Find object by its property in array of objects with angular way

Asked 2 years, 10 months ago   Active 12 months ago   Viewed 87k times

I have a array like below

▲

42

```
var myArray = [{'id':'73','name':'john'},{'id':'45','name':'Jass'}, etc.]
```

▼

Now I have an `id` 73 how to select this particular object from the array. I see I can do this in jQuery easily with **grep** Is there any

★ **angular way** of doing this?

7   Since most user developing app with angular always get data from array of objects(mostly for table) there should be a helper function for this?

So that I can change the data of the row with the ID of the row by updating the array of object.

I don't want to bind this in view. I want to manipulate the data and update the data withing function.

For Example. I have a table listing. If end user edit a row from the table I have the ID of the object so after end user hit save, I need to update the array and then back to table listing.

javascript   angularjs   arrays

edited Oct 28 '16 at 14:16   asked Oct 28 '16 at 14:09

rram
859   4   15   31

@JaredSmith I'm looking for any angular helper function(like angular.isNumber) for this, not javascript default array object methods. – rram   Oct 28 '16 at 14:30

1   that makes no sense at all. Why would you need a framework helper function to replace a native method supported all the way back to IE 9? Unless you give a reason otherwise, it just comes across as an XY problem meta.stackexchange.com/questions/66377/what-is-the-xy-problem. – Jared Smith Oct 28 '16 at 15:14

I can understand why you wouldn't want to include jquery if you're already using angular, but `.filter` is both native and well-supported. If you can give

a compelling reason why angular, and angular alone, **must** be used to solve this problem I'll retract the close vote. – Jared Smith Oct 28 '16 at 15:15 ✎

4   OP wanted an angularjs answer. The target dupe didn't answer the question. Just like a jQuery solution doesn't answer a javascript question. – Cerbrus
Nov 1 '16 at 13:29 ✎

This question is being discussed on meta – Cerbrus Nov 1 '16 at 13:33

4 Answers

you can use angular's `filter` https://docs.angularjs.org/api/ng/filter/filter

in your controller:

72

```
$filter('filter')(myArray, {'id':73})
```

or in your HTML

```
{{ myArray | filter : {'id':73} }}
```

answered Oct 28 '16 at 14:16

M B
2,050   14   26

Thanks +1 I was thinking to apply the filter only in view mode with markup, thanks for the idea to use it in controller too. – rram   Oct 28 '16 at 14:27 ✎

4   Just a note to fellow learners, We need to pass `$filter` in our controller in order to use this method. – rram   Oct 28 '16 at 14:36 ✎

17   Worth noting that the above code will filter based upon the ID being a substring. So filtering on an id of 73 would also find 733, 4573, 173 etc. To get a
strict comparison you must pass a third comparator parameter of 'true' as per the docs. – Hooligancat May 5 '17 at 18:54

Is there any way to store the result in HTML? I need that filter result in some loop. so this will reduce my filter process in each loop iteration. BTW i am
new to angular. Thanks. – Govind Totla Mar 29 at 13:04

How about plain JavaScript? More about `Array.prototype.filter()` .

41

```
var myArray = [{'id': '73', 'name': 'john'}, {'id': '45', 'name': 'Jass'}]

var item73 = myArray.filter(function(item) {
  return item.id === '73';
})[0];

// even nicer with ES6 arrow functions:
// var item73 = myArray.filter(i => i.id === '73')[0];

console.log(item73); // {"id": "73", "name": "john"}
```

Run code snippet   Expand snippet

answered Oct 28 '16 at 14:18

Timo
26k   10   62   68

Thanks +1 for the help. I already saw the JavaScript filter method, I was looking for angular way. – rram   Oct 28 '16 at 14:28

How about using `FIND` instead? – Dementic Mar 8 at 14:37

@Dementic `find` support was still pretty fresh in Chrome when the answer was written, and for example is not available in IE at all. `filter` on the other hand is. – Timo Mar 9 at 22:47

1   IE ? What is that ? :) – Dementic Mar 10 at 15:24

For complete `M B` answer, if you want to access to an specific attribute of this object already filtered from the array in your HTML, you will have to do it in this way:

6

```
{{ (myArray | filter : {'id':73})[0].name }}
```

So, in this case, it will print `john` in the HTML.

Regards!

answered Aug 8 '18 at 9:17

David Corral
1,865   2   17   25

The solucion that work for me is the following

0

```
$filter('filter')(data, {'id':10})
```

answered Sep 19 '18 at 14:17

Jorge Santos Neill
396　1　5