DEPARTMENT OF COMPUTER ENGINEERING & APPLICATIONS
Institute of Engineering & Technology

Lab Manual

Subject Name&Code: PHP Scripting Language Lab(BCSE 0284)

Sumbitted by :   **Sumbitted TO :**

Puroo Kulshrestha (201500535)   Mr. Akash Chaudhary

Course: B. Tech (C.S)

Year/Semester: 3/5

Section – L

Roll no. - 30

INDEX

10	Write a PHP program which iterates the integers from 1 to 100. For multiples of three print "FRONTEND" instead of the number and for the multiples of five print "BACKEND". For numbers which are multiples of both three and five print "FULLSTACK".	
11	Write a PHP script to create a registration form using various HTML form elements such as Input type(Text, email, password, radio buttons, check boxes, upload file option etc).	
12	Create a PHP dynamic web page to check whether a number is Armstrong or not using class –object concepts.	
13	Create a PHP script to arrange the elements of an array in ascending order as per key.	
14	Write a PHP script to show the working of various CRUD operations using object oriented connection approach.	
15	Write a PHP Script to store the registration form data into the Database using procedural query methods.	
16	Create a form that accepts the user's data. Create a script that saves the data to a file.	
17	Write various PHP scripts for different pages registration, login, home and logout to show the working of sessions how session variable created and how we can delete a session variable.	

EXPERIMENT NO: 1

Objective: Write a PHP script to create a static web page which display your CV in the formation of row and columns with the help of HTML table tag

Procedure/Algorithm/Steps/Syntax:
1. Use HTML tags for using CV.

Code:
```
<html>
<head>
<title>
CV</title>
</head>
<body bgcolor="white">
<center>
<table border="5" size="90">
<tr>
<td colspan=4>
<center><h1><u>CV</u></h1></center>
<b>Name:</b>PUROO KULSHRESTHA<br>
<b>Date of Birth:</b> 13 MAY 2003<br>
<b>Father's Name:</b> Dr. Mukesh Kumar Kulshrestha<br>
<b>Address:</b> Jawahar Navodya Vidayalaya, Joura, Morena, Madhya
Pradesh
</td>
</tr>
<tr>
<td colspan=5><h3>Academics:</h3></td>
</tr>
<tr>
<td><b>Degree</b></td>
<td><b>Institute</b></td>
<td><b>Percentage</b></td>
<td><b>Year</b></td>
</tr>
<tr>
<td>Btech</td>
<td>GLA University</td>
<td>7.04</td>
<td>2020-2024</td>
</tr>
<tr>
<td>Intermediate</td>
<td>Maa bharti Senior Secondary School</td>
<td>67.8</td>
<td>2019-20]</td>
</tr>
<tr>
<td>High School</td>
<td>Jawahar Navodaya Vidayalaya</td></td>
<td>77.2</td>
<td>2017-18</td>
</tr>
<tr>
```

```
<td colspan=5>
<h3>Achievements:</h3>
<ul>
<li>Successfully presented project on DBInserter using C</li>
</ul>
<h3>Skills:</h3>
<b>Technical</b>
<ul>
<li> Languages:C, Java, FLutter, HTML, Php</li>
<li>Tools: IntelliJ-IDEA,Visual Studio Code</li>
</ul>
<b>Professional</b>
<ul>
<li>Analytical Skills</li>
<li>Communication Skills</li>
<li>Presentation Skills</li>
</ul>
<h3>Extracurricular Activities:</h3>
<ul>
<li>Attend workshop on Flutter</li>
</ul>
</td>
</tr>
</center>
</body>
</html>
```

Sample Output:

CV

Name:PUROO KULSHRESTHA
Date of Birth: 13 MAY 2003
Father's Name: Dr. Mukesh Kumar Kulshrestha
Address: Jawahar Navodya Vidayalaya, Joura, Morena, Madhya Pradesh

Academics:

Degree	Institute	Percentage	Year
Btech	GLA University	7.04	2020-2024
Intermediate	Maa bharti Senior Secondary School	67.8	2019-20]
High School	Jawahar Navodaya Vidayalaya	77.2	2017-18

Achievements:

- Successfully presented project on DBInserter using C

Skills:

Technical

- Languages:C, Java, FLutter, HTML, Php
- Tools: IntelliJ-IDEA,Visual Studio Code

Professional

- Analytical Skills
- Communication Skills
- Presentation Skills

Extracurricular Activities:

- Attend workshop on Flutter

EXPERIMENT NO: 2

Objective: Write a PHP script to create a static web page to find the
maximum and minimum marks from the following set of
arrays.

