CIS-530: Bonus Assignment

Name: Prasanna Aditya Kandarpa

ID: 2896149

INTRODUCTION:

This Web Application is a simple User-Interface where the User can search the Employee details in a company by using Last Name. The User must enter the Last Name of the Employee and then it will retrieve the Employee details like First name, Last name, Sex and Department from the Company Database and display on the Web page.

SOFTWARES/TOOLS REQUIRED (Download links provided below):

- XAMPP Control Panel v3.3.0

  https://sourceforge.net/projects/xampp/files/XAMPP%20Windows/8.2.12/xampp-windows-x64-8.2.12-0-VS16-installer.exe

- Microsoft SQL Server 2022 Developer Edition

  https://go.microsoft.com/fwlink/p/?linkid=2215158&clcid=0x409&culture=en-us&country=us

- Microsoft SQL Server Management Studio 20.1

  https://aka.ms/ssmsfullsetup

  (Matching version of Microsoft Visual Studio should be pre-installed in the system)

- Visual Studio Code

  https://code.visualstudio.com/sha/download?build=stable&os=win32-x64-user

  (This is just an IDE to edit PHP/HTML scripts. We can edit in Notepad as well if its more convenient)

- Microsoft Drivers 5.12 for PHP for SQL Server

  https://go.microsoft.com/fwlink/?linkid=2258816

LANGUAGES USED:

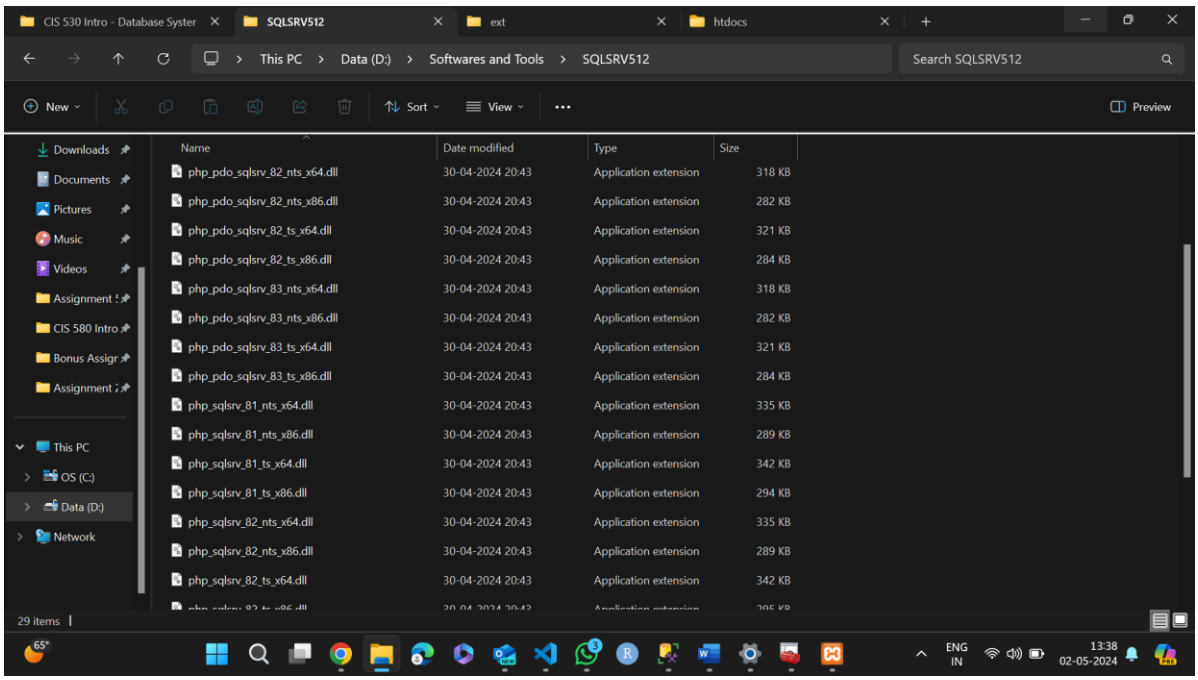
- HTML and CSS
- PHP
- JavaScript
- SQL

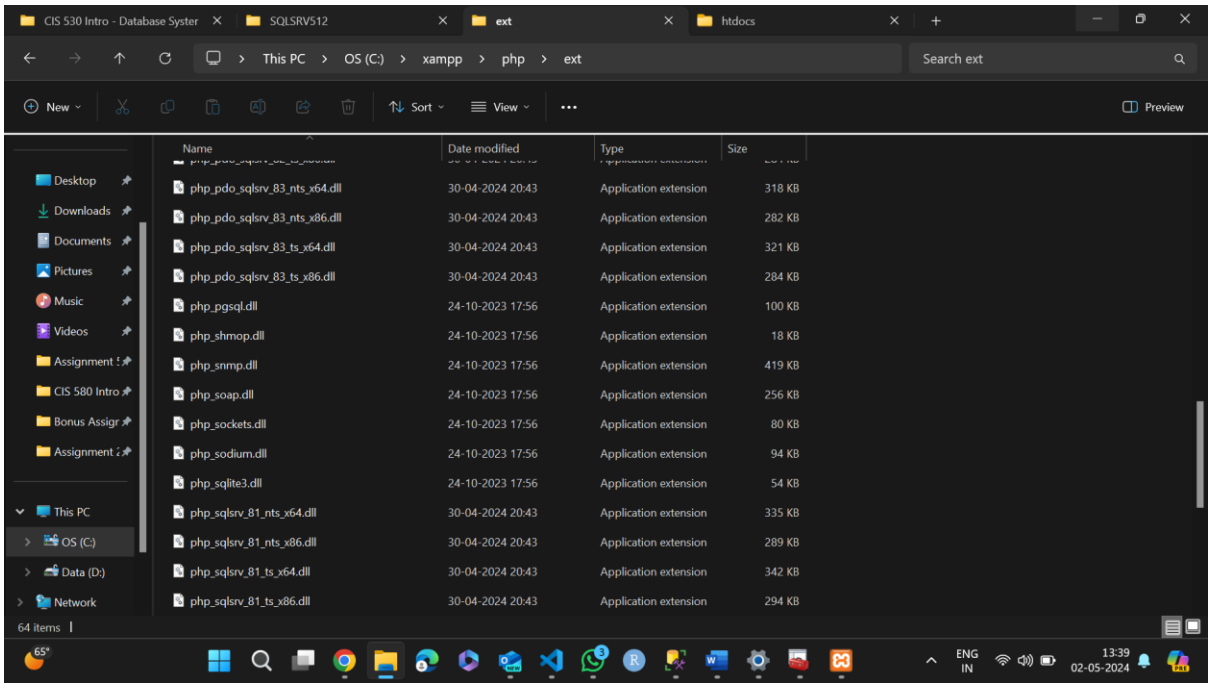
STEP 1 – Install and Setup XAMPP server in system:

- Install the XAMPP server which is downloaded from the first link given above.
- Follow onscreen instructions and complete the installation.

