

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](#).



How to flat an array in javascript

Posted on [April 15, 2020](#) | by [Prashant Yadav](#)

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

Learn how to flat an array in javascript.

Flattening an array is simple process of merging different N dimensional sub arrays to form a single array.

For example

Input:

```
[[[1, [1.1]], 2, 3], [4, 5]]
```

Output:

```
[1, 1.1, 2, 3, 4, 5]
```

Copy

ES2019 has introduced two new [array](#) methods `flat` and `flatMap` which can be used to flatten an array.

But as these methods are added recently they are supported only by newer versions of browsers. Firefox 62+, Chrome 69+, Edge 76+ and Safari 12+ currently have the support.

Advertisements



Practically
prepare for
your
JavaScript
interview

[JavaScript
Revision](#)

[JavaScript-
Concept Based
Problems](#)

[Data Structures](#)

[Algorithms](#)

[Machine
Coding](#)

[Web
Fundamentals](#)

Advertisements

You can check the [current browser support for it](#) before using it.



On the other hand if you are using [babel](#) then it will backport your code to previous version.

Flat an array in javascript

`flat` is the new array method which merges the multi dimensional sub arrays to create an one dimensional array with all the elements of sub array.

```
[1, 2, [3, 4]].flat();  
//[1, 2, 3, 4]
```

Copy

By default it only supports flattening for one level. But luckily it accepts an extra parameter which can be used to decide the level.

```
[1, 2, [3, 4]].flat(1);  
//[1, 2, 3, 4]  
  
[1, 2, [3, 4], [[5]]].flat(2);  
//[1, 2, 3, 4, 5]
```

Copy

Pass `Infinity` to flat till any level.

```
[[[1, [1.1]], 2, 3], [4, 5]].flat(Infinity);  
//[1, 1.1, 2, 3, 4, 5]
```

Copy

`flatMap` is combination of `flat` and `map` method of array. It keeps flattening all the array elements.

```
//map  
['Prashant Yadav', 'Learners Bucket'].map(e => e.split(' '));  
//[['Prashant', 'Yadav'], ['Learners', 'Bucket']];  
  
//flatMap  
['Prashant Yadav', 'Learners Bucket'].flatMap(e => e.split(' '));  
//['Prashant', 'Yadav', 'Learners', 'Bucket'];
```

Copy

Polyfill

In case you don't want to use any extra library for a single method. You can create your own custom function for array flattening.

This is the first method which uses modern ES6 features.

[Copy](#)

```
const flatten = (arr) => {  
  return arr.reduce((flat, toFlatten) => {  
    return flat.concat(Array.isArray(toFlatten) ? flatten(toFlatten) : toFlatten);  
  }, []);  
}
```

Second method which uses normal array functions.

[Copy](#)

```
const flatten = function(arr, result = []) {  
  for (let i = 0, length = arr.length; i < length; i++) {  
    const value = arr[i];  
    if (Array.isArray(value)) {  
      flatten(value, result);  
    } else {  
      result.push(value);  
    }  
  }  
  return result;  
};
```

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

[BEGIN LEARNING](#)

Recommended Posts:

[Chop string into chunks of given length](#)

[Set matrix zeroes](#)

[Detect Caps Lock in Javascript](#)

[How to validate JSON in javascript](#)

[useToggle\(\) hook in React](#)

[Aggregate the Input values](#)

[Shuffle deck of cards in JavaScript](#)

[How to capitalize first letter of string in javascript](#)

[Best way to compare strings in javascript](#)

[Count elements in nested array](#)

[Prev](#)[Next](#)

Advertisements



[About Us](#)

[Contact Us](#)

[Privacy Policy](#)

[Advertise](#)



Handcrafted with ♥ somewhere in **Mumbai**

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

