

Array.prototype.splice()

The **splice()** method changes the contents of an array by removing existing elements and/or adding new elements.

JavaScript Demo: Array.splice()

1

var months = ['Jan', 'March', 'April', 'June'];

2

months.splice(1, 0, 'Feb');

3

// inserts at 1st index position

4

console.log(months);

5

// expected output: Array ['Jan', 'Feb', 'March', 'April', 'June']

6

7

months.splice(4, 1, 'May');

8

// replaces 1 element at 4th index

9

console.log(months);

10

// expected output: Array ['Jan', 'Feb', 'March', 'April', 'May']

11

Run ›

Reset

Syntax

```
array.splice(start[, deleteCount[, item1[, item2[, ...]]]])
```

Parameters

start

Index at which to start changing the array (with origin 0). If greater than the length of the array, actual starting index will be set to the length of the array. If negative, will begin that many elements from the end of the array (with origin -1) and will be set to 0 if absolute value is greater than the length of the array.

deleteCount | Optional

An integer indicating the number of old array elements to remove.

If `deleteCount` is omitted, or if its value is larger than `array.length - start` (that is, if it is greater than the number of elements left in the array, starting at `start`), then all of the elements from `start` through the end of the array will be deleted.

If `deleteCount` is 0 or negative, no elements are removed. In this case, you should specify at least one new element (see below).

item1, item2, ... | Optional

The elements to add to the array, beginning at the `start` index. If you don't specify any elements, `splice()` will only remove elements from the array.

Return value

An array containing the deleted elements. If only one element is removed, an array of one element is returned. If no elements are removed, an empty array is returned.

Description

If you specify a different number of elements to insert than the number you're removing, the array will have a different length at the end of the call.

Examples

Remove 0 elements from index 2, and insert "drum"

```
1 | var myFish = ['angel', 'clown', 'mandarin', 'sturgeon'];
2 | var removed = myFish.splice(2, 0, 'drum');
3 |
4 | // myFish is ["angel", "clown", "drum", "mandarin", "sturgeon"]
5 | // removed is [], no elements removed
```

Remove 1 element from index 3

```
1 | var myFish = ['angel', 'clown', 'drum', 'mandarin', 'sturgeon'];
2 | var removed = myFish.splice(3, 1);
3 |
```

```
4 | // removed is ["mandarin"]
5 | // myFish is ["angel", "clown", "drum", "sturgeon"]
```

Remove 1 element from index 2, and insert "trumpet"

```
1 | var myFish = ['angel', 'clown', 'drum', 'sturgeon'];
2 | var removed = myFish.splice(2, 1, 'trumpet');
3 |
4 | // myFish is ["angel", "clown", "trumpet", "sturgeon"]
5 | // removed is ["drum"]
```

Remove 2 elements from index 0, and insert "parrot", "anemone" and "blue"

```
1 | var myFish = ['angel', 'clown', 'trumpet', 'sturgeon'];
2 | var removed = myFish.splice(0, 2, 'parrot', 'anemone', 'blue');
3 |
4 | // myFish is ["parrot", "anemone", "blue", "trumpet", "sturgeon"]
5 | // removed is ["angel", "clown"]
```

Remove 2 elements from index 2

```
1 | var myFish = ['parrot', 'anemone', 'blue', 'trumpet', 'sturgeon'];
2 | var removed = myFish.splice(myFish.length - 3, 2);
3 |
4 | // myFish is ["parrot", "anemone", "sturgeon"]
5 | // removed is ["blue", "trumpet"]
```

Remove 1 element from index -2

```
1 | var myFish = ['angel', 'clown', 'mandarin', 'sturgeon'];
2 | var removed = myFish.splice(-2, 1);
3 |
4 | // myFish is ["angel", "clown", "sturgeon"]
5 | // removed is ["mandarin"]
```

Remove all elements after index 2 (incl.)

```
1 | var myFish = ['angel', 'clown', 'mandarin', 'sturgeon'];
2 | var removed = myFish.splice(2);
3 |
4 | // myFish is ["angel", "clown"]
5 | // removed is ["mandarin", "sturgeon"]
```

Specifications

Specification	Status	Comment
ECMAScript 3rd Edition (ECMA-262)	<div><div></div><div>ST</div>Standard</div>	Initial definition. Implemented in JavaScript 1.2.
ECMAScript 5.1 (ECMA-262) The definition of 'Array.prototype.splice' in that specification.	<div><div></div><div>ST</div>Standard</div>	
ECMAScript 2015 (6th Edition, ECMA-262) The definition of 'Array.prototype.splice' in that specification.	<div><div></div><div>ST</div>Standard</div>	
ECMAScript Latest Draft (ECMA-262) The definition of 'Array.prototype.splice' in that specification.	<div><div></div><div>D</div>Draft</div>	

Browser compatibility

New compatibility tables are in beta ▼

Basic support		
<div><div></div><div></div></div>	<div><div></div><div></div></div>	1
<div><div></div><div></div></div>	<div><div></div><div></div></div>	Yes
<div><div></div><div></div></div>	<div><div></div><div></div></div>	1
<div><div></div><div></div></div>	<div><div></div><div></div></div>	5.5
<div><div></div><div></div></div>	<div><div></div><div></div></div>	Yes
<div><div></div><div></div></div>	<div><div></div><div></div></div>	Yes
<div><div></div><div></div></div>	<div><div></div><div></div></div>	Yes
<div><div></div><div></div></div>	<div><div></div><div></div></div>	Yes
<div><div></div><div></div></div>	<div><div></div><div></div></div>	Yes
<div><div></div><div></div></div>	<div><div></div><div></div></div>	Yes
<div><div></div><div></div></div>	<div><div></div><div></div></div>	4
<div><div></div><div></div></div>	<div><div></div><div></div></div>	Yes
<div><div></div><div></div></div>	<div><div></div><div></div></div>	Yes
<div><div></div><div></div></div>	<div><div></div><div></div></div>	?
<div><div></div><div></div></div>	<div><div></div><div></div></div>	Yes

- Full support
- Compatibility unknown

See also

- `push()` / `pop()` — add/remove elements from the end of the array
 - `unshift()` / `shift()` — add/remove elements from the beginning of the array
 - `concat()` — returns a new array comprised of this array joined with other array(s) and/or value(s)
-