Practical 1

1.Install and configure PHP web server

2.WAP to print message “welcome to PHP”

3 WAP PHP program using operators(bit wise)and Expression

Solution:

Download XAMPP for Linux

Head over to this website to https://www.apachefriends.org/index.html XAMPP. You’ll notice three download options for different platforms: Windows, Linux, and OS X. Click the middle option “XAMPP for Linux” and the download will start.

On a side note, since XAMPP can be installed on Windows and OS X too, this creates an excellent opportunity (in case you’re a pupil) to collaborate on a project with your Windows or OS X peers together.

Installing XAMPP On Linux

After you’ve downloaded XAMPP installation file, open a terminal program and browse to the directory where you downloaded the file. Usually, it’s in the Downloads folder so the below command would work on all Linux platforms unless you customized the download location yourself:

```
cd ~/Downloads
```

Change the file mode to executable:

chmod +x xampp-linux-x64-7.2.5-0-installer.run
Then using sudo, execute the xampp installation file to install it on your computer:

sudo ./xampp-linux-x64-7.2.5-0-installer.run
cd ~/Downloads
Change the file mode to executable:

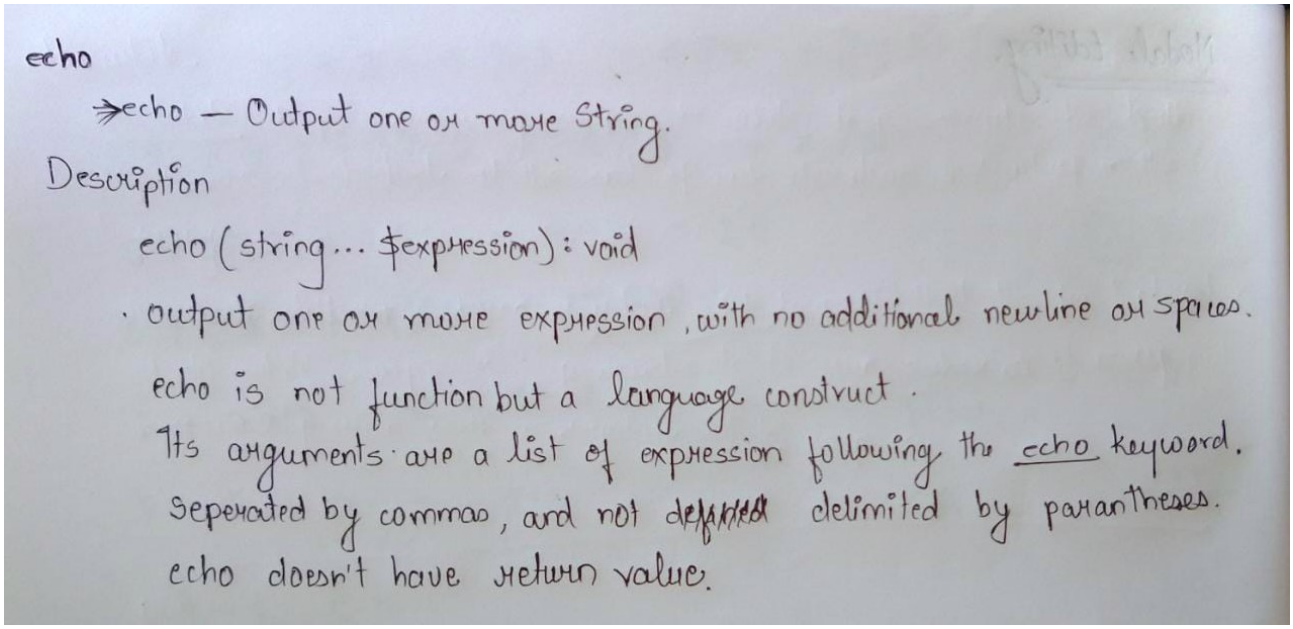
chmod +x xampp-linux-x64-7.2.5-0-installer.run
Then using sudo, execute the xampp installation file to install it on your computer:

sudo ./xampp-linux-x64-7.2.5-0-installer.run
Launching XAMPP

After installation, you might have clicked Finish by leaving the Launch XAMPP option ticked (and it'll launch). But unlike usual applications on your computer where you just launch them by point and click, XAMPP has no *.desktop* files, you'll have to launch it using a terminal emulator program. Also, XAMPP requires root privileges in order to be run successfully.
Launch a terminal program, then change your current directory to /opt/lampp and execute the below command to start xampp sudo ./manager-linux-x64.run After that, turn on **Apache web Server** on the Manage Servers tab and then launch your favorite web browser. On the URL bar, load this page http://localhost.com or http://127.0.0.1, you'll notice the similar output as shown in the screenshot below.


Welcome
Manage Servers
Application log
Server
Status
MySQL Database
Stopped
ProFTPD
Stopped
Start
Stop
Restart
Configure
Start All
Stop All
Restart All

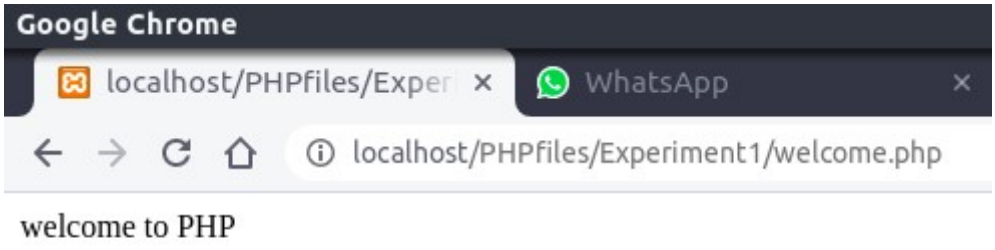
Defination:



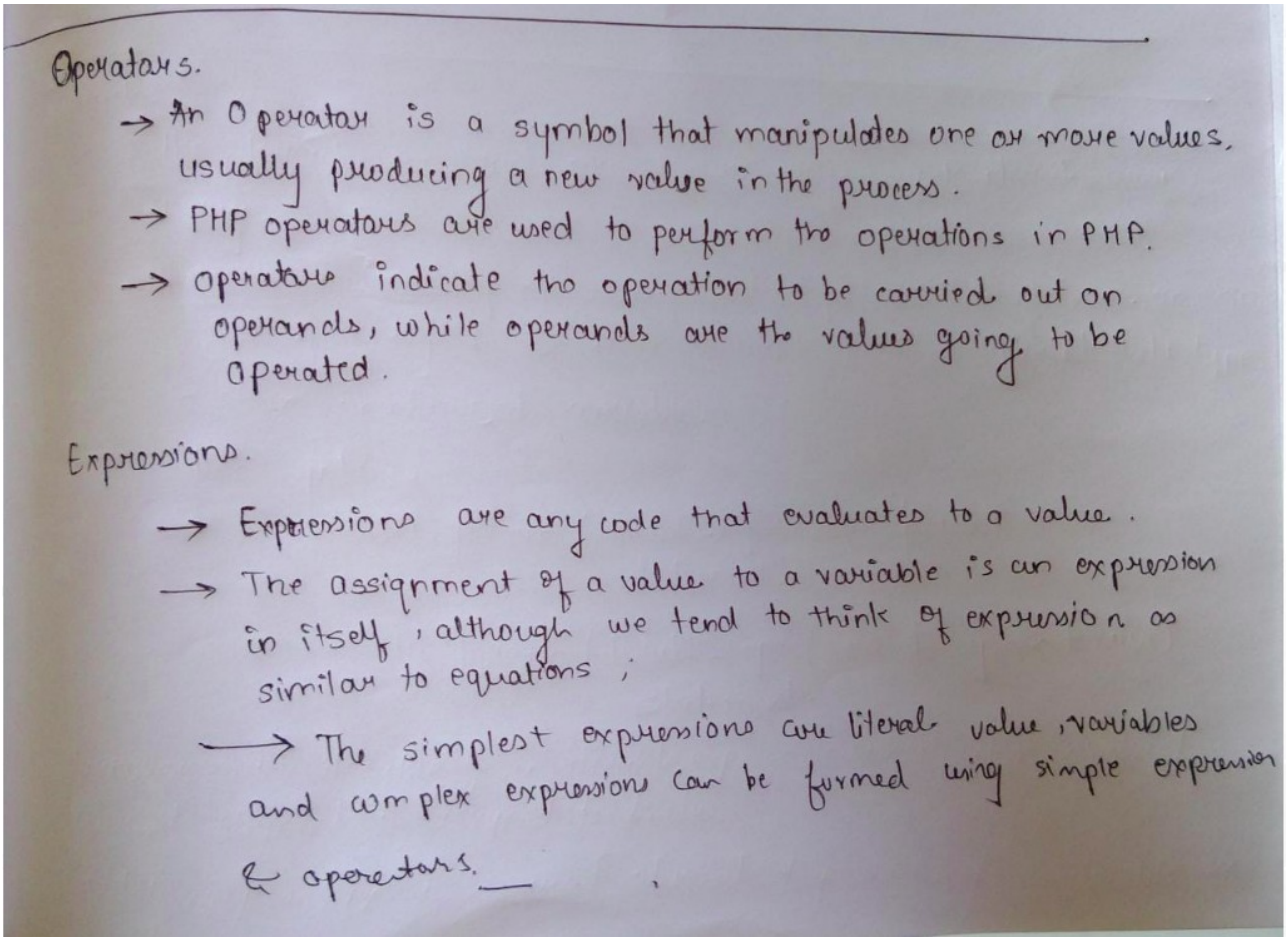
Code:

```
<?php
echo "welcome to PHP";
?>
```

Output:



Defination:

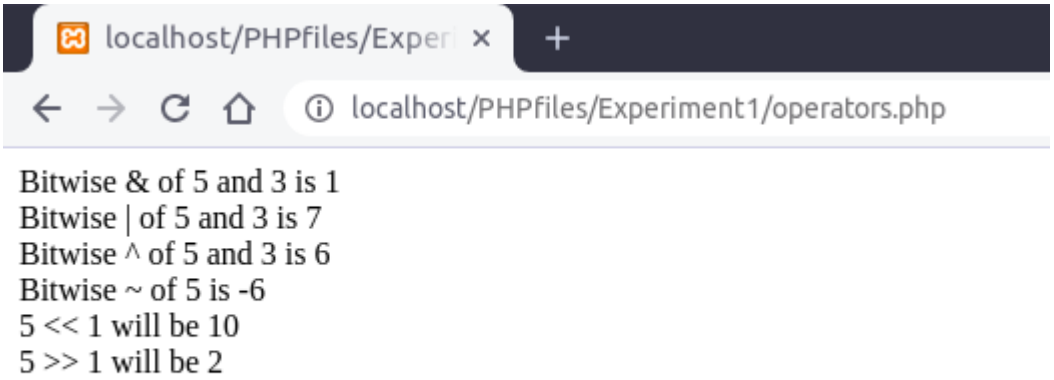


Code:

```
<?php
   // PHP code to demonstrate Bitwise Operator.

   // Bitwise AND
   $First = 5;
   $second = 3;
   $answer = $First & $second;

   print_r("Bitwise & of 5 and 3 is $answer<br/>");

   print_r("\n");

   // Bitwise OR
   $answer = $First | $second;
   print_r("Bitwise | of 5 and 3 is $answer<br/>");

   print_r("\n");

   // Bitwise XOR
```

```
    $answer = $First ^ $second;
    print_r("Bitwise ^ of 5 and 3 is $answer<br/>");

    print_r("\n");

    // Bitwise NOT
    $answer = ~$First;
    print_r("Bitwise ~ of 5 is $answer<br/>");

    print_r("\n");

    // Bitwise Left shift
    $second = 1;
    $answer = $First << $second;
    print_r("5 << 1 will be $answer<br/>");

    print_r("\n");

    // Bitwise Right shift
    $answer = $First >> $second;
    print_r("5 >> 1 will be $answer<br/>");

print_r("\n");
?>
```

Output:



Conclusion: We've successfully set up XAMPP and got it working! we can use it to test your PHP scripts or play around with MySQL. we can develop and test our web app before deploying it. , printed a welcome code and used operators(bitwise).

