**Task 2 Code Explanation**

**connection.php:**

<?php

$db\_host="db"; //localhost server

$db\_user="root"; //database username

$db\_password="csym019"; //database password

$db\_name="Recipe"; //database name

Try //Exception Handling

{

 $db=new PDO("mysql:host={$db\_host};dbname={$db\_name}",$db\_user,$db\_password);

 $db->setAttribute(PDO::ATTR\_ERRMODE, PDO::ERRMODE\_EXCEPTION); //A new PDO object is created and connected to MySQL using server, username and password

And setting it up so that database will show if there are any errors.

}

catch(PDOEXCEPTION $e)

{

 $e->getMessage(); //Getting the exception message

}

?>

**index.php:**

<?php

require\_once 'connection.php'; //Using the connection file

session\_start(); //Starting the session

if(isset($\_SESSION["user\_login"])) //check if session user\_login is set or not

{

 header("location: home.php"); //If session user\_login then redirect to home.php

}

if(isset($\_POST['login'])) //If login from the form is set and is not null

{

 $username =strip\_tags($\_POST["username"]); //Strips the string in username field of form from HTML,XML,PHP tags

 $email  =strip\_tags($\_POST["username"]); //Strips the string in username field of form from HTML,XML,PHP tags

 $password =strip\_tags($\_POST["password"]);   //Strips the string in password field of form from HTML,XML,PHP tags

 if(empty($username)){      //If username is empty

  $errorMsg[]="please enter username or email"; //Add the error message into errorMsg array

 }

 else if(empty($email)){ //If email is empty

  $errorMsg[]="please enter username or email"; //Add the error message into errorMsg array

 }

 else if(empty($password)){ //If password is empty

  $errorMsg[]="please enter password"; // Add the error message into errorMsg array

 }

 else

 {

  try //Exception Handling

  {

   $query=$db->prepare("SELECT \* FROM users WHERE username=? OR email=?"); //sql select query

   $query->execute(array($username,$email)); //execute query with bind parameter

   $row=$query->fetch(PDO::FETCH\_ASSOC); //Fetching the result as an associative array from database and storing in $row variable

   if($query->rowCount() > 0) //If Database record greater than 0

   {

    if($username==$row["username"] OR $email==$row["email"]) //check condition username or email matched from the database.

    {

     if($password== $row["password"]) //Check condition password matched from database password.

     {

      $\_SESSION["user\_login"] = $row["user\_id"]; //start the session with name "user\_login"

      $loginMsg = "Successfully Login...";  //Set user login success message

      header("refresh:2; home.php");  //Redirect to home.php

     }

     else

     {

      $errorMsg[]="wrong password"; //add the error message if passwords not matched

     }

    }

    else

    {

     $errorMsg[]="wrong username or email"; //add the error message if username or email not matched

    }

   }

   else

   {

    $errorMsg[]="wrong username or email"; //add the error message if username or email not matched

   }

  }

  catch(PDOException $e)

  {

   $e->getMessage(); //Handle any exceptions thrown and display the exception message

  }

 }

}

?>

<!DOCTYPE html> //HTML starts

<html>

  <head>

    <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css"> //Using Bootstrap css

    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"></script> //Using Jquery

    <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js"></script> //Using Bootstrap

    <link rel="stylesheet" href="layout.css"> //External stylesheet for the page

  </head>

  <body>

    <br> //Break

    <div class="login"> //Div element with class login

<form method="post" class="form-horizontal"> //Login Form with method Post

 <div class="form-group"> //Grouping label and input field

 <label class="col-sm-3 control-label">Username or Email</label> //Label for username or email

 <div class="col-sm-6"> //In small devices take width of 6 columns

 <input type="text" name="username" class="form-control" placeholder="Enter username or email" /> //Input field

 </div>

 </div>

 <div class="form-group"> //Grouping label and input field

 <label class="col-sm-3 control-label">Password</label> //Label for Password

 <div class="col-sm-6">

 <input type="password" name="password" class="form-control" placeholder="Enter passowrd" /> //input field

