

Ace your JavaScript Interview. [Get my ebook](#). 100 solved Javascript, 20 solved React, & 2 frontend system design questions (**1160+ copies sold**). Get a [Free preview](#). ×

Advertisements

■ ■ ■

Practically  
prepare for  
your  
JavaScript  
interview

[JavaScript  
Revision](#)

[JavaScript-  
Concept Based  
Problems](#)

[Data Structures](#)

[Algorithms](#)

[Machine  
Coding](#)

[Web  
Fundamentals](#)

# Remove an item from an array in javascript

Posted on [March 6, 2019](#) | by [Prashant Yadav](#)

Posted in [Arrays](#), [Javascript](#)

Learn how to remove single or multiple items from an given [array](#) in javascript.

We will see different ways in which an item can be removed with and without changing the original array.

## Without changing the array

Javascript provides many different ways to remove items from the array. The methods which we are going to use here will not change the original array.

### Using [slice\(\)](#)

If we know the index of the item we want to remove then we can use the [slice\(\)](#) method.

```
let arr = ['prashant', 'sachin', 'yogesh', 'panam', 'pranav'];
let index = 2;
let newArr = arr.slice(0, index).concat(arr.slice(index + 1, arr.length))
//OR
//ES6
let newArr = [...arr.slice(0, index), ...arr.slice(index + 1, arr.length)];

console.log(newArr);
//["prashant", "sachin", "panam", "pranav"]
```

Copy

[slice\(startIndex, endIndex\)](#) method takes two parameters [startIndex](#) and [endIndex](#) and returns all the elements between them(including startIndex's element) as a new array.

Advertisements

■ ■ ■



So we have taken all the elements from `0` to `2` and `3` to `5` and joined them using `concat()` method or [ES6](#) `...` [spread](#) operator.

## Removing multiple values

If we want to remove the multiple values then we can create a helper function and reuse it to remove the elements.

```
let arr = ['prashant', 'sachin', 'yogesh', 'panam', 'pranav'];

let removeItems = (arr, index) => {
  return [...arr.slice(0, index), ...arr.slice(index+1, arr.length)];
}

arr = removeItems(arr, 2);
console.log(arr);
//["prashant", "sachin", "panam", "pranav"]

arr = removeItems(arr, 3);
console.log(arr);
//["prashant", "sachin", "panam"]
```

Copy

Note: Here you must be aware that once you removed an item from the array then the index of other elements changes.

---

## Using [filter\(\)](#)

If we want to remove the element by the value, then we can use the [filter\(\)](#) method to remove the item.

```
let arr = ['prashant', 'sachin', 'yogesh', 'panam', 'pranav'];
let exclude = 'prashant';
let updated = arr.filter(e => e !== exclude);
console.log(updated);
//["sachin", "yogesh", "panam", "pranav"]
```

Copy

[filter\(callback\)](#) method returns a new array with the filtered value based on the given condition. It loops through each element of the array and adds the element in the filtered array only if the given condition is true.

Here we are using `=>` function.

We can also remove multiple values based on the above approach.

[Copy](#)

```
let arr = ['prashant', 'sachin', 'yogesh', 'panam', 'pranav'];
let exclude = ['prashant', 'yogesh'];

let updated = arr.filter(e => exclude.indexOf(e) === -1);
console.log(updated);
//["sachin", "panam", "pranav"]
```

We have created an array of values to be excluded and then used [indexOf](#) method to check if the element is present in the `exclude` array.

## By mutating the array

Javascript also provides another array method which can be used to remove an item from the array.

### Using [splice\(\)](#).

[Copy](#)

```
let arr = ['prashant', 'sachin', 'yogesh', 'panam', 'pranav'];
let index = 2;
arr.splice(index, 1);

console.log(arr);
//["prashant", "sachin", "panam", "pranav"]
```

[splice\(index, deleteCount\)](#) takes two parameters `index` and `deleteCount`. It removes no of element given through `deleteCount` starting from the given index. It modifies the original array i.e it removes the item in place.

### Using [pop\(\)](#).

Removing the last item from the array.

[Copy](#)

```
let arr = ['prashant', 'sachin', 'yogesh', 'panam', 'pranav'];
arr.pop();

console.log(arr);
//["prashant", "sachin", "yogesh", "panam"]
```

[pop\(\)](#) method removes the last item from the array.

### Using [shift\(\)](#).

Removing the first item from the array.

Copy

```
let arr = ['prashant', 'sachin', 'yogesh', 'panam', 'pranav'];
arr.shift();

console.log(arr);
//['sachin', 'yogesh', 'panam', 'pranav']
```

**pop()** method removes the first item from the array.

## By changing the array length

We can also change the array length to remove the last item from the array.

Copy

```
let arr = ['prashant', 'sachin', 'yogesh', 'panam', 'pranav'];
arr.length = 4;

console.log(arr);
//['prashant', 'sachin', 'yogesh', 'panam']
```

[Prepare for your JavaScript Interview practically on each Interview rounds and grab that job.](#)

BEGIN LEARNING

### Recommended Posts:

- [Javascript setInterval method](#)
- [Convert a string to lowercase in javascript](#)
- [Detect if window is scrolled to the bottom](#)
- [Sampling function in JavaScript](#)
- [How to find elements with indexof in javascript](#)
- [Sort object keys in JavaScript](#)
- [Shuffle deck of cards in JavaScript](#)
- [Promise.finally\(\).polyfill](#)
- [Find the element with k frequency in an array](#)
- [Maximum consecutive one's in a binary array](#)

## Leave a Reply

Your email address will not be published. Required fields are marked \*

Comment

Start typing...



Name\*

Name

Email\*

Email

POST COMMENT

Advertisements



[About Us](#)

[Contact Us](#)

[Privacy Policy](#)

[Advertise](#)



Handcrafted with  somewhere in **Mumbai**

© 2023 [LearnersBucket](#) | [Prashant Yadav](#)

