# PRACTICAL SET 4 – PHP & MYSQL

**[CO4] Write the server side scripts for designing web-based services with database connectivity.**

## AIM 1: Extend Practical set 3(4) by applying insert operation and display the student records with update and delete facility.

- **Theory:**

Before we can access data in the MySQL database, we need to be able to connect to the server connection object.

After a database and a table have been created, we can start adding data in them.

Here are some syntax rules to follow:

- The SQL query must be quoted in PHP
- String values inside the SQL query must be quoted
- Numeric values must not be quoted
- The word NULL must not be quoted

The INSERT INTO statement is used to add new records to a MySQL table:

INSERT INTO table_name (column1, column2, column3,...)
VALUES (value1, value2, value3,...)

The UPDATE statement is used to update existing records in a table:

UPDATE table_name
SET column1=value, column2=value2,...
WHERE some_column=some_value

The DELETE statement is used to delete records from a table:

DELETE FROM table_name
WHERE some_column = some_value.

- **Program:**

## 1. Connection.php

```php
<?php
// Enter your Host, username, password, database below.
// I left password empty because i do not set password on localhost.
$con=mysqli_connect('localhost','root','','myprograming') or die("connection
failed : ".mysqli_connect_error());
if ($con) {
// echo "Connection Successfully";
}
else{
echo "Sorry Some Mistakes is";
}
// Check connection
if (mysqli_connect_errno())
{
echo "Failed to connect to MySQL: " . mysqli_connect_error();
}
?>
```

## 2. Index.php

```html
<!DOCTYPE html>
<html>
<head>
<title>insert</title>
</head>
<body>
<div class="row text-center">
<div class="container">
<h1>Insert DATA</h1>
<form action="index.php" method="post">
<input type="text" name="firstname" placeholder="firstname"><br><br>
<input type="text" name="lastname" placeholder="lastname"><br><br>
<input type="gmail" name="gmail" placeholder="gmail"><br><br>
<input type="text" name="number" placeholder="number"><br><br>
<input type="text" name="address" placeholder="address"><br><br>

<input type="submit" name="submit" value="insert"><br><br>

</form>
```

```
<button><a href="show.php">show data</a></button>

    </div>
```

```
</div>
</body>
</html>
<?php
error_reporting(0);
include 'connection.php';
if (isset($_POST['submit'])) {
$firstname = $_POST['firstname'];
$lastname = $_POST['lastname'];
$gmail = $_POST['gmail'];
$number = $_POST['number'];
$address = $_POST['address'];
$sql = "INSERT INTO `reg` VALUES
('$id','$firstname','$lastname','$gmail','$number','$address')";
$data=mysqli_query($con,$sql);
if ($data) {
echo "insert";
}else
{   echo "sorry";  } }
?>
```

## 3. Show.php

```
<!DOCTYPE html>
<html>
<head>
<title>show table</title>
</head>
<body>
<?php
include ('connection.php');
$sql ="select * from reg";
$data =mysqli_query($con,$sql);
$total=mysqli_num_rows($data);
if ($total=mysqli_num_rows($data)) {
```

```php
?>
<table border="2">
<tr>    <th>id</th>
<th>firstname</th>
<th>lastname</th>
<th>gmail</th>
 <th>number</th>
<th>address</th>
<th>delete</th>
<th>update</th>
</tr>
<?php
while ($result = mysqli_fetch_array($data)) {
echo "
<tr>
<td>".$result['id']."</td>
<td>".$result['firstname']."</td>
<td>".$result['lastname']."</td>
<td>".$result['gmail']."</td>
<td>".$result['number']."</td>
<td>".$result['address']."</td>
<td><a href='update.php?id=$result[id] &
firstname=$result[firstname] & lastname=$result[lastname] &
gmail=$result[gmail]
& number=$result[number] &address=$result[address]'> update </a></td>
<td><a href='delete.php?id=$result[id] '>delete </a></td>
</tr>
";
}
}
else
{
echo "no record found";
}
?>
</table>
</body>
```

</html>
<!--------- echo "<br>".$result['id']." ".$result['firstname']."
".$result['lastname']."
".$result['gmail']." ".$result['number']." ".$result['address']."<br>";_----->

