

ԱՇԽԱՏԱՆՔ ՏՈՂԵՐԻ ՀԵՏ

Մաս 3

Հիմնական ֆունկցիաները

Strtolower(), mb_strtolower(), strtoupper(), mb_strtoupper()

և ucwords() ֆունկցիաները

Ֆունկցիայի տեսքը հետևյալն է.

strtolower(տող);

Strtolower ֆունկցիան տողը փոխում է ներքին ռեգիստրի:

Այս ֆունկցիան սխալ է աշխատում հայերեն այբուբենի հետ: Օգտագործեք mb_strtolower ֆունկցիան:

Տողը վերին ռեգիստրին փոխելու համար կիրառվում են strtoupper/mb_strtoupper ֆունկցիաները, որոնք աշխատում են նույն կերպ:

Օրինակ 1.

```
<?php
echo strtolower('ABCDE'); // 'abcde'
?>
```

```
<?php
echo mb_strtolower('ԱՔԳԴԵ'); // 'աքգդե'
?>
```

Ucwords ֆունկցիան տողի յուրաքանչյուր բառի առաջին սիմվոլը դարձնում է մեծատառ:

Օրինակ 2.

```
<?php
$foo = 'hello world!';

$foo = ucwords($foo);           // Hello World!

$bar = 'ARAM KARAPETYAN';

$bar = ucwords($bar);           // ARAM KARAPETYAN
```

```
$bar = ucwords(strtolower($bar)); // Aram Karapetyan
```

```
?>
```

Substr() և mb_substr() ֆունկցիաները

Ֆունկցիայի տեսքը հետևյալն է.

substr(\$str, \$start [, \$length])

Օգտագործելով substr() ֆունկցիան, կարելի է տողից ստանալ դրա որոշակի մասը: Այս ֆունկցիան կտրում է \$str տողը, սկսած \$start դիրքում գտնվող նիշից մինչև տողի ավարտը: Լրացուցիչ \$length կամընտրովի պարամետրով կարելի է սահմանել կտրվող նիշերի քանակը:

Հայերեն այբուբենի համար պետք է կիրառել mb_strtolower ֆունկցիան:

Օրինակ 3.

```
<?php
```

```
    $input = 'Հայաստան-Armenia Արարատ-Ararat';
```

```
    $subinput1 = mb_substr($input, 16);
```

```
    $subinput2 = mb_substr($input, 9, 7);
```

```
    echo $subinput1;
```

```
    echo "<br />";
```

```
    echo $subinput2;
```

```
?>
```

Արդյունքում կստանանք.

Արարատ-Ararat

Armenia

Երկրորդ պարամետրը կարող է բացասական լինել, այդ դեպքում հաշվարկը կսկսվի տողի վերջից, իսկ վերջին նիշի համարը կլինի -1:

Օրինակ 4.

```
<?php
```

```
    $input = 'Հայաստան-Armenia Արարատ-Ararat';
```

```

$subinput1 = mb_substr($input, -13);
$subinput2 = mb_substr($input, -13, 6);
echo $subinput1;
echo "<br />";
echo $subinput2;

?>

```

Արդյունքում կստանանք.

Արարատ-Ararat

Արարատ

Str_replace() և str_ireplace ֆունկցիաները

Ֆունկցիայի տեսքը հետևյալն է.

str_replace(\$search, \$replace, \$subject [, \$count])

Այս ֆունկցիան \$subject տողում բոլոր \$search ենթատողերը փոխարինում է \$replace ենթատողով, հաշվի առնելով ռեգիստրը:

Եթե \$count կամընտրովի պարամետրը տրված է, ապա այն ցույց կտա կատարված փոխարինումների քանակը:

Եթե search-ը և replace-ը զանգվածներ են, ապա str_replace() ֆունկցիան օգտագործում է համապատասխան զանգվածի յուրաքանչյուր արժեք՝ subject-ում որոնման և փոխարինման համար: Եթե replace զանգվածում ավելի քիչ տարրեր կան, քան search-ում, որպես փոխարինող տող մնացած արժեքների համար կօգտագործվի դատարկ տողը: Եթե search-ը զանգված է, իսկ replace-ը տող, ապա այդ տողը կօգտագործվի search զանգվածի յուրաքանչյուր տարրի փոխարինման համար: Հակառակ դեպքը իմաստ չունի:

Օրինակ 5.

```

<?php
echo str_replace('a', '!', 'abcabc');
// !bc!bc

```

```
echo str_replace(['a', 'b', 'c'], [1, 2, 3], 'abcabc');  
// 123123
```

```
$vowels = array("a", "e", "i", "o", "u", "A", "E", "I", "O", "U");  
echo $onlyconsonants = str_replace($vowels, "", "Hello World of PHP");  
// Hll Wrld f PHP
```

```
$phrase = "You should eat fruits, vegetables, and fiber every day.";  
$healthy = array("fruits", "vegetables", "fiber");  
$yummy = array("pizza", "beer", "ice cream");
```

```
echo $newphrase = str_replace($healthy, $yummy, $phrase);  
// You should eat pizza, beer, and ice cream every day.
```

```
echo $str = str_replace("ll", "", "good golly miss molly!", $count);  
echo "<br />";  
echo $count;  
// good goy miss moy!  
// 2
```

```
$bodytag = str_replace("%body%", "black", "<body text='%body%'>");  
// կվերագրի <body text='black'>
```

```
$str = "Snղ 1\\nSnղ 2\\rSnղ 3\\r\\nSnղ 4\\n";  
echo $str;  
echo "<br />". "<br />";  
$order = array("\\r\\n", "\\n", "\\r");  
$replace = '<br />';  
echo $newstr = str_replace($order, $replace, $str);
```

```
// Տող 1 Տող 2 Տող 3 Տող 4
```

```
// Տող 1
```

```
// Տող 2
```

```
// Տող 3
```

```
// Տող 4
```

```
?>
```

Կա նաև մի ֆունկցիա՝ `str_replace`, որը իրականացնում է նույն աշխատանքը, բայց առանց հաշվի առնելու ռեգիստրը:

Chr() և ord() ֆունկցիաները

`Chr` ֆունկցիան որպես փաստարկ է վերցնում ASCII կոդի \$ascii նիշը և վերադարձնում է այս կոդին համապատասխանող իրական նիշը:

```
<? php
```

```
echo chr(36); // $
```

```
echo chr(64); // @
```

```
?>
```

`Ord()` ֆունկցիան կատարում է `chr()` ֆունկցիայի հակառակը՝ վերադարձնում է նրան փոխանցված նիշի ASCII կոդը:

```
< ?php
```

```
echo ord ( '$' ); // 36
```

```
echo ord ( '@' ); // 64
```

```
?>
```