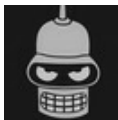


Методы массивов (простые)

<code>arr.indexOf(arg);</code>	Находит индекс первого найденного элемента в массиве
<code>arr.lastIndexOf(arg);</code>	Находит индекс последнего найденного элемента в массиве
Примечание <code>.index</code> и <code>lastIndex</code>	Если не нашло элемент - возвращает -1
<code>arr[arr.length - 1];</code>	Найти последний эл-т массива
<code>arr[arr.length] = el;</code>	Добавить эл-т в конец массива
<code>delete arr[3];</code>	Удаляет элемент, но оставляет дырку (undefined)
<code>x = arr.splice(3, 1);</code>	Вырезаем часть массива где первый аргумент - это индекс начала, а второй - показывает кол-во элементов (вправо), которые мы хотим вырезать (исходный массив будет изменен)
<code>x = arr.splice(-1, 1);</code>	Вырезаем часть массива где первый аргумент - это индекс начала (с конца), а второй - показывает кол-во элементов (вправо), которые мы хотим вырезать (исходный массив будет изменен)
<code>x = arr.splice(a, b, args);</code>	Если передать еще аргументы (args), то они будут вставлены на место вырезанных.



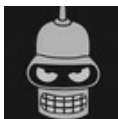
By **PavelChe**
cheatography.com/pavelche/

Published 14th May, 2017.
 Last updated 14th May, 2017.
 Page 1 of 4.

Sponsored by **Readable.com**
 Measure your website readability!
<https://readable.com>

Методы массивов (простые) (cont)

<code>x = arr.slice(3, 4);</code>	Копирует часть массива, не изменяя исходный. Первый аргумент начало фрагмента (включительно), второй аргумент - конец фрагмента (не включительно)
<code>x = arr.slice(-3, -1);</code>	Копирует часть массива, не изменяя исходный. Первый аргумент начало фрагмента (включительно с конца), второй аргумент - конец фрагмента (не включительно с конца)
<code>x = arr.slice(3);</code>	Копирует все после эл-та под индексом 3 (включительно)
<code>arr.push(-args);</code>	Добавить эл-ты в конец массива
<code>arr.pop();</code>	Вырезает последний эл-т массива
<code>arr.unshift('h','y');</code>	Добавить эл-ты в нач массива
<code>arr.shift();</code>	Вырезает первый эл-т массива
<code>arr.reverse()</code>	Перевернуть массив
<code>arr.length</code>	- длина массива (св-во)



By **PavelChe**

cheatography.com/pavelche/

Published 14th May, 2017.

Last updated 14th May, 2017.

Page 2 of 4.

Sponsored by **Readable.com**

Measure your website readability!

<https://readable.com>

Сортировка .sort(callback);

```
var array = [1, 3, 2, 15, 8, 7];
array.sort(function(a, b) {
    if (a > b) {
        return 1;
    } else if (a < b) {
        return -1;
    }
});
```

Результат: [1, 2, 3, 7, 8, 15];

a и b - сравниваемые числа
(Изменяет исходный массив)

Перебор .forEach(callback)

```
var arr = ['Яблоко', 'Апельсин', 'Груша'];
arr.forEach(function(item, i, arr) {
    console.log(i + ' : ' + item + ' : ' + arr);
});
```

Результат: Выведет поочередно эл-ты в

кон соль;

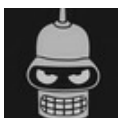
Производится перебор массива
item - элемент, i - индекс (счетчик), arr - перебираемый массив
(Не изменяет исходный массив)

Фильтр .filter(callback)

```
var arr = [1, -1, 2, -2, 3];
var positiveArr = arr.filter(function(num) {
    return num > 0; // вернет только положительные числа
});
```

Результат: positiveArr = [1, 2, 3];

Если return возвращает true - то item включен в новый массив
item - элемент, i - индекс (счетчик), arr - перебираемый массив
(не изменяет исходный массив)



By **PavelChe**
cheatography.com/pavelche/

Published 14th May, 2017.
Last updated 14th May, 2017.
Page 3 of 4.

Sponsored by **Readable.com**
Measure your website readability!
<https://readable.com>

Трансформация .map(callback);

```
var arr = [5, 1, 12, 75, 22];
var arr2 = arr.map(function(item) {
  return item * item;
});
```

//

Результат: [25, 1, 144, 5625, 484]

(Во звели в квадрат)

Трансформирует массив в новый исходя из переданного в return
item - элемент, i - индекс (счетчик), arr - перебираемый массив
(не изменяет исходный массив)

.compare(callback) и .some(callback)

```
var arr = [1, -1, 3, -2, 2];
var result = arr.every(function(item, i, arr) {
  return item > 0;
});
var result2 = arr.some(function(item, i, arr) {
  return item > 0;
});
```

.every() - в result вернет false. Если return вернет true для всех эл-тов массива, то в переменную вернется true
.some() - в result2 вернет true. Если return вернет true хотя-бы для одного эл-та массива, то в переменную вернется true
item - элемент, i - индекс (счетчик), arr - перебираемый массив

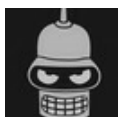
.reduce(callback, num)

```
var arr = [1, 2, 3, 4, 5];
var result = arr.reduce(function(previousValue, currentItem, i, arr) {
  return previousValue + currentItem;
}, 0);
```

// Второй аргумент (число) - значение previousValue для
первой итерации (по умолчанию = 0);

// Результат: result = 15;

Применяет функцию callback по очереди к каждому элементу массива слева-направо, сохраняя при этом промежуточный результат.
previousValue - промежуточный результат, currentItem - текущий эл-т, i - индекс (счетчик), arr - перебираемый массив.
.reduceRight(callback, num) - перебирает справа-налево



By PavelChe

cheatography.com/pavelche/

Published 14th May, 2017.

Last updated 14th May, 2017.

Page 4 of 4.

Sponsored by **Readable.com**

Measure your website readability!

<https://readable.com>