

# ASSIGNMENT 10

Name: Tarusi Mittal

RollNo: 1901CS65

**Assignment Overview** The basic objective of this assignment is to get familiar with PHP which is a server side scripting language and can be easily embedded into HTML. A web server is required to run any PHP code. Apache server is commonly used as web server

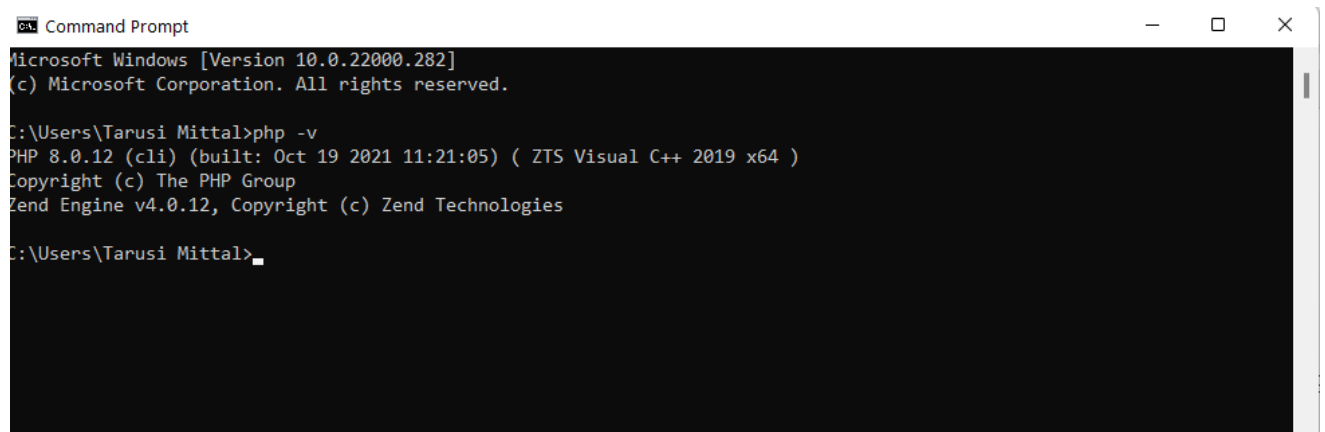
## TASK – A

Installing PHP:

As I have windows system so I downloaded php from the following link:

<https://windows.php.net/download#php-8.0>

and under this I downloaded the thread safe version



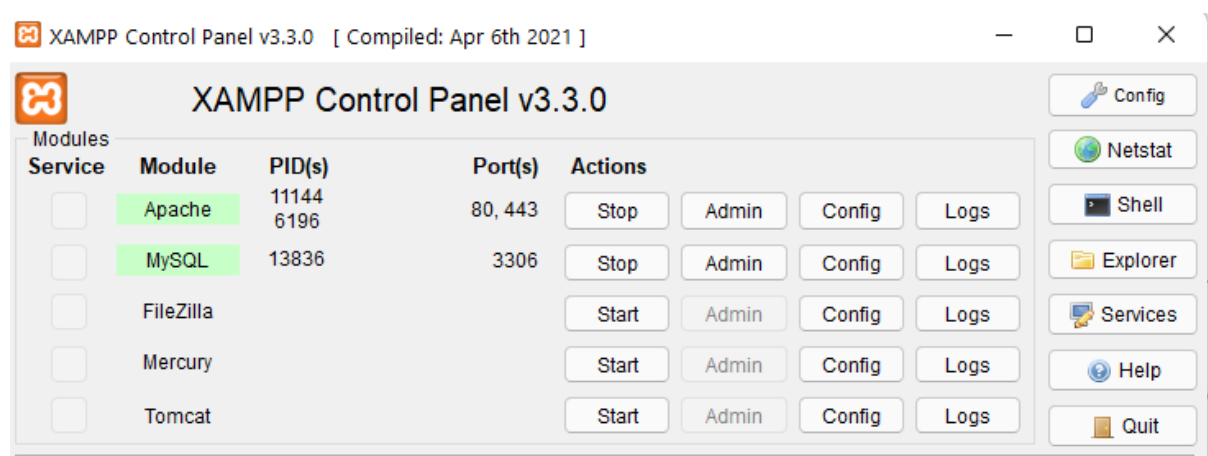
```
Microsoft Windows [Version 10.0.22000.282]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Tarusi Mittal>php -v
PHP 8.0.12 (cli) (built: Oct 19 2021 11:21:05) ( ZTS Visual C++ 2019 x64 )
Copyright (c) The PHP Group
Zend Engine v4.0.12, Copyright (c) Zend Technologies

C:\Users\Tarusi Mittal>
```

For downloading apache I went with Xampp and I downloaded that. Mysql was already downloaded on my system. The link used to download Xampp is

<https://www.apachefriends.org/index.html>



After downloading all the things I proceeded with task B

# TASK – B

As given in the assignment the task was to create a emp table first So for that I opened my localhost/phpMyAdmin

In that under the option of new io created a new **database by the name employee**

Now as given in the question I created an emp table with 8 columns with were as follows:

empID: varchar2(10) primary key

passwd: varchar2(255)

empName: varchar2(20)

DoJ: date

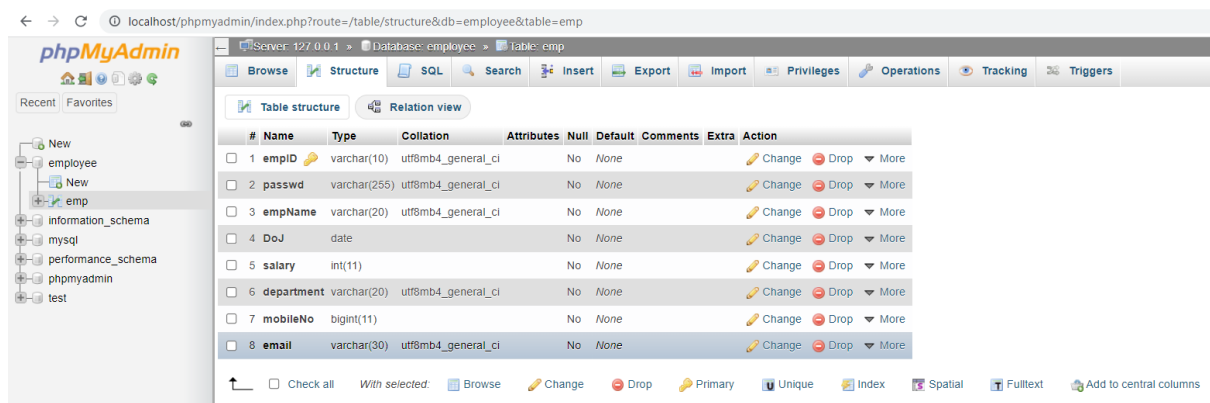
salary: int

department: varchar2(20)

mobileNo: int

email: varchar2(30)

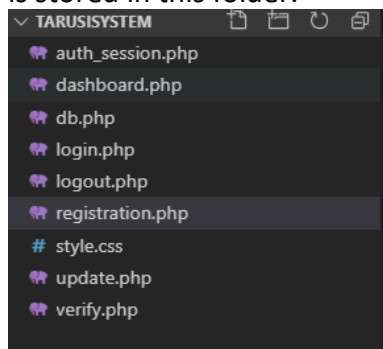
with primary key being the empID and the mobile number has to be taken as big int because of its nature.



