

PHP Strings

In this tutorial you will learn how to store and manipulate strings in PHP.

What is String in PHP

A string is a sequence of letters, numbers, special characters and arithmetic values or combination of all. The simplest way to create a string is to enclose the string literal (i.e. string characters) in single quotation marks ('), like this:

```
$my_string = 'Hello World';
```

You can also use double quotation marks ("). However, single and double quotation marks work in different ways. Strings enclosed in single-quotes are treated almost literally, whereas the strings delimited by the double quotes replaces variables with the string representations of their values as well as specially interpreting certain escape sequences.

The escape-sequence replacements are:

- `\n` is replaced by the newline character
- `\r` is replaced by the carriage-return character
- `\t` is replaced by the tab character
- `\$` is replaced by the dollar sign itself (`$`)
- `\"` is replaced by a single double-quote (`"`)
- `\\` is replaced by a single backslash (`\`)

Here's an example to clarify the differences between single and double quoted strings:

Example	Run this code »
<pre><?php \$my_str = 'World'; echo "Hello, \$my_str!
"; // Displays: Hello World! echo 'Hello, \$my_str!
'; // Displays: Hello, \$my_str! echo '<pre>Hello\tWorld!</pre>'; // Displays: Hello\tWorld! echo "<pre>Hello\tWorld!</pre>"; // Displays: Hello World! echo 'I\'ll be back'; // Displays: I'll be back ?></pre>	

Manipulating PHP Strings

PHP provides many built-in functions for manipulating strings like calculating the length of a string, find substrings or characters, replacing part of a string with different characters, take a string apart, and many others. Here are the examples of some of these functions.

Calculating the Length of a String

The `strlen()` function is used to calculate the number of characters inside a string. It also includes the blank spaces inside the string.

Example	Run this code »
<pre><?php \$my_str = 'Welcome to Tutorial Republic'; // Outputs: 28 echo strlen(\$my_str); ?></pre>	

Counting Number of Words in a String

The `str_word_count()` function counts the number of words in a string.

Example	Run this code »
<pre><?php \$my_str = 'The quick brown fox jumps over the lazy dog.'; // Outputs: 9 echo str_word_count(\$my_str); ?></pre>	

Replacing Text within Strings

The `str_replace()` replaces all occurrences of the search text within the target string.

Example	Run this code »

```
<?php
$my_str = 'If the facts do not fit the theory, change the facts.';

// Display replaced string
echo str_replace("facts", "truth", $my_str);
?>
```

The output of the above code will be:

If the truth do not fit the theory, change the truth.

You can optionally pass the fourth argument to the `str_replace()` function to know how many times the string replacements was performed, like this.

Example

[Run this code »](#)

```
<?php
$my_str = 'If the facts do not fit the theory, change the facts.';

// Perform string replacement
str_replace("facts", "truth", $my_str, $count);

// Display number of replacements performed
echo "The text was replaced $count times.";
?>
```

The output of the above code will be:

The text was replaced 2 times.

Reversing a String

The `strrev()` function reverses a string.

Example

[Run this code »](#)

```
<?php
$my_str = 'You can do anything, but not everything.';

// Display reversed string
echo strrev($my_str);
?>
```

The output of the above code will be:

```
.gnihtyreve ton tub ,gnihtyna od nac uoY
```

PHP String Reference

For a complete list of useful string functions, please check out [PHP String Reference](#).