Procedure/Algorithm/Steps/Syntax:
1. Declare two variables max and min to store maximum and minimum. Assume first array element as maximum and minimum both, say max = arr[0] and min = arr[0].
2. Iterate through array to find maximum and minimum element in array. Run loop from first to last array element i.e. 0 to size-1. Loop structure should look like for(i=0;i<size-1;i++).
3. Inside loop for each array element check for maximum and minimum. Assign current array element to max, if (arr[i] > max). Assign current array element to min if it is less than min i.e. perform `min = arr[i]` if (arr[i] < min).

Code:

```php
<?php

function max_arr($a)
{
  $n = count($a);
  $max = $a[0];
  for ($i = 1; $i < $n; $i++)
    if ($max < $a[$i])
        $max = $a[$i];
   return $max;
}


function min_arr($a)
{
  $n = count($a);
  $min = $a[0];
  for ($i = 1; $i < $n; $i++)
    if ($min > $a[$i])
        $min = $a[$i];
   return $min;
}


$arr = array(10, 22, 35, 47, 59);
echo ("Elements in array set : ");
foreach ($arr as $i) {
   echo "$i, ";
}
echo "<br>";
echo ("Maximum number in array set : ");
echo(max_arr($arr));
echo "<br>";
echo ("Minimum number in array set : ");
```

```
echo(min_arr($arr));
?>
```

Sample Input:

Input array elements : 10, 22, 35, 47, 59

Sample Output:

Elements in array set : 10, 22, 35, 47, 59,
Maximum number in array set : 59
Minimum number in array set : 10

.

EXPERIMENT NO: 3

Objective: Write a PHP script to create a static web page which will
convert decimal number to binary, binary number into
decimal, rounding of a given number to respected decimal

Procedure/Algorithm/Steps/Syntax:
    The **decbin() function** converts a decimal number to a binary number.
    The **bindec() function** converts a binary number to a decimal number.

Code:

```
<html>
  <title>Convert</title>
  <body>
    <form action="bianry_deci.php" method="post">
      <table>
      <tr><td> Decimal Value :  <input type="text" name="a"></tr></td>
      <tr><td> Binary Value :   <input type="text" name="b"></tr></td>
        <tr><td><input type="submit" name="sumbit" value="sumbit"/></tr></td>
      </table>
    </form>
  </body>
</html>

<?php

  if (isset($_POST['a'])){
    $num1 = $_POST['a'];
    echo decbin($num1);
  }
  echo "<br>";
  if (isset($_POST['b'])){
    $num2 = $_POST['b'];
    echo bindec($num2);
  }


?>
```

Sample Input:
            Decimal : 45
            Binary : 1011

Sample Output:

Decimal Value : 45

Binary Value : 1011

sumbit

101101
11

.

EXPERIMENT NO: 4

Objective: Write a PHP script to create a static web page to find the
maximum and minimum marks from the following set of
arrays.

Procedure/Algorithm/Steps/Syntax:
1. Declare two variables max and min to store maximum and minimum. Assume first array element as maximum and minimum both, say max = arr[0] and min = arr[0].
2. Iterate through array to find maximum and minimum element in array. Run loop from first to last array element i.e. 0 to size-1. Loop structure should look like for(i=0;i<size-1;i++).
3. Inside loop for each array element check for maximum and minimum. Assign current array element to max, if (arr[i] > max). Assign current array element to min if it is less than min i.e. perform `min = arr[i]` if (arr[i] < min).