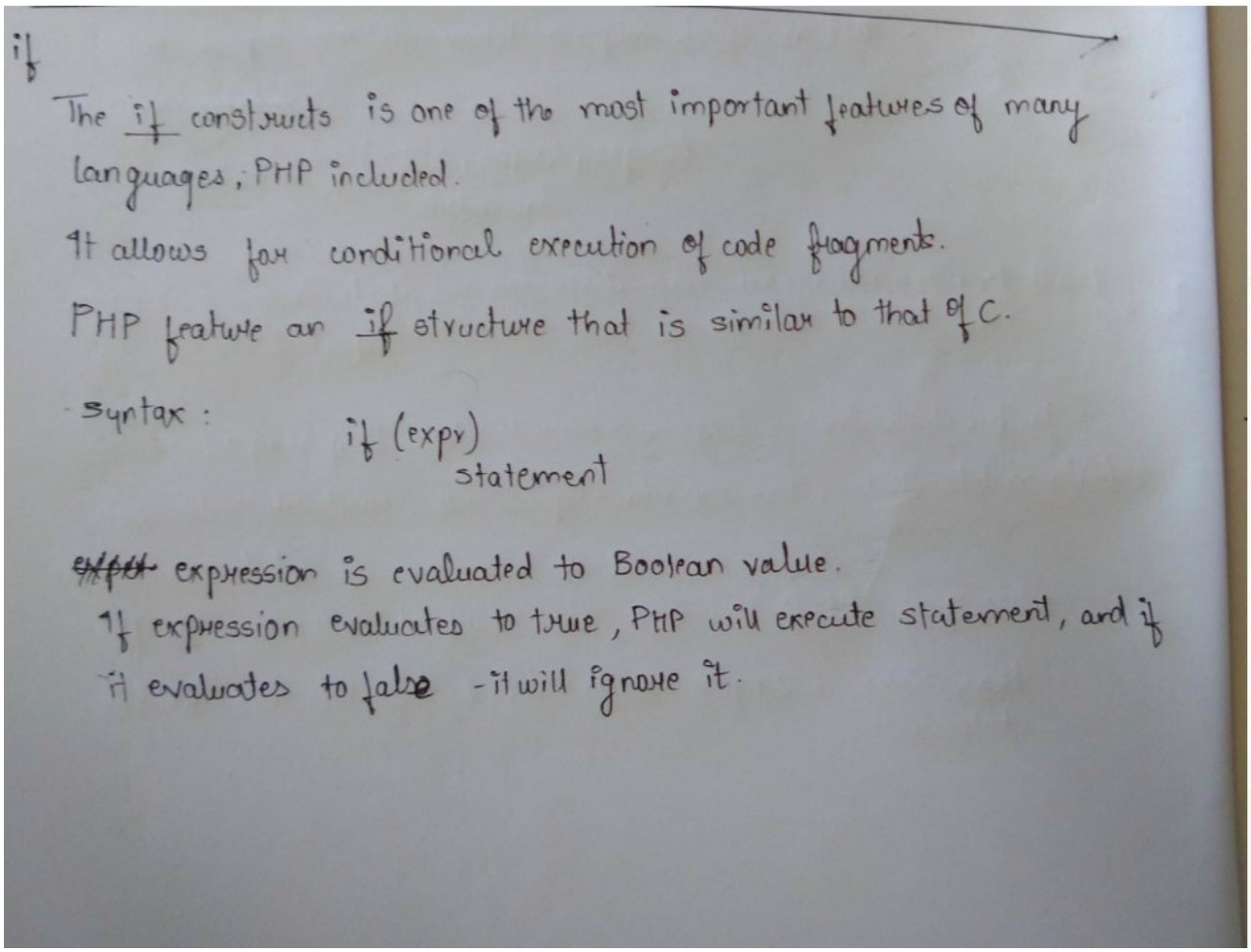
Practical 2

1.Print message “welcome TO PHP” by using if statement

2 WAP to check whether number is divisible by 9 or not using if-else

3 Any program using switch case

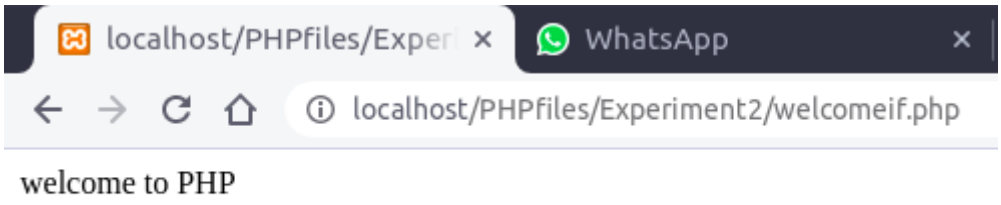
Defination:



Code:

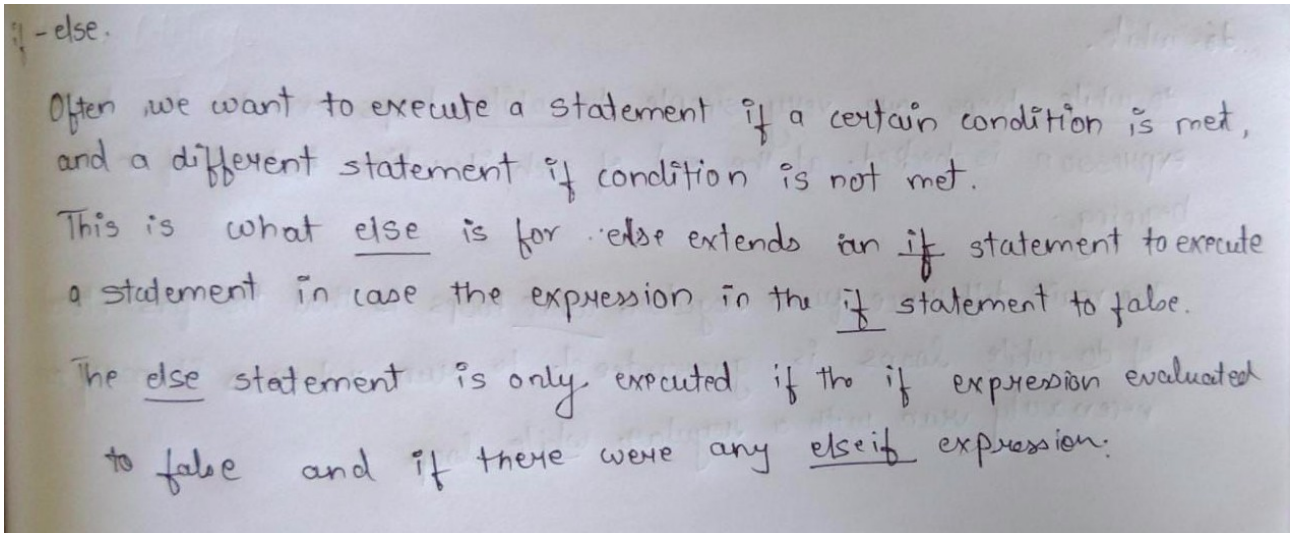
```
<?php
$user="name";
if ($user=="name")
{
   echo"welcome to PHP";
}
?>
```

Output:



welcome to PHP

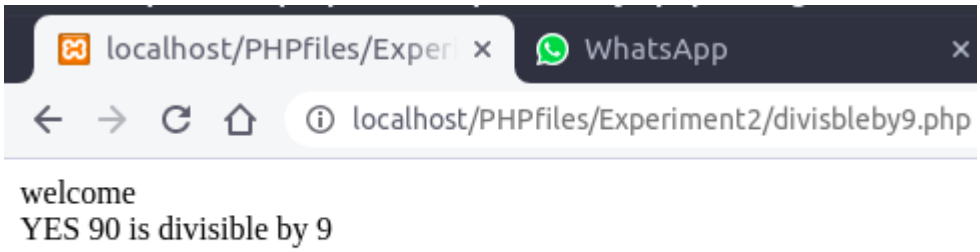
Defination:



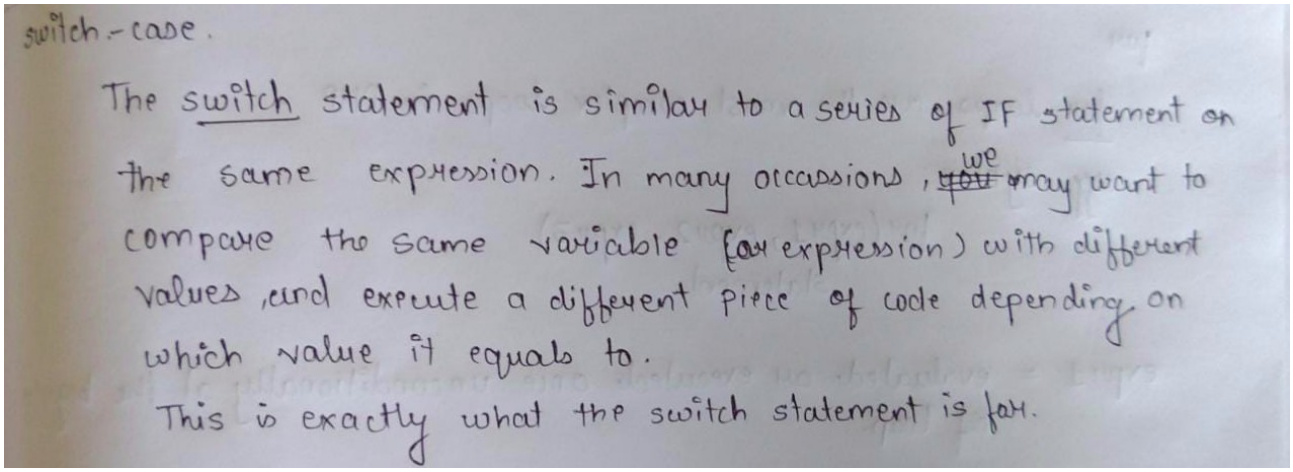
Code:

```
<?php
echo "welcome<br/>";
$number=90;
if($number%9==0)
{
 echo"YES ".$number." is divisible by 9<br/>";
}
else
{
echo"NO ".$number." is not  divisible by 9<br/>";
}
?>
```

Output:



Defination:


switch - case.

The switch statement is similar to a series of IF statement on the same expression. In many occassions, ~~you~~ we may want to compare the same variable (or expression) with different values, and execute a different piece of code depending on which value it equals to.

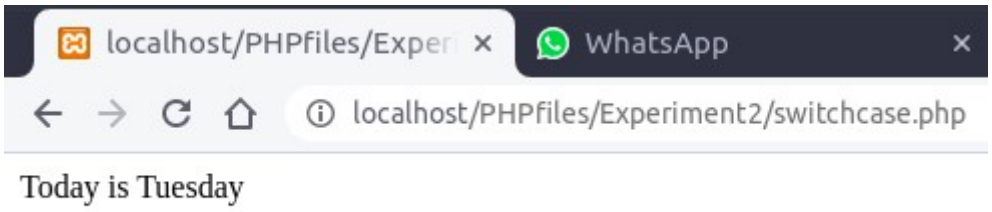
This is exactly what the switch statement is for.

Code:

```
<?php
$day="Tue";
switch($day)
{
case "Sun":
      echo "Today is Sunday";
   break;

case "Mon":
      echo "Today is Monday";
   break;

case "Tue":
      echo "Today is Tuesday";
   break;

case "Wed":
      echo "Today is Wednesday";
   break;

case "Thus":
      echo "Today is Thusday";
   break;

case "Fri":
      echo "Today is Friday";
   break;

case "Sat":
      echo "Today is Saturday";
      break;
default :
   echo "Something is wrong";
}
```

```
?>
```

Output:



Conclusion: we excuted a program that printed a messge using if ,checked whether a number is divisible or not and wrote a program to find which day is it using switch case.

Practical 3

1 WAP to print number in reverse order using while loop

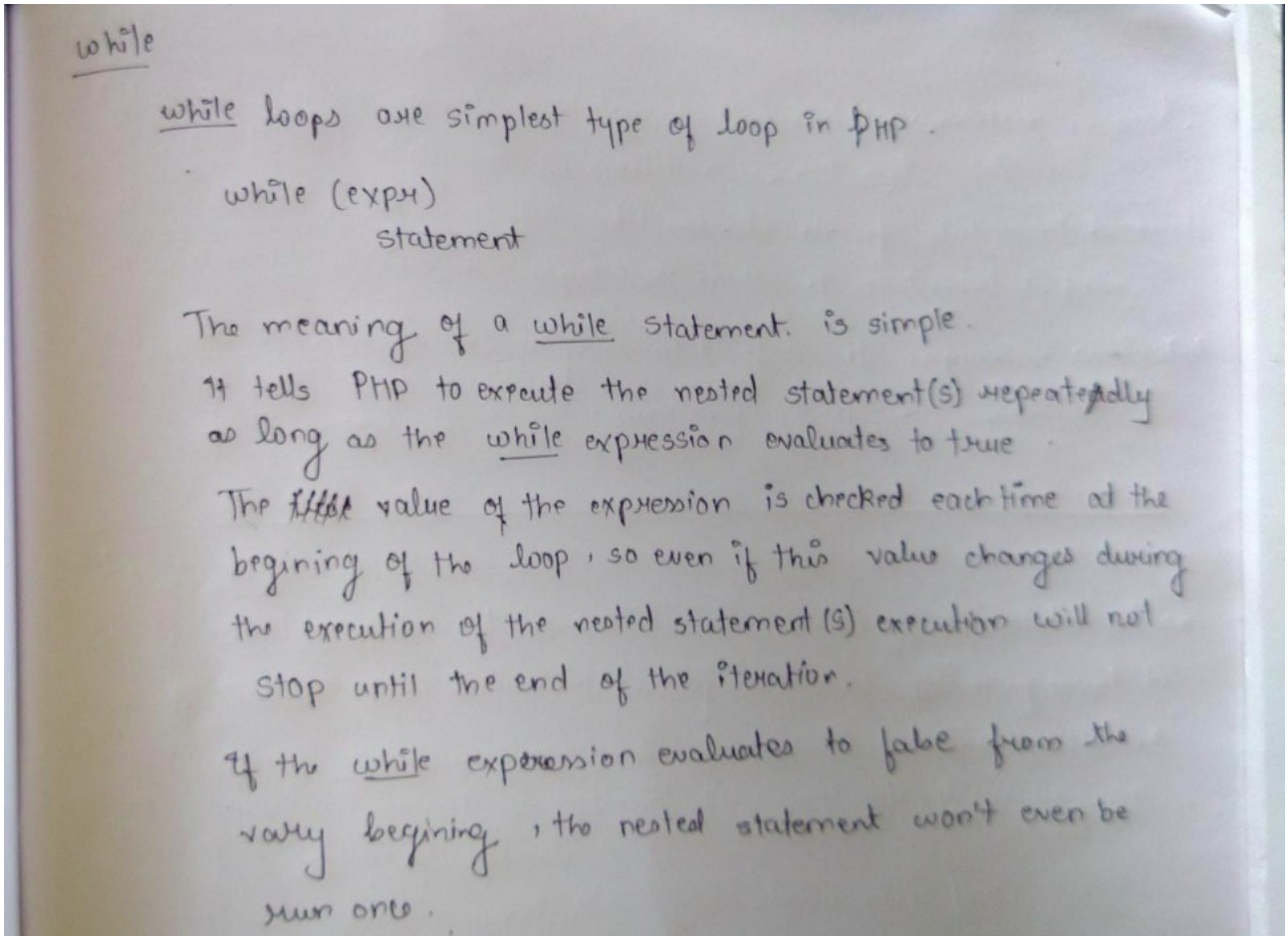
(number=1234 output=4321)

