

HTML5 AND BEYOND

JavaScript Check if Multiple Values are in an Array

1

1



Simon



November 9th, 2016

Arrays and checking whether certain values are present in an array is key part of JavaScript, and normally it's not too complex. Often it can be as simple as `array.indexOf(target)`, this then returns either a value of 0 if the value is present, and one of -1 if it's not. That's all you need in a lot of instances, but it doesn't offer much versatility if you want to check if multiple values are present within an array. If you wanted to check multiple values though, it becomes much less clear what the solution is. So today I'd like to share with you some functions that I modified from a [StackOverflow](#) answer. I've included 2 different solutions, the first has been modified slightly to remove the jQuery, so it's just plain old JavaScript, and the second has been modified into a function. All credit should go to [ggreiner](#) and [datacarl](#) for their original answers.

Solution 1

```
1 function containsAll(arr1, arr2){
2
3     for(var i = 0 , len = arr1.length; i < len; i++){
4         if(arr2.indexOf(arr1[i]) == -1) return false;
5     }
6
7     return true;
```

```
8  
9 }
```

- License - [CC BY-SA 3.0](#)

With this solution you need to set the array of values that you're looking for as the first parameter, and the array that's being searched as the second. This is so that it can loop through the second array, and then uses the index of the first to determine whether all the values are present. The use of **indexOf** means that it works in the same way as the example I used in the opening paragraph. So as it's looping through if any are not found in the array and therefore equal to -1 then it returns false. Otherwise if everything is found then it returns true.

Solution 2

```
1 function containsAll(arr1, arr2){  
2  
3     return arr1.every(function(v,i) {  
4         return arr2.indexOf(v) !== -1;  
5     })  
6  
7 }
```

- License - [CC BY-SA 3.0](#)

The way that this is structured in terms of being a function and with the parameters is identical to the first. So you need to set the array of values that you're looking for as the first parameter, and the array that's being searched as the second. Where this solution differs though is the use of the **every method** that's applied to the first array. The every method is used in a similar way to a loop, to test the values again using **indexOf**, much like the last example.

Working Examples

With this first example the values that we're checking for are 1 and 3, and they're both present in the test array. So this means that this first example will return true and therefore console log true.

```
1 var first = [1,3];
2 var second = [1,2,3,4,5];
3
4 if (containsAll(first, second)) {
5     console.log(true);
6 } else {
7     console.log(false);
8 }
```

1

Things are slightly different in the second example though. Here it's looking for the values 4, 5 and 6, and only 4 and 5 are present in the test array. That means that it'll return false because only one out of three of the items are present.

1

```
1 var first = [4,5,6];
2 var second = [1,2,3,4,5];
3
4 if (containsAll(first, second)) {
5     console.log(true);
6 } else {
7     console.log(false);
8     // Returns false because 5 is contained, but 6 isn't
9 }
```

Both of these solutions are perfect for using in IF statements like they have been in the working examples. One real world usage I can see these being especially useful for is eCommerce where you want to be able to specify multiple categories and tags that can be selected to narrow down a search. Each of the values would have to be found to narrow down the items on display, and this could be one potential solution.

I hope you've found this article useful. If you have then please subscribe to our mailing list to keep up to date with all our new content and articles.

JavaScript JavaScript Arrays

You May Also Like

Powered by Sumo

What Are the Costs
Associated with
Marijuana?
OHSonline.com

Get Your Free
eBook
creatureclinic.com

20 Inspiring African
Americans
Impacting The
World Through
Causeartist

Randolph Aviator
Sunglasses Free
Shipping & Free
Returns
World Expert On Sunglass..

1

1



Simon

I've been working in Web Development for the past 4 years in Manchester, UK. I have a passion for all things HTML, CSS, jQuery, JavaScript and WordPress. Plus the Ghost Platform since August 2016.

[VIEW COMMENTS](#)[NAVIGATION](#)

The content and advice discussed on this website is based purely on my opin-

This site is in no way linked or affiliated with the **World Wide Web Consortium**.

[Disclaimer](#)
[Privacy Policy](#)
[Cookie Policy](#)
[Contact Us](#)

ion. Professional advice should always be sought before following anything written on this website.

This website may contain affiliate marketing links, which means we may get paid commission on sales of those products or services we write about. Our editorial content is not influenced by advertisers or affiliate partnerships.

Copyright 2017 © HTML5 and Beyond | Web Development Blog - • All rights reserved.

1

1