#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	empID	varchar(10)	utf8mb4_general_ci		No	None		Primary	Change Drop More
2	passwd	varchar(255)	utf8mb4_general_ci		No	None			Change Drop More
3	empName	varchar(20)	utf8mb4_general_ci		No	None			Change Drop More
4	DoJ	date			No	None			Change Drop More
5	salary	int(11)			No	None			Change Drop More
6	department	varchar(20)	utf8mb4_general_ci		No	None			Change Drop More
7	mobileNo	bigint(11)			No	None		Primary	Change Drop More
8	email	varchar(30)	utf8mb4_general_ci		No	None			Change Drop More

After creating the database table I went to create a web server and link it to the database.

After creating the table In the downloaded xampp which was located on the C drive Inside that under the folder htdocs I made a new folder by the name TarusiSystem. Now every file is stored in this folder.



These are the files that were required for the project and I will be explaining them all one by one in order of their relevance but to give an overview I will start by explain the **db.php**.

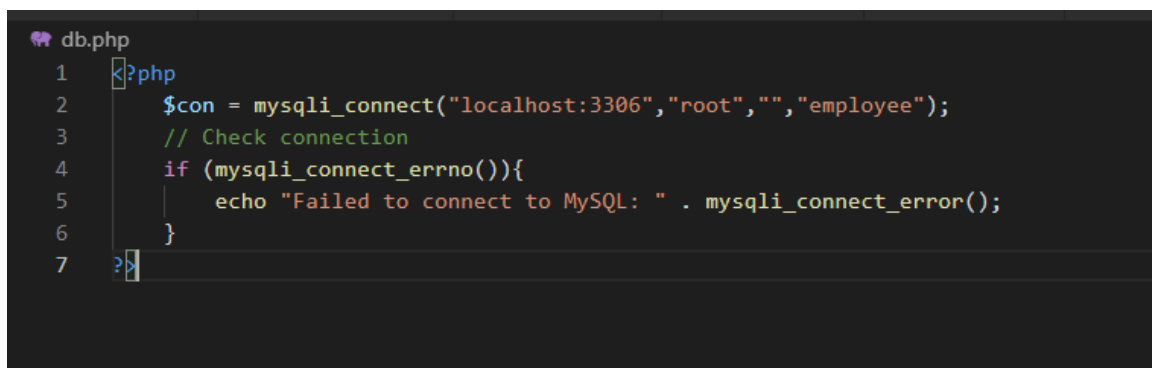
After that the **registration.php** file

Then the login file for which I first need the **auth\_session** file and after the **login.php** file I will move to the **dashboard.php**.

In dashboard I will explain the **logout.php**

After that we will move to the concept of **update.php** for which we first need **verify.php** for the user to verify itself. And in the end the **style.css**

For this the first file I Created was db.php -> The main purpose of this file is to connect to the database that has already being made. Now before this assignment my system got reset so mysql currently don't require password for root.



```
db.php
1 <?php
2 $con = mysqli_connect("localhost:3306","root","", "employee");
3 // Check connection
4 if (mysqli_connect_errno()){
5     echo "Failed to connect to MySQL: " . mysqli_connect_error();
6 }
7
```

This file will ensure that the connection is established whenever it will be included and if there is any error for establishing the error it will throw an error.

After this I created the **registration file**.

The things that are done in the registration page are:

1. We have to take inputs of all the variables that we have in our emp table
2. After taking the inputs we have to ensure that all the fields are injected in our database correctly
3. The password is a sensitive information therefore we need to encrypt it accordingly
4. After all the fields are taken input of and the user clicks register we need to take care of three situations
  - a. IF ALL FIELDS ARE FILLED: In this case we have to register the user successfully
  - b. IF SOME OR ALL FIELDS ARE MISSING: In this case our server should throw an error that Fields are missing and should not let user to proceed
  - c. IF THE EMPLOYEE ID THAT IS BEING CREATED ALREADY EXSITS: In this case also we should get an error that the employee Id already exsits and the user should not be allowed to register.

By taking these things into consideration I made my registration file.

```

39 <form class="form" action="" method="post">
40 <h1 class="login-title">Registration</h1>
41 <label for="empID ">Employee ID</label>
42 <input type="text" class="login-input" name="empID" placeholder="EmpID" required />
43 <label for=" password">Password</label>
44 <input type="password" class="login-input" name="password" placeholder="Password" required>
45 <label for=" empName ">Name</label>
46 <input type="text" class="login-input" name="empName" placeholder="Name" required>
47 <label for=" DoJ ">Date Of Joining</label>
48 <input type="date" class="login-input" name="DoJ" required>
49 <label for=" salary ">Salary</label>
50 <input type="number" class="login-input" name="salary" placeholder="Rs " required>
51 <label for=" department ">Department</label>
52 <input type="text" class="login-input" name="department" placeholder="Department" required>
53 <label for=" mobileNo ">Mobile Number</label>
54 <input type="number" class="login-input" name="mobileNo" placeholder="MobileNo">
55 <label for=" email ">Email</label>
56 <input type="email" class="login-input" name="email" placeholder="Email Adress">
57 <input type="submit" name="submit" value="Register" class="login-button">
58 <p class="link"><a href="login.php">Click to Login</a></p>
59 </form>

```

In these we have written required which will make sure that user has to input them

The above part of the code takes the input in the form.

The label marked shows the wording that appears on top of the input container and here as we can see 8 inputs are being taken as corresponding to the emp table.

```

require('db.php');
// When form submitted, insert values into the database.
if (isset($_REQUEST['empID'])) {
    $empID = $_REQUEST['empID'];
    $password = $_REQUEST['password'];
    $empName = $_REQUEST['empName'];
    $DoJ = $_REQUEST['DoJ'];
    $salary = $_REQUEST['salary'];
    $department = $_REQUEST['department'];
    $mobileNo = $_REQUEST['mobileNo'];
    $email = $_REQUEST['email'];
    $query = "INSERT into `emp` (empID, passwd, empName, DoJ, salary, department, mobileNo, email)
    VALUES ('$empID',md5('$password'), '$empName', '$DoJ', '$salary', '$department', '$mobileNo', '$email')";
    $result = mysqli_query($con, $query);
}

```

Password is encrypted

In the above part of the code first we are including the db.php file.

This ensures that the database is connected to this registration file and we can use the commands to inject the data in our database.

After that we see that empID is provided and as that is our primary key we will fill the whole table with that consideration.

Now according to the table variables we have declared new variables.

And finally by writing the INSERT query we are injecting the information in our database.

In the insert query the md5 before the password ensures that password is encrypted when seen from phpMyAdmin

The result insures that connection is there and query runned is correct.