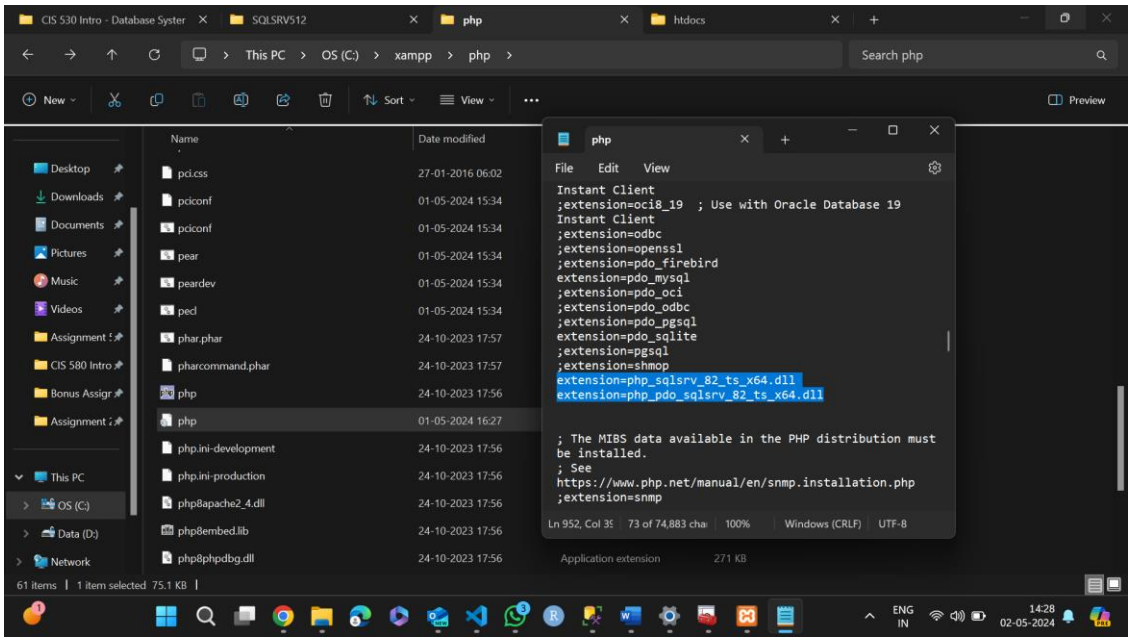
STEP 2 – Setup the PHP drivers for SQL Server

- Extract the SQLSRV512.zip file downloaded from the last link given above.
- Copy the php_pdo_sqlsrv_*.dll and php_sqlsrv_*.dll files from the extracted folder SQLSRV512 and paste them in the PHP extensions folder in XAMPP (C:\xampp\php\ext)





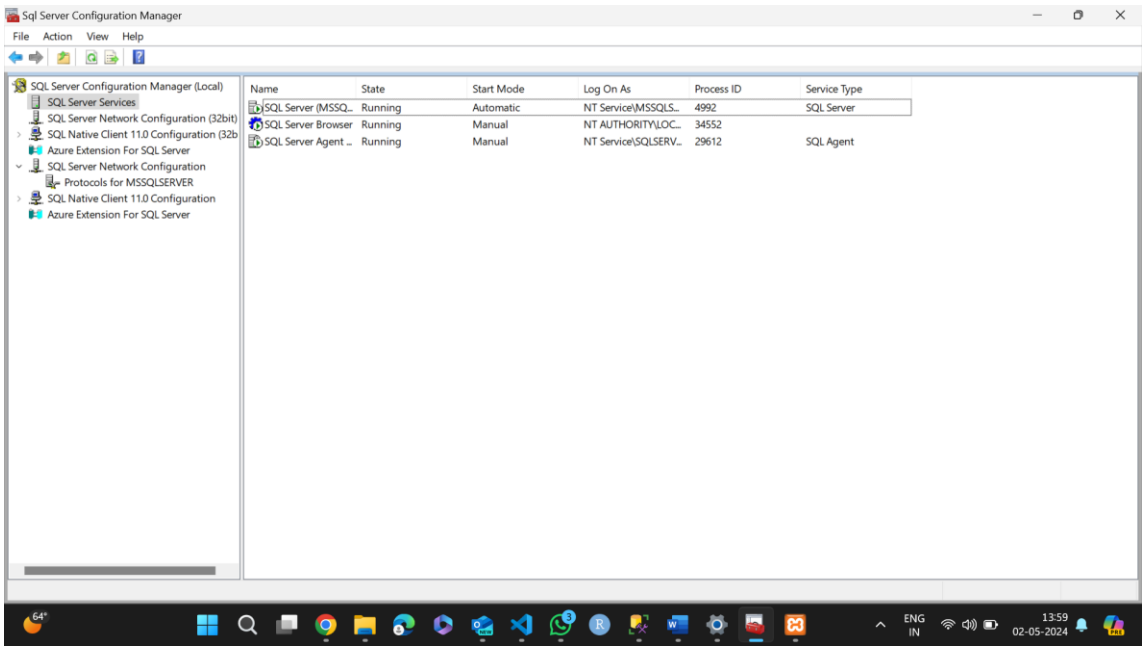
- Open the ‘php.ini’ file from C:\xampp\php in notepad and edit it via Notepad.
- Add the below lines as highlighted to configure the PHP drivers for SQL Server.



