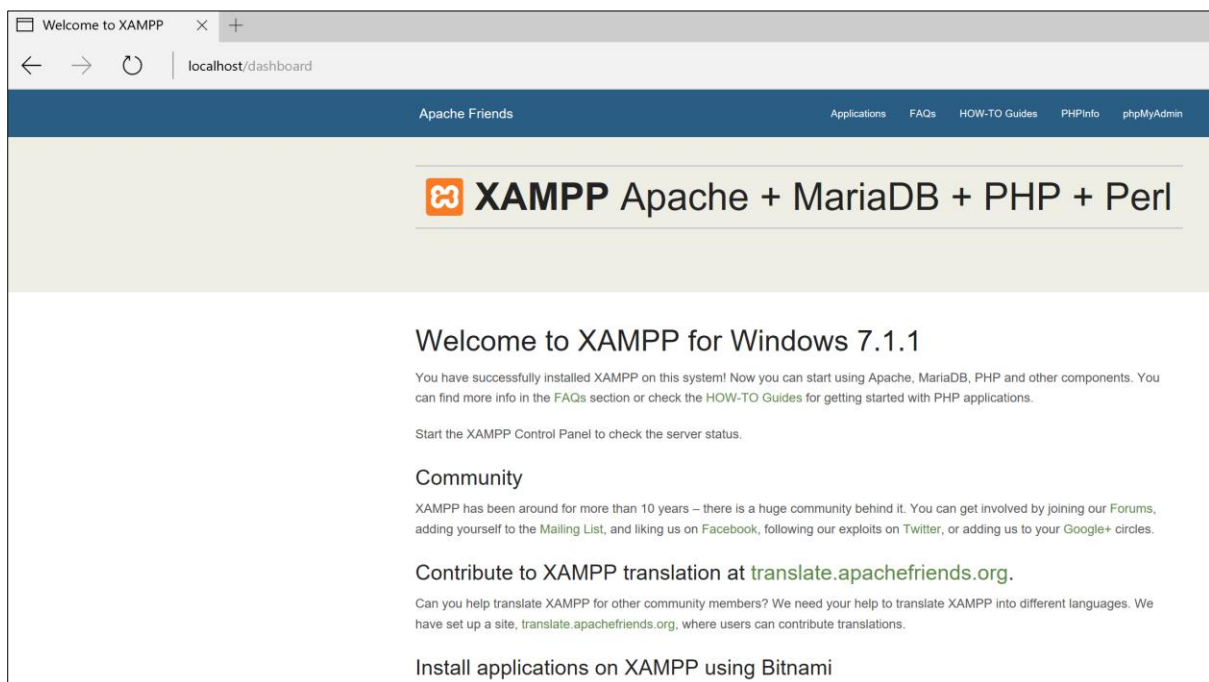
You should see the following.



We now have the Apache webserver up and running with MySQL.

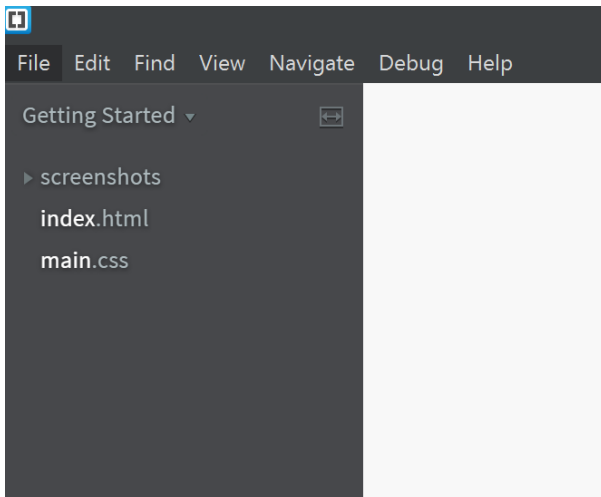
To check that it is working load up a browser and type localhost into the address bar.

You should see the following:

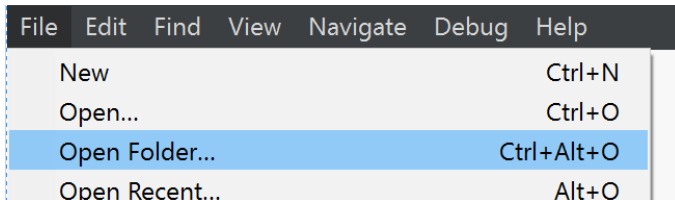


Now, this is all configuration information from Xampp, what we should do is clean out the folder htdocs, to put our own files. To do this, we will use brackets to clean out the folder.

