Functional Operators

PHP 8.5 and later supports one operator that works directly on callables. The `|ENTITY` operator, or “pipe,” accepts a single-parameter callable on the right and passes the left-side value to it, evaluating to the callable's result. The callable on the right may be any valid PHP callable: a `Closure`, a first-class callable, an object that implements __invoke(), etc.

That means the following two lines are logically equivalent.

Using `|ENTITY`

```
<?php
$result = "Hello World" |> strlen(...);
echo $result, PHP_EOL;

$result = strlen("Hello World");
echo $result, PHP_EOL;
?>
```

ENTITY

```
11
11
```

For a single call that is not especially useful. It becomes useful when multiple calls are chained together. That is, the following two code fragments are logically equivalent:

Chaining |ENTITY calls

```
<?php
$result = "PHP Rocks"
    |> htmlentities(...)
    |> str_split(...)
    |> (fn($x) => array_map(strtoupper(...), $x))
    |> (fn($x) => array_filter($x, fn($v) => $v != 'O'))
;
```

```
print_r($result);

$temp = "PHP Rocks";
$temp = htmlentities($temp);
$temp = str_split($temp);
$temp = array_map(strtoupper(...), $temp);
$temp = array_filter($temp, fn($v) => $v != 'O');
$result = $temp;
print_r($result);
?>
```

ENTITY

```
Array
(
    [0] => P
    [1] => H
    [2] => P
    [3] =>
    [4] => R
    [6] => C
    [7] => K
    [8] => S
)
Array
(
    [0] => P
    [1] => H
    [2] => P
    [3] =>
    [4] => R
    [6] => C
    [7] => K
    [8] => S
)
```

The left-hand side of the pipe may be any value or expression. The right-hand side may be any valid PHP callable that takes a single parameter, or any expression that evaluates to such a callable. Functions with more than one required parameter are not allowed and will fail as if the

function were called normally with insufficient arguments. Functions that take a variable by reference are not allowed. If the right-hand side does not evaluate to a valid callable it will throw an Error.

> [!NOTE] Be aware that, to avoid syntax ambiguity, arrow functions MUST be wrapped in parentheses when used with a pipe operator, as in the examples above. Failing to do so will result in a fatal error.

ENTITY

`Closure`

□□□ #2
□ □□□ □□□ 2026-10-31 17:41:18 UTC
□ □□□ □□□ 2026-10-31 17:41:20 UTC