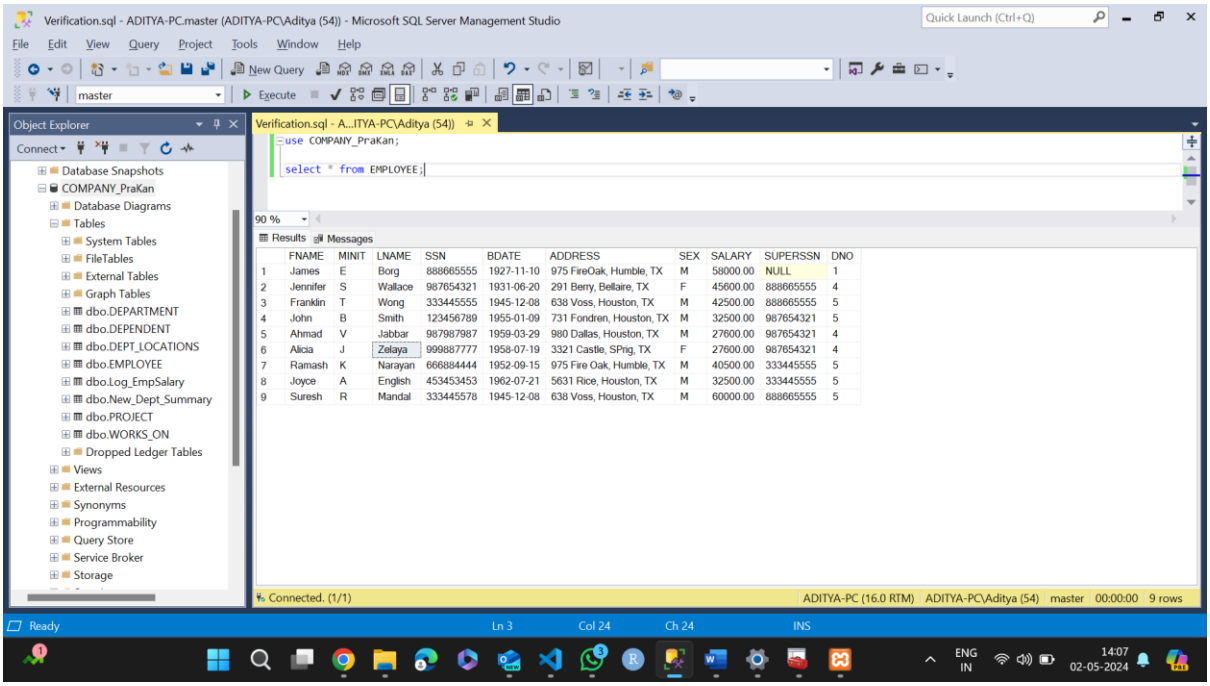
- Here we have used php_pdo_sqlsrv_82_ts_x64.dll and php_sqlsrv_82_ts_x64.dll as we are using Thread Safe (TS) 64-bit PHP drivers and the version of PHP installed via XAMPP is 8.2.12.
- Refer to the Additional Steps Section on more details on how to verify the PHP version and how to check if the SQL Server drivers are correctly configured for PHP.

STEP 3 – Run the SQL Server and SQL Server Management Studio:

- Start the Microsoft SQL Server Configuration Manager and make sure the SQL Server Services are in running state: SQL Server, SQL Server Browser and SQL Server Agent.

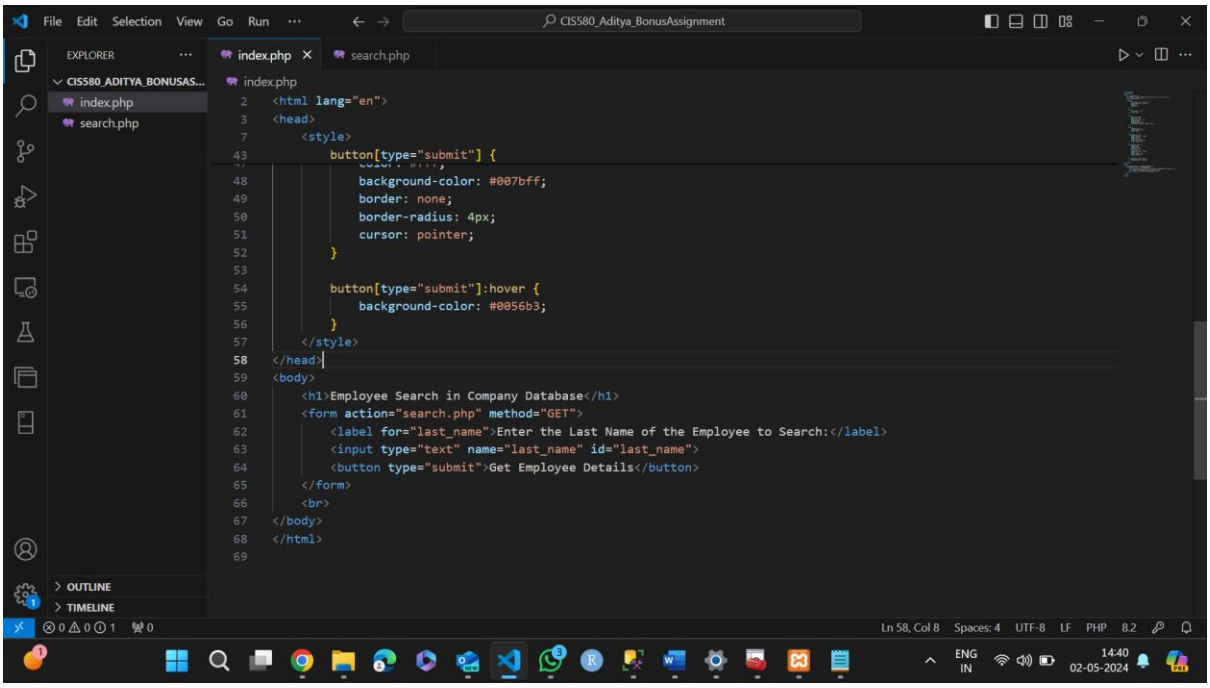


- Open the SQL Server Management Studio and connect to the Company Database “COMPANY_PraKan” which is already created and setup in the previous Labs using Windows Authentication.
- Retrieve the records in the EMPLOYEE table (Just to verify the existing records).