After this we run a if else condition to check whether we should allow the user to register or should we throw an exception:

```

25         //echo $result;
26         if ($result) {
27             echo "<div class='form'>
28                 <h3>You are registered successfully.</h3><br/>
29                 <p class='link'>Click here to <a href='login.php'>Login</a></p>
30                 </div>";
31         } else {
32             echo "<div class='form'>
33                 <h3>Employee ID already exists.</h3><br/>
34                 <p class='link'>Click here to <a href='registration.php'>registration</a> again.</p>
35                 </div>";
36         }
37     } else {
38         //
39     }

```

This takes care that if the empID is repeated or if any of the fields are missing there should be an error that the employeeid is repeated and the user should not be allowed to register.

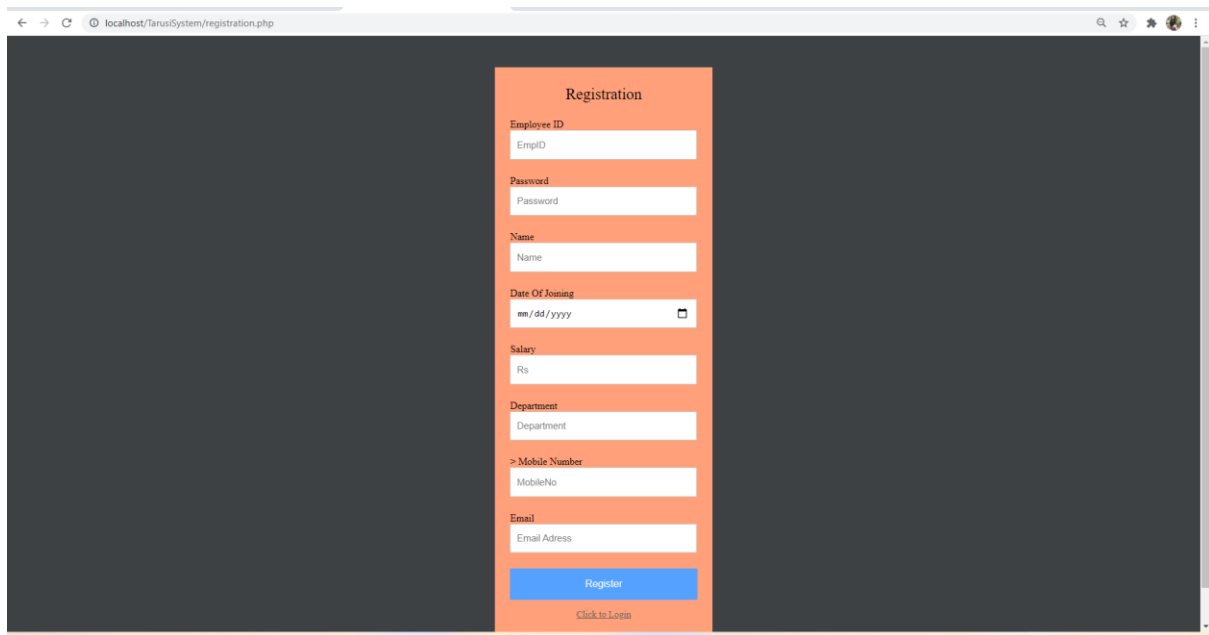
The full code Of registration page is shown below:

```

1  registration.php
2  <!DOCTYPE html>
3  <html>
4  <head>
5      <meta charset="utf-8"/>
6      <title>Registration</title>
7      <link rel="stylesheet" href="style.css"/>
8  </head>
9  <body>
10     <?php
11         require('db.php');
12         // When form submitted, insert values into the database.
13         if (isset($_REQUEST['empID'])) {
14             $empID = $_REQUEST['empID'];
15             $password = $_REQUEST['password'];
16             $empName = $_REQUEST['empName'];
17             $DoJ = $_REQUEST['DoJ'];
18             $salary = $_REQUEST['salary'];
19             $department = $_REQUEST['department'];
20             $mobileNo = $_REQUEST['mobileNo'];
21             $email = $_REQUEST['email'];
22             $query = "INSERT into `emp` (empID, passwd, empName, DoJ, salary, department, mobileNo, email)
23                 VALUES ('$empID','$password','$empName','$DoJ','$salary','$department','$mobileNo','$email')";
24             $result = mysqli_query($con, $query);
25             //echo $result;
26             if ($result) {
27                 echo "<div class='form'>
28                     <h3>You are registered successfully.</h3><br/>
29                     <p class='link'>Click here to <a href='login.php'>Login</a></p>
30                     </div>";
31             } else {
32                 echo "<div class='form'>
33                     <h3>Employee ID already exists.</h3><br/>
34                     <p class='link'>Click here to <a href='registration.php'>registration</a> again.</p>
35                     </div>";
36             }
37         } else {
38             //
39             <form class="form" action="" method="post">
40                 <h1 class="login-title">Registration</h1>
41                 <label for="empID">Employee ID</label>
42                 <input type="text" class="login-input" name="empID" placeholder="EmpID" required />
43                 <label for="password">Password</label>
44                 <input type="password" class="login-input" name="password" placeholder="Password" required>
45                 <label for="empName">Name</label>
46                 <input type="text" class="login-input" name="empName" placeholder="Name" required>
47                 <label for="DoJ">Date Of Joining</label>
48                 <input type="date" class="login-input" name="DoJ" required />
49                 <label for="salary">Salary</label>
50                 <input type="number" class="login-input" name="salary" placeholder="Rs " required>
51                 <label for="department">Department</label>
52                 <input type="text" class="login-input" name="department" placeholder="Department" required>
53                 <label for="mobileNo">Mobile Number</label>
54                 <input type="number" class="login-input" name="mobileNo" placeholder="MobileNo">
55                 <label for="email">Email</label>
56                 <input type="email" class="login-input" name="email" placeholder="Email Address">
57                 <input type="submit" name="submit" value="Register" class="login-button">
58                 <p class="link"><a href="login.php">Click to Login</a></p>
59             </form>
60         <?php
61         }
62     >
63 </body>
64 </html>

```

The following snapshot shows the registration page:

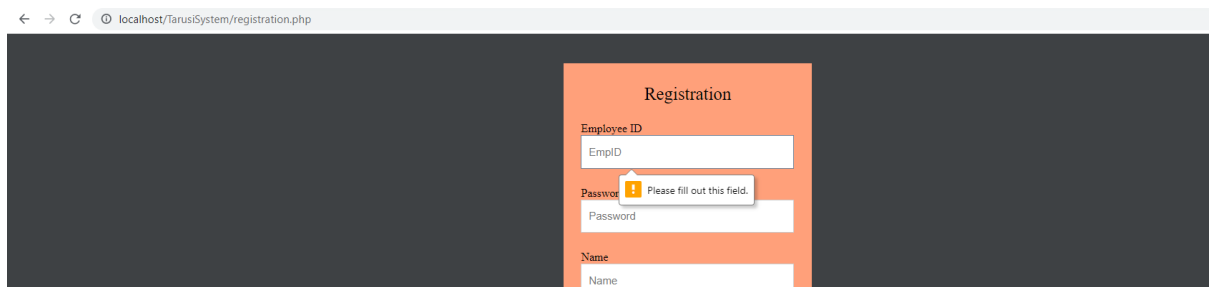


The screenshot shows a web browser window with the URL `localhost/TarusiSystem/registration.php`. The page has a dark grey background. In the center, there is an orange rectangular box containing the registration form. The form is titled "Registration" and includes the following fields: "Employee ID" (with a placeholder "EmpID"), "Password", "Name", "Date Of Joining" (with a date picker icon), "Salary" (with a placeholder "Rs"), "Department", "Mobile Number" (with a placeholder "MobileNo"), and "Email" (with a placeholder "Email Adress"). At the bottom of the orange box is a blue "Register" button and a link that says "Click to Login".