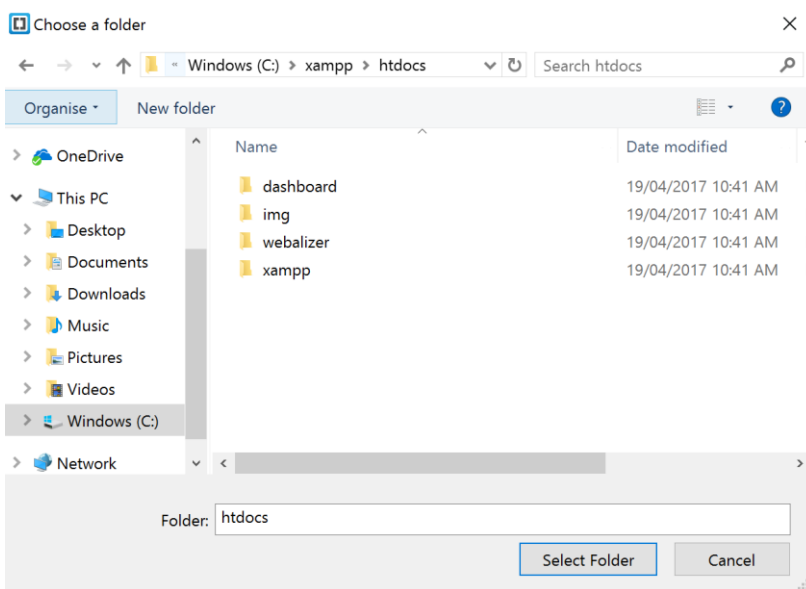
Load up brackets



From here, go to open folder

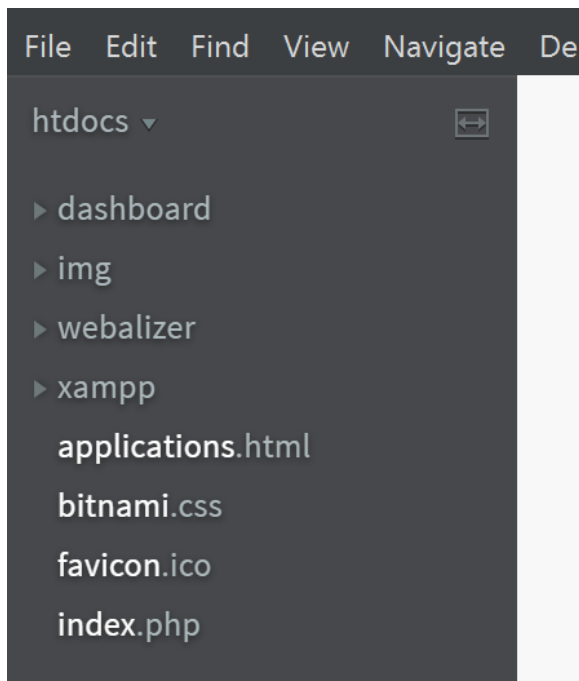


Navigate to c:\xampp\htdocs

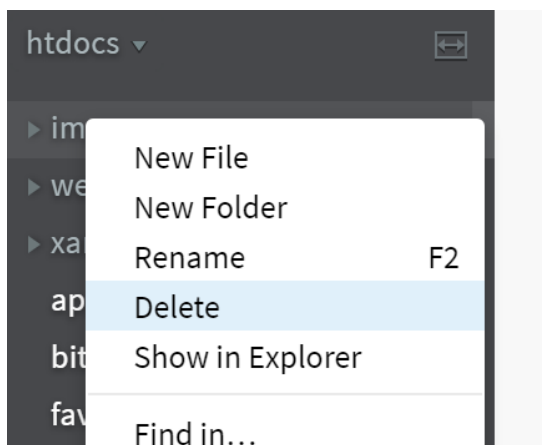


Click Select

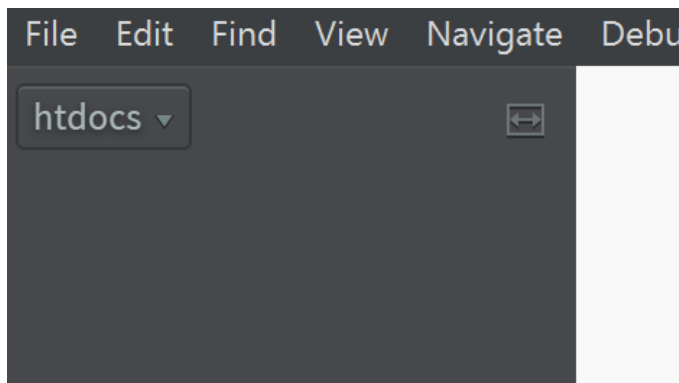
This should give you the following set of files in the navigation pane of brackets:



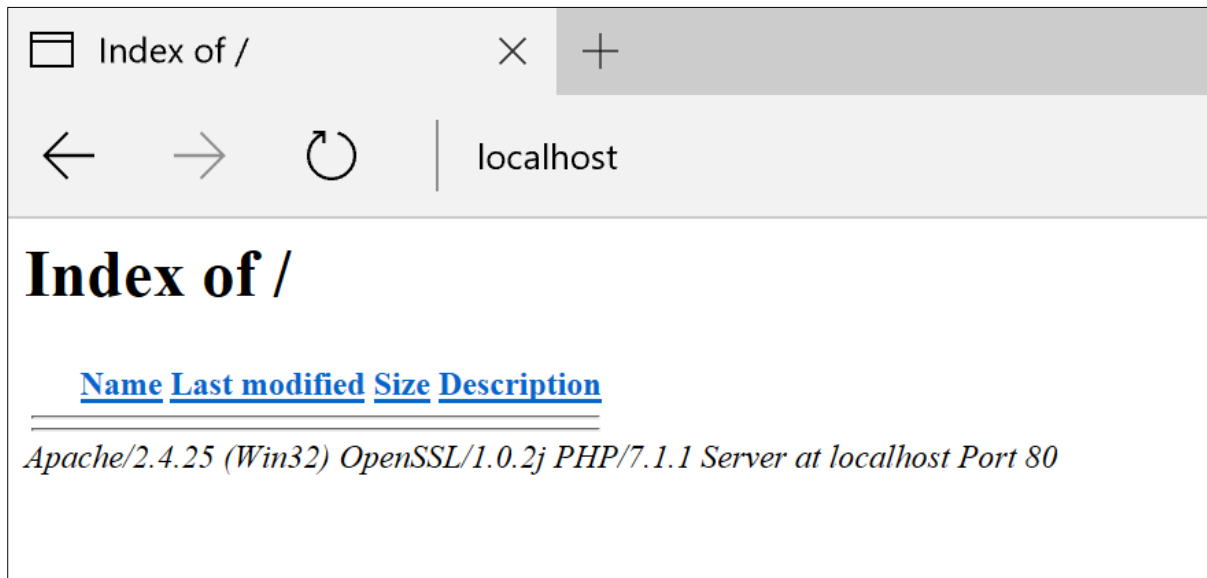
Highlight and right click on the navigation elements and delete everything in the folder



Once you have emptied the folder it should look like



Now, we check to ensure that the system is clean, go to your browser and navigate to <http://localhost/> You should see the following



Now we have a clean server to start creating with.

Creating your php Website

From here, right click and create a new file called index.php and put in the following content:

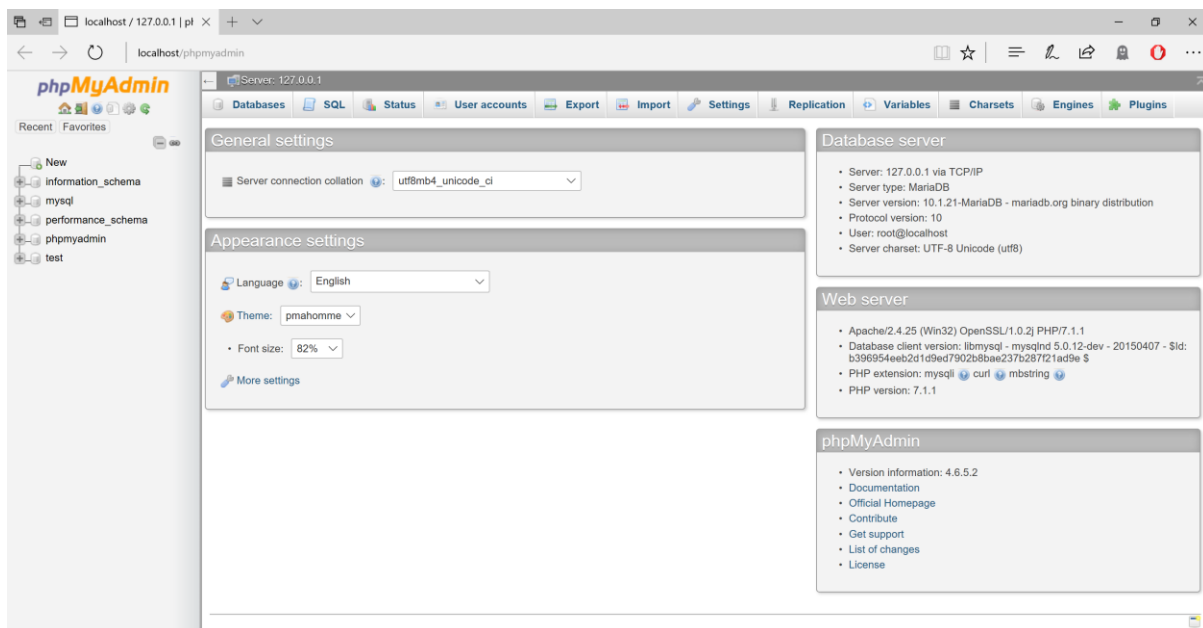


```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title></title>
5     <style type="text/css">
6   </style>
7   </head>
8   <body>
9   </body>
10 </html>
```

In this lesson we will be communicating with a database through our php files.