Code:

```php
<?php

function max_arr($a)
{
  $n = count($a);
  $max = $a[0];
  for ($i = 1; $i < $n; $i++)
    if ($max < $a[$i])
        $max = $a[$i];
   return $max;
}


function min_arr($a)
{
  $n = count($a);
  $min = $a[0];
  for ($i = 1; $i < $n; $i++)
    if ($min > $a[$i])
        $min = $a[$i];
   return $min;
}


$arr = array(10, 22, 35, 47, 59);
echo ("Elements in array set : ");
foreach ($arr as $i) {
   echo "$i, ";
}
echo "<br>";
echo ("Maximum number in array set : ");
echo(max_arr($arr));
```

```
echo "<br>";
echo ("Minimum number in array set : ");
echo(min_arr($arr));
?>
```

Sample Input:

Input array elements : 10, 22, 35, 47, 59

Sample Output:

Elements in array set : 10, 22, 35, 47, 59,
Maximum number in array set : 59
Minimum number in array set : 10

.

EXPERIMENT NO: 5

Objective: Write a PHP program which iterates the integers from 1 to 100. For multiples of three print "FRONTEND" instead of the number and for the multiples of five print "BACKEND". For numbers which are multiples of both three and five print "FULLSTACK".

Procedure/Algorithm/Steps/Syntax:
1. First start the loop from 1 to given number n (n=100).
2. Then inside the loop put the if-else condition as per given condition.
3. If (i%3==0 and i%5==0) print fullstack, else if (i%3==0) print frontend, else if (i%5==0) print backend,

Code:

```php
<?php
  for ($i=1; $i<=100;$i++)
  {
    if ($i%3==0 && $i%5==0 )
    {
      echo ("Fullstack");
      echo ("<br>");
    }

    else if ( $i%3 == 0 )
    {
      echo ("Frontend");
      echo ("<br>");
    }

    else if ( $i%5 == 0 )
    {
      echo ("Backend");
      echo ("<br>");
    }

    else{
      echo ($i);
      echo ("<br>");
    }
  }
?>
```

Sample Input:
        1-100

Sample Output:

1
2
Frontend
4
Backend
Frontend
7
8
Frontend
Backend
11
Frontend
13
14
Fullstack
16
17
Frontend
19
Backend
Frontend
22
23
Frontend
Backend
26
Frontend
28
29
Fullstack
31
32
Frontend
34
Backend
Frontend
37
38
Frontend
Backend
41
Frontend
43
44
Fullstack
46
47
Frontend

52
53
Frontend
Backend
56
Frontend
58
59
Fullstack
61
62
Frontend
64
Backend
Frontend
67
68
Frontend
Backend
71
Frontend
73
74
Fullstack
76
77
Frontend
79
Backend
Frontend
82
83
Frontend
Backend
86
Frontend
88
89
Fullstack
91
92
Frontend
94
Backend
Frontend
97
98
Frontend
Backend

EXPERIMENT NO: 6

Objective: Write a PHP script to create a static web page which displays the greatest among four numbers using ternary operator.

Procedure/Algorithm/Steps/Syntax:
1. First create the static web page with html tags for taking input of four numbers from user.
2. Now for php part you have to store all the four number in local variables then make a logic of finding largest number from ternary operator.

Code:

```
<html>
   <title>Largest number</title>
   <body>
      <form action="largest_ternary.php" method="post">
         <table>
         <tr><td> First number :  <input type="text" name="a"></tr></td>
         <tr><td> Second number :   <input type="text" name="b"></tr></td>
         <tr><td> Third number :  <input type="text" name="c"></tr></td>
         <tr><td> fourth number :   <input type="text" name="d"></tr></td>
           <tr><td><input type="submit" name="sumbit" value="sumbit"/></tr></td>
         </table>
      </form>
   </body>
</html>

<?php
   if (isset($_POST['a'])){
      $n1 = $_POST['a'];
   }

   if (isset($_POST['b'])){
      $n2 = $_POST['b'];
   }

   if (isset($_POST['c'])){
      $n4 = $_POST['c'];
   }

   if (isset($_POST['d'])){
      $n4 = $_POST['d'];
   }

   $max = ($n1 > $n2 && $n1 > $n3 && $n1 > $n4) ? $n1 : (($n2 > $n3 && $n2 > $n4) ? $n2 : ($n3
> $n4 ? $n3 : $n4));


 echo ("Largest number : ");
 echo ($max);
?>
```

Sample Input:

Input array elements : 10, 22, 35, 47

Sample Output:

First number : [   ]
Second number : [   ]
Third number : [   ]
fourth number : [   ]
[sumbit]

Warning: Undefined variable $n3 in **D:\XAMPP\htdocs\php_codes\largest_ternary.php** on line **33**

Warning: Undefined variable $n3 in **D:\XAMPP\htdocs\php_codes\largest_ternary.php** on line **33**
Largest number : 47

EXPERIMENT NO: 7

Objective: Write a PHP script using nested for loop that creates a chess board as shown below. Use table width="270px" and take 30px as cell height and width.