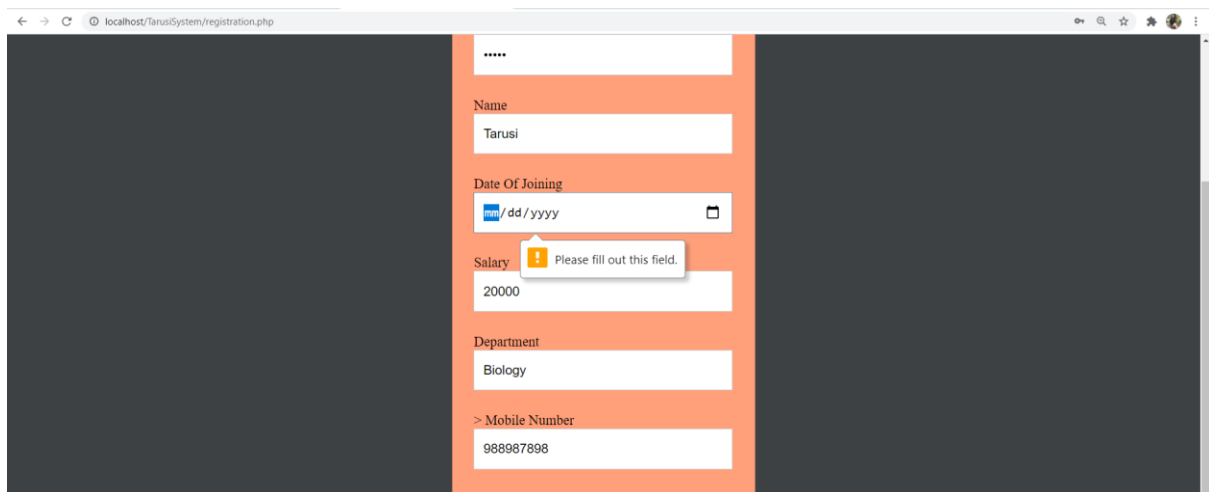
On the registration page on the bottom there is an option to login as well in case the user has already registered and wants to login.

In case of registration failed it shows the following message:

1. This is the case when any field is not filled. If there is any field that is missing it will show that this field is required.



This screenshot shows the registration form with a validation error. A yellow tooltip with an exclamation mark icon and the text "Please fill out this field." is displayed next to the "Password" field. The other fields are empty.



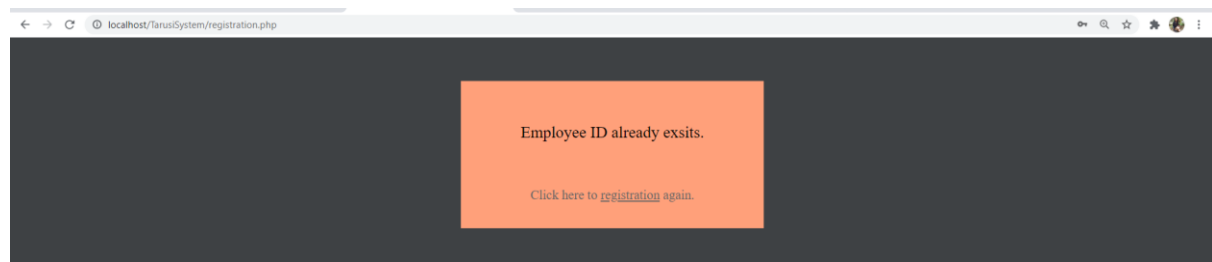
This screenshot shows the registration form with a validation error. A yellow tooltip with an exclamation mark icon and the text "Please fill out this field." is displayed next to the "Salary" field. The "Name" field is filled with "Tarusi", the "Date Of Joining" field has a date picker, the "Department" field is filled with "Biology", and the "Mobile Number" field is filled with "988987898".

But if the mobile No or email address is not filled it will not show an error because we have given the user an option to update it later so that fields can remain empty for user to fill out later or update them

2. Now if the employee id is not unique:

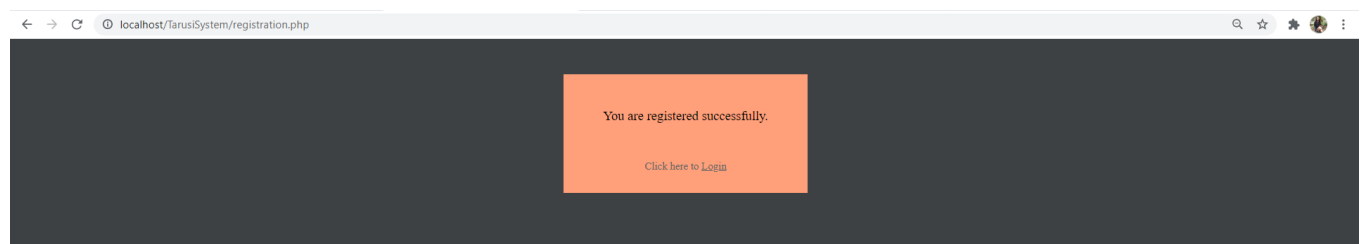
NOTE:THE EMPLOYEE ID IS NOT CASE SENSITIVE EVEN IF SOMEONE WILL TRY TO ENTER THE ID BY WRITING Pixie to pixie it will show id already exists

In this case it shows the button of registration again



3. If the registration is successful it shows:

In this case it shows the button of Login page



As given in the assignment I created 5 successful registrations the emp table looks like this:

Also we can see that the password is encrypted in this.

Showing rows 0 - 4 (5 total, Query took 0.0007 seconds)

```
SELECT * FROM `emp`
```

Options: empID, passwd, empName, DoJ, salary, department, mobileNo, email

	empID	passwd	empName	DoJ	salary	department	mobileNo	email
<input type="checkbox"/>	ashu989	e85d6f02f169df377173b5cc77ae4138	Ashwani Kumar	2021-11-04	200000	Commerce	6789098789	ashwani@gmail.com
<input type="checkbox"/>	Pixie	a9069d07b3714ea1611fa7e377ba7fba	Tarusi Mittal	2021-11-04	200000	Computers	9855664085	tarusimittal@gmail.com
<input type="checkbox"/>	Puja345	85af2647520e645bd90547aa9a98efb6	Puja Goyal	2021-11-03	100000	Chemistry	9876598765	puja@gmail.com
<input type="checkbox"/>	Teja204	f6cbf32f52dc7c592865b90b3eaa912d	Tejas Mittal	2021-11-02	90000	Robotics	9898989898	tejas@gmail.com
<input type="checkbox"/>	Tipi105	e82629f99de37d3beeb197f70816ee83	Tripati	2021-10-06	409700	CSE	9875890809	tripati@gmail.com