To make things easy we will be introducing AJAX as the main method of entering data into the system.

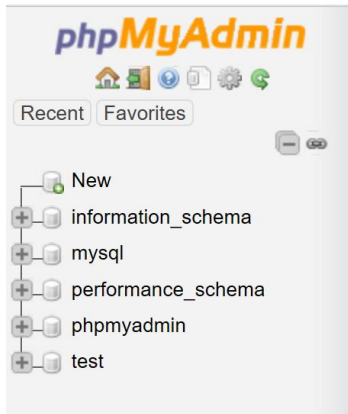
To get access to the web interface, open a browser and go to the address localhost/phpmyadmin and you should see the following:



This is the normal screen in which you can manipulate all of the databases you require for most web projects.

We will run through the interface, to increase your ability to navigate and understand the way it works and how it will work for your code.

- The following pane, contains a list of all currently existing databases within the MySQL system.



-
- Each of the databases can be viewed in more by expanding via the plus sign. Also note, that there is a home (house) icon under the title, this returns you to this page.

- This part of the interface:



- Allows you to access specific elements of MySQL. There are only a few of them that we will require for this lesson.
 - Database – This is where you can create and delete databases
 - User Accounts – This is where you will create/delete user accounts that allow access to the database. This is a security element that is very important, if your server has multiple databases and each database is access via one user account, if there is a security breach then only one database can be compromised. So always create a specific user for a database.
 - Export – This is where you can make a backup of the database. When saving your work/submitting your assignment. You will need to export your database to a .sql file.
 - Import – This is where you bring a .sql file back into the MySQL system. So, if you have worked on a database, exported the work and saved it to USB, if the computer removes all the server details (All university computers are cleaned, so your files will be removed each week) it means that when you return to the computer, you will be able to import all the database information (Tables and data). Thereby removing the potential requirement to recreate and re-enter.

Now that we have a basic overview of the interface, let's create the database structure, account, and tables within the database

Click on the database link, you should see the following:

Databases

Create database

Database name Collation Create

Database	Collation	Action
<input type="checkbox"/> information_schema	utf8_general_ci	Check privileges
<input type="checkbox"/> mysql	latin1_swedish_ci	Check privileges
<input type="checkbox"/> performance_schema	utf8_general_ci	Check privileges
<input type="checkbox"/> phpmyadmin	utf8_bin	Check privileges
<input type="checkbox"/> test	latin1_swedish_ci	Check privileges
Total: 5	latin1_swedish_ci	

Check all With selected: [Drop](#)

Note: Enabling the database statistics here might cause heavy traffic between the web server

- [Enable statistics](#)

From here enter in the database name and click create. NB When it comes to your assignments, try using your student number as your database name. In this case, we will name the database lesson11.

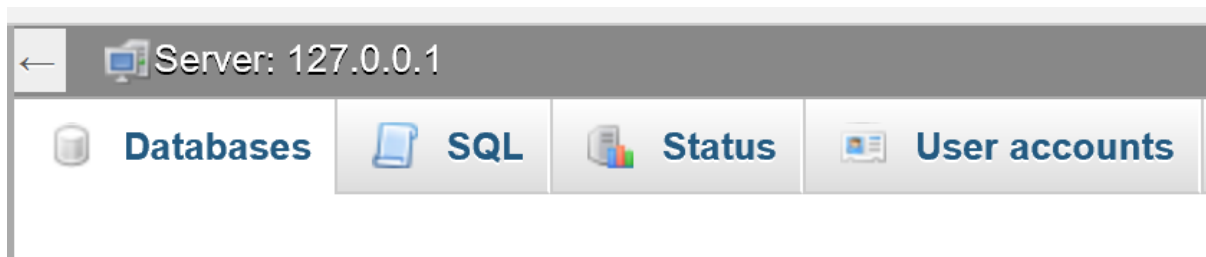
You should have a structure looking like this once done. Once the database was created, you might be automatically taken into the database, click on the home icon and then database link to see the following.

New

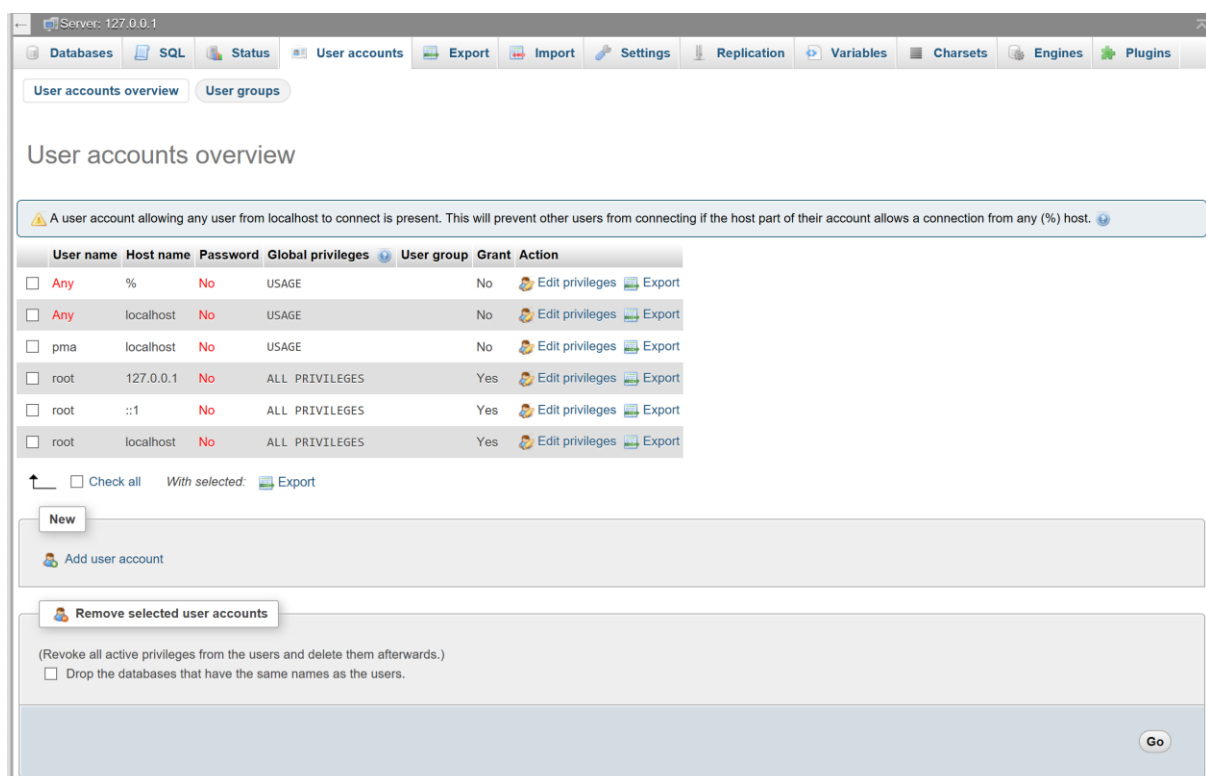
- information_schema
- lesson9
- lesson10
- lesson11
- mysql
- performance_schema
- phpmyadmin
- test

As you can see, the database is listed in the left navigation bar as well as in the list of databases.

