

Встроенные функции для работы с массивами

Преподаватель – Борисова Татьяна Андреевна

Сегодня на занятии:

1. Функция `print_r`
2. Функция `count`
3. Функция `array_push`
4. Функция `array_pop`
5. Функция `array_unshift`
6. Функция `array_shift`
7. Функция `explode`
8. Функция `implode`

Функция `print_r`

Встроенная функция **`print_r()`** выводит массивы в удобочитаемом формате. Чтобы использовать функцию, передайте в нее массив в качестве аргумента:

```
$my_array = array(25, "or", 6, "to", 4);
```

```
print_r($my_array);
```

```
/* Array
```

```
( [0] => 25
```

```
[1] => or
```

```
[2] => 6
```

```
[3] => to
```

```
[4] => 4 ) */
```


Функция count

Встроенная функция **count()** принимает в качестве аргумента массив и возвращает количество элементов в этом массиве:

```
$devices = array("watch", "phone", "tablet", "laptop");
```

```
$number_of_devices = count($devices);
```

```
echo $number_of_devices;
```

```
// выведет: 4
```

Функция `array_push`

Встроенная функция **`array_push()`** принимает массив в качестве первого аргумента. Последующие аргументы - это элементы, которые должны быть добавлены в массив.

Функция добавляет каждый из элементов в конец массива и возвращает новое количество элементов в массиве:

```
$some_numbers = ["1", 2, 3.0];
```

```
echo array_push($some_numbers, "four");
```

```
// выведет: 4 // $some_numbers теперь состоит из таких элементов: ["1", 2, 3.0, "four"]
```

Функция `array_pop`

Встроенная функция **`array_pop()`** принимает в качестве аргумента массив. Она удаляет последний элемент массива и возвращает удаленный элемент:

```
$some_numbers = ["1", 2, "three"];
```

```
echo array_pop($some_numbers); // выведет: "three"
```

```
// массив $some_numbers теперь выглядит так: ["1", 2]
```

Функция `array_unshift`

Встроенная функция **`array_unshift()`** принимает массив в качестве первого аргумента. Последующие аргументы - это элементы, которые должны быть добавлены в массив.

Функция добавляет каждый из элементов в начало массива и возвращает новое количество элементов в массиве:

```
$joke = [7, "ate", "nine!"];
```

```
$scared = array_unshift($joke, "Why is 6", "afraid of 7?", "Because");
```

```
echo $scared; // выведет: 6
```

```
// массив $joke теперь: ["Why is 6", "afraid of 7?", "Because", 7, "ate", "nine!"]
```


Функция `array_shift`

Встроенная функция **`array_shift()`** принимает массив в качестве аргумента, удаляет первый элемент из массива и возвращает удаленный элемент. Все элементы в массиве будут сдвинуты на индекс на один меньше, чем их предыдущий индекс:

```
$some_numbers = ["1", 2, "three"];
```

```
echo array_shift($some_numbers); // выведет: 1
```

```
// теперь массив $some_numbers выглядит так: [2, "three"]
```


Функция explode

Встроенная функция **explode()** разбивает строку с помощью разделителя. Функция принимает два обязательных аргумента: разделитель и строку, которую необходимо разделить на части:

```
$str = "1 2 3 4 5";
```

```
$arr = explode(" ", $str);
```

```
print_r($arr);
```

Функция explode

Функция возвращает массив, поэтому в результате выполнения кода выше мы увидим на экране:

Array (

[0] => 1

[1] => 2

[2] => 3

[3] => 4

[4] => 5

)

Функция implode

Встроенная функция **implode()** создает строку из массива. Функция принимает два аргумента: строку для размещения между каждым элементом массива и массив, который нужно объединить:

```
$pieces = [1, "2", "three", 4.0];
```

```
$glue = " and a ";
```

```
echo implode($glue, $pieces);
```

```
// Выведет: 1 and a 2 and a three and a 4
```

*СПАСИБО ЗА
ВНИМАНИЕ!*