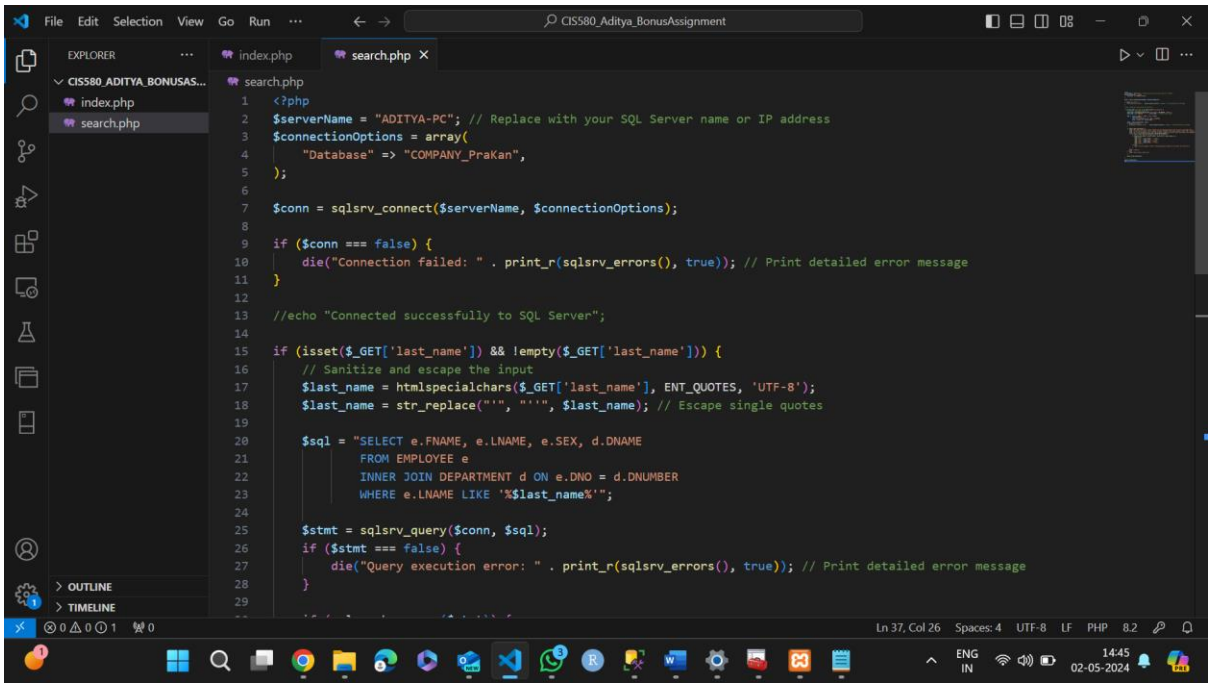


STEP 4 – Create the ‘index.php’ and ‘search.php’ files:

- Open VSCode and write the script for the HomePage of the Application which contains a simple form to enter the Last name of the Employee and a button to search for the respective details. Save this file as ‘index.php’

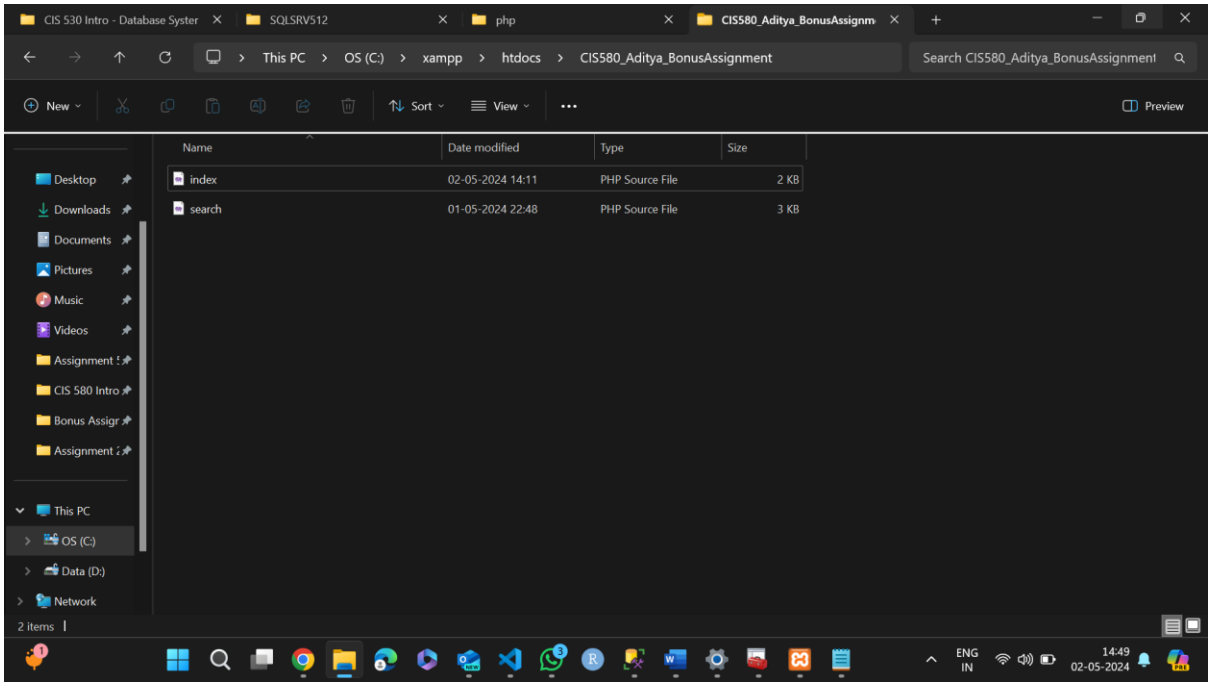


- Write another script for the search functionality which will connect to the Company Database in the SQL server and retrieve the Employee details with the given Last name and display the details on another page. Save this file as ‘search.php’