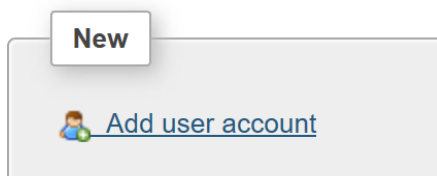
Next click on the user accounts link.



You should see the following:



This is a list of all current user accounts within the MySQL system. To ensure that we work securely with our database, we need to create a new account. As such, click on Add User Account:



Here you will be prompted for details to enter. Use the following:

- Username: advweb
- Host: localhost
- Password: password

Also, ensure that you click the check all part. Once the details are filled out scroll down and click go. This will create the user and apply its permissions.

It should look like the following:

Add user account

Login Information

User name:

Host name:

Password:

Re-type:

Authentication Plugin:

Generate password:

Database for user account

Create database with same name and grant all privileges.

Grant all privileges on wildcard name (username_%).

Global privileges **Check all**

Note: MySQL privilege names are expressed in English.

<input checked="" type="checkbox"/> Data	<input checked="" type="checkbox"/> Structure	<input checked="" type="checkbox"/> Administration	Resource limits
<input checked="" type="checkbox"/> SELECT <input checked="" type="checkbox"/> INSERT <input checked="" type="checkbox"/> UPDATE <input checked="" type="checkbox"/> DELETE <input checked="" type="checkbox"/> FILE	<input checked="" type="checkbox"/> CREATE <input checked="" type="checkbox"/> ALTER <input checked="" type="checkbox"/> INDEX <input checked="" type="checkbox"/> DROP <input checked="" type="checkbox"/> CREATE TEMPORARY TABLES	<input checked="" type="checkbox"/> GRANT <input checked="" type="checkbox"/> SUPER <input checked="" type="checkbox"/> PROCESS <input checked="" type="checkbox"/> RELOAD <input checked="" type="checkbox"/> SHUTDOWN	<i>Note: Setting these options to 0 (zero) removes the limit.</i> MAX QUERIES PER HOUR <input type="text" value="0"/> MAX UPDATES PER HOUR <input type="text" value="0"/>

Once you have clicked go, you should be redirected to a page that indicates a successful user addition. If there is an error message that appears regards `c:\xampp\mysql\lib\plugins` and the message states `dir not found`. You will have to create the folder structure. Open file explorer and navigate to `c:\xampp`. Once there, go inside the `mysql` folder and create a new folder called `lib`. Once this has been done, go inside the newly created `lib` folder and create the folder called `plugins`.

After the creation of these folders, go back to the web browser and re-create the user. It should work successfully this time.

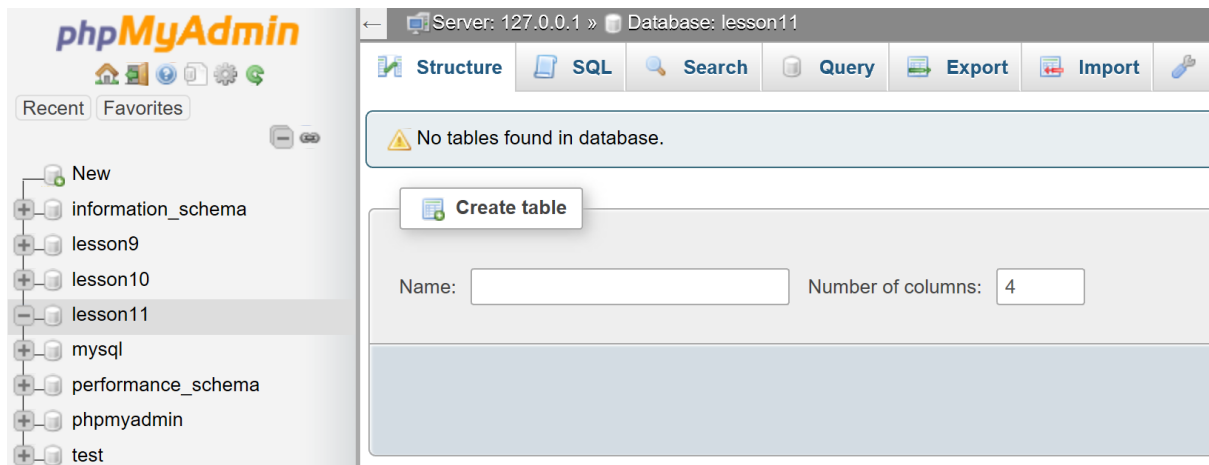
You should be directed to the following page:

The screenshot shows the MySQL Workbench interface for editing user privileges. At the top, a green banner states "You have added a new user." Below this, a SQL statement is displayed: `CREATE USER 'advweb'@'localhost' IDENTIFIED VIA mysql_native_password USING '***';GRANT ALL PRIVILEGES ON *.* TO 'advweb'@'localhost' REQUIRE NONE WITH GRANT OPTION MAX_QUERIES_PER_HOUR 0 MAX_CONNECTIONS_PER_HOUR 0 MAX_UPDATES_PER_HOUR 0 MAX_USER_CONNECTIONS 0;`. The page title is "Edit privileges: User account 'advweb'@'localhost'". A "Global privileges" section is active, with a "Check all" checkbox. The privileges are organized into four columns: Data, Structure, Administration, and Resource limits. The "Data" column includes SELECT, INSERT, UPDATE, DELETE, and FILE. The "Structure" column includes CREATE, ALTER, INDEX, DROP, CREATE TEMPORARY TABLES, SHOW VIEW, CREATE ROUTINE, ALTER ROUTINE, EXECUTE, CREATE VIEW, EVENT, and TRIGGER. The "Administration" column includes GRANT, SUPER, PROCESS, RELOAD, SHUTDOWN, SHOW DATABASES, LOCK TABLES, REFERENCES, REPLICATION CLIENT, REPLICATION SLAVE, and CREATE USER. The "Resource limits" section includes input fields for MAX_QUERIES_PER_HOUR, MAX_UPDATES_PER_HOUR, MAX_CONNECTIONS_PER_HOUR, and MAX_USER_CONNECTIONS, all set to 0. An "SSL" section is also present with radio buttons for REQUIRE NONE (selected), REQUIRE SSL, REQUIRE X509, and SPECIFIED, along with input fields for REQUIRE_CIPHER, REQUIRE_ISSUER, and REQUIRE_SUBJECT.