Also I am attaching a case where the mobile No and email ID are left empty initially

employee`emp`

New

employee

New

emp

Information\_schema

mysql

performance\_schema

phpmyadmin

test

SELECT \* FROM `emp`

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all

Number of rows: 25

Filter rows: Search this table

Sort by key: None

Options

empID

passwd

empName

DoJ

salary

department

mobileNo

email

<input type="checkbox"/>	Edit	Copy	Delete	ashu989	e85d6f02f169df377173b5cc77ae4138	Ashwani Kumar	2021-11-04	200000	Commerce	6789098789	ashwani@gmail.com
<input type="checkbox"/>	Edit	Copy	Delete	Pixie	a9069d07b3714ea1611fa7e377ba7fba	Tarusi Mittal	2021-11-04	200000	Computers	9781337500	tarusi_1901cs65@iitp.ac.in
<input type="checkbox"/>	Edit	Copy	Delete	Puja345	85af2647520e645bd90547aa9a98efb6	Puja Goyal	2021-11-03	100000	Chemistry	9876598765	puja@gmail.com
<input type="checkbox"/>	Edit	Copy	Delete	Taru1	0250bb838e608ac179e688027dd90c71	Taru	2021-10-31	300200	Physics	0	
<input type="checkbox"/>	Edit	Copy	Delete	Teja204	f6cbf32f52dc7c592865b90b3eaa912d	Tejas Mittal	2021-11-02	90000	Robotics	9898989898	tejas@gmail.com
<input type="checkbox"/>	Edit	Copy	Delete	Tipi105	e82629f99de37d3beeb197f70816ee83	Tripati	2021-10-06	409700	CSE	9875898089	tripati@gmail.com

☐ Check all With selected: Edit Copy Delete Export

Now coming to the [Login Page](#):

In the login page we need to take care of the following things:

1. First we will start a user session in which the user will be identified by its empID.
2. Then we need to make sure that the password matches with the password that was earlier provided by the user
3. If the login is not successful it should throw an error
4. Otherwise it should take the user to its dashboard where all the users info is displayed.

For creating the login page first I had to create the auth\_session page the main purpose of that page is basically to start a session and verify that the empID is there as this page will be taken as required in the login page

```
auth_session.php
1  <?php
2      session_start();
3      if(!isset($_SESSION["empID"])) {
4          header("Location: login.php");
5          exit();
6      }
7  ?>
```

First it will take the inputs from the user as described in the code below:



```

35  >>
36  <form class="form" method="post" name="login">
37      <h1 class="login-title">Login</h1>
38      <input type="text" class="login-input" name="empID" placeholder="empID" autofocus="true"/>
39      <input type="password" class="login-input" name="password" placeholder="Password"/>
40      <input type="submit" value="Login" name="submit" class="login-button"/>
41      <p class="link"><a href="registration.php">New Registration</a></p>
42  </form>
43  <?php
44      }
45  >>
46  </body>
47  </html>

```

Here for login page we only need the emp Id and the password which will be provided by the user

After that it will extract the information of the empID that is provided by the user. The code for that is shown below:

```

10  require('db.php');
11  session_start();
12  // When form submitted, check and create user session.
13  if (isset($_REQUEST['empID'])) {
14      $empID = stripslashes($_REQUEST['empID']); // removes backslashes
15      $empID = mysqli_real_escape_string($con, $empID);
16      $password = stripslashes($_REQUEST['password']);
17      $password = mysqli_real_escape_string($con, $password);
18      // Check user is exist in the database
19      $query = "SELECT * FROM `emp` WHERE empID='$empID'";
20      $result = mysqli_query($con, $query);
21      $rows = mysqli_fetch_assoc($result);

```

In the query we have selected all things of the empId and stored it in the result now we will match the password with the password that the user has provided this time:

```

22      if (md5($password) === $rows['passwd']) {
23          $_SESSION['empID'] = $empID;
24          $_SESSION['mobileNo'] = $rows['mobileNo'];
25          // Redirect to user dashboard page
26          header("Location: dashboard.php");
27          //echo "login successfull";
28      } else {
29          echo "<div class='form'>
30              <h3>Incorrect Username/password.</h3><br/>
31              <p class='link'>Click here to <a href='login.php'>Login</a> again.</p>
32              </div>";
33      }
34  } else {

```

Now if the password matches it will take the user to the login page else it will throw an error.

The full code for login page is shown below:

```

login.php
2  <html>
3  <head>
4      <meta charset="utf-8"/>
5      <title>Login</title>
6      <link rel="stylesheet" href="style.css"/>
7  </head>
8  <body>
9  <?php
10     require('db.php');
11     session_start();
12     // When form submitted, check and create user session.
13     if (isset($_REQUEST['empID'])) {
14         $empID = stripslashes($_REQUEST['empID']); // removes backslashes
15         $empID = mysqli_real_escape_string($con, $empID);
16         $password = stripslashes($_REQUEST['password']);
17         $password = mysqli_real_escape_string($con, $password);
18         // Check user is exist in the database
19         $query = "SELECT * FROM `emp` WHERE empID='$empID'";
20         $result = mysqli_query($con, $query);
21         $rows = mysqli_fetch_assoc($result);
22         if (md5($password) == $rows['password']) {
23             $_SESSION['empID'] = $empID;
24             $_SESSION['mobileNo'] = $rows['mobileNo'];
25             // Redirect to user dashboard page
26             header("Location: dashboard.php");
27             //echo "login successful";
28         } else {
29             echo "<div class='form'>
30                 <h3>Incorrect Username/password.</h3><br/>
31                 <p class='link'>Click here to <a href='login.php'>Login</a> again.</p>
32             </div>";
33         }
34     } else {
35     }
36     <?php
37     <form class="form" method="post" name="login">
38         <h1 class="login-title">Login</h1>
39         <input type="text" class="login-input" name="empID" placeholder="empID" autofocus="true"/>
40         <input type="password" class="login-input" name="password" placeholder="Password"/>
41         <input type="submit" value="Login" name="submit" class="login-button"/>
42         <p class="link"><a href="registration.php">New Registration</a></p>
43     </form>
44 <?php
45 }
46 </body>
47 </html>

```

The login page looks like this:

The screenshot shows a web browser window with the URL 'localhost/TarusiSystem/login.php'. The page displays a login form with the following elements:

- A title 'Login' at the top of the form box.
- An input field for 'empID'.
- An input field for 'Password'.
- A blue button labeled 'Login'.
- A link labeled 'New Registration' at the bottom of the form box.

At the end of the login page there is a button for new registration which basically takes the user back to the registration page in case the user wants to do a new registration.

