PHP › String

String ref ››

str_replace (search, replace, str)

```
$str = str_replace('foo', 'bar', 'foo bar'); // bar bar
```

Multiple string

```
$arr = array('e', 'o', 'd');
$str = str_replace($arr, '', 'Hello World'); // Hll Wrd
```

strtoupper (str) | strtolower (str)

```
$str = strtoupper('foo'); // FOO
$str = strtolower('FOO'); // foo
```

ucfirst(str) | ucwords(str)

ucfirst: uppercase first letter, **ucwords**: uppercase the first letter of each words

```
$str = ucfirst('hello world'); // Hello world
$str = ucwords ('hello world'); // Hello World
```

trim(str) | ltrim(str) | rtrim(str)

trim strp white space at the beginning and end of string, **ltrim** (left trim) at beginning, and **rtrim** (right trim) at end of string. **Optional**: second parameter is character to be trimmed

```
$str = trim('  foo  '); // 'fpp'
$str = ltrim('  foo  '); // 'foo  '
$str = rtrim('  foo  '); // '  foo'
$str = rtrim('foo+ ' , '+ '); // 'foo'
```

strpos (str, needle) | stripos (str, needle)

strpos and stripos are used to find first occurrence of a substring in a string. strops (case sensitive), stripos (case insensitive)

```
$pos = strpos('foo', 'o'); // 1
$str = 'foo';
if (strpos($str, 'foo') !== false) {
      echo 'exists'
} // 'exists'
```

substr (str, start, length)

Take part of string

```
$str = substr('abcdef', 0, 3); // abc
$str = substr('abcdef', -3); // def
$str = substr('abcdef', 4); // ef
```

strlen (str)

Get string length

```
$str = strlen('hello world'); // 11
```

strstr (str, needle, before_string = false)

Find first occurrence of a string

```
$str= strstr('name@email.com','@');// @email.com
$str= strstr('name@email.com','@', true);// name
```

nl2br (str, is_xhtml = true)

New line to HTML break (<br/>)

```
$str = nl2br ("foo\r\nbar"); // foo<br />bar
```

explode (delimiter, str)

implode (string = '', array)

explode: split string into array, implode: join array values into a string

```
$str = explode (',', 'foo, bar'); // Array ([0]=>foo [1]=>bar)
$arr = array('foo', 'bar')
$str = implode (', ', $arr); // foo, bar
$str = implode ($arr); // foobar
```

. (dot)

Used to concat strings

```
$str = 'Hello';
$str = $str . ' World'; // Hello World
$str .= 'PHP' // Hello World PHP
```

'' | ""

Single quote only store string, double quote can store variable

```
$var = 'Hello';
$str = $var . ' World'; // Hello World
$str = "$var bar"; // Hello World
```

echo | print_r | var_dump

echo for print string only. print_r and var_dump for print array dan object. var_dump will display the data tipe and data length

```
echo 'foo bar';
$arr = array('foo', 'bar');
echo '<pre>'; print_r($arr); // Array ( [0] => foo [1] => bar
echo '<pre>'; var_dump($arr); // array(2) { [0]=> string(3) "foo" [1]=> string(3) "bar" }
```

is_string (var)

Test whether the type of variable is a string

```
if (is_string($var)) {
      echo 'string';
}
```