Now that the user and database have been created, we need to add some tables to the database, to do this, we need to access the database. On the left hand navigation panel, click on the database called lesson11.

The screenshot shows the left-hand navigation tree in MySQL Workbench. The tree is expanded to show the following items: "New", "information_schema", "lesson9", "lesson10", "lesson11" (which is highlighted in grey), "mysql", "performance_schema", "phpmyadmin", and "test". Each item has a small icon to its left, and "lesson11" has a minus sign icon, indicating it is currently selected.

This will bring you to the following screen:



Now that we are inside the database we need to create a table to hold information.

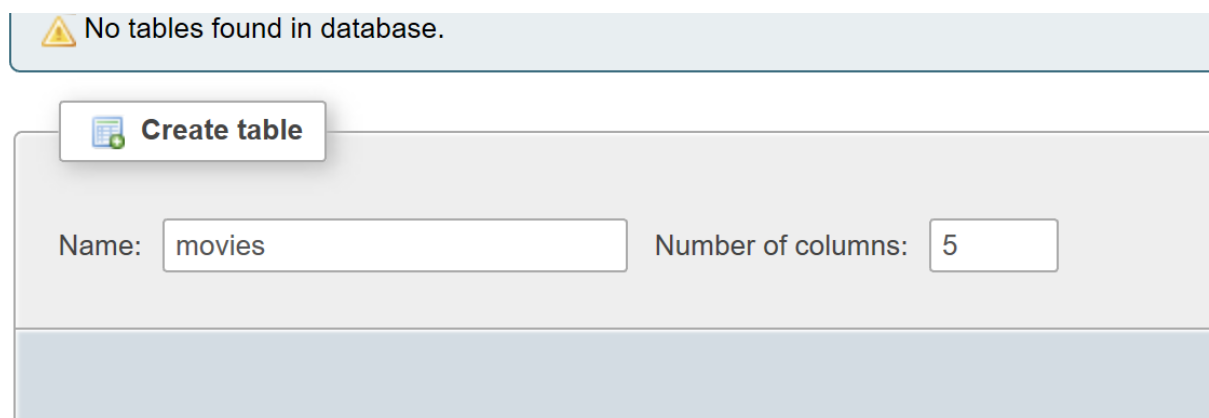
In this lesson, we will be focusing on a simple database transactions using AJAX, as such we will build a system to hold movie information.

We will store the following information:

- Id
- Name of Movie
- Synopsis
- Year
- Running time

We could always add a lot more, but this way we can focus on the important aspect of getting the system using ajax.

To start with we will build the table movies, and then add the fields id, name, synopsis, year and length.



The table will have

- Id – primary key, auto increment
- Name – varchar (200)
- Synopsis - text
- Year – varchar(4)
- Running_time – varchar(6)

Table name: Add column(s)

Name	Type	Length/Values	Default	Collation	Attributes	Null	Index	A_I	Comments
<input type="text" value="id"/>	INT	<input type="text"/>	None	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	PRIMARY	<input checked="" type="checkbox"/>	<input type="text"/>
<small>Pick from Central Columns</small>									
<input type="text" value="name"/>	VARCHAR	200	None	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	---	<input type="checkbox"/>	<input type="text"/>
<small>Pick from Central Columns</small>									
<input type="text" value="synopsis"/>	TEXT	<input type="text"/>	None	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	---	<input type="checkbox"/>	<input type="text"/>
<small>Pick from Central Columns</small>									
<input type="text" value="year"/>	VARCHAR	4	None	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	---	<input type="checkbox"/>	<input type="text"/>
<small>Pick from Central Columns</small>									
<input type="text" value="running_time"/>	VARCHAR	6	None	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	---	<input type="checkbox"/>	<input type="text"/>
<small>Pick from Central Columns</small>									

Table comments:
Collation:
Storage Engine: InnoDB

PARTITION definition:

Partition by: ()

Partitions:

Once entered, click on save. Then we will go to brackets to start coding.

Brackets PHP – MySQL

From here we are going to add a simple form to the index page, this form will be used to gather information to insert into the database.

To start with, we need to code up the database connection code and form. As before, we've used tinymce to enhance the textarea code, we will implement it again, providing a more robust data entry screen.

We need to get information into the database, so create the following index and functions file

Index.php

```
1 <?php
2     include_once('functions/functions.php');
3     $dbConnect = dbLink();
4     if ($dbConnect)
5         echo '<!-- database connection established -->';
6 ?>
7 <!DOCTYPE html>
8 <html>
9     <head>
10        <title></title>
11        <style type="text/css">
12            body{background-color: aliceblue;}
13            input{padding: 10px 10px 10px 10px; margin: 10px 10px 10px 10px; border-radius: 15px;}
14            #wrapper{width: 80%; margin: 0 auto;}
15        </style>
16        <script src="https://cloud.tinymce.com/stable/tinymce.min.js"></script>
17        <script>tinymce.init({ selector:'textarea' });</script>
18    </head>
19    <body>
20        <div id="wrapper">
21            <h2>Add a Movie</h2>
22            <form method="post" action="addMovie.php">
23                <input type="text" name="mov_name" placeholder="Enter Movie Name"><br><br>
24                <h3>Enter Synopsis</h3>
25                <textarea name="mov_synopsis"></textarea>
26                <br><br>
27                <input type="text" name="mov_year" placeholder="Year Published">
28                <input type="text" name="mov_rt" placeholder="Enter Running time">
29                <input type="submit" value="Enter Details">
30            </form>
31            <div id="movieDetails">Movie listing here...</div>
32        </div>
33    </body>
34 </html>
```

The main function at the moment is the dbLink function. As always, to speed up coding, grab the function from the previous tutorials and make the modifications needed.

Functions.php

```
1 <?php
2
3 function dbLink()
4 {
5     $db_user = "advweb";
6     $db_pass = "password";
7     $db_host = "localhost";
8     $db = "lesson11";
9     try{
10        $db = new PDO("mysql:host=$db_host;dbname=$db",$db_user,$db_pass);
11    } catch (Exception $e){
12        echo 'Unable to access database';
13        exit;
14    }
15    error_reporting(0);
16    return $db;
17 }
18
19
20 ?>
```

Result

localhost

Add a Movie

Enter Movie Name

Enter Synopsis

File Edit View Format

← → Formats **B** *I* ☰ ☷ ☹ ☺ ☻ ☼

Year Published Enter Running time Enter Details

Movie listing here...