 </div>

 </div>

 <div class="form-group">

 <div class="col-sm-offset-3 col-sm-9 m-t-15"> //In sm devices set offset of 3 cols and take width of 9 cols

 <input type="submit" name="login" class="btn btn-success" value="Login"> //Login Button

 </div>

 </div>

 <div class="form-group">

 <div class="col-sm-offset-3 col-sm-9 m-t-15">

 You don't have a account register here? <a href="register.php"><p class="text-info">Register Account</p></a>   //Register Link

 </div>

 </div>

</form>

<?php

if(isset($errorMsg)) //If errorMsg is set or not null

{

//The below foreach is used to loop through the errorMsg array

 foreach($errorMsg as $error)

 {

 ?>

  <div class="alert alert-danger">

   <strong><?php echo $error; ?></strong> //Displaying the errors in red colour

  </div>

    <?php

 }

}

if(isset($loginMsg)) //If loginMsg is set or not null

{

?>

 <div class="alert alert-success">

  <strong><?php echo $loginMsg; ?></strong> //Display the login message in green color

 </div>

<?php

} //End of loop

?>

</div>

</body> //End of body

</html> //End of html

**register.php:**

<?php

require\_once "connection.php"; //Use connection.php

if(isset($\_POST['btn\_register'])) //If btn\_register in form is set or not null

{

 $username = strip\_tags($\_POST['username']); //Strips the string in username field of form from HTML,XML,PHP tags

 $email  = strip\_tags($\_POST['email']); //Strips the string in email field of form from HTML,XML,PHP tags

 $password = strip\_tags($\_POST['password']); //Strips the string in password field of form from HTML,XML,PHP tags

 if(empty($username)){ //If username is empty

  $errorMsg[]="Please enter username"; //Add the error message to errorMsg array

 }

 else if(empty($email)){ //If email is empty

  $errorMsg[]="Please enter email"; //Add the error message to errorMsg array

 }

 else if(empty($password)){ //If Password is empty

  $errorMsg[]="Please enter password"; //Add the error message to errorMsg array }

 else if(strlen($password) < 6){ //If password is less than 6 characters

  $errorMsg[] = "Password must be atleast 6 characters"; //Add the error message to errorMsg array }

 else

 {

  Try //Exception Handling

  {

   $query=$db->prepare("SELECT username, email FROM users

          WHERE username=? OR email=?"); // sql select query

   $query->execute(array($username, $email)); //execute query

   $row=$query->fetch(PDO::FETCH\_ASSOC); //Fetching the result as an associative array from database and storing in $row variable

   if($row["username"]==$username){ //Check if username already exists

    $errorMsg[]="Sorry username already exists"; //Add the error message to errorMsg array

   }

   else if($row["email"]==$email){ //Check if email already exists

    $errorMsg[]="Sorry email already exists"; //Add the error message to errorMsg array

   }

   else if(!isset($errorMsg)) //check if there’s no errorMsg

   {

    $insert\_stmt=$db->prepare("INSERT INTO users (username,email,password) VALUES

                (:uname,:uemail,:upassword)");   //sql insert query

    if($insert\_stmt->execute(array( ':uname' =>$username,

                                    ':uemail'=>$email,

                                    ':upassword'=>$password))){

     $registerMsg="Register Successfully..... Please Click On Login Account Link"; //Set execute query success message to registerMsg

    }

   }

  }

  catch(PDOException $e) //Catch any exceptions

  {

   echo $e->getMessage(); //Display the exception messsages

  }

 }

}

?>

<!DOCTYPE html> //HTML starts

<html>

  <head>

    <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css"> //Using Bootstrap.css

    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"></script> //Using JQuery

    <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js"></script> //Using Bootstrap

