

Array Methods

30+ Methods



Part 01

12 Methods

JS



@CodeBustler
@coder_aishya



JS Array Methods

1. toString()
2. join()
3. push()
4. pop()
5. shift()
6. unshift()
7. concat()
8. sort()
9. splice()
10. slice()
11. reverse()
12. isArray()
13. indexOf()
14. lastIndexOf()
15. find()
16. findIndex()
17. includes()
18. entries()
19. every()
20. some()
21. fill()
22. copyWithin()
23. valueOf()
24. forEach()
25. map()
26. filter()
27. reduce()
28. reduceRight()
29. flat()
30. flatMap()
31. from()
32. keys()
33. delete Operator
34. length property

1. toString()

Returns **string with array values** separated by commas(,).

Does overwrites original ? No ❌

```
let array = [1, 2, 3];  
log(array.toString()); // "1, 2, 3"
```

2. join(separator)

Return array as string, **Any separator can** be specified. The default is comma (,)

Does overwrites original ? No ❌

```
let array = ["Hello", "World!"];  
log(array.join("-")); // Hello-World!  
log(array.join("**")); // Hello**World!
```


3. pop()

Removes the last element of an array,
Returns Removed element.

Does overwrites original ? Yes 

```
let array = ["a", "b", "c"];  
log(array.pop()); // c (Removed)  
log(array); // ["a", "b"]
```

4. push(item1, item2 ... n)

Adds new element to the end of an array,
Returns new Length of array.

Does overwrites original ? Yes 

```
let array = ["a", "b", "c"];  
log(array.push("z")); // 4 (Length)  
log(array); // ["a", "b", "c", "z"]
```

5. shift()

Removes the first element of an array,
Returns Removed element.

Does overwrites original ? Yes 

```
let array = ["a", "b", "c"];  
log(array.shift()); // a (Removed)  
log(array); // ["b", "c"]
```

6. unshift(item1, item2 ... n)

Adds new element to the start of an array,
Returns new Length of array.

Does overwrites original ? Yes 

```
let array = ["a", "b", "c"];  
log(array.unshift("X")); // 4 (Length)  
log(array); // ["X", "a", "b", "c"]
```

7. concat()

Concatenates (joins) **two or more arrays**.

Returns a new array, joined arrays

Does overwrites original ? No ❌

```
let ary1 = [1, 2], ary2 = [3, 4], ary3 = ["a"];  
let concatArray = ary1.concat(ary2, ary3);  
log(concatArray); // [1,2,3,4,"a"]
```

8. sort()

Sorts the elements of an array, **strings in alphabetical** and ascending order.

Does overwrites original ? Yes ✅

```
let nums = [5, 3, 6, 2], str = ["b", "a", "z"];  
log(nums.sort()); // [2, 3, 5, 6]  
log(str.sort()); // ["a", "b", "z"]
```


9. splice(index*, count, e1, e2..)

Adds and/or removes array elements.

Returns removed items in array (if any).

Does overwrites original ? Yes 

```
let array = ["a", "b", "c", "d"];  
log(array.splice(1, 2, "X")); // ["b", "c"]  
log(array); // ["a", "X", "d"]
```

10. slice(start, end) @CodeBustler


Returns **selected elements in an array**, as a new array, default start = 0, end = last E

Does overwrites original ? No 

```
let numbers = [11, 22, 33, 44, 55];  
let sliceArray = numbers.slice(2, 4);  
log(sliceArray); // [33, 44] | New Array
```

11. reverse()

Reverses the order of the elements in an array,

Does overwrites original ? Yes 

```
let array = [1, 2, 3, 4, 5];  
log(array.reverse()); // [5, 4, 3, 2, 1]
```

12. isArray(array)

Returns true if an object is an array, otherwise false.

Does overwrites original ? No 

```
let array5 = [1, 2, 3, 4, 5];  
let str = "this is string";  
log(Array.isArray(array5)); // True  
log(Array.isArray(str)); // False
```