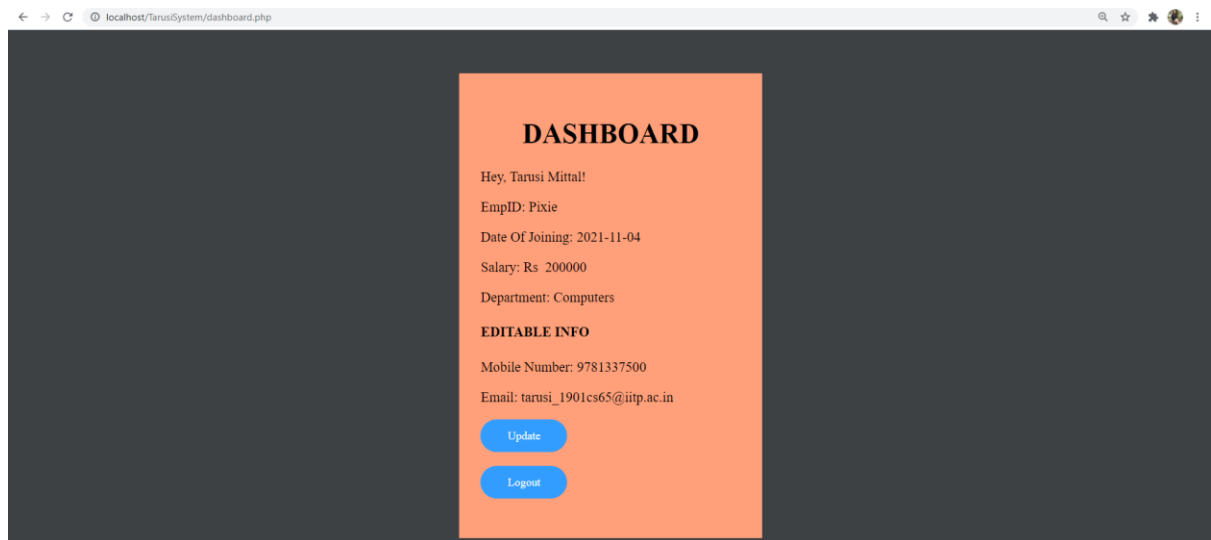
If login is unsuccessful it shows like this:

The screenshot shows the same web browser window, but the login form now displays an error message:

- The text 'Incorrect Username/password.' is shown in the center of the form box.
- Below the error message is a link that says 'Click here to Login again.'

And we can directly go to login again by clicking the login button

If login is successful it takes to the dashboard which looks like this:



For making the dashboard the following things were done:

```
dashboard.php
1  <?php
2  //include auth_session.php file on all user panel pages
3  include("auth_session.php");
4  require('db.php');
5  $var1 = $_SESSION['empID'];
6  $sql = "SELECT * from `emp` where empID= '$var1' " ;
7  $query = mysqli_query($con, $sql);
8  $row = mysqli_fetch_assoc($query);
9  ?>
10 <!DOCTYPE html>
11 <html>
12 <head>
13     <meta charset="utf-8">
14     <title>Dashboard - Client area</title>
15     <link rel="stylesheet" href="style.css" />
16 </head>
17 <body>
18     <div class="form">
19         <h1> <center> DASHBOARD </center> </h1>
20         <p>Hey, <?php echo $row['empName']; ?>!</p>
21         <p>EmpID: <?php echo $row['empID']; ?></p>
22         <p>Date Of Joining: <?php echo $row['DoJ']; ?></p>
23         <p>Salary: Rs&nbsp;<?php echo $row['salary']; ?></p>
24         <p>Department: <?php echo $row['department']; ?></p>
25         <h4> EDITABLE INFO </h4>
26         <p>Mobile Number: <?php echo $row['mobileNo']; ?></p>
27         <p>Email: <?php echo $row['email']; ?></p>
28         <p><a href="verify.php" class="login-but">Update</a></p>
29         <p><a href="logout.php" class="login-but">Logout</a></p>
30     </div>
31 </body>
32 </html>
```

The session which was started in auth\_session was continued and all the information related to the provided empId was extracted and displayed on the dashboard except the password.

To extract the information first a variable had to be created which I named var1. After that that variable was used to connect to the empID which was there in auth\_session and then from that variable all the information was extracted using fetch and stored in new variable called row.

Then while displaying the information the variable row was used.

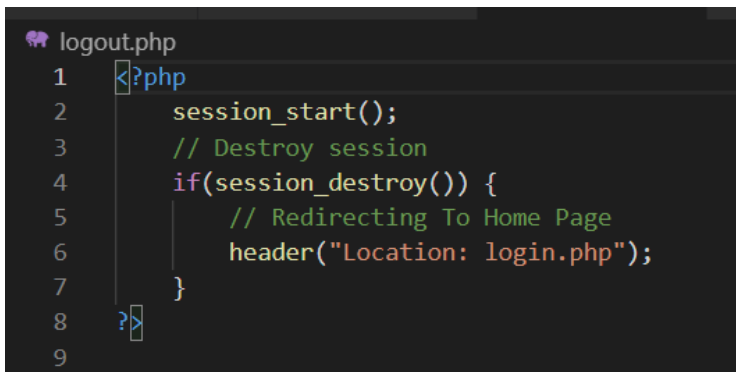
To understand this we can think that row is like an array which has now stored the information according to the index which are nothing but the fields of the emp table. So now when we will call row[index] it will give us the information corresponding to that empID which was there in login (provided by the auth\_session).

Now after displaying the information, we need to have two buttons

1. Which will take us to the update page where we can update our mobile no and email address
2. The logout page from where we can go outside the dashboard

### THE LOGOUT PAGE:

If we click on the logout button it takes us back to the login page again and the code for that is shown below:



```
1 <?php
2     session_start();
3     // Destroy session
4     if(session_destroy()) {
5         // Redirecting To Home Page
6         header("Location: login.php");
7     }
8 ?>
```

### THE UPDATE PAGE:

For the update page I have first made a verify page where basically I will be asking the user to verify his empID and password, If the details will be correct it will move the user to the update page otherwise it will throw an error.

The verify page is almost like the login page except for some changes that the session started here is new

```

verify.php
1 <!DOCTYPE html>
2 <html>
3 <head>
4     <meta charset="utf-8"/>
5     <title>Verify</title>
6     <link rel="stylesheet" href="style.css"/>
7 </head>
8 <body>
9 <?php
10     require('db.php');
11     session_start();
12     // When form submitted, check and create user session.
13     if (isset($_REQUEST['empID'])) {
14         $empID = stripslashes($_REQUEST['empID']); // removes backslashes
15         $empID = mysqli_real_escape_string($con, $empID);
16         $password = stripslashes($_REQUEST['password']);
17         $password = mysqli_real_escape_string($con, $password);
18         // Check user is exist in the database
19         $query = "SELECT passwd FROM `emp` WHERE empID='$empID'";
20         $result = mysqli_query($con, $query);
21         $rows = mysqli_fetch_assoc($result);
22         if (md5($password) == $rows['passwd']) {
23             $_SESSION['empID'] = $empID;
24             // Redirect to user update page
25             header("Location: update.php");
26             //echo "login successfull";
27         } else {
28             echo "<div class='form'>
29                 <h3>Incorrect Username/password.</h3><br/>
30                 <p class='link'>Click here to <a href='login.php'>Login</a> again.</p>
31             </div>";
32         }
33     } else {
34     ?>
35     <form class="form" method="post" name="login">
36         <h1 class="login-title">Verify</h1>
37         <input type="text" class="login-input" name="empID" placeholder="empID" autofocus="true"/>
38         <input type="password" class="login-input" name="password" placeholder="Password"/>
39         <input type="submit" value="Verify" name="submit" class="login-button"/>
40     </form>
41 <?php
42     }
43 ?>

```

The verify page looks like:

Only when the user verifies itself it will be directed to the update page otherwise it will be thrown to the logon page.