# 4. Update.php

```
<!DOCTYPE html>
<html>
<head>
<title>update</title>
</head>
<body>
<form action="" method="get">
<input type="text" name="id" placeholder="id" value="<?php echo
$_GET['id'];
?>"><br><br>
<input type="text" name="firstname" placeholder="firstname"
value="<?php
echo $_GET['firstname']; ?>"><br><br>
<input type="text" name="lastname" placeholder="lastname"
value="<?php echo
$_GET['lastname']; ?>" ><br><br>
<input type="gmail" name="gmail" placeholder="gmail" value="<?php
echo
$_GET['gmail']; ?>"><br><br>
<input type="text" name="number" placeholder="number" value="<?php
echo
$_GET['number']; ?>"><br><br>
<input type="text" name="address" placeholder="address" value="<?php
echo
$_GET['address']; ?>"><br><br>
<input type="submit" name="submit" value="update">
</form>
<?php
error_reporting(0);
```

```php
include ('connection.php');
if ($_GET['submit'])
{
$id = $_GET['id'];
$firstname = $_GET['firstname'];
$lastname = $_GET['lastname'];
$gmail = $_GET['gmail'];
$number = $_GET['number'];
$address = $_GET['address'];
$sql="UPDATE reg SET firstname='$firstname' , lastname='$lastname',
gmail='$gmail' , number='$number', address='$address' WHERE id='$id'";
$data=mysqli_query($con, $sql);
if ($data) {
//echo "record update";
header('location:show.php');  }
else{   echo "not update";  }
}else
{   echo "click on button to save the change";  }
?>
</body>
</html>
```

# 5. Delete.php

```php
<?php
include ('connection.php');
$id = $_GET['id'];
$sql ="DELETE FROM `reg` WHERE id='$id'";
$data = mysqli_query($con,$sql);
if ($data) {
echo "deleted";
header('location:show.php');
}else
{
echo "error";
} ?>
```

- **Output**



| ID | FirstName | LastName | Number | Email | Delete | Update |
|----|-----------|----------|----------|------------------|--------|--------|
| 1 | Parth | Vaghela | 1146846 | parth@ldce.com | Delete | Update |
| 2 | Veer | Vaghela | 46848466 | veer@ldce.com | Delete | Update |
| 3 | Dhruv | Patel | 165468 | dhruv@ldce.com | Delete | Update |
| 4 | karan | saravaiya | 5465466 | karan@gamail.com | Delete | Update |

## Update

**ID : 4**

**First Name**

Deep

**Last Name**

sarvaiya

**Phone No :**

513546

**City**

gandhinagar

**Address**

Gandhi Nagar GUjrat

UPDATE

| ID | FirstName | LastName | Number | Email | Delete | Update |
|----|-----------|----------|--------|-------|--------|--------|
| 1 | Parth | Vaghela | 1146846 | parth@ldce.com | Delete | Update |
| 2 | Veer | Vaghela | 46848466 | veer@ldce.com | Delete | Update |
| 3 | Dhruv | Patel | 165468 | dhruv@ldce.com | Delete | Update |
| 4 | Deep | saravaiya | 513546 | karan@gamail.com | Delete | Update |

**Signature of Faculty:**                                                              **Grade:**

**AIM 2: The user is first served a login page which takes users name and password. After submitting the details, the server checks these values against the data from a database and takes the following decisions**

    a. **If name and password match, serves a welcome page with user's full name.**

    b. **If name matches and password does not match, then serves password mismatch page.**

    c. **If name not found in database, serves a registration page where users full name is asked and on submitting the full name, it stores, the login name, password and full name in the database (HINT: Use session for storing the submitted login name and password.)**

## Theory:

A session is a way to store information (in variables) to be used across multiple pages.

Unlike a cookie, the information is not stored on the user's computer.

**What Is Session??**

When you work with an application, you open it, do some changes, and then you close it. This is much like a Session. The computer knows who you are. It knows when you start the application and when you end. But on the internet there is one problem: the web server does not know who you are or what you do, because the HTTP address doesn't maintain state.

Session variables solve this problem by storing user information to be used across multiple pages (e.g. username, favourite colour, etc.). By default, session variables last until the user closes the browser.

So; Session variables hold information about one single user, and are available to all pages in one application.

**Start a PHP Session**

A session is started with the session_start() function.

Session variables are set with the PHP global variable: $_SESSION.

**Get PHP Session Variable Values**

Notice that session variables are not passed individually to each new page, instead they are retrieved from the session we open at the beginning of each page (session_start()).

Also notice that all session variable values are stored in the global $_SESSION variable.