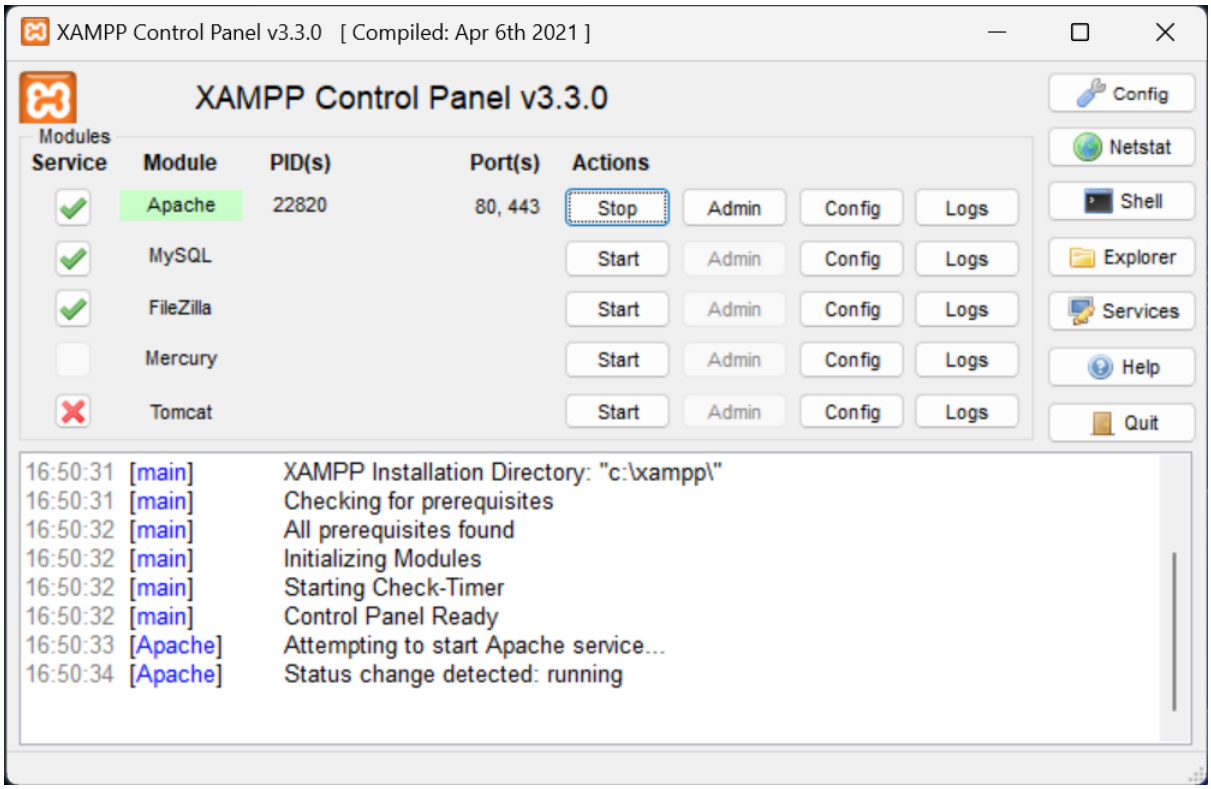
```php
<?php
$serverName = "ADITYA-PC"; // Replace with your SQL Server name or IP address
$connectionOptions = array(
    "Database" => "COMPANY_PraKan",
);

$conn = sqlsrv_connect($serverName, $connectionOptions);

if ($conn === false) {
    die("Connection failed: " . print_r(sqlsrv_errors(), true)); // Print detailed error message
}

//echo "Connected successfully to SQL Server";

if (isset($_GET['last_name']) && !empty($_GET['last_name'])) {
    // Sanitize and escape the input
    $last_name = htmlspecialchars($_GET['last_name'], ENT_QUOTES, 'UTF-8');
    $last_name = str_replace("'", "''", $last_name); // Escape single quotes

    $sql = "SELECT e.FNAME, e.LNAME, e.SEX, d.DNAME
            FROM EMPLOYEE e
            INNER JOIN DEPARTMENT d ON e.DNO = d.DNUMBER
            WHERE e.LNAME LIKE '%$last_name%'";

    $stmt = sqlsrv_query($conn, $sql);
    if ($stmt === false) {
        die("Query execution error: " . print_r(sqlsrv_errors(), true)); // Print detailed error message
    }
```

- Detailed explanation of the code in these two files is given in a separate section below named ‘Explanation of the Code’.
- Place both these files in the Web Server Directory (C:\xampp\htdocs)
  Create a folder ‘CIS580_Aditya_BonusAssignment’ in the htdocs folder and place both the index.php and search.php files.

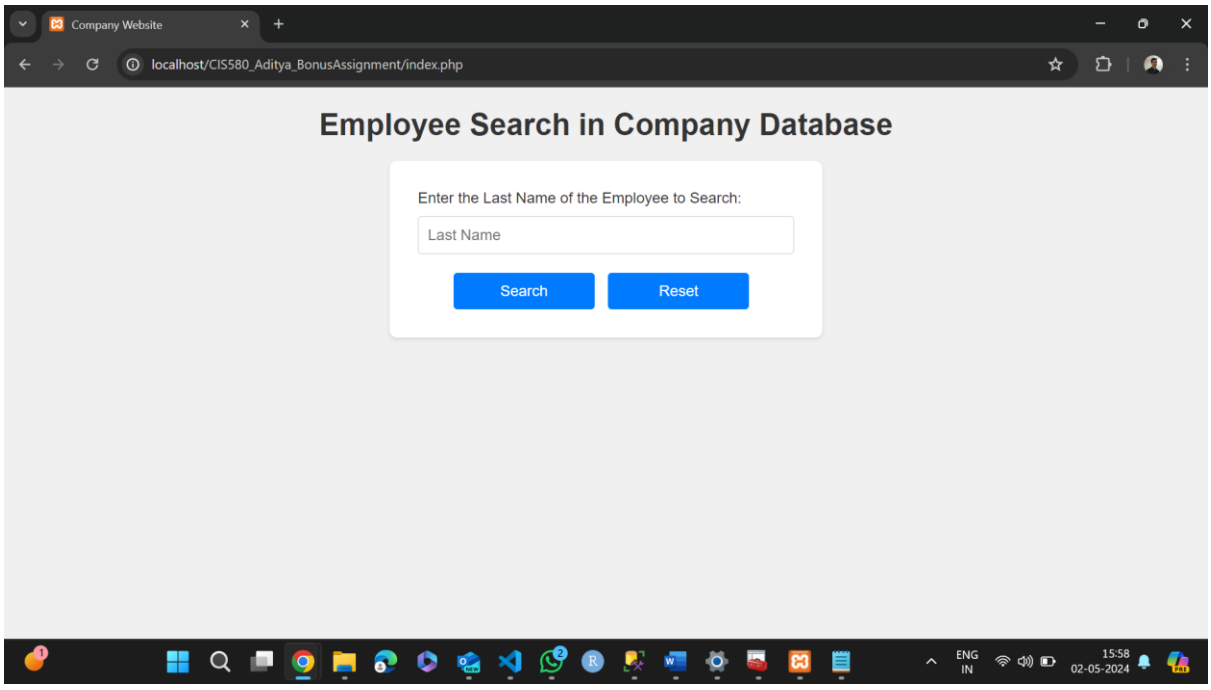


STEP 5 – Start the Server and launch the web page:

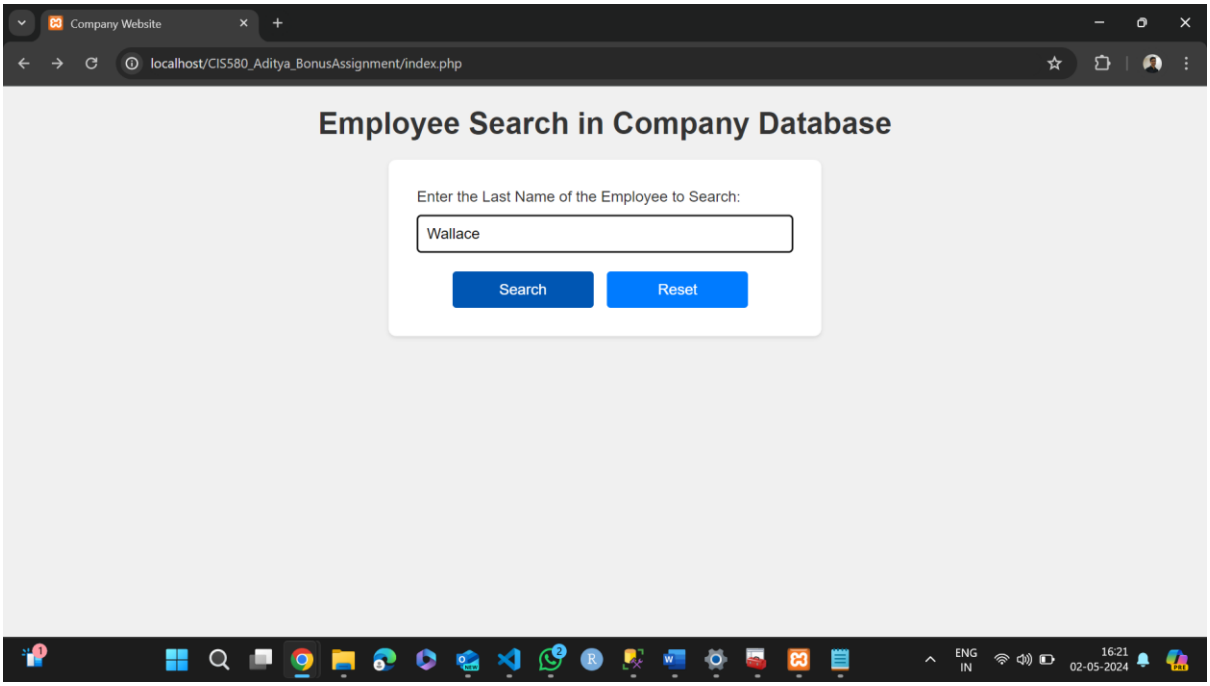
- Open the XAMPP Control panel and start the Apache Service.
  Note: Start the XAMPP Control Panel as Administrator for



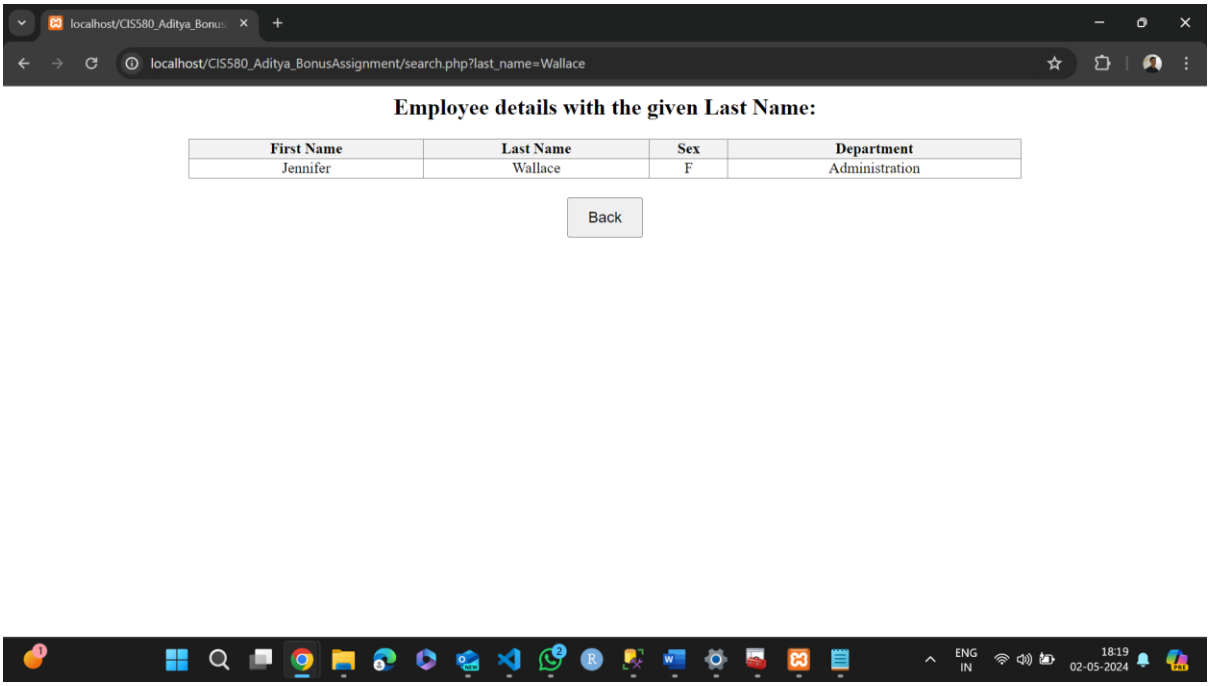
- Open the Browser and go to the below site
  'http://localhost/CIS580_Aditya_BonusAssignment/index.php'



