



# PHP Numbers

[< Previous](#)[Next >](#)

In this chapter we will look in depth into Integers, Floats, and Number Strings.

## PHP Numbers

One thing to notice about PHP is that it provides automatic data type conversion.

So, if you assign an integer value to a variable, the type of that variable will automatically be an integer. Then, if you assign a string to the same variable, the type will change to a string.

This automatic conversion can sometimes break your code.

## PHP Integers

2, 256, -256, 10358, -179567 are all integers.

An integer is a number without any decimal part.

An integer data type is a non-decimal number between -2147483648 and 2147483647 in 32 bit systems, and between -9223372036854775808 and 9223372036854775807 in 64 bit systems. A value greater (or lower) than this, will be stored as float, because it exceeds the limit of an integer.

**Note:** Another important thing to know is that even if  $4 * 2.5$  is 10, the result is stored as float, because one of the operands is a float (2.5).

Here are some rules for integers:

- An integer must have at least one digit
- An integer must NOT have a decimal point
- An integer can be either positive or negative
- Integers can be specified in three formats: decimal (10-based), hexadecimal (16-based - prefixed with 0x) or octal (8-based - prefixed with 0)

PHP has the following predefined constants for integers:

- PHP\_INT\_MAX - The largest integer supported
- PHP\_INT\_MIN - The smallest integer supported
- PHP\_INT\_SIZE - The size of an integer in bytes

PHP has the following functions to check if the type of a variable is integer:

- is\_int()
- is\_integer() - alias of is\_int()
- is\_long() - alias of is\_int()

## Example

Check if the type of a variable is integer:

```
<?php
$x = 5985;
var_dump(is_int($x));

$x = 59.85;
var_dump(is_int($x));

?>
```

Try it Yourself »

# PHP Floats

A float is a number with a decimal point or a number in exponential form.

2.0, 256.4, 10.358, 7.64E+5, 5.56E-5 are all floats.

The float data type can commonly store a value up to 1.7976931348623E+308 (platform dependent), and have a maximum precision of 14 digits.

PHP has the following predefined constants for floats (from PHP 7.2):

- PHP\_FLOAT\_MAX - The largest representable floating point number
- PHP\_FLOAT\_MIN - The smallest representable positive floating point number
- PHP\_FLOAT\_MAX - The smallest representable negative floating point number
- PHP\_FLOAT\_DIG - The number of decimal digits that can be rounded into a float and back without precision loss
- PHP\_FLOAT\_EPSILON - The smallest representable positive number x, so that  $x + 1.0 \neq 1.0$

PHP has the following functions to check if the type of a variable is float:

- is\_float()
- is\_double() - alias of is\_float()

## Example

Check if the type of a variable is float:

```
<?php
$x = 10.365;
var_dump(is_float($x));
?>
```

[Try it Yourself »](#)

## PHP Infinity

A numeric value that is larger than `PHP_FLOAT_MAX` is considered infinite.

PHP has the following functions to check if a numeric value is finite or infinite:

- [`is\_finite\(\)`](#).
- [`is\_infinite\(\)`](#).

However, the PHP `var_dump()` function returns the data type and value:

### Example

Check if a numeric value is finite or infinite:

```
<?php
$x = 1.9e411;
var_dump($x);
?>
```

[Try it Yourself »](#)

## PHP NaN

NaN stands for Not a Number.

NaN is used for impossible mathematical operations.

PHP has the following functions to check if a value is not a number:

- [`is\_nan\(\)`](#).

However, the PHP `var_dump()` function returns the data type and value:

## Example

Invalid calculation will return a NaN value:

```
<?php
$x = acos(8);
var_dump($x);
?>
```

[Try it Yourself »](#)

---

## PHP Numerical Strings

The PHP `is_numeric()` function can be used to find whether a variable is numeric. The function returns true if the variable is a number or a numeric string, false otherwise.

## Example

Check if the variable is numeric:

```
<?php
$x = 5985;
var_dump(is_numeric($x));

$x = "5985";
var_dump(is_numeric($x));

$x = "59.85" + 100;
var_dump(is_numeric($x));

$x = "Hello";
var_dump(is_numeric($x));
?>
```

[Try it Yourself »](#)

**Note:** From PHP 7.0: The `is_numeric()` function will return `FALSE` for numeric strings in hexadecimal form (e.g. `0xf4c3b00c`), as they are no longer considered as numeric strings.

## PHP Casting Strings and Floats to Integers

Sometimes you need to cast a numerical value into another data type.

The `(int)`, `(integer)`, or `intval()` function are often used to convert a value to an integer.

### Example

Cast float and string to integer:

```
<?php
// Cast float to int
$x = 23465.768;
$int_cast = (int)$x;
echo $int_cast;

echo "<br>";

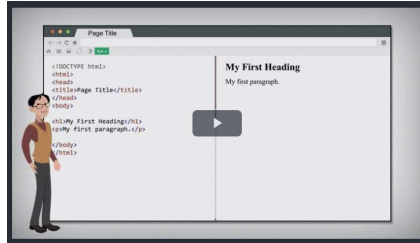
// Cast string to int
$x = "23465.768";
$int_cast = (int)$x;
echo $int_cast;
?>
```

Try it Yourself »

[< Previous](#)[Next >](#)

**NEW**

We just launched  
W3Schools videos



Explore now

COLOR PICKER



Get certified  
by completing  
a PHP  
course today!



Get started

## CODE GAME



Play Game

---

Report Error

Forum

About

Buy Certificate

---

### Top Tutorials

[HTML Tutorial](#)

[CSS Tutorial](#)

[JavaScript Tutorial](#)

[How To Tutorial](#)

[SQL Tutorial](#)

[Python Tutorial](#)

[W3.CSS Tutorial](#)

[Bootstrap Tutorial](#)



[PHP Tutorial](#)  
[Java Tutorial](#)  
[C++ Tutorial](#)  
[jQuery Tutorial](#)

## Top References

[HTML Reference](#)  
[CSS Reference](#)  
[JavaScript Reference](#)  
[SQL Reference](#)  
[Python Reference](#)  
[W3.CSS Reference](#)  
[Bootstrap Reference](#)  
[PHP Reference](#)  
[HTML Colors](#)  
[Java Reference](#)  
[Angular Reference](#)  
[jQuery Reference](#)

## Top Examples

[HTML Examples](#)  
[CSS Examples](#)  
[JavaScript Examples](#)  
[How To Examples](#)  
[SQL Examples](#)  
[Python Examples](#)  
[W3.CSS Examples](#)  
[Bootstrap Examples](#)  
[PHP Examples](#)  
[Java Examples](#)  
[XML Examples](#)  
[jQuery Examples](#)

## Web Courses

[HTML Course](#)  
[CSS Course](#)  
[JavaScript Course](#)  
[Front End Course](#)  
[SQL Course](#)  
[Python Course](#)  
[PHP Course](#)  
[jQuery Course](#)  
[Java Course](#)  
[C++ Course](#)  
[C# Course](#)  
[XML Course](#)

[Get Certified »](#)

W3Schools is optimized for learning and training. Examples might be simplified to improve reading and learning. Tutorials, references, and examples are constantly reviewed to avoid errors, but we cannot warrant full correctness of all content. While using W3Schools, you agree to have read and accepted our terms of use, cookie and privacy policy.

Copyright 1999-2022 by Refsnes Data. All Rights Reserved.  
W3Schools is Powered by W3.CSS.

