

PHP - Global Variables

In PHP, any variable that can be accessed from anywhere in a PHP script is called as a **global variable**. If the variable is declared outside all the functions or classes in the script, it becomes a global variable.

While global variables can be accessed directly outside a function, they aren't automatically available inside a function.

Example

In the script below, **\$name** is global for the function **sayhello()**.

</>

Open Compiler

```
<?php
$name = "Amar";
function sayhello() {
    echo "Hello " . $name;
}
sayhello();
?>
```

However, the variable is not accessible inside the function. Hence, you will get an **error message** "Undefined variable \$name".

Hello

PHP Warning: Undefined variable \$name in /home/cg/root/93427/main.php on line 5

Example

To get access within a function, you need to use the "global" keyword before the variable.

</>

Open Compiler

```
<?php
$name = "Amar";
function sayhello() {
```

```
GLOBAL $name;
echo "Hello " . $name;
}
sayhello();
?>
```

It will produce the following **output** –

Hello Amar

If a function accesses a global variable and modifies it, the modified value is available everywhere after the function call is completed.

Let us change the value of **\$name** inside the **sayhello()** function and check its value after the function is called.

Explore our [latest online courses](#) and learn new skills at your own pace. Enroll and become a certified expert to boost your career.

Example

Take a look at this following example –

```
</> Open Compiler

<?php
$name = "Amar";
function sayhello() {
    GLOBAL $name;
    echo "Global variable name: $name" .PHP_EOL;
    $name = "Amarjyot";
    echo "Global variable name changed to: $name" .PHP_EOL;
}
sayhello();
echo "Global variable name after function call: $name" .PHP_EOL;
?>
```

It will produce the following **output** –

Global variable name: Amar

Global variable name changed to: Amarjyot

Global variable name after function call: Amarjyot

The \$GLOBALS Array

PHP maintains an associative array named **\$GLOBALS** that holds all the variables and their values declared in a global scope. The \$GLOBALS array also stores many predefined variables called as superglobals, along with the user defined global variables.

Any of the global variables can also be accessed inside any function with the help of a regular syntax of accessing an array element. For example, the value of the global variable **\$name** is given by **\$GLOBALS["name"]**.

Example

In the following example, two global variable \$x and \$y are accessed inside the addition() function.

[Open Compiler](#)

```
<?php
    $x = 10;
    $y = 20;

    function addition() {
        $z = $GLOBALS['x']+$GLOBALS['y'];
        echo "Addition: $z" .PHP_EOL;
    }
    addition();
?>
```

It will produce the following **output** –

Addition: 30

Example

You can also add any local variable into the global scope by adding it in the \$GLOBALS array. Let us add **\$z** in the global scope.



Open Compiler

```
<?php
    $x = 10;
    $y = 20;
    function addition() {
        $z = $GLOBALS['x']+$GLOBALS['y'];
        $GLOBALS['z'] = $z;
    }
    addition();
    echo "Now z is the global variable. Addition: $z" .PHP_EOL;
?>
```

It will produce the following **output** –

```
Now z is the global variable. Addition: 30
```

Including One PHP Script in Another

You can include one PHP script in another. Variables declared in the included script are added in the global scope of the PHP script in which it is included.

Here is "a.php" file –

```
<?php
    include 'b.php';
    function addition() {
        $z = $GLOBALS['x']+$GLOBALS['y'];
        echo "Addition: $z" .PHP_EOL;
    }
    addition();
?>
```

It includes "b.php" that has the **\$x** and **\$y** variables, so they become the global variables for the **addition()** function in "a.php" script.

```
<?php
    $x = 10;
    $y = 20;
?>
```

Global variables are generally used while implementing singleton patterns, and accessing registers in embedded systems and also when a variable is being used by many functions.