    <link rel="stylesheet" href="layout.css"> //Using external stylesheet for page

  </head>

  <body> //Start of Body

    <br> //Break

    <div class="login"> //Division Element with class login

<form method="post" class="form-horizontal"> //Registration Form with method Post

 <div class="form-group"> //grouping label and input field

 <label class="col-sm-3 control-label">Username</label> //Label for username

 <div class="col-sm-6">

 <input type="text" name="username" class="form-control" placeholder="Enter username" /> //Input field for username

 </div>

 </div>

 <div class="form-group"> //grouping label and input field

 <label class="col-sm-3 control-label">Email</label> //Label for email

 <div class="col-sm-6">

 <input type="text" name="email" class="form-control" placeholder="Enter email" /> //Input field for email

 </div>

 </div>

 <div class="form-group"> //grouping label and input field

 <label class="col-sm-3 control-label">Password</label> //Label for password

 <div class="col-sm-6">

 <input type="password" name="password" class="form-control" placeholder="Enter passowrd" /> //input field for password

 </div>

 </div>

 <div class="form-group">

 <div class="col-sm-offset-3 col-sm-9 m-t-15">

 <input type="submit"  name="btn\_register" class="btn btn-primary " value="Register"> //Button Register with classes btn btn-primary

 </div>

 </div>

 <div class="form-group">

 <div class="col-sm-offset-3 col-sm-9 m-t-15">

 You have a account register here? <a href="index.php"><p class="text-info">Login Account</p></a>  //Link to login account

 </div>

 </div>

</form> //Form ends

<?php

if(isset($errorMsg)) //If errorMsg is not null and is set

{

//The below foreach is used to loop through the errorMsg array

 foreach($errorMsg as $error)

 {

 ?>

  <div class="alert alert-danger">

   <strong>WRONG ! <?php echo $error; ?></strong> //Displaying each error message in red color

  </div>

    <?php

 }

}

if(isset($registerMsg)) //If registerMsg is set or not null

{

?>

 <div class="alert alert-success">

  <strong><?php echo $registerMsg; ?></strong> //Display the error message in green color

 </div>

<?php

} //End of for loop

?>

</div>

</body> //End of body

</html> //End of HTML

**Home.php:**

<?php

 require\_once 'connection.php'; //Use connection.php

 session\_start(); //Start the session

 if(!isset($\_SESSION['user\_login'])) //check unauthorize user not access in "home.php" page

 {

  header("location: index.php"); //If session is not user\_login then redirect to index.php

 }

 $id = $\_SESSION['user\_login']; //Getting the id

 $select\_stmt = $db->prepare("SELECT \* FROM users WHERE user\_id=:uid"); //SQL select query

 $select\_stmt->execute(array(":uid"=>$id)); //Executing the query and fetching the data where user\_id = session id

 $row=$select\_stmt->fetch(PDO::FETCH\_ASSOC); //Fetching the result as an associative array from database and storing in $row variable

 if(isset($\_SESSION['user\_login'])) //If user\_login is set and not null

 {

 ?>

<!DOCTYPE html> //HTML starts

<html>

  <head>

    <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css">

    <link rel="stylesheet" href="layout.css"> //Using Bootstrap.css

    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"></script> //using JQuery

    <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js"></script> //Using Bootstrap Js

  </head>

  <body>

  <header> //Start of header section

            <a class="home" href="home.php"><h3>CSYM019 - BBC GOOD FOOD RECIPES</h3></a>

            Welcome,

<?php

   echo $row['username']; //Prints welcome message

 }

?>

<?php

if(isset($errorMsg)) //If errorMsg is set or not null

{

 ?>

  <div class="alert alert-danger">

   <strong><?php echo $errorMsg; ?></strong> //Display error messages in red color

  </div>

    <?php

}

?>

            <a class="logout" href="logout.php">Logout</a> //Link for logout of session

        </header> //End of header section

 <nav> //Start of nav section

            <ul> //Unordered List

                <li><a href="./recipeSelectionForm.php">Recipe Report</a></li> //Link to Recipe Report page

