

# PHP - Spaceship Operator

The Spaceship operator is one of the many new features introduced in PHP with its 7.0 version. It is a three-way comparison operator.

The conventional comparison operators (<, >, !=, ==, etc.) return true or false (equivalent to 1 or 0). On the other hand, the spaceship operator has three possible return values: -1, 0, or 1. This operator can be used with integers, floats, strings, arrays, objects, etc.

## Syntax

The symbol used for spaceship operator is "<=>".

```
$retval = operand1 <=> operand2
```

Here, \$retval is -1 if operand1 is less than operand2, 0 if both the operands are equal, and 1 if operand1 is greater than operand2.

The spaceship operator is implemented as a combined comparison operator. Conventional comparison operators could be considered mere shorthands for <=> as the following table shows –

Operator	<=> equivalent
\$a < \$b	(\$a <=> \$b) === -1
\$a <= \$b	(\$a <=> \$b) === -1    (\$a <=> \$b) === 0
\$a == \$b	(\$a <=> \$b) === 0
\$a != \$b	(\$a <=> \$b) !== 0
\$a >= \$b	(\$a <=> \$b) === 1    (\$a <=> \$b) === 0
\$a > \$b	(\$a <=> \$b) === 1

## Example 1

The following example shows how you can use the spaceship operator in PHP –

[Open Compiler](#)

```
<?php
    $x = 5;
    $y = 10;
    $z = $x <=> $y/2;

    echo "$x <=> $y/2 = $z";
?>
```

It will produce the following **output** –

```
5 <=> 10/2 = 0
```

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

## Example 2

Change  $x=4$  and check the result –

```
</>
```

[Open Compiler](#)

```
<?php
    $x = 4;
    $y = 10;
    $z = $x <=> $y/2;

    echo "$x <=> $y/2 = $z";
?>
```

It will produce the following **output** –

```
4 <=> 10/2 = -1
```

## Example 3

Change  $y=7$  and check the result again –

```
</>
```

[Open Compiler](#)

```
<?php
    $x = 7;
    $y = 10;
    $z = $x <=> $y/2;

    echo "$x <=> $y/2 = $z";
?>
```

It will produce the following **output** –

```
7 <=> 10/2 = 1
```

## Example 4

When used with string operands, the spaceship operand works just like the strcmp() function.

&lt;/&gt;

Open Compiler

```
<?php
    $x = "bat";
    $y = "ball";
    $z = $x <=> $y;

    echo "$x <=> $y = $z";
?>
```

It will produce the following **output** –

```
bat <=> ball = 1
```

## Example 5

Change \$y = "baz" and check the result –

&lt;/&gt;

Open Compiler

```
<?php
    $x = "bat";
    $y = "baz";
    $z = $x <=> $y;

    echo "$x <=> $y = $z";
?>
```

It will produce the following **output** –

```
bat <=> baz = -1
```

## Spaceship Operator with Boolean Operands

The spaceship operator also works with Boolean operands –

```
true <=> false returns 1
false <=> true returns -1
true <=> true as well as false <=> false returns 0
```