PHP básico

Índice

[PHP 3](#_Toc71647416)

[Variables 3](#_Toc71647417)

[Enteros (Integer) 3](#_Toc71647418)

[Numero de coma flotante (double) 3](#_Toc71647419)

# PHP

## Variables

En PHP las variables se representan con un signo de dólar seguido por el nombre de la variable. El nombre de la variable es sensible a minúsculas y mayúsculas.

Los nombres de variables siguen las mismas reglas que otras etiquetas en PHP. Un nombre de variable válido tiene que empezar con una letra o un carácter de subrayado (underscore), seguido de cualquier número de letras, números y caracteres de subrayado. Como expresión regular se podría expresar como: '[a-zA-Z\_\x7f-\xff][a-zA-Z0-9\_\x7f-\xff]\*'

## Enteros (Integer)

Los Integer pueden especificarse mediante notación decimal (base 10), hexadecimal (base 16), octal (base 8) o binaria (base 2), opcionalmente precedidos por un signo (- o +).

Los literales de tipo Integer binarios están disponibles desde PHP 5.4.0.

Para utilizar la notación octal, se antepone al número un 0 (cero). Para utilizar la notación hexadecimal, se antepone al número 0x. Para utilizar la notación binaria, se antepone al número 0b.

## Numero de coma flotante (double)

Formalmente a partir de PHP 7.4.0 (anteriormente, no se permitían los guiones bajos):

LNUM [0-9]+(\_ [0-9]+) \*

DNUM ([0-9]\*(\_ [0-9]+) \*[\.]{LNUM}) | ({LNUM}[\.][0-9]\*(\_[0-9]+)\*)

EXPONENT\_DNUM (({LNUM} | {DNUM}) [eE] [+-]? {LNUM})

El tamaño de un 'float' depende de la plataforma, aunque un valor común consiste en un máximo de aproximadamente 1.8e308 con una precisión cercana a los 14 dígitos decimales (el formato de 64 bit del IEEE).

## Cadena de caracteres

Un String, o cadena, es una serie de caracteres donde cada carácter es lo mismo que un byte. Esto significa que PHP solo admite un conjunto de 256 caracteres, y de ahí que no ofrezca soporte nativo para Unicode. Véanse los [detalles del tipo String](https://www.php.net/manual/es/language.types.string.php#language.types.string.details).

## Valores binarios (Boolean)

Una variable de tipo boolean solo puede contener dos valores: Verdadero o Falso. Se suelen utilizar para comprobar si se cumple una condición en el programa.

## Valores nulos (NULL)

Es un tipo especial de datos que solo contiene el valor NULL. Aunque se suele utilizar por norma en mayúscula, PHP lo acepta escrito de diversas formas: NULL, null, NuLl.

## Arrays

Un array en PHP es en realidad un mapa ordenado. Un mapa es un tipo de datos que asocia valores con claves. Este tipo se optimiza para varios usos diferentes; se puede emplear como un array, lista (vector), tabla asociativa (tabla hash - una implementación de un mapa), diccionario, colección, pila, cola, y posiblemente más. Ya que los valores de un array pueden ser otros arrays, también son posibles árboles y arrays multidimensionales.

Una explicación sobre tales estructuras de datos está fuera del alcance de este manual, aunque se proporciona al menos un ejemplo de cada uno de ellos. Para más información, consulte la extensa literatura que existe sobre este amplio tema.

### In\_array ()

Busca dentro de un array un valor pasado como parámetro, y si lo encuentra devuelve el valor TRUE, si no, devuelve FALSE. Toma dos argumentos, el valor a buscar y el array donde buscar.

### List ()

Asigna a las variables indicadas como parámetros los valores del array al que se iguala.

## Variables

En PHP las variables se representan con un signo de dólar seguido por el nombre de la variable. El nombre de la variable es sensible a minúsculas y mayúsculas.

Los nombres de variables siguen las mismas reglas que otras etiquetas en PHP. Un nombre de variable válido tiene que empezar con una letra o un carácter de subrayado (underscore), seguido de cualquier número de letras, números y caracteres de subrayado. Como expresión regular se podría expresar como: '[a-zA-Z\_\x7f-\xff][a-zA-Z0-9\_\x7f-\xff]\*'

### Funciones de variables

#### Isset () y unset ()

La función Isset () nos indica si la variable existe y para eliminarla de la memoria necesitamos utilizar unset ().

#### Is\_integer (), is\_double (), is\_string ()

Estas funciones devuelven TRUE si la variable pasada coincide con el tipo que indica la función.

#### Intval (), doubleval (), strval ()

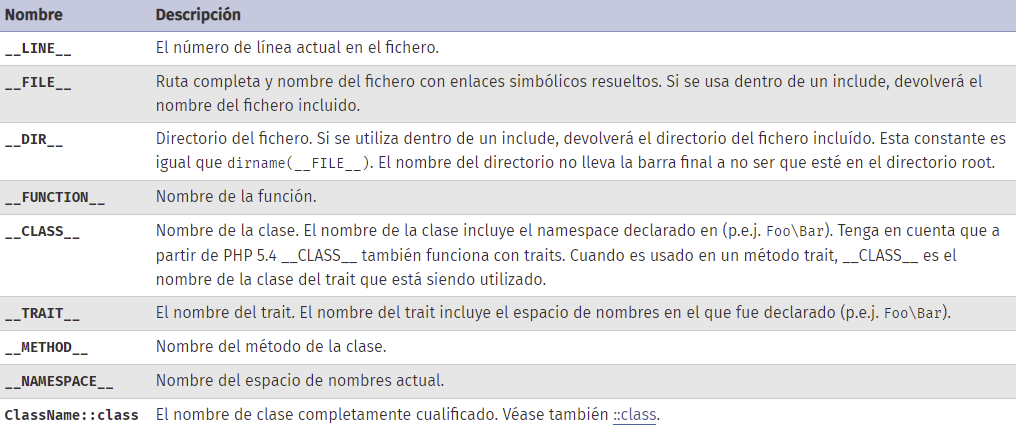
Convierte el valor de una variable al tipo indicado en la función. Estas funciones nos permiten la conversión a los tipos Object o array.

## Constante

Una constante es un identificador (nombre) para un valor simple. Como el nombre sugiere, este valor no puede variar durante la ejecución del script (a excepción de las [constantes mágicas](https://www.php.net/manual/es/language.constants.predefined.php), que en realidad no son constantes). Por defecto, una constante distingue mayúsculas y minúsculas. Por convención, los identificadores de constantes siempre se declaran en mayúsculas.

El nombre de una constante sigue las mismas reglas que cualquier otra etiqueta de PHP. Un nombre de constante válido empieza por una letra o guion bajo, seguido por cualquier número de letras, números o guiones bajos. Usando una expresión regular, se representaría de la siguiente manera: [a-zA-Z\_\x7f-\xff][a-zA-Z0-9\_\x7f-\xff]\*

### Constantes predefinidas



### Defined ()

La función defined () comprueba si una constante está definida o no.