**Program 7**

**PART-A**

**A.1 Aim:** Implement factorial,fib,Armstrong,

Implement tic toe game(use random)

Form Processing with PHP

**A.2 Prerequisite:** HTML, CSS, Javascript, Basic programming language

**A.3 Outcome:**

After successful completion of this experiment students will be able to

1. Understand basic PHP programming

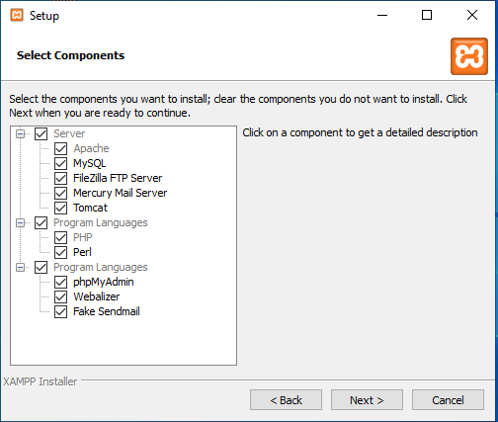
2. Understand and implement PHP as server side code for form processing

**A.4 Theory:**

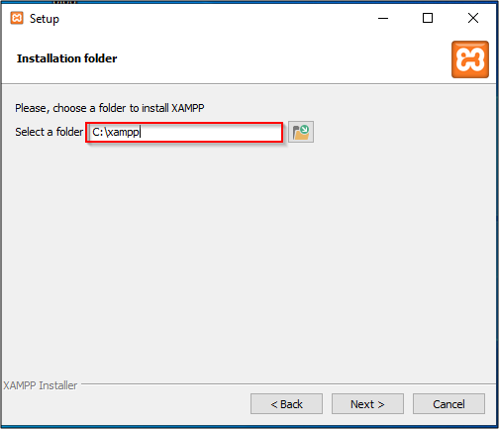
1. Xampp is the most popular PHP development environment for Windows, OS X, and Linux platforms. Xampp stands for Cross platform(x), Apache(a), Maria db(m), PHP(p), Pearl(p) which is a software distribution server which makes developer’s work eaiser for testing and deploying by creating a local web server.

## ****How to install Xampp?****

It is completely free and easy to install Apache distribution containing MySQL, PHP, and Perl. First, download XAMP from <https://www.apachefriends.org/download.html>. In the first page, select the components you want to install.

1. 

Select the installation directory so that all the components that you choose will be installed in this directory.

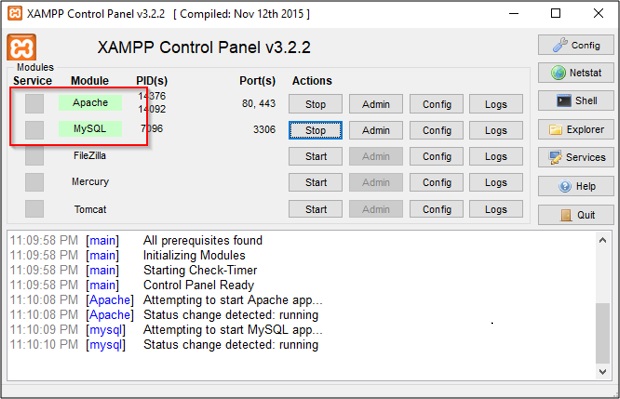


## ****How to run PHP program in Xampp step-by-step?****

* Write php program in a notepad/sublime text and save it as file.php or any other name.

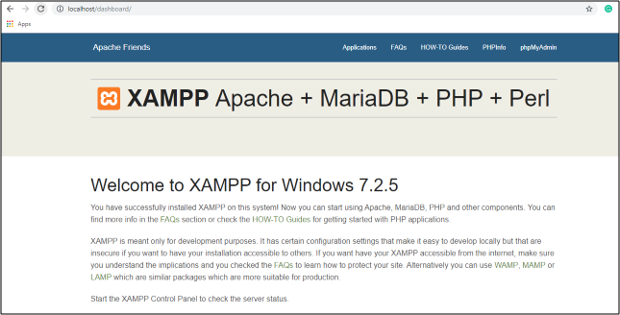
After completion of the installation, you can use the XAMPP Control Panel to start/ stop all servers.

* Start Mysql and Apache servers.



