

# PHP - \$GLOBALS

\$GLOBALS is one of the "**superglobal**" or "**automatic global**" variables in PHP. It is available in all scopes throughout a script. There is no need to do "**global \$variable;**" to access it within functions or methods.

\$GLOBALS is an associative array of references to all globally defined variables. The names of variables form keys and their contents are the values of an associative array.

## Example

This example shows \$GLOBALS array containing the name and contents of global variables –

&lt;/&gt;

Open Compiler

```
<?php
    $var1="Hello";
    $var2=100;
    $var3=array(1,2,3);

    echo $GLOBALS["var1"] . "\n";
    echo $GLOBALS["var2"] . "\n";
    echo implode($GLOBALS["var3"]) . "\n";
?>
```

It will produce the following **output** –

```
Hello
100
123
```

## Example

In the following example, **\$var1** is defined in the global namespace as well as a local variable inside the function. The global variable is extracted from the \$GLOBALS array.

&lt;/&gt;

Open Compiler

```
<?php
function myfunction() {
    $var1="Hello PHP";
    echo "var1 in global namespace: " . $GLOBALS['var1']. "\n";
    echo "var1 as local variable: ". $var1;
}
$var1="Hello World";
myfunction();
?>
```

It will produce the following **output** –

```
var1 in global namespace: Hello World
var1 as local variable: Hello PHP
```

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

## Example

Prior to PHP version 8.1.0, global variables could be modified by a copy of \$GLOBALS array.

```
</>
```

[Open Compiler](#)

```
<?php
$a = 1;
$globals = $GLOBALS;
$globals['a'] = 2;
var_dump($a);
?>
```

It will produce the following **output** –

```
int(1)
```

Here, **\$globals** is a copy of the \$GLOBALS superglobal. Changing an element in the copy, with its key as "a" to 2, actually changes the value of **\$a**.

It will produce the following **output** –

```
int(2)
```

## Example

As of PHP 8.1.0, \$GLOBALS is a read-only copy of the global symbol table. That is, global variables cannot be modified via their copy. The same operation as above won't change **\$a** to 2.

[Open Compiler](#)

```
<?php
    $a = 1;
    $globals = $GLOBALS;
    $globals['a'] = 2;
    var_dump($a);
?>
```

It will produce the following **output** –

```
int(1)
```