Procedure/Algorithm/Steps/Syntax:
1. To create Chess in php we have to run 2 loops each will create 8 blocks.
2. The inner-loop will generate table-row with black/white background-color based upon a value.
3. If value is even, Black background is generated.
4. If value is odd, White background is generated

Code:

```
<!DOCTYPE html>
<html>
<body>
   <table width="270px" border="1px" cellspacing="30px" >
     <?php
       $value = 0;
       for($col = 0; $col < 8; $col++){
          echo "<tr>";
          $value = $col;
          for($row = 0; $row < 8; $row++){
             if($value%2 == 0){
                echo "<td height=-30px width=-40px bgcolor=black></td>";
                $value++;
             }
             else {
                echo "<td height=-30px width=-40px bgcolor=white></td>";
                $value++;
             }
          }
          echo "</tr>";
       }
     ?>
   </table>
</body>

</html>
```

Sample Output:

EXPERIMENT NO: 8

Objective: Write a PHP script to create various functions for various operations such as factorial of a number, number is prime or not, calculate the area of square. While reading the number from user using switch case statement.

Procedure/Algorithm/Steps/Syntax:
        we take input from user and in function.php there is user defined function to perfrom given operation
Code:

```
<html>
   <title>Various_function</title>
   <body>
     <form action="various_func.php" method="post">
       <table>
       <tr><td> Enter the Value :  <input type="text" name="a"></tr></td>
       <tr><td> Select the Operation :  1.Factorial 2.Prime 3.Area of square<br><input type="text"
name="b" placeholder="Provide only Value"></tr></td>
         <tr><td><input type="submit" name="sumbit" value="sumbit"/></tr></td>
       </table>
     </form>
   </body>
</html>


<?php
   if (isset($_POST['a'])){
     $num1 = $_POST['a'];
   }
   if (isset($_POST['b'])){
     $num2 = $_POST['b'];

   }

   function fact($num1){
     $fact = 1;
     for ($i=$num1;$i>=1;$i--){
       $fact = $fact*$i;
     }
     echo "Factorial of $num1 is $fact";
   }

   function Prime($num1){
     if ($num1 == 1)
       echo ("Not a prime number");
     for($i=2;$i<$num1;$i++){
       if($num1 % $i == 0)
          echo ("Prime number");
     }
     echo ("Not a prime number");
   }

   function Area($num1){
```
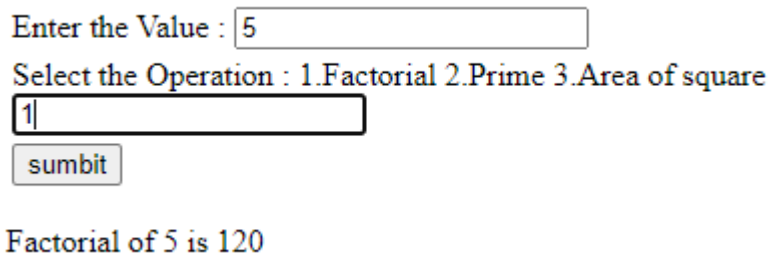
```
    $res = $num1 * $num1;
    echo "Area of square is $res";
  }

  switch ($num2){
    case 1:
      fact($num1);
      break;
    case 2:
      Prime($num1);
      break;
    case 3:
      Area($num1);
      break;
    default:
      echo ("Not valid submission");
  }

?>
```

Sample Input:
   Enter the number – 5
   Select the Operation – 1 (factorial)

Sample Output:

Enter the Value : 5

Select the Operation : 1.Factorial 2.Prime 3.Area of square
1

sumbit

Factorial of 5 is 120

EXPERIMENT NO: 9

Objective: Write a PHP script that creates the following table (use for loops)

Procedure/Algorithm/Steps/Syntax:
        We have to use table tag with one loop-nested loop.
Code:

```php
<?php
  echo "<table>";
  for($row=1;$row<=9;$row++)
  {
   echo "<tr>";
   for($col=1;$col<=9;$col++)
   {
    echo "<td style='border:0px solid black'>";
    $res = $row*$col;
    echo "$res";
    echo "</td>";
   }
   echo "<br>";
   echo "</tr>";
  }
  echo "</table>";
?>
```

Sample Output:

1	2	3	4	5	6	7	8	9	10
2	4	6	8	10	12	14	16	18	20
3	6	9	12	15	18	21	24	27	30
4	8	12	16	20	24	28	32	36	40
5	10	15	20	25	30	35	40	45	50
6	12	18	24	30	36	42	48	54	60
7	14	21	28	35	42	49	56	63	70
8	16	24	32	40	48	56	64	72	80
9	18	27	36	45	54	63	72	81	90
10	20	30	40	50	60	70	80	90	100

EXPERIMENT NO: 10

Objective: Write a PHP script to check adam number concept while read the value from user and provide a button.

Procedure/Algorithm/Steps/Syntax:
1. Read or initialize a number N.
2. Find the square of the number.
3. Find the reverse of the square.
4. Reverse the number N.
5. Find the square of the reversed number.
6. If the reverse of the square of the number is equal to the square of the reverse of the number, the number is an Adam number, else not.

Code:

```
< <html>
   <title>Convert</title>
   <body>
     <form action="adam.php" method="post">
       <table>
       <tr><td>Enter the Number : <input type="text" name="a"></tr></td>
         <tr><td><input type="submit" name="sumbit" value="sumbit"/></tr></td>
       </table>
     </form>
   </body>
</html>


<?php

function reverseDigits($num)
{
   $rev = 0;
   while ($num > 0)
   {
     $rev = $rev * 10 + $num % 10;
     $num = (int) $num / 10;
   }
   return $rev;
}

function checkAdamNumber($num)
{

   $a = $num * $num;
   $b = reverseDigits($num);
   $b = $b * $b;

   if ($a == reverseDigits($b))
      return 0;
   return -1;
}
```

```
if (isset($_POST['a'])){
    $num = $_POST['a'];
}

if (checkAdamNumber($num))
    echo "$num is Adam Number";
else
    echo "$num is not a Adam Number";



?>
```

Sample Input:
    Enter the number – 12

Sample Output:

12 is Adam Number

EXPERIMENT NO: 11

Objective: Write a PHP script to create a registration form using various HTML form elements such as Input type(Text, email, password, radio buttons, check boxes, upload file option etc)..

Procedure/Algorithm/Steps/Syntax:
we simply use the html tags in echo.

