

Array methods

метод	делает	возвращает	аргументы	мутирует/создаёт новый	пример	
unshift(a)	добавляет элемент в начало	новую длину массива	a - элемент, который добавляем	мутирует	<pre>let arr = [1, 2, 3, 4, 5]; let result = arr.unshift(0); // original array [ 0, 1, 2, 3, 4, 5 ] return 6</pre>	
shift()	извлекает элемент из начала	извлечённый элемент	-	мутирует	<pre>let arr = [1, 2, 3, 4, 5]; let result = arr.shift(); // original array [ 2, 3, 4, 5 ] return 1</pre>	
push(a)	добавляет элемент в конец	новую длину массива	a - элемент, который добавляем	мутирует	<pre>let arr = [1, 2, 3, 4, 5]; let result = arr.push(100); //original array [ 1, 2, 3, 4, 5, 100 ] return 6</pre>	
pop()	извлекает элемент из конца	извлечённый элемент	-	мутирует	<pre>let arr = [1, 2, 3, 4, 5]; let result = arr.pop(); //original array [ 1, 2, 3, 4 ] return 5</pre>	
split(a)	преобразует строку в массив	массив	a = “,” - разделитель в строке	создаёт новый	<pre>let str = "kittens,puppies,chickens"; let result = str.split(","); // return [ 'kittens', 'puppies', 'chickens' ]</pre>	
join(a)	преобразует массив в строку	строку	a = “,” - разделитель в будущей строке	создаёт строоку	<pre>let arr = ["kittens", "puppies", "chickens"]; let result = arr.join("&amp;"); // return kittens&amp;puppies&amp;chickens</pre>	
slice(a,b)	возвращает часть массива	часть исходного массива	a - индекс начала (включительно) b - индекс конца (не включительно) (если отрицательные, то считается с конца)	создаёт новый	<pre>let arr = [1, 2, 3, 4, 5]; let result = arr.slice(2, 4); //original arr [ 1, 2, 3, 4, 5 ] return [ 3, 4 ]</pre>	
splice(a,b, ...c)	вырезает часть массива и вставляет новую	вырезанные элементы массива	a - индекс начала (включительно). если отрицательные, то считается с конца b - количество вырезаемых элементов (может быть 0) ...c - вставляемые элементы (вместо вырезанных)	мутирует	<pre>let arr = [1, 2, 3, 4, 5]; let result = arr.splice(-4, 2, "blabla", "hello"); //original arr [ 1, 'blabla', 'hello', 4, 5 ] return [ 2, 3 ]</pre>	
concat(...a)	объединяет массивы	новый объединённый массив	... a - массивы, которые присоединяются к исходному	создаёт новый	<pre>const male = ["Vasya", "Igor"]; const female = ["Anna", "Marina"]; let res = male.concat(female, ["Max", "Nataly"]); //[ 'Vasya', 'Igor', 'Anna', 'Marina', 'Max', 'Nataly' ]</pre>	
reverse()	переворачивает массив (элементы в обратном порядке)	перевернутый массив	-	мутирует	<pre>const female = ["Anna", "Marina", 1, 2, 3]; let femRev = female.reverse(); //result [ 3, 2, 1, 'Marina', 'Anna' ] original array [ 3, 2, 1, 'Marina', 'Anna' ]</pre>	
some(cb)	проверяет, чтобы хоть один элемент массива удовлетворял условию в cb	true/false	cb - функция, которая проверяет элементы (удовлетворяет ли элемент заданным условиям)	-	<pre>let badParty = company.some((el) =&gt; el.age &gt; 18); //true</pre>	<pre>const company = [ {   name: "Oleg",   age: 15, }, {   name: "Shamil",   age: 28, }, {   name: "Alya",   age: 22, }, ];</pre>
every(cb)	проверяет, чтобы все элементы массива удовлетворял условию в cb	true/false	cb - функция, которая проверяет элементы (удовлетворяет ли элемент заданным условиям)	-	<pre>let party = company.every((el) =&gt; el.age &gt; 18); //false</pre>	
filter(cb)	Создаёт новый массив со всеми элементами, удовлетворяющими заданному условию	отфильтрованный массив	cb - функция, которая проверяет элементы (удовлетворяет ли элемент заданным условиям)	создаёт новый	<pre>let adults = company.filter((el) =&gt; el.name &gt; 18); //[ { name: 'Shamil', age: 28 }, { name: 'Alya', age: 22 } ]</pre>	
map(cb)	Создаёт новый массив с результатом вызова функции	новый, изменённый массив	cb - функция, которая применяется к каждому элементу массива	создаёт новый	<pre>const arr = [1, 2, 3, 4]; const newArr = arr.map((el) =&gt; el * 2); //result [ 2, 4, 6, 8 ]</pre>	
sort(cb)	Сортирует массив	отсортированный массив	<b>(a, b) =&gt; a - b</b>	мутирует	<pre>const arr = [15, 22, 110, 0, 1, 2, 3, 4]; const newArr = arr.sort((a, b) =&gt; a - b); //result [   0,  1,  2,   3,   4, 15, 22, 110 ]</pre>	
reduce(cb)	применяет функцию к каждому значению массива (слева-направо), сводя его к одному значению (аккумулятору).	значение	cb - функция, которая применяется к каждому элементу массива	-	<pre>const arr = [1, 2, 3]; let res = arr.reduce((acc, el) =&gt; (acc = acc + el), 0); //6</pre>	
forEach(cb)	применяет функцию к каждому значению массива	-	cb - функция, которая применяется к каждому элементу массива	-	<pre>arr.forEach((el, i) =&gt; {   console.log(el);   if (el === 2) {     return console.log("hello", i);   } }); //1 2 hello 1 3 4</pre>	