2 WAP to find factorial of a number using for loop

3.WAP to print no 1 to 10 using do-while

4WAP using foreach and display array elements

Defination:



while

while loops are simplest type of loop in PHP.

while (expr)
    statement

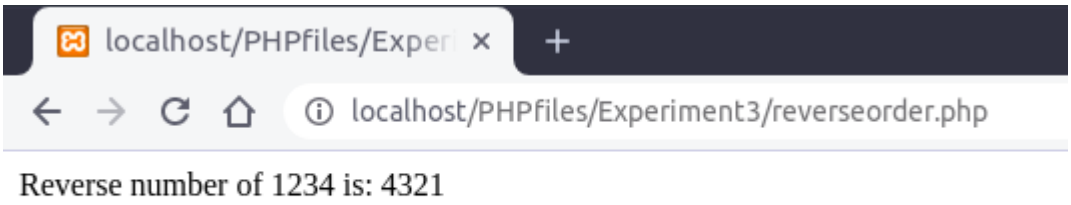
The meaning of a while statement is simple.
It tells PHP to execute the nested statement(s) repeatedly as long as the while expression evaluates to true.
The value of the expression is checked each time at the begining of the loop, so even if this value changes during the execution of the nested statement (s) execution will not stop until the end of the iteration.
If the while expression evaluates to false from the very begining, the nested statement won't even be run once.

Code:

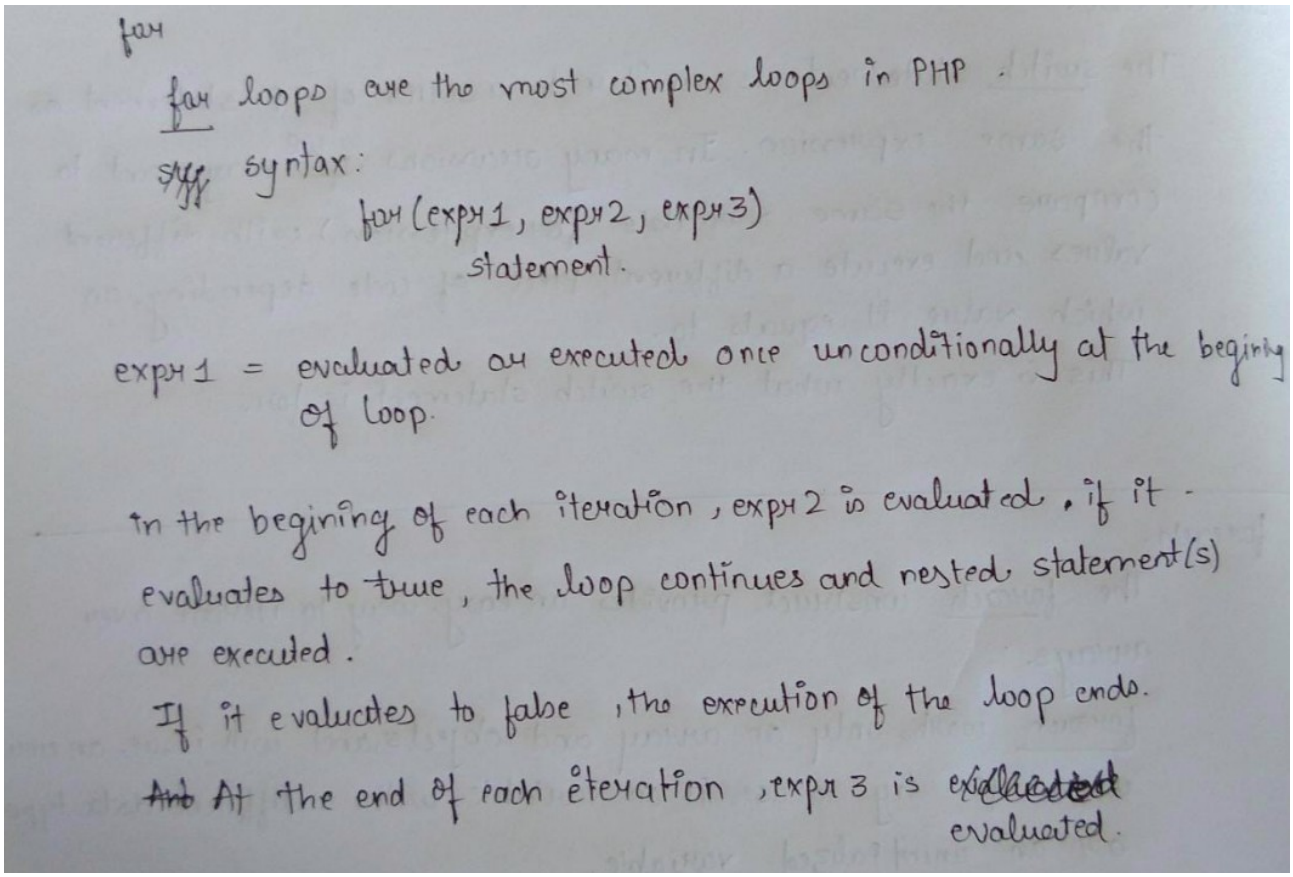
```
<?php
$num = 1234;
$revnum = 0;
while ($num > 1)
{
$rem = $num % 10;
```

```
$revnum = ($revnum * 10) + $rem;
$num = ($num / 10);
}
echo "Reverse number of 1234 is: ". $revnum;
?>
```

Output:



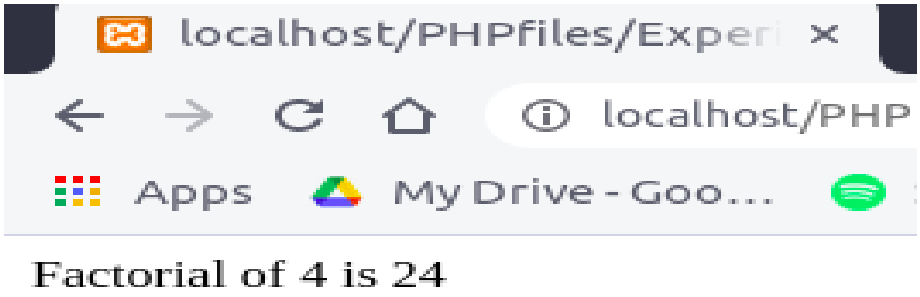
Defination:



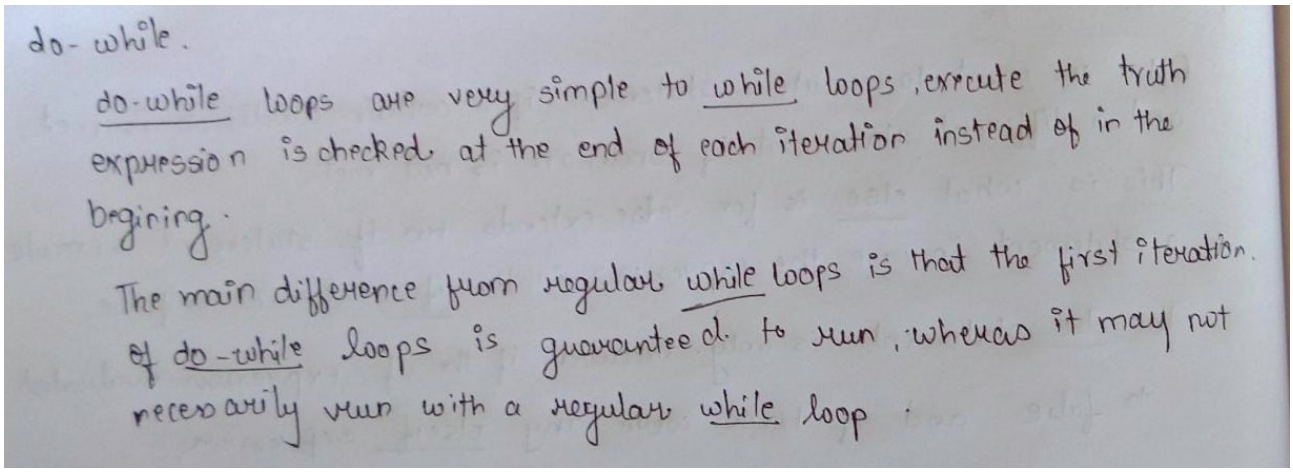
Code:

```
<?php
$num = 4;
$factorial = 1;
for ($x=$num; $x>=1; $x--)
{
  $factorial = $factorial * $x;
}
echo "Factorial of $num is $factorial";
?>
```

Output:



Defination:


do-while.

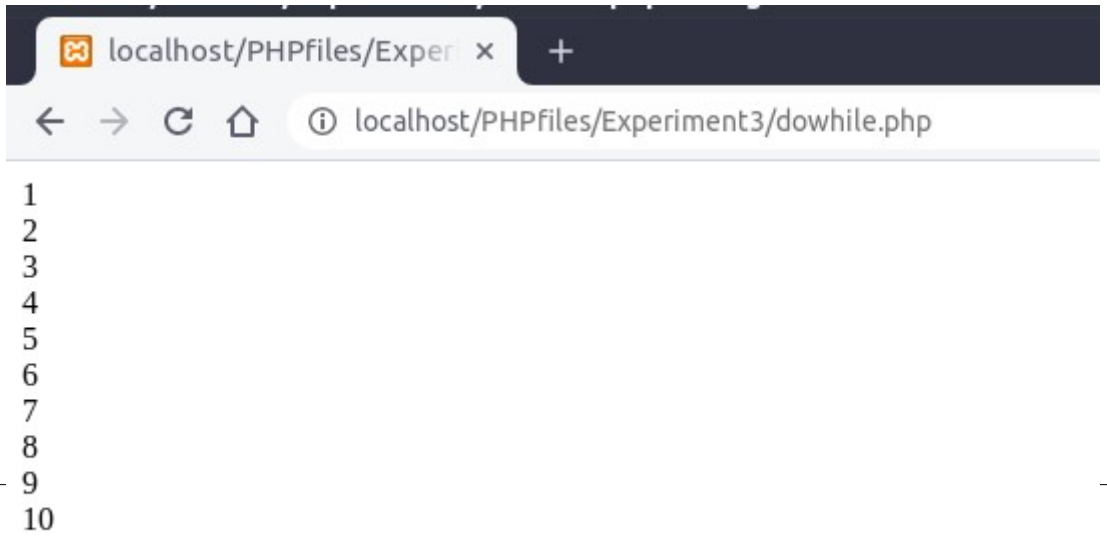
do-while loops are very simple to while loops, execute the truth expression is checked at the end of each iteration instead of in the begining.

The main difference from regular while loops is that the first iteration of do-while loops is guaranteed to run, whereas it may not necessarily run with a regular while loop.

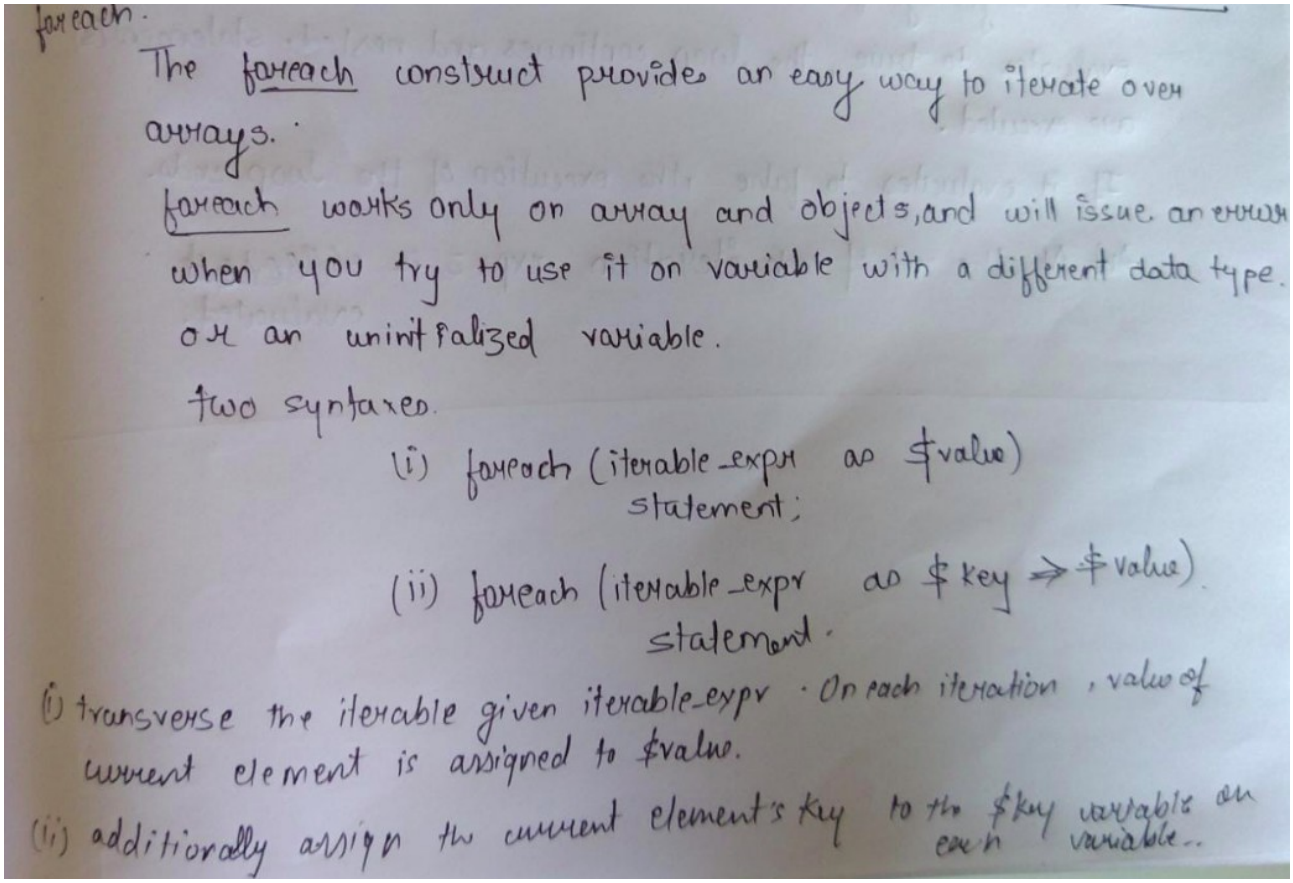
Code:

```
<?php
$i=1;
do {
 echo $i."<br/>";

    $i++;
}
while($i<11);
?>
```