- **Program:**

## 1. Form.html

```
<form action="home.php" method="post">
Login<br />
User name: <input type="text" name="user" /><br />
Password: <input type="password" name="pass" /><br />
<input type="submit" value="Login" />
</form>
```

# 2. Home.php

```php
<?php
session_start();
$user = $_POST['user'];
$pass = $_POST['pass'];
$_SESSION['luser'] = $user;
$_SESSION['lpass'] = $pass;
$database = "users";
$tablename = "fulluser";
if ($user&&$pass)
{
$connect=mysqli_connect('localhost','root','','users') or die("connection
failed : ".mysqli_connect_error());
$sql = "SELECT * FROM $tablename WHERE username='$user' and
password='$pass'";
$query = mysqli_query($connect,$sql);
$numrows = mysqli_num_rows($query);
if ($numrows!=0)
{
while ($row = mysqli_fetch_assoc($query))
{
print "Hello " . $row['firstname'] . " " . $row['middlename'] . " " .
$row['lastname'];}
}
$query = mysqli_query("SELECT * FROM $tablename WHERE
username='$user' and
```

```
password!='$pass'");
$numrows = mysqli_num_rows($query);
if ($numrows!=0)
{
print "Password is wrong";
}
$query = mysqli_query("SELECT * FROM $tablename WHERE
username='$user'");
$numrows = mysqli_num_rows($query);
if ($numrows==0)
{
header("Location:C:\xampp\htdocs\php_practical\practical_7\reg.php");}
}
else
die("please enter a username and password!");
?>
```

# 3. Reg.php

```
<?php
session_start();
?>
<form action="insert.php" method="post">
Enter First Name<input type="text" name="fn"><br>
Enter Middle Name <input type="text" name="mn"><br>
Enter Last Name <input type="text" name="ln"><br>
<input type="submit" value=" Register ">
</form>
```

# 4. Insert.php

```
<?php
session_start();
$user = $_SESSION['luser'];
$pass = $_SESSION['lpass'];
$server = "localhost";
```

```
$username = "root";
$password = "mysql";
$database = "users";
$tablename = "fulluser";
$firstname = $_POST['fn'];
$middlename = $_POST['mn'];
$lastname = $_POST['ln'];
$connect = mysql_connect($server,$username,$password) or die("not
connecting");
mysql_select_db($database) or die("no db :");
mysql_query("insert into $tablename
values('$user','$pass','$firstname','$middlename','$lastname')");
 print "Registered successfully";
session_destroy();
?>
```
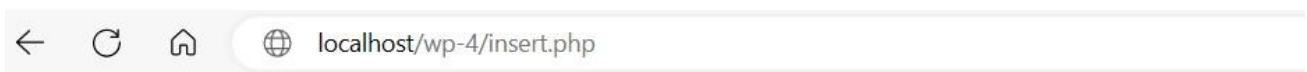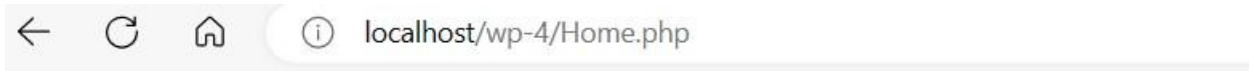
- **Output**

localhost/wp-4/Home.php

Enter First Name PARTH
Enter Middle Name PANKAJ
Enter Last Name VAGHELA

Register

←  C  ⌂  ⓘ  localhost/wp-4/Home.php

# Password is wrong

←  C  ⌂  ⓘ  127.0.0.1:3000/insert.html

# Login
## User name: PARTH
## Password: •••••  ◉

[ Login ]

←  C  ⌂  ⊕  localhost/wp-4/insert.php

# Registered successfully

**Signature of Faculty:**                                          **Grade:**

**AIM 3: Write a program that keeps track of how many times a visitor has loaded the page.**

**Theory:**

A **session** mechanism can be used to store page views which increment on each refresh and show the count on a webpage. A session is user specific and for every user, a separate session is created along with a separate session variable which is associated with that session. Using this mechanism, for every user the session variable is set to 1 initially for the first visit. On consecutive visits, the value of this session variable is incremented and displayed on the output webpage.

- **Program:**

```php
<html>
<body>
<h1>Page loaded <?php
session_start();
if(isset($_SESSION['visit']))
{
$_SESSION['visit']+=1;
}
else
{
$_SESSION['visit']=1;
}
echo $_SESSION['visit'];
?> times</h1>
</body>
</html>
```

- **Output :**

```
Page Load Frequencies : 4
```

**Signature of Faculty:**                                                                 **Grade:**