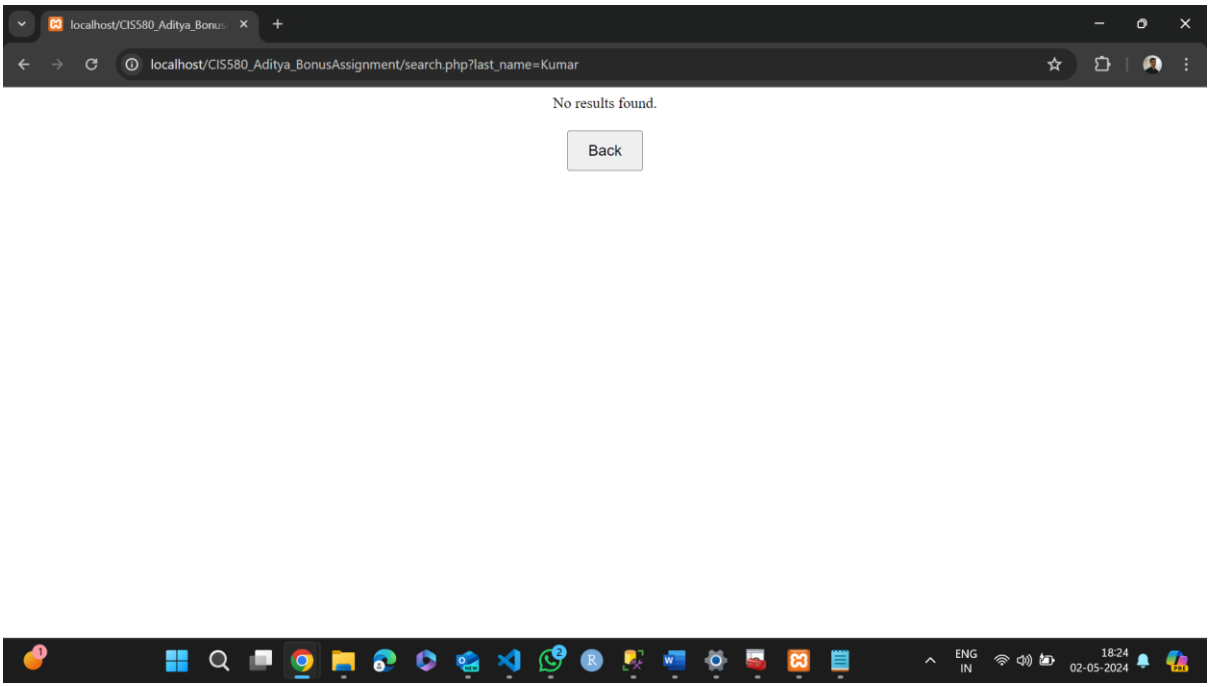
- Enter any Last Name which was inserted in the Employee table and click on ‘Search’



- After Clicking on Search, the Employee details with the given Last Name are displayed by the script.

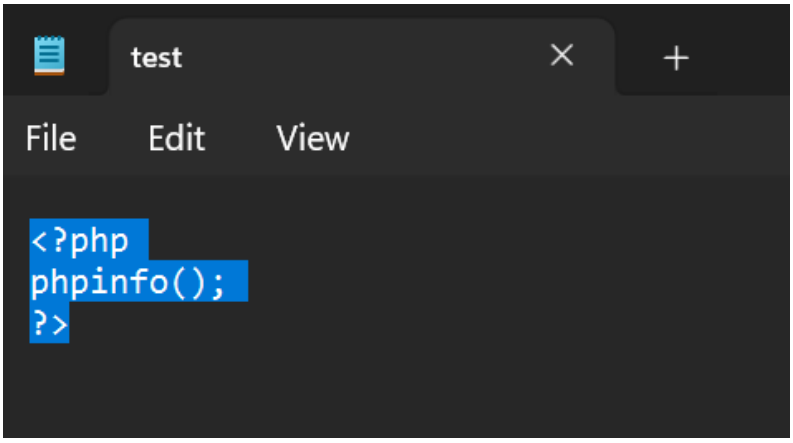


- If the given Last Name does not exist in the Employee table, i.e., there are no records with the given Last Name, then the script displays the below message:

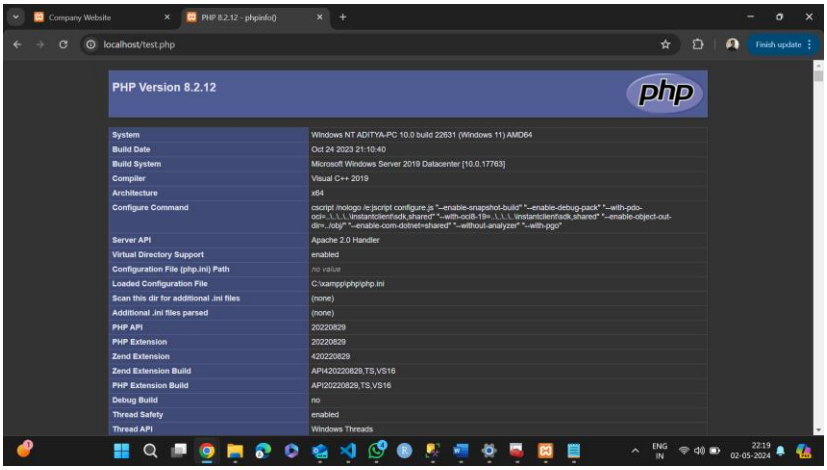


ADDITIONAL STEPS (Optional)

- Create a small file 'test.php' with the below lines and place it in the Web Server Directory (C:\xampp\htdocs)



- Open the below link in the browser (XAMPP Control Panel should be up, and Apache Service should be running).
  http://localhost/test.php



- This will verify that PHP is correctly configured and the version is 8.2.12.
- Look for pdo_sqlsrv and sqlsrv sections in this webpage.

pdo_sqlsrv

pdo_sqlsrv support	enabled	
ExtensionVer	5.12.0+17729	
Directive	**Local Value**	**Master Value**
pdo_sqlsrv.client_buffer_max_kb_size	10240	10240
pdo_sqlsrv.log_severity	0	0
pdo_sqlsrv.report_additional_errors	1	1

sqlsrv

sqlsrv support	enabled	
ExtensionVer	5.12.0+17729	
Directive	**Local Value**	**Master Value**
sqlsrv.ClientBufferMaxKBSize	10240	10240
sqlsrv.LogSeverity	0	0
sqlsrv.LogSubsystems	0	0
sqlsrv.WarningsReturnAsErrors	On	On

- The details in the above two sections verify that the SQL drivers are correctly setup for PHP in the system.

