Introduction to function

Objective

In this chapter learner able to understand:

- Function
- Passing Parameters to functions
- Return value through function
- Date and Time with PHP
- PHP “include” and “require”

Function in PHP

Function can be defined as small peace of code within your program which performs a predefined task within program.

Function can be understood as smallest execution unit of your program.

Type of the functions:

1.Built-In Functions: These are the function which are provided to us by language itself we just use these function within our program.

array() , strstr() , include() , require() etc.

2. User-Defined Functions:These are the function which definition is defined by the user itself according to their need.

Creating Function

Syntax:

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

Rule:

Give the function a name that reflects what the function does. The function name can start with a letter or underscore (not a number)

User-Defined Function Type

1-Normal function
2-function with return value
3-function with parameter
4-function with parameter and return value
5-function with reference parameter
6-function with return multiple value
7-Conditional functions
8-function with default parameter
9-Variable Function

Functions - Adding parameters

A parameter is just like a variable.

❖Parameters are specified after the function name, inside the parentheses.

Syntax:

```
function functionname(parameter1, parameter1…….. )
{
  //code
}
```

Example

```
<?php
    function writeName($fname)
    {
        echo $fname." Verma.<br />";
    }
    echo "My name is ";
    writeName(“AAAA");
    echo "My sister's name is ";
    writeName(“EEEE");
    echo "My brother's name is ";
    writeName(“DDDD");
?>
```

Functions - Return values

To let a function return a value, use the return statement.

```
<?php
function add($x,$y)
{
$total=$x+$y;
return $total;
}
echo "1 + 16 = " . add(1,16);
?>
```

Returning Values By Reference

PHP also allows you to return variables by reference using & symbol.

```
function &myfun($name)
{
        $name+=1;
        return $name;
}
$num = 10;
$value =& myfun($num);
print $value . “<br>";
```

Function arguments by reference

```
<?php
    function add_some_extra(&$string)
    {
        $string .= ”and something extra.”;
    }
    $str = ”This is a string, ”;
    add_some_extra($str);
    echo $str;
 ?>
```

Conditional functions

When a function is defined in a conditional manner such as the two examples shown. Its definition must be processed prior to being called.

Example :-

```
<?php
$maketest = true;
show();
test();          //Fatal Error function not define
if ($maketest)
        {
         function test()
        {
             echo "I don't exist until program execution reaches me.<br>";
       }
}
test();
function show()
{           echo "I exist immediately upon program start.<br>“;    ?>
```

Function default argument

```
<?php
	function printname($name = "PHP")
	{
		return "My name is $name.<br>";
	}
	echo printname();
	echo printname(null);
	echo printname(“Rahul");
?>
```

Incorrect usage of default arguments

```
<?php
    function printname($sirname = "PHP",$name)
    {
        return "My name is $name $sirname<br>";
    }
    echo printname(“Anuj");
?>
```

Correct usage of default arguments

```
<?php
    function printname($name,$sirname = "PHP")
    {
        return "My name is $name $sirname<br>";
    }
    echo printname(“Web");
?>
```

Returning an array to get multiple values

```
<?php
	function small_numbers()
	{
		return array (0, 1, 2);
	}
	list ($first, $second, $third) = small_numbers();
	echo $second;
	$arr=small_numbers();
	print_r($arr);
?>
```

Variable functions

PHP supports the concept of variable functions.
Example :-

```
<?php
function first() {    echo "In first<br />";   }
function secondfun($arg = 'second')
{    echo "In second fun() argument was $arg.<br />“;  }
function thirdfun($string)
{    echo $string;  }
      $myfun = first;
      $myfun();
      $myfun = "secondfun";
      $myfun();
      $myfun = 'thirdfun';
      $myfun('test');
?>
```

Some use full built-in functions

- **include():**

  The **include()** statement includes and evaluates the specified file.

  Example

  vars.php

```
<?php
$color = 'green';
$fruit = 'apple';
?>
```

  test.php

```
<?php
echo "A $color $fruit";
// A
include 'vars.php';
echo "A $color $fruit";
// A green apple
?>
```

Built-in functions (count..)

require() :

The require() statement includes and evaluates the specific file.

require() examples:

```
<?php
        require 'prepend.php';
        require $somefile;
        require ('somefile.txt');
?>
```

Built-in functions (count..)

require_once()

- The **require_once()** statement includes and evaluates the specified file during the execution of the script.
- This is a behavior similar to the **require()** statement
- difference being that if the code from a file has already been included, it will not be included again.

Example:

```
<?php
        require_once "a.php";
        // this will include a.php
        require_once "A.php";
        // this will include a.php again on
        Windows! (PHP 4 only)
?>
```

Built-in functions (count..)

- PHP has nice built in date function which allows you to display dates in human readable formats.

  Date Function Syntax: date(format, timestamp)
- format - Always required. Specify the format to display the in.
- timestamp - Optional. Specify UNIX time stamp. If not passed, the current timestamp is used.
- Function Parameters
- d - The day of the month, i.e. 01-31
- m - Month representation in numbers, i.e. 01-12
- Y - Year in four digits

Built-in functions (count..)

Date Function Formatting

1. **DAYS**
   d - day of the month 2 digits (01-31)
   j - day of the month (1-31)
   D - 3 letter day (Mon - Sun)
   l - full name of day (Monday - Sunday)
   N - 1=Monday, 2=Tuesday, etc (1-7)
   S - suffix for date (st, nd, rd)
   w - 0=Sunday, 1=Monday (0-6)
   z - day of the year (1=365)
2. **WEEK**
   W - week of the year (1-52)

Built-in functions (count..)

MONTH
F - Full name of month (January - December)
m - 2 digit month number (01-12)
n - month number (1-12)
M - 3 letter month (Jan - Dec)
t - Days in the month (28-31)

YEAR
L - leap year (0 no, 1 yes)
Y - four digit year (Ex. 1979, 2006)
y - two digit year (Ex. 79, 06)

Built-in functions (count..)

TIME

- a - am or pm
- A - AM or PM
- g - 12 hour (1-12)
- G - 24 hour c (0-23)
- h - 2 digit 12 hour (01-12)
- H - 2 digit 24 hour (00-23)
- i - 2 digit minutes (00-59)
- s - 2 digit seconds (00-59)

Chapter Summary

In this chapter, you have learned about:

- A function is a self-contained block of statements that performs a specific task of some kind
- A function is capable to return some value to the calling statement.
- We can pass parameters to the function and return appropriate value to the calling statements.
- The getdate() function of PHP gives an array as output.
- PHP date() function makes a timestamp to a more readable date- time.
- include() function uses all the contents of one file into another file where it included.
- The functionality of require() function is similar to include() means it uses all the contents of one file into another file where it included.

Thank you