* Copy file.php to htdocs (C:/Program Files/XAMPP/htdocs)
* You can also create any folders inside htdocs folder and save our codes over there.

In order to get the dashboard for localhost: search [http://localhost](http://localhost/) in any browser.



* Now to run your code, open browser and type localhost/file.php then it gets executed.

2. You need to work with basic html form. Under form action write name of the **php file** that generates response

3. The server-side code for form validation and processing username, email, preferences should be written in the **php file**

4. Sample code for empty username password

function Login()

{

if(empty($\_POST['username']))

{

$this->HandleError("UserName is empty!");

return false;

}

if(empty($\_POST['password']))

{

$this->HandleError("Password is empty!");

return false;

}

$username = trim($\_POST['username']); /// Jst to remove trim the strings

$password = trim($\_POST['password']);

}

4. Sample code to retrieve multiple checkbox values

<form action="#" method="post">

<input type="checkbox" name="check\_list[]" value="C/C++"><label>C/C++</label><br/>

<input type="checkbox" name="check\_list[]" value="Java"><label>Java</label><br/>

<input type="checkbox" name="check\_list[]" value="PHP"><label>PHP</label><br/>

<input type="submit" name="submit" value="Submit"/>

</form>

<?php

if(isset($\_POST['submit']))

{

if(!empty($\_POST['check\_list']))

{

// Loop to store and display values of individual checked checkbox.

foreach($\_POST['check\_list'] as $selected)

{

echo $selected."</br>";

}

}

}

?>

Output Expected on the browser:

You NAME :………………..

Your email:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Your Gender: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Your preferred courses: \_\_\_\_\_\_\_\_\_\_\_, \_\_\_\_\_\_\_\_\_\_\_\_\_\_, \_\_\_\_\_\_\_\_\_\_\_\_

**PART B**

**(PART B: TO BE COMPLETED BY STUDENTS)**

**(Students must submit the soft copy as per following segments within two hours of the**

**practical. The soft copy must be uploaded on the Blackboard or emailed to the concerned**

**lab in charge faculties at the end of the practical in case the there is no Black board access**

**available)**

**B.1 Software Code written by student:**

**(Students must paste the code here)**

**Task 1:**

**HTML**

<!DOCTYPE html>  
<html>  
<head>  
 <title>LAB\_7\_Task1</title>  
</head>  
<body>  
  
<form action="http://localhost/lab\_7.php" method="POST">  
 <label>Fibonacci, Armstrong and Factorial for: </label>  
 <input type="text" name="num"><br>  
 <input type="submit" name="submit">  
</body>  
</html>

**PHP**

<?*php*$number = $\_POST['num'];  
  
echo "Fibonacci series for first $number numbers: ";  
for ($counter = *0*; $counter < $number; $counter++){  
 echo Fibonacci($counter),' ';  
}  
echo "<br>";  
  
$flag = Armstrong($number);  
if ($flag == *1*)  
 echo "Yes, $number is an Armstrong number <br>";  
else  
 echo "No, $number is not an Armstrong number <br>";  
  
  
$fact = Factorial($number);  
echo "Factorial for $number is $fact ";  
  
  
function Fibonacci($number){  
 if ($number == *0*)  
 return *0*;  
 else if ($number == *1*)  
 return *1*;  
 else  
 return (Fibonacci($number-*1*) +  
 Fibonacci($number-*2*));  
}  
  
function Armstrong($number){  
 $sum = *0*;  
 $x = $number;  
 while($x != *0*)  
 {  
 $rem = $x % *10*;  
 $sum = $sum + $rem \* $rem \* $rem;  
 $x = $x / *10*;  
 }  
 if ($number == $sum)  
 return *1*;  
 return *0*;  
}  
  
function Factorial($number){  
 $factorial = *1*;  
 for ($i = *1*; $i <= $number; $i++){  
 $factorial = $factorial \* $i;  
 }  
 return $factorial;  
}  
  
  
?>

**Task 2:**

**HTML**

<!DOCTYPE html>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <title>LAB\_7\_Task2</title>  
</head>  
<body>  
 <form action="http://localhost/lab\_7\_2.php" method="POST">  
 <label>Enter username: </label><br>  
 <input type="text" name="username" id="username"><br>  
  
 <label>Enter Email: </label><br>  
 <input type="text" name="email" id="email"><br>  
  