Code:

```html
        <html>
   <title>Registration</title>
   <body>
     <form action="registration.php" method="post">
       <table>
       <tr><td> Name :      <input type="text" name="uname"></tr></td>
       <tr><td> Age :       <input type="text" name="age" size="3"></tr></td>
       <tr><td> Email :     <input type="text" name="mail"
placeholder="@gmail.com"></tr></td>
       <tr><td> Password :   <input type="text" name="pass"></tr></td>
       <tr><td> Resume :     <input type="file" id="resume"></tr></td>
       <tr><td> Gender :    Male<input type="radio" name="gender" value="Male">
Female<input type="radio" name="gender" value="Female"></tr></td>
       <tr><td> Martial Status :    Married<input type="checkbox" name="marry"
value="Married">Single<input type="checkbox" name="marry" value="Single"></tr></td>

       <tr><td> <tr><td><input type="submit" name="sumbit"
value="sumbit"/></tr></td></tr></td>
       </table>
     </form>
   </body>
</html>

<?php
   if (isset($_POST['uname'])){
     $num1 = $_POST['uname'];
     echo ("Name : $num1");
   }
   echo "<br>";
   if (isset($_POST['age'])){
     $num2 = $_POST['age'];
     echo ("Age : $num2");
   }
   echo "<br>";
   if (isset($_POST['mail'])){
     $num3 = $_POST['mail'];
     echo ("Email : $num3");
```
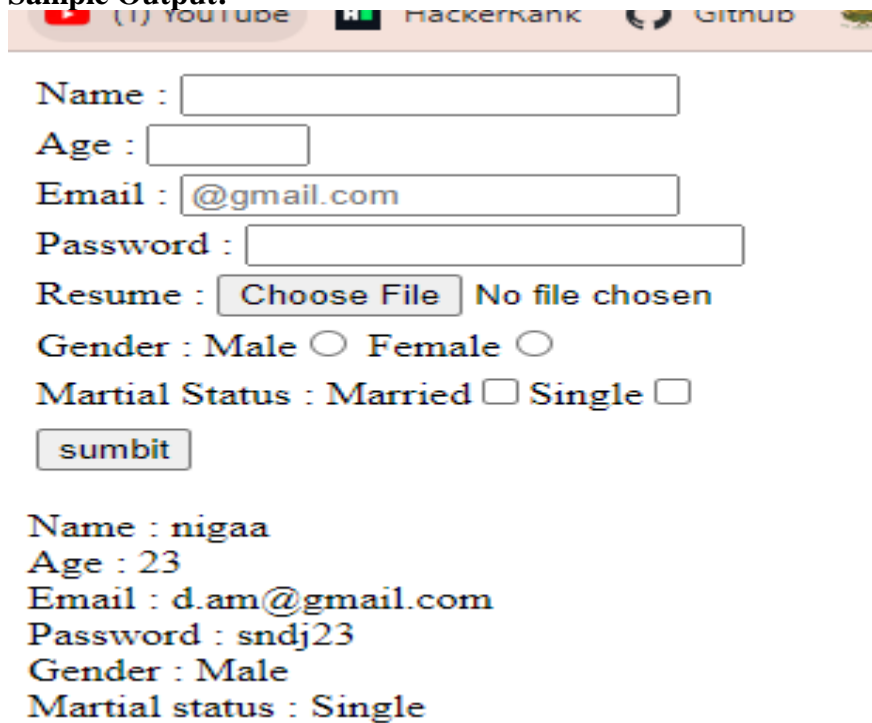
```
        }
    echo "<br>";
    if (isset($_POST['pass'])){
        $num4 = $_POST['pass'];
        echo ("Password : $num4");
    }
    echo "<br>";
    if (isset($_POST['gender'])){
        $num5 = $_POST['gender'];
        echo ("Gender : $num5");
    }
    echo "<br>";
    if (isset($_POST['marry'])){
        $num6 = $_POST['marry'];
        echo ("Martial status : $num6");
    }
?>
```

Sample Output:

Name : nigaa
Age : 23
Email : d.am@gmail.com
Password : sndj23
Gender : Male
Martial status : Single

EXPERIMENT NO: 12

Objective: Create a PHP dynamic web page to check whether a number is Armstrong or not using class –object concepts

Procedure/Algorithm/Steps/Syntax:
1. The number of digits in num is determined and found out.
2. The sum of digits of a number in Python or individual digit sums are got by performing num mod 10, where mod is called the remainder operation.
3. The individual digit is then raised to the power (number of digits) and stored.
4. The number is then divided by 10 in order to obtain the second digit.
5. All the above 3-steps numbered Steps 2, 3 and 4 are performed until the value of num is greater than 0.
6. When the num is less than 0, end the while loop.
7. Check the sum obtained or Armstrong value is the same as the original number
8. When yes, the number is labelled an Armstrong number

