

# Array.prototype.concat()

Share:  Twitter  Facebook  Google+

The **concat()** method returns a new array comprised of the array on which it is called joined with the array(s) and/or value(s) provided as arguments.

## Syntax

```
var new_array = old_array.concat(value1[, value2[, ...[, valueN]]])
```

## Parameters

**valueN**


Arrays and/or values to concatenate into a new array. See the description below for details.

## Description

`concat` creates a new array consisting of the elements in the object on which it is called, followed in order by, for each argument, the elements of that argument (if the argument is an array) or the argument itself (if the argument is not an array).

`concat` does not alter `this` or any of the arrays provided as arguments but instead returns a shallow copy that contains copies of the same elements combined from the original arrays. Elements of the original arrays are copied into the new array as follows:

- Object references (and not the actual object): `concat` copies object references into the new array. Both the original and new array refer to the same object. That is, if a referenced object is modified, the changes are visible to both the new and original arrays.
- Strings and numbers (not `String` and `Number` objects): `concat` copies the values of strings and numbers into the new array.

 **Note:** Concatenating array(s)/value(s) will leave the originals untouched. Furthermore, any operation on the new array will have no effect on the original arrays, and vice versa.

## Examples

### Concatenating two arrays

The following code concatenates two arrays:

```
1 | var alpha = ['a', 'b', 'c'],
2 |     numeric = [1, 2, 3];
3 |
4 | var alphaNumeric = alpha.concat(numeric);
5 |
6 | console.log(alphaNumeric); // Result: ['a', 'b', 'c', 1, 2, 3]
```

### Concatenating three arrays

The following code concatenates three arrays:

```
1 | var num1 = [1, 2, 3],
2 |     num2 = [4, 5, 6],
3 |     num3 = [7, 8, 9];
4 |
5 | var nums = num1.concat(num2, num3);
6 |
7 | console.log(nums); // Result: [1, 2, 3, 4, 5, 6, 7, 8, 9]
```

### Concatenating values to an array

The following code concatenates three values to an array:

```
1 | var alpha = ['a', 'b', 'c'];
2 |
3 | var alphaNumeric = alpha.concat(1, [2, 3]);
4 |
5 | console.log(alphaNumeric);
6 | // Result: ['a', 'b', 'c', 1, 2, 3]
```

## Specifications

Specification	Status	Comment
<a href="#">ECMAScript 3rd Edition (ECMA-262)</a>	<div><div></div><div>st</div>Standard</div>	Initial definition. Implemented in JavaScript 1.2.
<a href="#">ECMAScript 5.1 (ECMA-262)</a> The definition of 'Array.prototype.concat' in that specification.	<div><div></div><div>st</div>Standard</div>	
<a href="#">ECMAScript 2015 (6th Edition, ECMA-262)</a> The definition of 'Array.prototype.concat' in that specification.	<div><div></div><div>st</div>Standard</div>	

## Browser compatibility

	Desktop	Mobile				
Feature	Chrome	Firefox (Gecko)	Internet Explorer	Opera	Safari	
Basic support	1.0	1.0 (1.7 or earlier)	5.5	(Yes)	(Yes)	

## See also

- [push](#) / [pop](#) — add/remove elements from the end of the array
- [unshift](#) / [shift](#) — add/remove elements from the beginning of the array
- [splice](#) — add/remove elements from the specified location of the array
- [String.prototype.concat\(\)](#)