                <li><a href="./newRecipe.php">New Recipe</a></li> //Link to New Recipe page

            </ul>

        </nav> //End of navigation section

        <main> //Start of main section

<?php

        try //Exception Handling

  {

   $query=$db->prepare("SELECT \* FROM Recipes"); // sql select query

   $query->execute(); //execute query

   $results=$query->fetchAll(); //Fetch All the rows from the databaseand store in results variable

   if(empty($results)){ //If results is empty

    $errorMsg="Sorry No recipes Exists"; //Set the error message

   }

   else //check no "$errorMsg" show then continue

   {

     ?>

    <h1 class="titles">Recipes</h1> //H1 element with class “titles”

    <table> //Table start

    <thead class='tbl-header'> //Table Headings

      <tr>

        <th>No.</th> //No. heading

        <th>Name</th> //Name

        <th>Author</th> //Author

        <th>More Details</th> //More Details

        <th>Delete</th> //Delete

      </tr>

    </thead> //End of table headings

    <tbody> //Table Body

    <?php

    $c=1; //Set the counter to 1

//The below foreach loops through the results fetched from database

    foreach($results as $row){

      ?>

      <tr> //Table row

<?php

      echo '<td>' . $c++ . '</td>'; //Display and increment counter

      echo '<td>' . $row['name'] . '</td>'; //Display the name of recipe

      echo '<td>' . $row['author'] . '</td>'; //Display Author of Recipe

//Displays Prep Time, Cooking Time, Ratings, Description, Ingredients, Nutrition and Method

      echo '<td>Prep Time: ' . $row['prep'] . '<br>Cook Time: ' . $row['cook'] . '<br>Serves: ' . $row['serves']

      . '<br>Ratings: ' . $row['ratings'] . '<br>Description: ' . $row['description'] . '<br><br>Ingredients: ' .

      $row['ingredients'] . '<br><br>Kcal: ' . $row['kcal'] . '&emsp;Fat: ' . $row['fat'] . 'g&emsp;Saturates: ' . $row['saturates'] .

      'g&emsp;Carbs: ' . $row['carbs'] . 'g&emsp;Sugars: ' . $row['sugars'] . 'g&emsp;Fibre: ' . $row['fibre'] . 'g&emsp;Protein: '

      . $row['protein'] . 'g&emsp;Salts: ' . $row['salt'] . 'g<br><br>Method: ' . $row['method'] . '<br></td>';?>

      <td><a href="delete.php?id=<?php echo $row['id']?>" class="btn btn-danger btn-sm">Delete</a> </td> //Display Delete button and sending the id of recipe to delete.php

      </tr> //End of table row

<?php

    }

    ?>

    </tbody> //End of table body

    </table> //End of table

<?php

    }

   }

  catch(PDOException $e) //Catch any exceptions

  {

   echo $e->getMessage(); //Display the exception messages

  }

?>

</main> //End of main section

        <footer>&copy; CSYM019 2022</footer> //footer section

</div>

</body> //End of body

</html> //End of HTML

**delete.php:**

<?php

require\_once 'connection.php'; //Use connection.php

session\_start(); //start session

 if(!isset($\_SESSION['user\_login'])) //check unauthorize user not access in "home.php" page

 {

  header("location: index.php"); //Redirect if session is not set

 }

if (isset($\_GET['id'])){ //If Recipe ID is set or not null

    $ID = (int)$\_GET['id']; //Get the recipe id, convert to int and store in $ID

    $delete= $db->prepare("DELETE FROM Recipes WHERE id=$ID LIMIT 1"); //SQL query

    $delete->execute(); //Execute the query

    if ($delete){ //If deleted

        echo '<script>alert("Recipe deleted successfully !")</script>'; //Alert that Recipe is deleted

        echo '<script>window.location.href="home.php";</script>'; //Redirect to home.php

    }

}

?>

**Application Testing:**