Output:



Defination:


foreach.

The foreach construct provides an easy way to iterate over arrays.

foreach works only on array and objects, and will issue an error when you try to use it on variable with a different data type or an uninitialized variable.

two syntaxes.

(i) foreach (iterable_expr as $value)
statement;

(ii) foreach (iterable_expr as $key => $value)
statement.

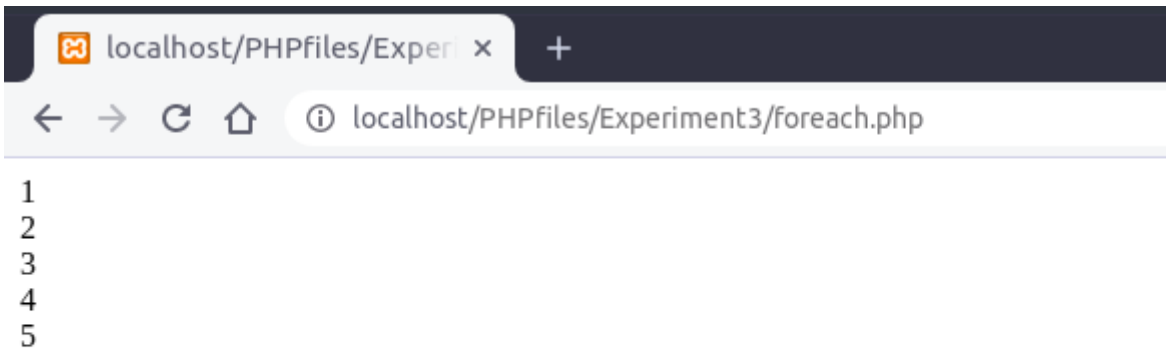
(i) transverse the iterable given iterable_expr. On each iteration, value of current element is assigned to $value.

(ii) additionally assign the current element's key to the $key variable on each iteration.

Code:

```
<?php
$arr=array(1,2,3,4,5);
{
foreach ($arr as $value)
    echo $value."<br/>";
}
?>
```

Output:



Conclusion: implemented looping structures in multiple programs.

Practical 4

1.WAP to find largest element of an array using index array

b.Any program using associative array

C.Any program using multidimensional array

Defination :

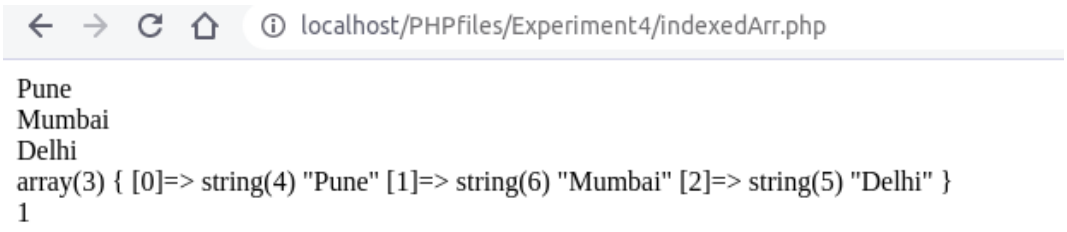
PHP indexed array is an array which is represented by an index number by default. All elements of array are represented by an index number which starts from 0.

PHP indexed array can store numbers, strings or any object. PHP indexed array is also known as numeric array.

Code:

```
<?php
$city=array(0=>"Pune",1=>"Mumbai",2=>"Delhi");
echo $city[0]."<br/>".$city[1]."<br/>".$city[2]."<br/>";
var_dump($city);

echo "</br>".is_array($city);
?>
```

Output:



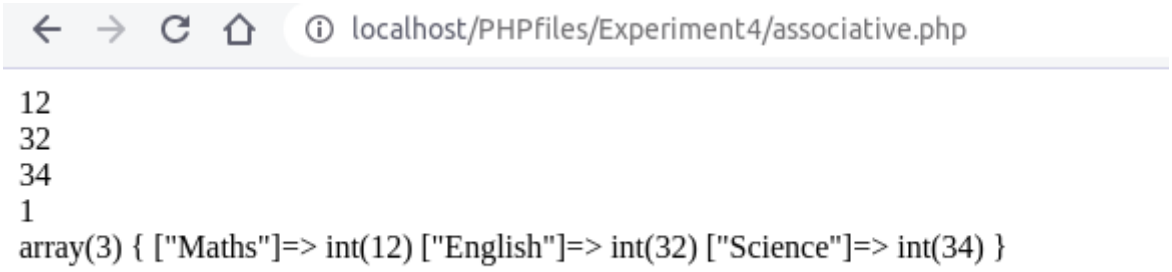
Defination :

PHP allows you to associate name/label with each array elements in PHP using => symbol. Such way, you can easily remember the element because each element is represented by label than an incremented number.