Now we create the files to add data to the database. Though, we start with ensuring all data is traversing from index to addMovie. To do this we will use the showMem function we have written before.

addMovie.php

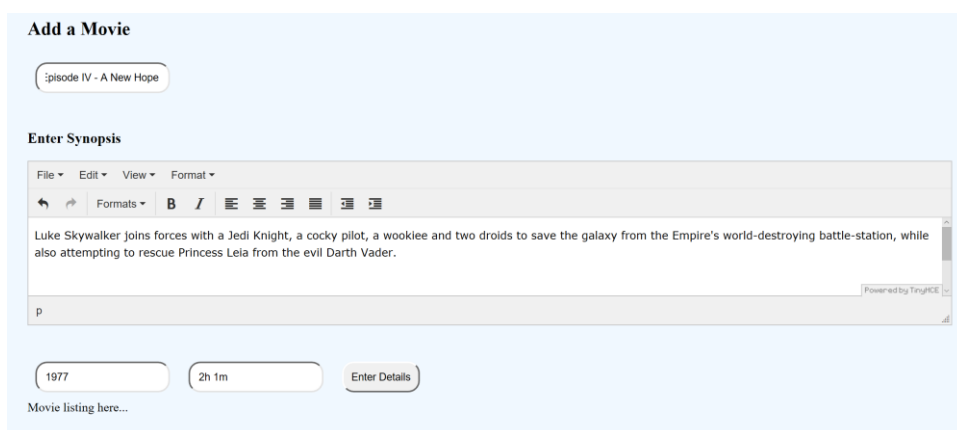
```
1  <?php
2      include_once('functions/functions.php');
3      $dbConnect = dbLink();
4      showMem();
5  ?>
```

functions.php

showMem function

```
19  function showMem()
20  ▼ {
21      echo '<h3>Get Memory</h3>';
22      print_r($_GET);
23      echo '<h3>Post Memory</h3>';
24      print_r($_POST);
25      echo '<h3>Session Memory</h3>';
26      print_r($_SESSION);
27  }
```

Result – index.php



The screenshot shows a web form titled "Add a Movie" on a light blue background. At the top, there is a text input field containing "Episode IV - A New Hope". Below this is a section titled "Enter Synopsis" which contains a rich text editor. The rich text editor has a menu bar with "File", "Edit", "View", and "Format" options. The text area contains the synopsis: "Luke Skywalker joins forces with a Jedi Knight, a cocky pilot, a wookiee and two droids to save the galaxy from the Empire's world-destroying battle-station, while also attempting to rescue Princess Leia from the evil Darth Vader." Below the rich text editor, there are two more input fields: one for the year "1977" and one for the runtime "2h 1m". To the right of these fields is a button labeled "Enter Details". At the bottom of the form, there is a placeholder text "Movie listing here..."

Result – addMovie.php

```
Get Memory
Array ( )

Post Memory
Array ( [mov_name] => Star Wars: Episode IV - A New Hope [mov_synopsis] =>
Luke Skywalker joins forces with a Jedi Knight, a cocky pilot, a wookiee and two droids to save the galaxy from the Empire's world-destroying battle-station, while also attempting to rescue Princess Leia from the evil Darth Vader.
[mov_year] => 1977 [mov_rt] => 2h 1m )

Session Memory
```

Now that we have confirmation that all of the data is coming across, we will add this to the database.

addMovie.php

```
1 <?php
2     include_once( 'functions/functions.php' );
3     $dbConnect = dbLink();
4     showMem();
5     $name = $_POST[ 'mov_name' ];
6     $synopsis = $_POST[ 'mov_synopsis' ];
7     $year = $_POST[ 'mov_year' ];
8     $rt = $_POST[ 'mov_rt' ];
9     $resultInsert = insertMovie($dbConnect,$name,$synopsis,$year,$rt);
10    if($resultInsert)
11        echo '<br>Add successful';
12    ?>
```

functions.php

insertMovie function

```
29 function insertMovie($dbConnect,$name,$synopsis,$year,$rt)
30 {
31     $q = "INSERT into movies (id,name,synopsis,year,running_time) VALUES(NULL,:name,:syn,:year,:rt)";
32     $query = $dbConnect->prepare($q);
33     $query->bindParam(":name",$name);
34     $query->bindParam(":syn",$synopsis);
35     $query->bindParam(":year",$year);
36     $query->bindParam(":rt",$rt);
37     $result = $query->execute();
38     return $result;
39 } // eo of function
```

Result – addMovie.php

```
Get Memory
Array ( )

Post Memory
Array ( [mov_name] => Star Wars: Episode IV - A New Hope [mov_synopsis] =>
Luke Skywalker joins forces with a Jedi Knight, a cocky pilot, a wookiee and two droids to save the galaxy from the Empire's world-destroying battle-station, while also attempting to rescue Princess Leia from the evil Darth Vader.
[mov_year] => 1977 [mov_rt] => 2h 1m )

Session Memory

Add successful
```

Result – phpmyadmin browse of movies table

The screenshot shows the phpMyAdmin interface for the 'movies' table. At the top, it displays 'Server: 127.0.0.1 » Database: lesson11 » Table: movies'. Below this is a navigation bar with buttons for Browse, Structure, SQL, Search, Insert, Export, Import, Privileges, Operations, and Tracking. A green status bar indicates 'Showing rows 0 - 0 (1 total, Query took 0.0005 seconds.)'. The SQL query editor shows 'SELECT * FROM `movies`'. Below the query editor are controls for 'Show all', 'Number of rows: 25', and 'Filter rows: Search this table'. A '+ Options' section is visible. The table data is shown with columns: id, name, synopsis, year, and running_time. The first row contains: 1, Star Wars: Episode IV - A New Hope, <p>Luke Skywalker joins forces with a Jedi Knight,...., 1977, 2h 1m. Below the table are controls for 'Check all', 'With selected: Edit, Copy, Delete, Export', and another set of 'Show all', 'Number of rows: 25', and 'Filter rows: Search this table' controls.

Now that we now this happens, we can write a redirect and add it to addMovie, so this way it looks far more efficient to the end user. When the end user now adds elements to the database, they won't see addMovie.php, they'll just be redirected back to the index.php.

addMovie.php

```
1 <?php
2     include_once('functions/functions.php');
3     $dbConnect = dbLink();
4     //showMem();
5     $name = $_POST['mov_name'];
6     $synopsis = $_POST['mov_synopsis'];
7     $year = $_POST['mov_year'];
8     $rt = $_POST['mov_rt'];
9     $resultInsert = insertMovie($dbConnect,$name,$synopsis,$year,$rt);
10    // if($resultInsert)
11    //     echo'<br>Add successful';
12    ?>
13    <!DOCTYPE html>
14    <html>
15    <head>
16        <script type="text/javascript">
17            function redirect()
18            {
19                window.location.href = "index.php";
20            }
21        </script>
22    </head>
23    <body onload="redirect()">
24    </body>
25    </html>
```

Result – index.php (before addition)

Add a Movie

Enter Synopsis

File Edit View Format

Formats **B** *I* [List of icons]

After the rebels have been brutally overpowered by the Empire on their newly established base, Luke Skywalker takes advanced Jedi training with Master Yoda, while his friends are pursued by Darth Vader as part of his plan to capture Luke.

Powered by TinyMCE

Movie listing here...

Result – index.php (after addition)

Add a Movie

Enter Synopsis

File Edit View Format

Formats **B** *I* [List of icons]

This domain is not registered with TinyMCE Cloud. [Get a free API key to disable this message.](#)

Powered by TinyMCE

Movie listing here...

