

Experiment No: 06

Write a simple PHP program to demonstrate use of simple function and parameterized function.

Resources required:

Hardware	Software
Computer System	Any database tools such as XAMPP

Practical Significance:

PHP functions are similar to other programming languages. A function is a piece of code which takes one more input in the form of parameter and does some processing and returns a value.

Theoretical Background:

- They are built-in functions but PHP gives you option to create your own functions as well.
- A function will be executed by a call to the function. You may call a function from anywhere within a page.
- There are two parts which should be clear to you
- Creating a PHP Function
- Calling a PHP Function
- It's very easy to create your own PHP function. Suppose you want to create a PHP function which will simply write a simple message on your browser when you will call it. Following example creates a function called writeMessage() and then calls it just after creating it.
- A user-defined function declaration starts with the word function :

Syntax:

```
function functionName()  
{  
    code to be executed;  
}
```

PHP Functions with Parameters:

PHP gives you option to pass your parameters inside a function. You can pass as many as parameters you like. These parameters work like variables inside your function. Following example takes two integer parameters and add them together and then print them.

Program Code:

```
<?php

function palin ($str)
{
$str2= strrev("MOM");
if ($str == $str2)
{
    echo "Given string is plaineindrome";
}
else
{
    echo "Given string is not plaineindrome";
}
}
echo palin("MOM");

?>
```

Output :

Given string is plaineindrome

Practical related questions:

1. What is anonymous function?

There are times when you need to create a small localized throw-away function consisting of a few lines for a specific purpose, such as a callback. It is unwise to pollute the global namespace with these kind of single use functions. For such an event you can create **anonymous or lambda functions using create_function ()**. Anonymous functions allow the creation of functions which have no specified name. An example is as follows :

Example 1 :

```
<?php
$str = "hello world!";
$lambda = create_function('$match', 'return "friend!";');
$str = preg_replace_callback('/world/', $lambda, $str);
echo $str ;
?>
```

Output :

```
hello friend!
```

2. Write the difference between built in function & user defined function.

Built in function is a function which is created prior, and all code blocks is **predefine** while downloading php. To use built in function **user just have call the function** for example all **string function** and **array function**.

User defined is a function in which **developer must create the function first then write the code block and then developer have to call the function**.

3. What is variable function?

- PHP supports the concept of variable functions. This means that if a variable name has parentheses appended to it, PHP will look for a function with the same name as whatever the variable evaluates to, and will attempt to execute it. Among other things, this can be used to implement callbacks, function tables and so forth.
- Variable functions won't work with language constructs such as echo, print, unset(), isset(), empty(), include, require and the like. Utilize wrapper functions to make use of any of these constructs as variable functions.

Example 1 :

```
<?php
function simple()
{
    echo "In simple()<br />\n";
}
function data($arg = "")
{
    echo "In data(); argument was '$arg'.<br />\n";
}
$func = 'simple';
$func();    // This calls simple()

$func = 'data';
$func('test'); // This calls data()
?>
```

Output :

```
In simple()
In data(); argument was 'test'.
```

Exercise:

1. Write a code to perform addition of 3 numbers using function.

CODE-:

```
<?PHP

function gnum ($n1,$n2,$n3)
{
    if ($n1>$n2 & $n3)
    {
        echo $n1." is the largest number";
    }
    elseif ($n2>$n1 & $n3)
    {
        echo $n2." is the largest number";
    }
    else
    {
        echo $n3." is the largest number";
    }
}

echo gnum(5,3,8)

?>
```

OUTPUT-:

8 is the largest number

2. Write a PHP program to check whether number is even or odd using function.

CODE-:

```
<?PHP

function numeven ($n1)
{
    if ($n1%2==0)
    {
        echo $n1." is even number";
    }
    else
    {
        echo $n1." is odd number";
    }
}

echo numeven (8)

?>
```

OUTPUT-:

8 is even number

3. Write a PHP program to print factorial of number using function.

CODE-:

```
<?PHP

function facto ($n)
{
    if($n <= 1)
    {
        return 1;
    }
    else
    {
        return $n * facto( $n - 1);
    }
}

echo "Factorial of given number is ". facto (8)

?>
```

OUTPUT-:

Factorial of given number is 40320

4. Write PHP program to calculate the sum of digits using function.

CODE-:

```
<?PHP

function sumofnum ($n)
{
    $sum=0;
    $rem=0;

    for ($i =0; $i<=strlen($n);$i++)
    {
        $rem=$n%10;
        $sum = $sum + $rem;
        $n=$n/10;
    }

    return $sum;
}

echo "Sum of given number is ". sumofnum (852548)

?>
```

OUTPUT-:

Sum of given number is 32

5. PHP program to check whether a number is prime or Not.

CODE:-

```
<?php

function primenum($n){
    if ($n == 1)
        return 0;
    for ($i = 2; $i <= $n/2; $i++){
        if ($n % $i == 0)
            return 0;
    }
    return 1;
}

// Driver Code

$flag = primenum(17);

if ($flag == 1)
    echo "Given number is Prime";
else
    echo "Given number is Not Prime"
?>
```

OUTPUT:-

Given number is Prime