



Type Boolean Integer String Array Object/Class String functions conversion Array functions conversion
 Class definition member declaration member visibility Date/Time functions formats Predefined Variables \$_SERVER \$_FILES

Type: Boolean

Values (case insensitive): TRUE, FALSE, true, false

Casts: (bool), (boolean)

The value -1 is considered TRUE, like any other non-zero (whether negative or positive) number!

Type: Integer

Casts: (int), (integer)

If PHP encounters a number beyond the bounds of the integer type, it will be interpreted as a float instead.

Type: String

When a string is specified in double quotes (") the variables are parsed within it.

Converting to string: with cast (string) or *strval()*

String Functions:

array explode(string \$delimiter , string \$string , [int \$limit]) - Returns an array of strings, each of which is a substring of string formed by splitting it on boundaries formed by the string delimiter

string implode(string \$glue , array \$pieces) - Join array elements with a glue string

string htmlentities(string \$string , [int \$quote_style , [string \$charset , [bool \$double_encode]]]) - Convert all applicable characters to HTML entities

string html_entity_decode(string \$string , [int \$quote_style , [string \$charset]]) - Convert all HTML entities to their applicable characters

int strlen(string \$string) - Get string length

string strpos(string \$haystack , mixed \$needle , [bool \$before_needle]) - Find first occurrence of a string

int strpos(string \$haystack , mixed \$needle , [int \$offset]) - Returns the numeric position of the first occurrence of needle in the haystack string

string substr(string \$string , int \$start , [int \$length]) - Return part of a string

Type: Array

An array can be created by the *array()* language construct.

The *unset()* function allows removing keys from an array.

Converting to array: *(array)\$scalarValue* or *array(\$scalarValue)*

Array Functions:

array array_diff(array \$array1 , array \$array2 , [array \$...]) - Compares array1 against array2 and returns the difference

array array_keys(array \$input , [mixed \$search_value , [bool \$strict]]) - Return all the keys of an array

array array_map(callback \$callback , array \$arr1 , [array \$...]) - Applies the callback to the elements of the given arrays

bool array_walk(array &\$array , callback \$funcname , [mixed \$userdata]) - Apply a user function to every member of an array

array array_merge(array \$array1 , [array \$array2 , [array \$...]]) - Merge one or more arrays

mixed array_pop(array &\$array) - Pop the element off the end of array

int array_push(array &\$array , mixed \$var , [mixed \$...]) - Push one or more elements onto the end of array

mixed array_shift(array &\$array) - Shift an element off the beginning of array

mixed array_unshift(array &\$array , mixed \$var , [mixed \$...]) - Prepend one or more elements to the beginning of an array

array array_slice(array \$array , int \$offset , [int \$length , [bool \$preserve_keys]]) - Returns the sequence of elements from the array array as specified by the offset and length parameters

array array_splice(array &\$input , int \$offset , [int \$length , [mixed \$replacement]]) - Removes the elements designated by offset and length from the input array, and replaces them with the elements of the replacement array, if supplied

array in_array(mixed \$needle , array \$haystack , [bool \$strict]) - Checks if a value exists in an array, searches haystack for needle

Type: Object/Class

To instantiate a class use the *new* statement.

If a value of any other type is converted to an object, a new instance of the `stdClass` built-in class is created.

Setting constants inside of class with **const** keyword.

Class definition:

general

```
1 class SampleClass {
2 }
```

abstract - It is not allowed to create an instance of a class that has been defined as abstract

```
1 abstract class SampleClass {
2 }
```

final - Prevents the class to be extended

```
1 final class SampleClass {
2 }
```

Class members declaration:

static - Declaring class members or methods as static makes them accessible without needing an instantiation of the class

```
1 class SampleClass {
2
3     public static $class_property = "foo";
4
5     public static class_method() {
6         //code here
7     }
8 }
```

final - Prevents child classes from overriding a method

```
1 class SampleClass {
2
3     final public class_method() {
4         //code here
5     }
6 }
```

Class members visibility:

public - Public declared items can be accessed everywhere

```
1 class SampleClass {
2
3     public $class_property = "foo";
4
5     public class_method() {
6         //code here
7     }
8 }
```

private - Private limits visibility only to the class that defines the item

```
1 class SampleClass {
2
3     private $class_property = "foo";
4
5     private class_method() {
6         //code here
7     }
8 }
```

protected - Protected limits access to inherited and parent classes (and to the class that defines the item)

```
1 class SampleClass {
2
3     protected $class_property = "foo";
4
5     protected class_method() {
6         //code here
7     }
8 }
```

Date/Time

Functions:

string date(string \$format , [int \$timestamp]) - Returns a string formatted according to the given format string using the given integer timestamp or the current time if no timestamp is given

void date_add(DateTime \$object , DateInterval \$interval) - Adds an amount of days, months, years, hours, minutes and seconds to a DateTime object

int mktime([int \$hour , [int \$minute , [int \$second , [int \$month , [int \$day , [int \$year , [int \$is_dst]]]]]]) - Returns the Unix timestamp corresponding to the arguments given

int strtotime(string \$time , [int \$now]) - The function expects to be given a string containing a US English date format and will try to parse that format into a Unix timestamp

int time() - Return current Unix timestamp

Date/Time Formats:

Day

d - 01 - 31

j - 1 - 31

D - Mon - Sun

l - Monday - Sunday

Month

M - Jan - Dec

F - January - December

m - 01 - 12

n - 1 - 12

Year

Y - 2003, 2008

y - 03, 08

Time

a - am, pm

A - AM, PM

g - 12 hours - 1 - 12

h - 12 hours - 00 - 12

H - 24 hours - 00 - 23

G - 24 hours - 0 - 23

i - minutes - 00 - 59

s - seconds - 00 - 59

Predefined Variables

\$_SERVER:

The values in the array are provided by the server, there's no guarantee that all the values will be available on your configuration. In this list you can find the most used values only.

PHP_SELF - the filename of the currently executing script, relative to the document root

SERVER_ADDR - the IP address of the server

SERVER_NAME - the name of the server

REQUEST_METHOD - which request method was used to access the page: GET, HEAD, POST, PUT

QUERY_STRING - the query string, if any, via which the page was accessed

DOCUMENT_ROOT - the document root directory under which the current script is executing

HTTP_REFERER - the address of the page (if any) which referred the user agent to the current page

REMOTE_ADDR - the IP address from which the user is viewing the current page

SCRIPT_FILENAME - the absolute pathname of the currently executing script

REQUEST_URI - The URI which was given in order to access this page; e.g. */index.html*

\$_FILES:

\$_FILES['userfile']['name'] - the original name of the file on the client machine

\$_FILES['userfile']['type'] - the mime type of the file, if the browser provided this information

\$_FILES['userfile']['size'] - the size, in bytes, of the uploaded file

\$_FILES['userfile']['tmp_name'] - the temporary filename of the file in which the uploaded file was stored on the server

\$_FILES['userfile']['error'] - The error code associated with this file upload

Error codes:

UPLOAD_ERR_OK or value **1** - no error

UPLOAD_ERR_INI_SIZE or value **2** - the uploaded file exceeds the `upload_max_filesize` directive in php.ini

UPLOAD_ERR_PARTIAL or value **3** - the uploaded file was only partially uploaded

UPLOAD_ERR_NO_FILE or value **4** - no file was uploaded

UPLOAD_ERR_NO_TMP_DIR or value **5** - missing temporary folder

UPLOAD_ERR_CANT_WRITE or value **7** - failed to write file to disk