Result – phpmyadmin

Server: 127.0.0.1 » Database: lesson11 » Table: movies

Browse Structure SQL Search Insert Export Import Privileges Operations Tracking

Showing rows 0 - 1 (2 total, Query took 0.0006 seconds.)

```
SELECT * FROM `movies`
```

Profiling [Edit inline] [Edit] [Expl

Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

+ Options

	id	name	synopsis	year	running_time
<input type="checkbox"/>	1	Star Wars: Episode IV - A New Hope	<p>Luke Skywalker joins forces with a Jedi Knight,...	1977	2h 1m
<input type="checkbox"/>	2	Star Wars: Episode V - The Empire Strikes Back	<p>After the rebels have been brutally overpowered...	1980	2h 4m

Check all | With selected: Edit Copy Delete Export

Now, let's add AJAX to read back the content of the database. To do this we will add the code to index.php

```
1  <?php
2      include_once('functions/functions.php');
3      $dbConnect = dbLink();
4      if ($dbConnect)
5          echo '<!-- database connection established -->';
6  ?>
7  <!DOCTYPE html>
8  <html>
9      <head>
10         <title></title>
11         <style type="text/css">
12             body{background-color: aliceblue;}
13             input{padding: 10px 10px 10px 10px; margin: 10px 10px 10px 10px; border-radius: 15px;}
14             #wrapper{width: 80%; margin: 0 auto;}
15         </style>
16         <script src="https://cloud.tinymce.com/stable/tinymce.min.js"></script>
17         <script>tinymce.init({ selector:'textarea' });</script>
18
19         <script> /* AJAX Read */
20             function showMovies(){
21                 var xmlhttp;
22                 if (window.XMLHttpRequest)
23                     { // code for IE7+, Firefox, Chrome, Opera, Safari
24                         xmlhttp=new XMLHttpRequest();
25                     }
26                 else
27                     { // code for IE6, IE5
28                         xmlhttp=new ActiveXObject("Microsoft.XMLHTTP");
29                     }
30                 xmlhttp.onreadystatechange=function()
31                 {
32                     if (xmlhttp.readyState==4 && xmlhttp.status==200)
33                     {
34                         document.getElementById("movieDetails").innerHTML=xmlhttp.responseText;
35                     }
36                 }
37                 xmlhttp.open("GET","getMovie.php",true);
38                 xmlhttp.send();
39             }
40         </script>
41     </head>
42     <body>
43         <div id="wrapper">
44             <h2>Add a Movie</h2>
45             <form method="post" action="addMovie.php">
46                 <input type="text" name="mov_name" placeholder="Enter Movie Name"><br><br>
47                 <h3>Enter Synopsis</h3>
48                 <textarea name="mov_synopsis"></textarea>
49                 <br><br>
50                 <input type="text" name="mov_year" placeholder="Year Published">
51                 <input type="text" name="mov_rt" placeholder="Enter Running time">
52                 <input type="submit" value="Enter Details">
53             </form>
54
55             <p onclick="showMovies();" style="cursor:pointer;">Click here to View Movies</p>
56             <div id="movieDetails">Movie listing here...</div>
57         </div>
58     </body>
59 </html>
```


getMovie.php

```
1 <?php
2     include_once('functions/functions.php');
3     $dbConnect = dbLink();
4     if ($dbConnect !== NULL)
5         echo '<!-- db Connection established -->';
6     echo '<table><tr><th colspan="3">Movies</th></tr>';
7     $sql = "SELECT * FROM movies";
8     foreach ($dbConnect->query($sql) as $row) {
9         echo '<tr>';
10        $id = $row['id'];
11        echo '<td>'. $row['name']. '</td>';
12        echo '<td>Released: '. $row['year']. '</td>';
13        echo '<td>Running Time: '. $row['running_time']. '</td>';
14        echo '</tr>';
15        echo '<tr>';
16        echo '<td colspan="3">Synopsis<br>'. $row['synopsis']. '</td>';
17        echo '</tr>';
18    }
19    echo "</table>";
20 ?>
```

Result -index.php (before click)

Add a Movie

Enter Synopsis

File Edit View Format

← → Formats **B** *I* [List Icons]

This domain is not registered with TinyMCE Cloud. [Get a free API key](#) to disable this message. ✕

Powered by TinyMCE

[Click here to View Movies](#)

Movie listing here...

Result – index.php (after click)

Add a Movie

Enter Synopsis

File ▾ Edit ▾ View ▾ Format ▾

← → Formats ▾ **B** *I* [List Icons]

This domain is not registered with TinyMCE Cloud. [Get a free API key](#) to disable this message. ✕

Powered by TinyMCE

[Click here to View Movies](#)

Movies

Star Wars: Episode IV - A New Hope
Synopsis

Released: 1977

Running Time: 2h 1m

Luke Skywalker joins forces with a Jedi Knight, a cocky pilot, a wookiee and two droids to save the galaxy from the Empire's world-destroying battle-station, while also attempting to rescue Princess Leia from the evil Darth Vader.

Star Wars: Episode V - The Empire Strikes Back
Synopsis

Released: 1980

Running Time: 2h 4m

After the rebels have been brutally overpowered by the Empire on their newly established base, Luke Skywalker takes advanced Jedi training with Master Yoda, while his friends are pursued by Darth Vader as part of his plan to capture Luke.

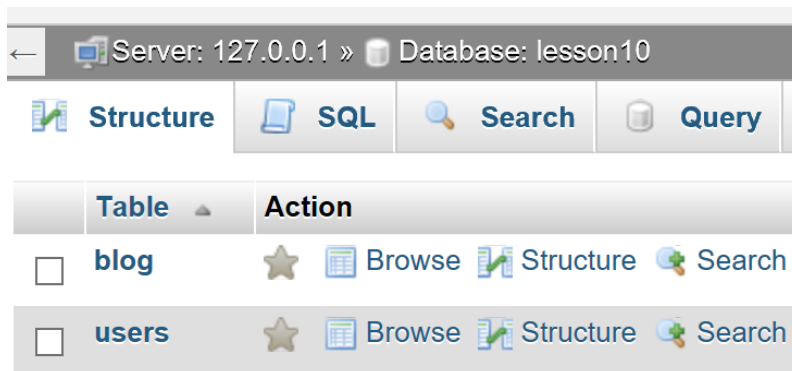
Backup

This concludes tutorial.

