

PHP Variables Scope

[Back to ACP page](#)

Table of Contents

- [PHP Variables Scope](#)
 - [PHP 1 Global Scope](#)
 - [Example 1](#)
 - [PHP 2 Local Scope](#)
 - [Example 2](#)
 - [PHP 3](#)
 - [Example 3](#)
 - [PHP 4 The static Keyword](#)
 - [Example 4](#)
 - [Document](#)
 - [Reference](#)

In PHP, variables can be declared anywhere in the script.

The scope of a variable is the part of the script where the variable can be referenced/used.

PHP has three different variable scopes:

- local
- global
- static

PHP 1 Global Scope

```
<h4>Global_Scope</h4>
<p>Variable with global scope: </p>
<?php
    $x = 5; // global scope

    function myTest() {
        // using x inside this function will generate an error
        echo "<p>Variable x inside function is: $x</p>";
    }
    myTest();

    echo "<p>Variable x outside function is: $x</p>";

?>
```

Global_Scope

Variable with global scope:

Variable x inside function is:

Variable x outside function is: 5

Example 1

Result [View Example](#)

PHP 2 Local Scope

A variable declared within a function has a LOCAL SCOPE and can only be accessed within that function:

```
<h2>Local_Scope</h2>
<p>Variable with local scope: </p>
<?php
function myTest() {
    $x = 5; // local scope
    echo "<p>Variable x inside function is: $x</p>";
}
myTest();

// using x outside the function will generate an error
echo "<p>Variable x outside function is: $x</p>";
?>
```

Local_Scope

Variable with local scope:

Variable x inside function is: 5

Variable x outside function is:

Example 2

Result [View Example](#)

PHP 3

Lorem ipsum dolor sit, amet consectetur adipisicing elit.

```
<h2>The_global_Keyword</h2>
<p>From w3schools.com, Experiment by Teeratus_R </p>
<?php
$x = 5;
$y = 10;

function myTest() {
global $x, $y;
$y = $x + $y;
}

myTest(); // run function
echo $y; // output the new value for variable $y
?>
```

The_global_Keyword

From w3schools.com, Experiment by Teeratus_R

15

Example 3

Result [View Example](#)

PHP 4 The static Keyword

Normally, when a function is completed/executed, all of its variables are deleted. However, sometimes we want a local variable NOT to be deleted. We need it for a further job.

To do this, use the static keyword when you first declare the variable:

```
<h2>The_static_Keyword</h2>
<p>To do this, use the static keyword when you first declare the variable:
</p>
<?php
function myTest() {
static $x = 0;
echo $x;
$x++;
}

myTest();
echo "<br>";
myTest();
echo "<br>";
```

```
myTest();  
?>
```

The `_static_` Keyword

To do this, use the static keyword when you first declare the variable:

```
0  
1  
2
```

Example 4

Result [View Example](#)

Document

Document in project

You can [Download PDF](#) file.

Reference