

[floatval »](#)
[« doubleval](#)

- [Manual de PHP](#)
- [Referencia de funciones](#)
- [Extensiones relacionadas con variable y tipo](#)
- [Manejo de variables](#)
- [Funciones de manejo de variables](#)

Change language: Spanish ▼

[Submit a Pull Request](#) [Report a Bug](#)

empty

(PHP 4, PHP 5, PHP 7, PHP 8)

empty — Determina si una variable está vacía

Descripción ¶

empty([mixed](#) \$var): bool

Determina si una variable es considerada vacía. Una variable se considera vacía si no existe o si su valor es igual a **false**. **empty()** no genera una advertencia si la variable no existe.

Parámetros ¶

var

La variable que se comprueba

Nota:

Antes de PHP 5.5, **empty()** sólo soportaba variables; cualquier otra cosa provocaría un error del intérprete. En otras palabras, lo siguiente no funcionaría:
empty(trim(\$nombre)). En su lugar, utilice **trim(\$nombre) == false**.

No se genera una advertencia si la variable no existe. Esto significa que **empty()** es esencialmente el equivalente conciso de **!isset(\$var) || \$var == false**.

Valores devueltos ¶

Devuelve **false** si var existe y tiene un valor no vacío, distinto de cero. De otro modo devuelve **true**.

Los siguientes valores son considerados como vacíos:

- "" (una cadena vacía)
- 0 (0 como un integer)
- 0.0 (0 como un float)
- "0" (0 como un string)
- null
- false
- array() (un array vacío)

Historial de cambios ¶

Versión

Descripción

5.5.0 **empty()** ahora soporta expresiones, en vez de únicamente variables.

5.4.0 Verificación de índices no numéricos de strings devuelve **true**.

Ejemplos ¶

Ejemplo #1 Una simple comparación empty() / [isset\(\)](#)

```
<?php
$var = 0;

// Se evalúa a true ya que $var está vacía
if (empty($var)) {
    echo '$var es o bien 0, vacía, o no se encuentra definida en absoluto';
}

// Se evalúa como true ya que $var está definida
if (isset($var)) {
    echo '$var está definida a pesar que está vacía';
}
?>
```

Ejemplo #2 empty() en índices de string

PHP 5.4 cambia como **empty()** se comporta cuando se pasan índices de string.

```
<?php
$expected_array_got_string = 'somestring';
var_dump(empty($expected_array_got_string['some_key']));
var_dump(empty($expected_array_got_string[0]));
var_dump(empty($expected_array_got_string['0']));
var_dump(empty($expected_array_got_string[0.5]));
var_dump(empty($expected_array_got_string['0.5']));
var_dump(empty($expected_array_got_string['0 Mostel']));
?>
```

Salida del ejemplo anterior en PHP 5.3:

```
bool(false)
bool(false)
bool(false)
bool(false)
bool(false)
bool(false)
bool(false)
```

Salida del ejemplo anterior en PHP 5.4:

```
bool(true)
bool(false)
bool(false)
bool(false)
bool(true)
bool(true)
bool(true)
```

Notas ¶