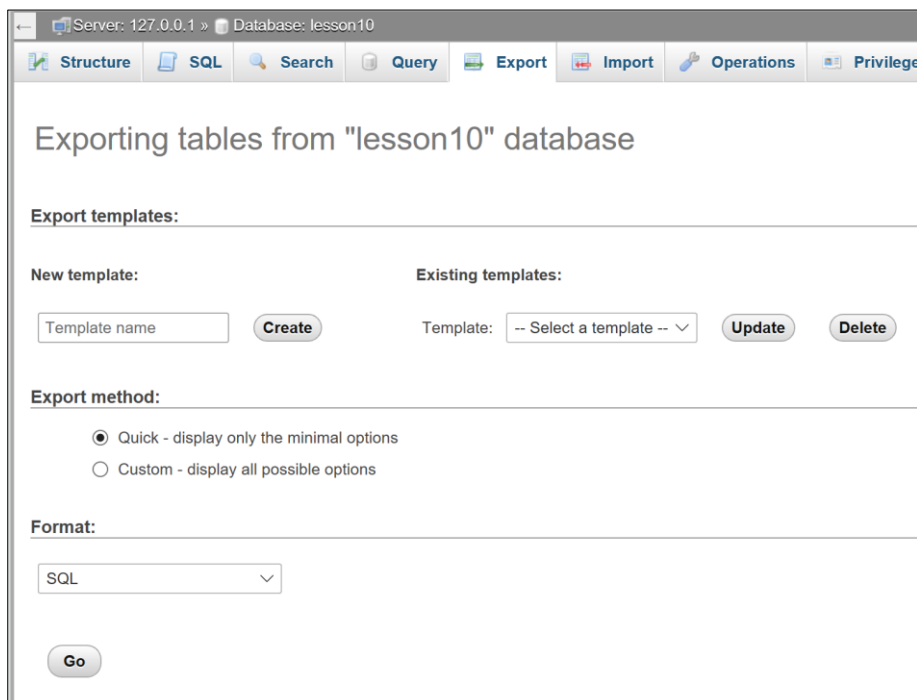
As we are now using servers, and server information, there are two areas to backing up your information.

The first area is the htdocs folder. This is all of the php/html/css/javascript files. These are all needed to view the website.

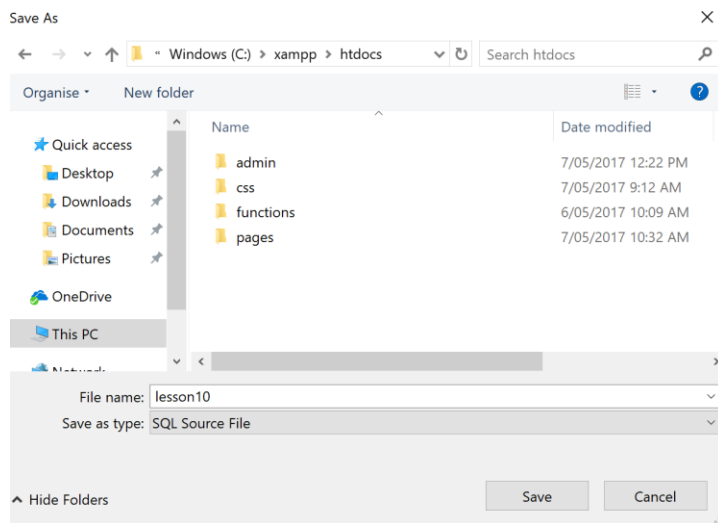
Secondly, the database itself. To back up the database, go to phpmyadmin and navigate to the root area of the database



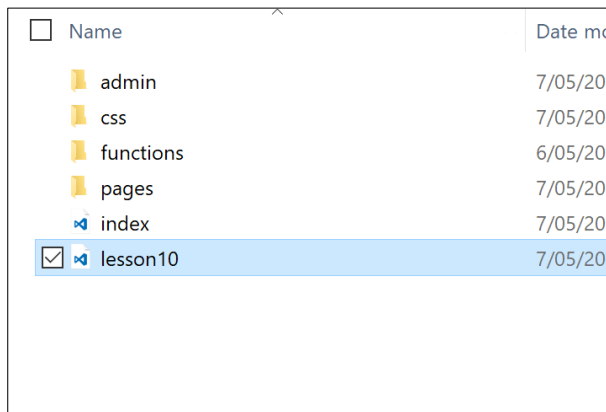
From here, click on export:



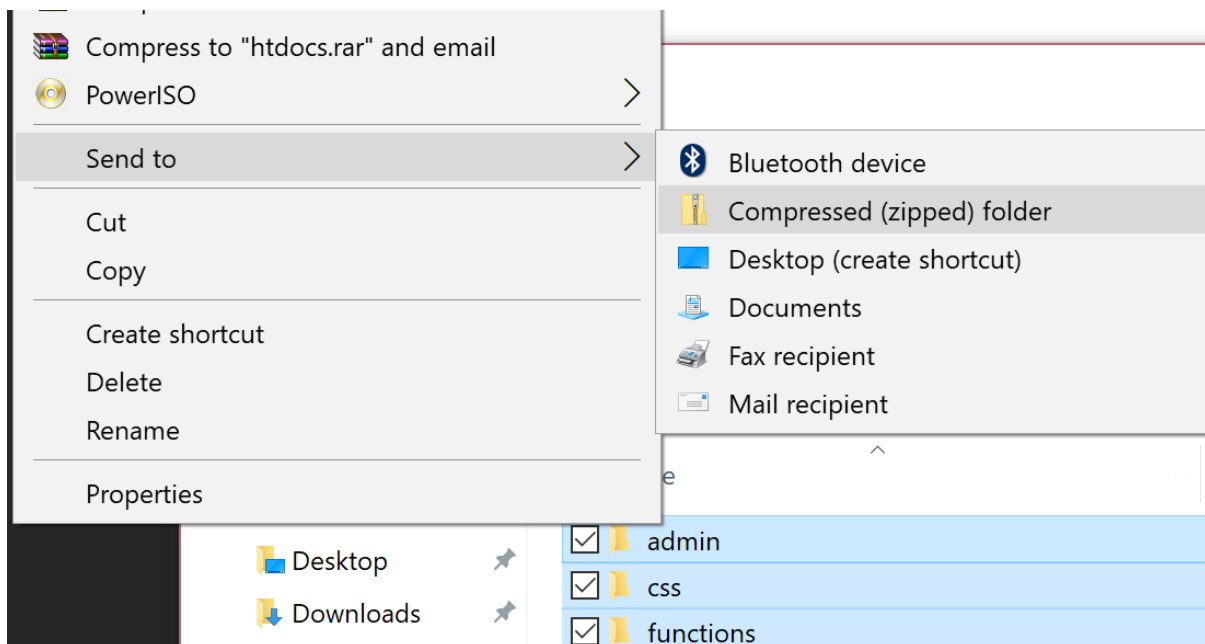
Click on go and save this .sql file into the htdocs folder



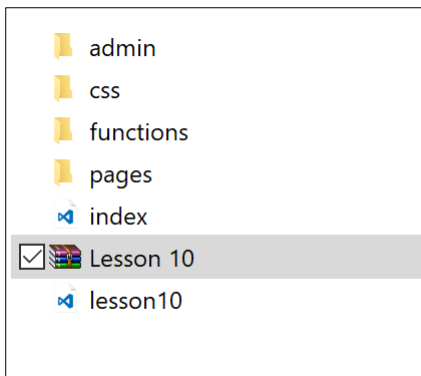
When viewing the htdocs folder you should see



Highlight everything and right click, then select send to compressed file



It will save a zip file, name this file appropriately.



If the icon is different, this is because on the machine this was created on, I use winrar as my compression tool.

Save your work (c:\xampp\htdocs) to USB, student drive, Onedrive, Dropbox or zip and email the folder to yourself.