

# Раздаточный материал по предмету “Язык программирования PHP”. Лекция 4.

## Функции

### *Передача аргументов по ссылке*

```
function plus_some(&$inSr)
{
    $inSr.= ' and some added value.';
}
$str = 'First initiated string, ';
plus_some($str);
echo $str;    // Выведет 'First initiated string, and some added value.'
```

### *Использование значений по умолчанию в определении функции*

```
function maketea($type = "Green"){
    return "Making a cup of $type tea.";
}
echo maketea ();
echo maketea ("Earl grey");
```

Результат работы приведенного выше кода будет таким:

```
Making a cup of Green tea.
Making a cup of Earl grey tea.
```

### *Глобальные переменные*

```
$glVar=1;

function globExample(){
    global $glVar;
    $glVar = $glVar *2;
    echo 'в результате работы функции $glVar =', $glVar;
}
echo 'вне функции $glVar =', $glVar,', ' ';
globExample();
echo '<br />вне функции $glVar =', $glVar,', ' ';
globExample();
```

В результате работы этого скрипта получим:

```
вне функции $glVar =1, в результате работы
  функции $glVar =2
вне функции $glVar =2, в результате работы
  функции $glVar =4
```

## Статические переменные

```
<?
function statExample(){
static $a = 1;
// нельзя присваивать выражение или ссылку
    $a = $a*2;
    echo $a;
}
statExample(); // выведет 2
echo $a; // ничего не выведет, так как
        // $a доступна только
        // внутри функции
statExample(); // внутри функции $a=2, поэтому
        // результатом работы функции
        // будет число 4
?>
```

## Массивы

### Индексированные и ассоциированные массивы

```
<?php
$arr = array("type" => "apple", 7 => true);

echo $arr["type"]; // apple
echo $arr[7];      // 1
?>
```

### Массив может содержать данные любого нужного типа

```
<?php
$arr = array("somearray" => array(6 => 5, 13 => 9, "a" => 42));

echo $arr["somearray"][6]; // 5
echo $arr["somearray"][13]; // 9
echo $arr["somearray"]["a"]; // 42
```

### Пример работы с массивом

```
<?php
$arr = array(5 => 1, 12 => 2);

$arr[] = 56; // В этом месте скрипта это
            // эквивалентно $arr[13] = 56;

$arr["x"] = 42; // Это добавляет к массиву новый
               // элемент с ключом "x"

unset($arr[5]); // Это удаляет элемент из массива

unset($arr); // Это удаляет массив полностью
?>
```

## Операции с массивами

```
<?
$a = array("и"=>"Информатика",
           "м"=>"Математика");
$b = array("и"=>"История", "м"=>"Биология",
           "ф"=>"Физика");
$c = $a + $b;
$d = $b + $a;
print_r($c);
/* получим: Array([и]=>Информатика [м]=>Математика [ф]=>Физика) */
print_r($d);
/* получим: Array([и]=>История [м]=>Биология [ф]=>Физика) */
?>
```

## Сравнение массивов

```
<?
$a = array("и"=>"Информатика",
           "м"=>"Математика");
$b = array("м"=>"Математика",
           "и"=>"Информатика");
if ($a == $b) echo "Массивы равны и";
else echo "Массивы НЕ равны и ";
if ($a === $b) echo " эквивалентны";
else echo " НЕ эквивалентны";
// получим echo "Массивы равны и
                НЕ эквивалентны"
?>
```

## Сортировка массивов с использованием встроенной функции sort()

```
<?php
$fruits = array("lemon", "orange", "banana", "apple");
sort($fruits);
reset($fruits);
while (list($key, $val) = each($fruits)) {
    echo "fruits[" . $key . "] = " . $val . "\n";
}
?>
```

## Дополнительные примеры работы с массивами

Тип массив в PHP является очень гибким, поэтому мы приведем здесь несколько примеров, чтобы продемонстрировать вам многие возможности массивов.

```
<?php
// это
$a = array( 'color' => 'red',
           'taste' => 'sweet',
           'shape' => 'round',
           'name' => 'apple',
           4 // ключом будет 0
           );

// полностью соответствует
$a['color'] = 'red';
$a['taste'] = 'sweet';
$a['shape'] = 'round';
$a['name'] = 'apple';
$a[] = 4; // ключом будет 0
```

```
$b[] = 'a';
$b[] = 'b';
$b[] = 'c';
// создаст массив array(0 => 'a' , 1 => 'b' , 2 => 'c'),
// или просто array('a', 'b', 'c')
?>
```

## Использование array()

```
<?php
// Массив как карта (свойств)
$map = array( 'version'    => 4,
              'OS'        => 'Linux',
              'lang'      => 'english',
              'short_tags' => true
            );

// исключительно числовые ключи
$array = array( 7,
               8,
               0,
               156,
               -10
            );
// это то же самое, что и array(0 => 7, 1 => 8, ...)

$switching = array(      10, // ключ = 0
                    5    => 6,
                    3    => 7,
                    'a'  => 4,
                    11,  // ключ = 6 (максимальным числовым индексом был 5)
                    '8'  => 2, // ключ = 8 (число!)
                    '02' => 77, // ключ = '02'
                    0    => 12 // значение 10 будет перезаписано на 12
                );

// пустой массив
$empty = array();
?>
```