Code:

```
<html>
  <title>Armstrong</title>
  <body>
    <form action="Armstrong_dyna.php" method="post">
      <table>
      <tr><td>Enter the Number : <input type="text" name="a"></tr></td>
        <tr><td><input type="submit" name="sumbit" value="sumbit"/></tr></td>
      </table>
    </form>
  </body>
</html>

<?php
  class Armstrong{
    function arm($n){
      $k=$n;
      $res=0;
      while($k!=0)
      {
        $rem=$k%10;
        $res=$res+($rem*$rem*$rem);
        $k=$k/10;
      }

      if($res==$n)
        echo ("Armstrong number");
```
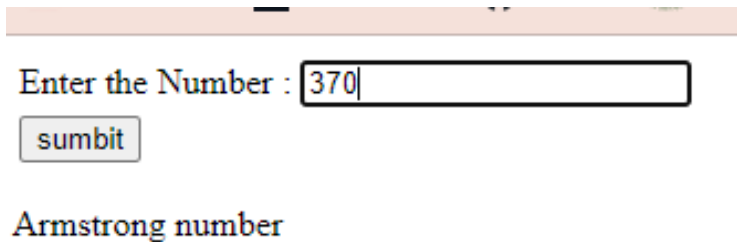
```
        else
            echo ("Not a armstrong number");
        }

    }

    if (isset($_POST['a'])){
        $n = $_POST['a'];
        $obj = new Armstrong();
        $obj->arm($n);
    }
?>
```

Sample Input:
   Number : 370

Sample Output:

Enter the Number : 370

sumbit

Armstrong number

EXPERIMENT NO: 13

Objective: Create a PHP script to arrange the elements of an array in ascending order as per key.

Procedure/Algorithm/Steps/Syntax:
1. The ksort() function sorts an associative array in ascending order, according to the key.
2. It sorts in a way that the relationship between the indices and values is maintained.

Code:

```php
<?php
 $names = [
   1001 => 'Puroo',
   1010 => 'Kulshrestha',
   101 => 'Gla',
   191 => 'Universisty',
   10 => 'Mathura',
 ];
 ksort($names);
 foreach ($names as $key => $val) {
   echo "$key = $val\n"."<br>";
}
?>
```

Sample Input:
Input array elements : 1001 => 'Puroo', 1010 => 'Kulshrestha', 101 => 'Gla', 191 => 'Universisty', 10 => 'Mathura',

Sample Output:__

```
10 = Mathura
101 = Gla
191 = Universisty
1001 = Puroo
1010 = Kulshrestha
```

EXPERIMENT NO: 15

Objective: Write a PHP Script to store the registration form data into the Database using procedural query methods.

Procedure/Algorithm/Steps/Syntax:
1. Contain the structure of the database table
2. used for add a record in the database
3. Used for read the record from database
4. Used for updating a record.

Code:

```
<html>
   <title></title>
</html>

<body>
   <form action="config.php" method="post">
     <table>
        <tr><td><input type="text" name="uname" placeholder="Username"></td></tr>
        <tr><td><input type="email" name="Email" placeholder="Email" required=""></td></tr>
        <tr><td><input type="text" name="password" placeholder="Password"
required=""></td></tr>
        <tr><td>
             <input type="submit" name="save" value="submit">
             <input type="reset" name="reset" value="reset">
             <input type="submit" name="display" value="display">
        </td></tr>

     </table>
   </form>
</body>

<?php
   $name = $_POST['uname'];
   $em = $_POST['Email'];
   $pass = $_POST['password'];

   // echo $name.$em.$pass;

   $connection = mysqli_connect("localhost","root","","dbpuroo");

   if(!$connection){
      die("Problem in Connection");
   }
```

```php
    if(isset($_POST['save']))
    {
        $insert_qr = "INSERT INTO userdb(UserName,Email,Password) VALUES
('$name','$em','$pass')";
        if(mysqli_query($connection,$insert_qr)){
            echo "Query is Sumbitted";
            header("refersh:5,url=form_1.html");
        }
    }

    if(isset($_POST['reset']))
    {
        echo "Rewrite your Belongings";
    }

    if(isset($_POST['display']))
    {
        $display_qr = "Select * from userdb";
        $res = mysqli_query($connection,$display_qr);
        if(mysqli_num_rows($res)>0)
        {
            while($data = mysqli_fetch_assoc($res)){
                echo $data['id']." ";
                echo $data['UserName']." ";
                echo $data['Email']." ";
                echo $data['Password']."<br>";
            }
        }
        else{
            echo "No record found";
        }
    }

?>
```

Sample Input:

```
1 sknx sjb@gaj 1jb2b
2 aahs aj@xah jaugqy1
124 SHama ajsh@ja 1sdhhu2
125 purdsk aj@xah akskajn12
```

Sample Output:

```
1 sknx sjb@gaj 1jb2b
2 aahs aj@xah jaugqy1
124 SHama ajsh@ja 1sdhhu2
125 purdsk aj@xah akskajn12
```

EXPERIMENT NO: 16

Objective: Create a form that accepts the user's data. Create a script that saves the data to a file.