Code:

```
<?php
$marks=array("Maths"=>12,"English"=>32,"Science"=>34);
echo $marks['Maths']."<br/>".$marks['English']."<br/>".$marks['Science']."<br/>";
echo is_array($marks)."<br>";
var_dump($marks);

?>
```

Output:



Defination :

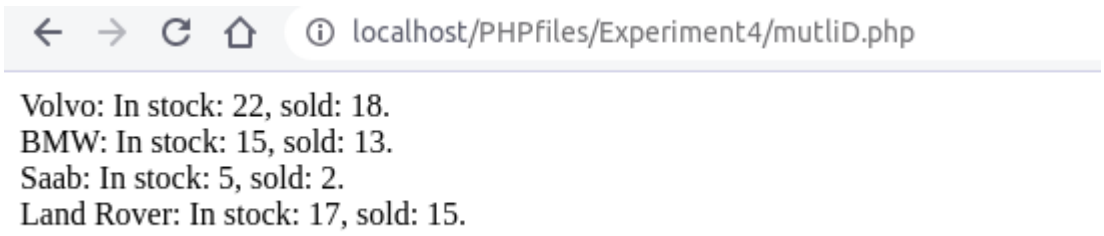
A multidimensional array is an array containing one or more arrays.

PHP supports multidimensional arrays that are two, three, four, five, or more levels deep. However, arrays more than three levels deep are hard to manage for most people.

Code:

```
<?php
$cars = array (
  array("Volvo",22,18),
  array("BMW",15,13),
  array("Saab",5,2),
  array("Land Rover",17,15)
);

echo $cars[0][0].": In stock: ".$cars[0][1].", sold: ".$cars[0][2].".<br>";
echo $cars[1][0].": In stock: ".$cars[1][1].", sold: ".$cars[1][2].".<br>";
echo $cars[2][0].": In stock: ".$cars[2][1].", sold: ".$cars[2][2].".<br>";
echo $cars[3][0].": In stock: ".$cars[3][1].", sold: ".$cars[3][2].".<br>";
?>
```

Output:



Conclusion:executed all types of arrays in multiple programs

Practical 5

Aim:

1.calculate length of string without using built in function

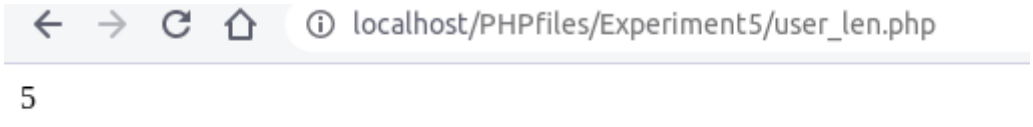
2.count the number of words in string-without using string built in function

3.write a simple PHP program to demonstrate use of various built in string function

Code:

```
<?php
function mystrlen($str)
{
$count = 0;
for($i=0;$i<1000000;$i++)
{
if(@$str[$i] != "")
$count++;
else break;
}
return $count;
}
echo mystrlen("Akash");
?>
```

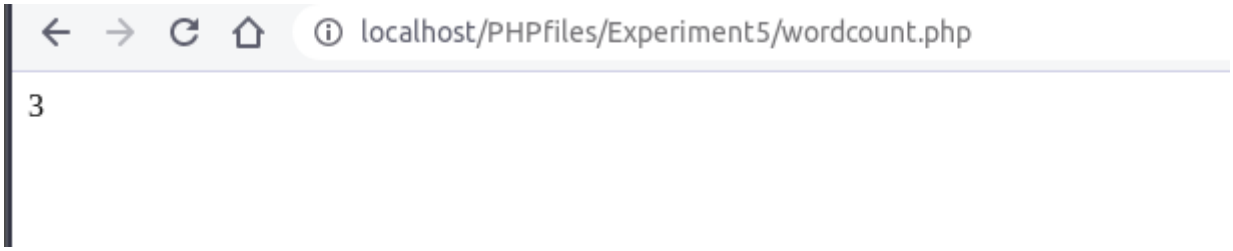
Output:



Code:

```
<?php
// PHP program to count number
// of words in a string

// Function to count the words
function get_num_of_words($string) {
    $string = preg_replace('/\s+/', ' ', trim($string));
    $words = explode(" ", $string);
    return count($words);
}

$str = "  Geeks  for    Geeks  ";

// Function call
$len = get_num_of_words($str);

// Printing the result
echo $len;
?>
```

Output:



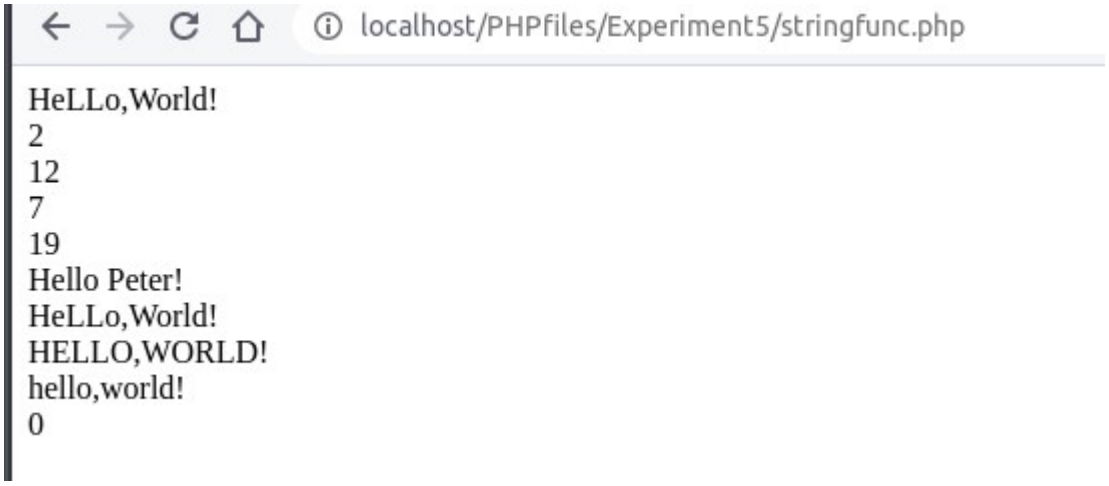
Defination :

1. **str_word_count()**: returns number of word used in the string
2. **strlen()**:returns number of character i.e length of the string
3. **strpos()**:find the position of the first occurence of a substring in a string
4. **strrpos()**:find the position last occurence of the substring in the main string.
5. **str_replace()**:replaces some characters with other characters
6. **ucwords()**:converts the first character to uppercase.
7. **stroupper()**:converts the entire string into uppercase.
8. **strolower()**:converts the entire string into lowercase.
9. **strcmp()**:compares two strings and tells which one is greater ,less and equal .

Code:

```
<?php
$word="HeLLo,World!";
echo $word."</br>";
echo str_word_count($word)."<br>";
echo strlen($word)."</br>";
echo strpos("I love php, I love php too!","php")."</br>";
echo strrpos("I love php, I love php too!","php")."</br>";
echo str_replace("world","Peter","Hello world!")."</br>";
echo ucwords($word)."</br>";
echo strtoupper($word)."</br>";
echo strtolower($word)."</br>";
echo strcmp("Hello world!","Hello world!")."</br>";
?>
```

Output:



Conclusion:

implemented all the string manupilation programns with and without built-in functions.