 <label>Enter Password: </label><br>  
 <input type="password" name="password" id="password"><br>  
  
 <label>Checkboxes here: </label><br>  
 <input type="checkbox" name="checkbox[]" value="Checkbox1" id="checkbox1">Checkbox 1<br>  
 <input type="checkbox" name="checkbox[]" value="Checkbox2" id="checkbox2">Checkbox 2<br>  
  
 <label>Radio buttons here: </label><br>  
 <input type="radio" name="radio" value="Radio1" id="radio1" checked>Radio 1<br>  
 <input type="radio" name="radio" value="Radio2" id="radio2">Radio 2<br><br>  
  
 <input type="submit" name="submit">  
  
</body>  
</html>

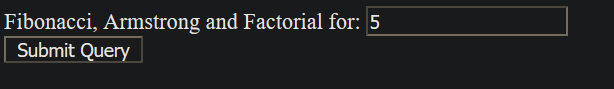
**PHP**

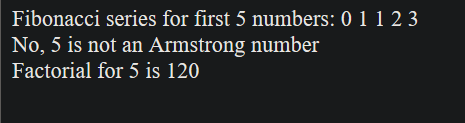
<?*php*$username = trim($\_POST['username']);  
$password = trim($\_POST['password']);  
$email = trim($\_POST['email']);  
  
  
if(empty($username)){  
 echo "Username is Empty!!";  
}  
else{  
 echo "Username entered is: " . $username;  
}  
  
if(empty($password)){  
 echo "<br>Password is Empty!!";  
}  
else{  
 $a = preg\_match("/^(?=.{8,})(?=.\*[a-z])(?=.\*[A-Z])(?=.\*[^\w\d]).\*$/",$\_POST["password"]);  
 if(!$a)  
 {  
 echo "<br>Invalid password";  
 }  
 else {  
 echo "<br>Valid password";  
 }  
}  
  
if(empty($email)){  
 echo "<br>Email is Empty!!";  
}  
else{  
 if(filter\_var($email, FILTER\_VALIDATE\_EMAIL)) {  
 echo("<br>Valid email address: " . $email);  
 }  
 else{  
 echo("<br>Invalid Email");  
 }  
}  
  
  
if(!empty($\_POST['checkbox'])){  
 echo "<br>Checked values: ";  
  
 foreach($\_POST['checkbox'] as $selected){  
 echo $selected." ";  
 }  
}  
else{  
 echo "<br>No Checkbox Selected!!";  
}  
  
echo "<br>Radio button picked: " . $\_POST['radio']  
  
?>

**B.2 Input and Output**

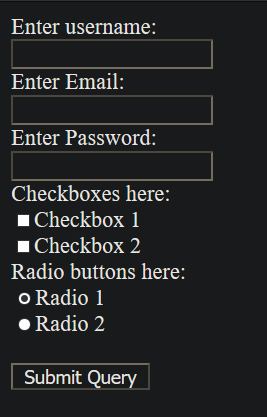
**(Students must paste input and output here)**

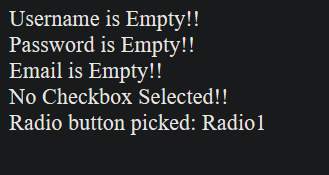
**Task 1:**

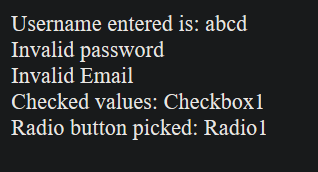
****

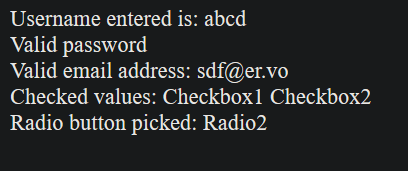
****

**Task 2:**

****

****

****

****

**B.3. Observations and Learning**

**(Students are expected to comment on the output obtained with clear observations and learning for each task/ sub part assigned**

1. Understood basic PHP programming

2. Understood and implemented mathematical operations with PHP

3. Understood and implemented PHP as server-side code for form processing

**B.4. Conclusion**

**(Students must write the conclusion as per the attainment of individual outcome listed above and learning/observation noted in Sec. B.3)**

PHP can be used to perform mathematical operations and also for form processing, these can be particularly useful when the client has disabled JavaScript on their browser for security concerns or something else.