In the update page the empID will be taken from the session of verify so as to make sure that the user is able to update its information.

The code for the update page is as below:

```

36     }
37     } else {
38     ?>
39     <form class="form" action="" method="post">
40     <h1 class="login-title">UPDATE</h1>
41     <input type="number" class="login-input" name="mobileNo" placeholder="MobileNo">
42     <input type="email" class="login-input" name="email" placeholder="Email Address">
43     <input type="submit" name="submit" value="UPDATE" class="login-button">
44     <p class="link"><a href="dashboard.php">Back</a></p>
45     </form>
46     <?php
47     }
48     ?>
49     </body>
50     </html>

```

In this it will take the input from the user about the Mobile Number and the Email Address

The next part of the code makes sure of the following things:

1. The empID is same as of verify session so that correct user's info is updated
2. None of the fields the Mobile No or the Email is empty
3. The information provided is updated in the database under that empID .

The code for that is:

```

10     require('db.php');
11     session_start();
12     // When form submitted, insert values into the database.
13     if (isset($_REQUEST['mobileNo'])) {
14
15         $empID = $_SESSION['empID'];
16         $mobileNo = $_REQUEST['mobileNo'];
17         $email = $_REQUEST['email'];
18         if($email==NULL){
19             echo "<div class='form'>
20                 <h3>Required fields are missing.</h3><br/>
21                 </div>";
22         } else{
23             $query = "UPDATE `emp` SET mobileNo = $mobileNo, email = '$email' where empID='$empID'";
24
25             $result = mysqli_query($con, $query);
26             echo $result;
27             if ($result) {
28                 header("Location: dashboard.php");
29             } else {
30                 //echo mysqli_error($con);
31                 echo "<div class='form'>
32                     <h3>Required fields are missing.</h3><br/>
33                     </div>";
34
35             }
36         }
37     } else {

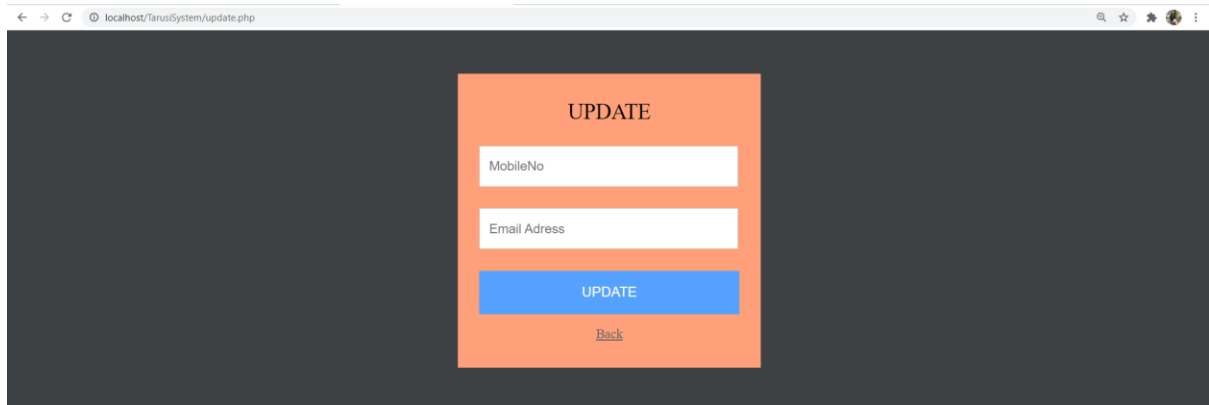
```

Here we can see firstly we have included the db.php so that anything we change is reflected in our database.

Then it first checks if Mobile Number is provided by the user Otherwise it will show that fields are missing. If mobile number is provided it check if email is NULL or not. If email is

NULL it will show that fields are missing otherwise if both fields are there it will show that update is successful and the database will be updated. Also in the update page there is an option if the user wants to go back to the dashboard.

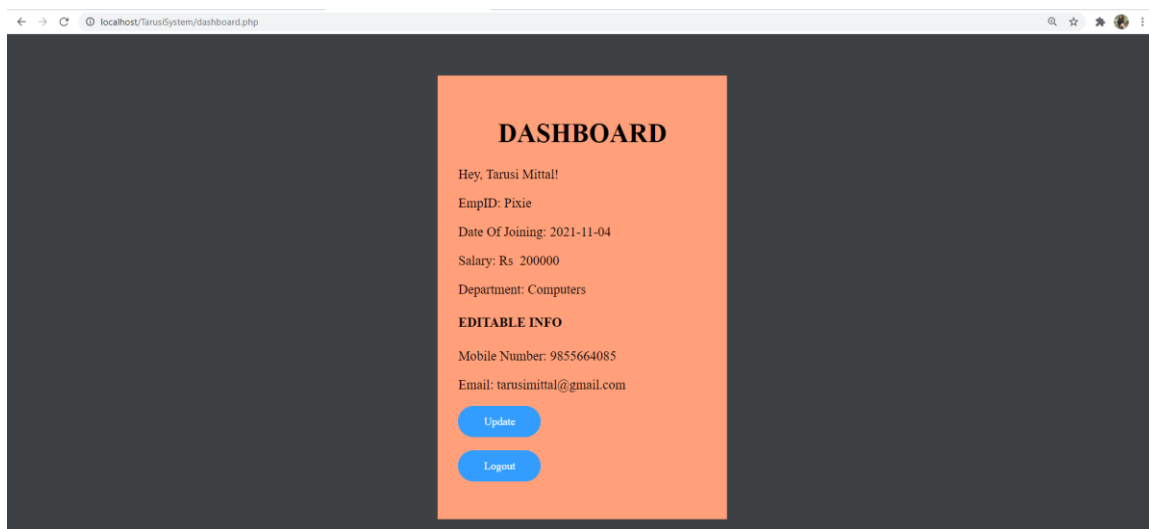
The update page looks like:



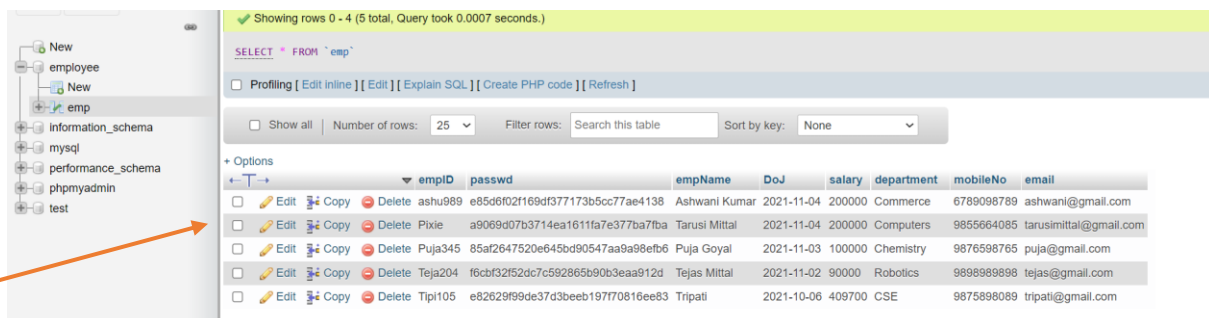
If we update a user the changes are reflected on the dashboard also and in the database as well which can be seen from the examples:

**Before update:**

The DashBoard



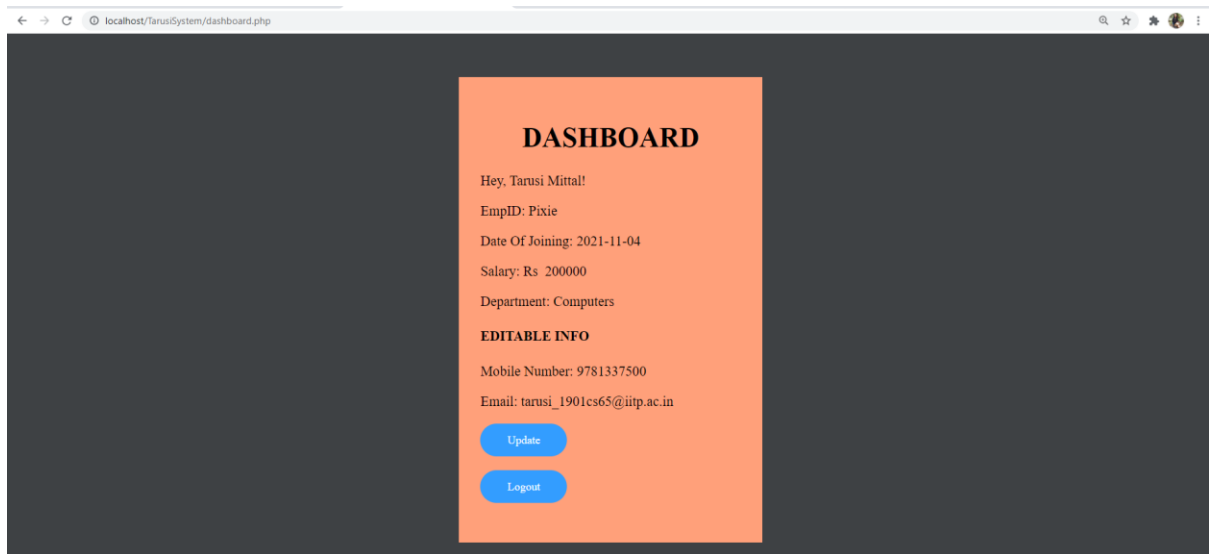
The database:



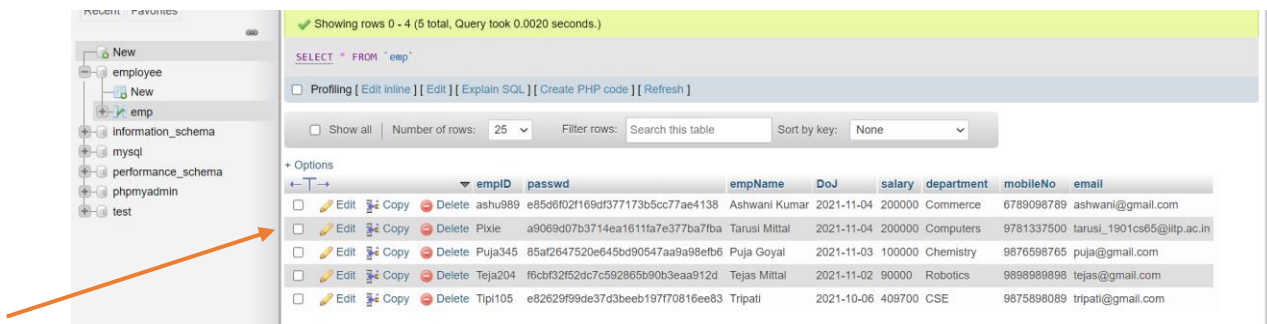
	empID	passwd	empName	DoJ	salary	department	mobileNo	email
<input type="checkbox"/>	ashu989	e85d6f02169df377173b5cc77ae4138	Ashwani Kumar	2021-11-04	200000	Commerce	6789098789	ashwani@gmail.com
<input type="checkbox"/>	Pixie	a9069d07b3714ea1611fa7e377ba7fba	Tarusi Mittal	2021-11-04	200000	Computers	9855664085	tarusimittal@gmail.com
<input type="checkbox"/>	Puja345	85af2647520e645bd90547aa9a98efb6	Puja Goyal	2021-11-03	100000	Chemistry	9876598765	puja@gmail.com
<input type="checkbox"/>	Teja204	f6cbf32f52dc7c592865b90b3eaa912d	Tejas Mittal	2021-11-02	90000	Robotics	9898989898	tejas@gmail.com
<input type="checkbox"/>	Tipi105	e82629f99de37d3beeb197170816ee83	Tripathi	2021-10-06	409700	CSE	9875898089	tripati@gmail.com

## After Update:

## The Dashboard:



## The Database



Just in case the user provides leaves the any field of the update page empty the page shows:





This was about the php code and the database but to make the webpages I required CSS also and the code file for that is shown below:

```
# style.css > body
1
2  ∨ body {
3
4      background: #3e4144;
5
6  }
7  ∨ .form {
8      margin: 50px auto;
9      width: 300px;
10     padding: 30px 25px;
11     background: #FFA07A;
12 }
13 ∨ h1.login-title {
14     color: rgb(0, 0, 0);
15     margin: 0px auto 25px;
16     font-size: 25px;
17     font-weight: 300;
18     text-align: center;
19 }
20 ∨ .login-input {
21     font-size: 15px;
22     border: 1px solid #ccc;
23     padding: 10px;
24     margin-bottom: 25px;
25     height: 25px;
26     width: calc(100% - 23px);
27 }
28 ∨ .login-input:focus {
29     border-color: #6e8095;
30     outline: none;
31 }
32 ∨ .login-button {
33     color: #fff;
34     background: #55a1ff;
35     border: 0;
```

```

# style.css > body
35     border: 0;
36     outline: 0;
37     width: 100%;
38     height: 50px;
39     font-size: 16px;
40     text-align: center;
41     cursor: pointer;
42 }
43 .login-but {
44     display: inline-block;
45     margin-right: 10px;
46     background: #339DFF;
47     color: #fff;
48     text-decoration: none;
49     font-size: 13px;
50     line-height: 38px;
51     border-radius: 50px;
52     -webkit-transition: all 0.3s;
53     transition: all 0.3s;
54     width: 100px;
55     text-align: center;
56 }
57 .link {
58     color: #666;
59     font-size: 15px;
60     text-align: center;
61     margin-bottom: 0px;
62 }
63 .link a {
64     color: #666;
65 }
66 h3 {
67     font-weight: normal;
68     text-align: center;
69 }

```

This covers my assignment which included the

1. Registration Page
2. Login Page
3. Update Page
4. Database Connection
5. Errors prompts wherever check was needed
6. A CSS to design it

END