Procedure/Algorithm/Steps/Syntax:
We have to take data from the web to store inside a file.
Code:
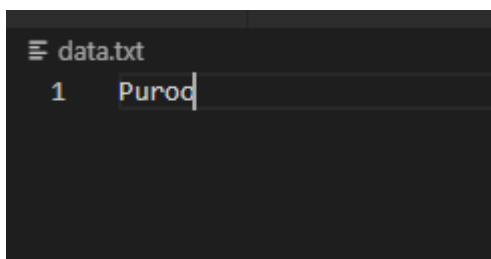
```
<html>
<head>
<title>Document</title>
</head>
<body>
<form action="16_1.php" method="post">
<input type="text" name="data" placeholder="Enter Data">
<input type="submit" name="submit">
</form>
</body>
</html>
<?php
$data = $_POST['data'];
$var = fopen("data.txt","w");
fwrite($var, $data);
?>
```

Sample Input:
Name - Puroo

Sample Output:

EXPERIMENT NO: 17

Objective: Write various PHP scripts for different pages registration, login, home and logout to show the working of sessions how session variable created and how we can delete a session variable.

Procedure/Algorithm/Steps/Syntax:
we created the four pages login , logout , home and product.

Code:

```php
<html>
<body>
<form action="17.php" method="post">
<table>
<tr><th>Login</th></tr>
<tr><td>Username</td><td><input type="email" name="uname"></td></tr>
<tr><td>Pssword</td><td><input type="password" name="psw"></td></tr>
<tr><td colspan="2"><input type="submit" name="login" value="login"></td></tr>
</table>
</form>
</body>
</html>
<?php
$username = "admin@com";
$passw = "admin";
session_start();
if(isset($_SESSION['user']))
{
echo "welcome"."<h3 style='color:red' align='right'>".$_SESSION['user']."</h3>";
echo "<h4>You are scrolling home page data</h4>";
echo "<a href='./product.php'>Click here for navigate to Product Details Page</a>";
echo "<br><a href='logout.php'><input type=submit name=logout value=logout></a>";
}
else{
if($username == $_POST['uname'] and $passw == $_POST['psw'])
{
$_SESSION['user'] = $username;
echo "<script>location.href='home.php'</script>";
}
else{
echo "<script>alert('invalid username and password!!!')</script>";
echo "<script>location.href='login.html'</script>";
}
}
?>
<?php
session_start();
if(isset($_SESSION['user']))
```
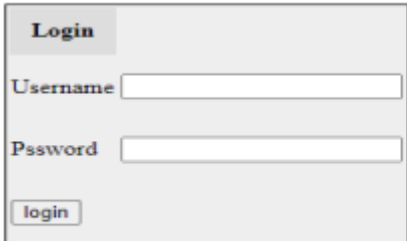
```
{
echo "welcome"."<h3 style='color:red' align='right'>".$_SESSION['user']."</h3>";
echo "<h4>You are scrolling Details page data</h4>";
echo "<a href='home.php'>Click here for navigate to Home Page</a>";
echo "<br><a href='logout.php'><input type=submit name=logout value=logout></a>";
}
else{
echo "<script>location.href='login.html'</script>";
}
?>
<?php
session_start();
if(isset($_SESSION['user']))
{
unset($_SESSION['user']);
echo "<script>location.href='login.html'</script>";
}
else{
echo "<script>location.href='login.html'</script>";
}
?>
```
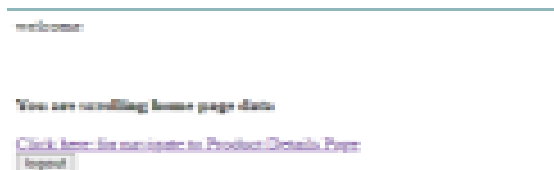
Sample Output:
Login Page

Home Page:

Product Page :