EXPLANATION OF THE CODE:

The section is the detailed explanation of the code written to develop and run this web page. Please refer to the actual code files which are shared separately with this document.

index.php -

This PHP script creates a basic webpage using HTML and CSS where users can input the last name and search for employees in the company's database. Let's analyse the code in detail:

1. **<!DOCTYPE html>**: This declaration specifies the document type and the version of HTML being used.
2. **<html lang="en">**: This is the opening tag for the HTML script. **lang="en"** indicates that the script is written in English.

3. **<head>**: This section contains meta-information about the document and links to external resources such as CSS files and JavaScript files. It contains the below sub tags:

   <meta charset="UTF-8">: Defines the character encoding for the document as UTF-8

   <meta name="viewport" content="width=device-width, initial-scale=1.0">: Sets the viewport for responsive design, ensuring the page renders properly on different devices and sets the width and zoom level**.**

   <title>Company Website</title>: Sets the title of the web page which is displayed on the browser's title tab.

   <style>: This block contains the CSS styling of the web page.

4. **<body>**: This section contains all the visible elements of the web page. Below are each sections description:

   <h1>Employee Search in Company Database</h1>: Heading element which displays the title of the web page.

   <form action="search.php" method="GET">: Form element that defines a form for user input which sends the data to a PHP script named **search.php** via the GET method when the form is submitted. It contains the below elements:

      <label for="last_name">Enter the Last Name of the Employee to Search:</label>: Label which provides description for the input field and improves accessibility.

      <input type="text" name="last_name" id="last_name" placeholder="Last Name" required>: Text input field where the users can input the Last name of the employee to search in the database.

         type="text": Specifies the input type as text.

         name="last_name": Provides a name for the input field used to identify the input data when the form is submitted.

         id="last_name": Provides an id for the input field, used for scripting purposes.

         placeholder="Last Name": Displays a placeholder text inside the input field.

         required: Indicates that the input field is mandatory and must be filled out before submitting the form.

      ** **: Container for the buttons.

         <button type="submit">Search</button>: Submit button, when clicked, submits the form data to the PHP script specified in the form's action attribute.

         <button type="reset">Reset</button>: Reset button, when clicked, resets all form controls to their initial values.

5. **</form>**: This tag closes the form element.

6. **</body>**: This tag closes the body element.
7. **</html>**: This tag closes the HTML document.

The CSS styles defined in the **<style>** block define the appearance of the page:

- **body:** Sets the font family, background colour, margin, and padding for the entire page.
- **h1:** Styles the heading with centred alignment, colour, and margin.
- **form:** Styles the form with a maximum width, background colour, padding, border radius, and box shadow.
- **label:** Styles the labels for the input fields.
- **input[type="text"]:** Styles the text input fields with width, padding, font size, border, border radius, and box sizing.
- **.button-container:** Styles the container for buttons with margin and text alignment.
- **button[type="submit"] and button[type="reset"]:** Styles the submit and reset buttons with width, padding, font size, colour, background colour, border, border radius, cursor, and hover effect.

search.php-

This PHP script connects to a SQL Server database, executes a query based on the provided last name, displays the results in an HTML table, and provides a button to go back to the previous page. Let's analyze the code in detail:

1. **Setting up Database Connection**

- This section establishes a connection to a SQL Server using **sqlsrv_connect()** function.
- **$serverName** contains the name or IP address of the SQL Server.
- **$connectionOptions** contains an array specifying the database name.
- **sqlsrv_connect()** returns a connection resource if successful or false if failed.

2. **Checking Connection**

- If the connection fails, it prints an error message along with the detailed error information.

3. **Handling GET request**

- This checks if the **'last_name'** parameter is not empty.

4. **Sanitizing the input**

- **htmlspecialchars()** is used to convert special characters to HTML entities.
- **str_replace()** is used to escape single quotes by doubling them to prevent SQL injection.

5. **Executing the SQL Query**

- This section executes an SQL query to select Employee details based on the provided last name.
- It joins the EMPLOYEE table with the DEPARTMENT table and filters based on the last name.

6. **Checking and Displaying Query Results**

- This section first checks if the query has returned any rows.
- Then it fetches each row and displays the details from the EMPLOYEE table.
- If no rows are found, it displays a message that no results are found.

7. **Freeing Resources and Closing connection**

- This section releases the statement resources after the execution and closes the database connection.

8. **JavaScript Function to Go back**

- This section creates a button which calls a JavaScript function, when clicked, to go back to the previous page using the **'window.history.back()**' method.

Verification.sql-

- This is a small optional SQL script just to check the Data in EMPLOYEE table using the SELECT Query in the SQL Server Management Studio.

⇨ DESCRIPTION OF THE PROBLEM, INPUT AND OUTPUT:

Problem Description:

We are creating a Web Application where users can search for an Employee by entering their last name. The Script will perform the below two main functionalities:

- Search the Employee by Last Name
- Retrieve the Results from the Database and Display them on the Web Page.

Input and Output:

The Input is the Last Name of the Employee whose data needs to be displayed.

The Output is the Table with Employee details like First name, Last name, Sex, and Department which are retrieved from the Database and Displayed on the Web Page.

⇨ TESTING OF THE PROGRAM

The Application need to be tested to ensure all the functionalities are working properly and it handles all the input scenarios.

- Valid/Invalid Input: If the entered Last Name exists in the Database, then it retrieves the data and displays it on the web page. If the entered Last Name does not exist in the Database, the Script displays the Error message “No Results found”.
- Empty Input: If the Last name is not entered in the web page and the Search button is pressed, the Script prompts a message that the field is required.
- Connectivity Testing: The Script successfully establishes connection with the Database and retrieves the data. If the connection fails, it will display the Error message as “Connection failed”.
- Cross-Browser Testing: The Web site runs without any issues in all browsers, Chrome, Edge and Firefox.