How can I detect if a selector returns null?

Ask Question

252

85

What is the best way to detect if a jQuery-selector returns an empty object. If you do:

```
alert($('#notAnElement'));
```

you get [object Object], so the way I do it now is:

```
alert($('#notAnElement').get(0));
```

which will write "undefined", and so you can do a check for that. But it seems very bad. What other way is there?

jquery jquery-selectors

edited Jan 22 '13 at 6:48
Ata Iravani
1,006 5 21 34

asked May 28 '09 at 10:46
peirix
18.6k 21 80 116

Home

PUBLIC

Stack Overflow

Tags

Users

8 Answers

My favourite is to extend jQuery with this tiny convenience:

```
$.fn.exists = function () {
    return this.length !== 0;
```

469

✓

```
}
```

Used like:

```
$("#notAnElement").exists();
```

More explicit than using length.

edited Jun 23 '11 at 5:43

answered May 28 '09 at 11:10
Magnar
24.5k 7 53 62

1 that's looking very sweet! Even though I marked this as answered, I'd still like to see more suggestions... (: – peirix May 28 '09 at 11:13

2 Why did you wrap 'this' inside a $() ? wouldn't 'this' in this context actually point to jQuery object ? – Adham Atta Mar 28 '11 at 11:25

1 Great! That's what I wanted. – Pascal Gagneur Dec 18 '12 at 15:23

5 If you want to maintain the ability to chain, should the function return true, or if you would like to use this method to assign `this` to a variable, I would recommend changing the return statement to what CSharp has. – Hanna Nov 6 '13 at 22:52

3 @Adam Yes. But in that case, you should just drop the exists-check, given how jQuery works. It will just not do something if there is no element matching the selector. – Magnar Mar 10 '16 at 7:15

189

```
if ( $("#anid").length ) {
  alert("element(s) found")
}
else {
  alert("nothing found")
}
```

edited Jul 27 '15 at 19:26

Joshua Pinter
25k 9 140 170

answered May 28 '09 at 10:51

duckyflip
11k 4 27 34

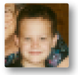
61

The selector returns an array of jQuery objects. If no matching elements are found, it returns an empty array. You can check the `.length` of the collection returned by the selector or check whether the first array element is 'undefined'.

You can use *any* the following examples inside an IF statement and they all produce the same result. True, if the selector found a matching element, false otherwise.

```
$('#notAnElement').length > 0
$('#notAnElement').get(0) !== undefined
$('#notAnElement')[0] !== undefined
```

edited Jan 21 '15 at 3:14



Evan Carroll
36.3k 24 147 249

answered May 28 '09 at 11:06

Jose Basilio
43.7k 9 107 116

1 I use `.length` unless the code is hot. It's certainly the most clear with intent. Also, [`.size()`](https://stackoverflow.com) is deprecated and has been for years (since 1.8). I'm just going to edit your answer and remove that, feel free to add it back

with a note but it's so old, I think it's better gone. – Evan Carroll Jan 21 '15 at 3:13

Technically it returns a jQuery object that contains no DOM nodes. Saying it returns an empty array isn't right. – Vigrant Apr 6 '17 at 19:53

38

I like to do something like this:

```
$.fn.exists = function(){
    return this.length > 0 ? this : false;
}
```

So then you can do something like this:

```
var firstExistingElement =
    $('#iDontExist').exists() ||      //<-returns false;
    $('#iExist').exists() ||          //<-gets assigned to the variab
    $('#iExistAsWell').exists();      //<-never runs

firstExistingElement.doSomething();   //<-executes on #iExist
```

http://jsfiddle.net/vhbSG/

answered Jun 2 '12 at 10:21

CSharp
569 5 6

it is duplicate of accepted answer – Ehsan Sajjad Jun 5 '15 at 6:36

4 @Ehsan Actually it's not a duplicate of the accepted answer...note that the accepted answer only returns true/false while this answer returns either *the original object* ("this") or false. This answer allows you to do additional stuff with the return value (such as additional function chaining) if it is non-false. – RSW Jun 10 '15 at 13:07

1 I have exactly the same code in my standard toolkit. This form is equivalent to object.presence in Rails and super useful in the fallthrough construct that @CSharp describes. – inopinatus Aug 27 '15 at 23:12

This is a good answer, but it would be helpful if you explain what happens if NONE of the tested elements exists. Short answer: `TypeError: firstExistingElement.doSomething is not a function` . You can wrap the entire variable assignment/test in an if() and only do something if an element is found.... – Stephen R May 2 '18 at 20:14

9

I like to use `presence` , inspired from Ruby on Rails:

```
$.fn.presence = function () {
    return this.length !== 0 && this;
}
```

Your example becomes:

```
alert($('#notAnElement').presence() || "No object found");
```

I find it superior to the proposed `$.fn.exists` because you can still use boolean operators or `if` , but the truthy result is more useful. Another example:

```
$ul = $elem.find('ul').presence() || $('<ul class="foo">').appendTo($
$ul.append('...')
```

edited Apr 19 '13 at 7:49

Scott C Wilson
10.9k 8 44 72

answered Feb 21 '13 at 22:44

Marc-André Lafortune
63.3k 12 139 158

My preference, and I have no idea why this isn't already in jQuery:

7

```
$.fn.orElse = function(elseFunction) {
  if (!this.length) {
    elseFunction();
  }
};
```

Used like this:

```
$('#notAnElement').each(function () {
  alert("Wrong, it is an element")
}).orElse(function() {
  alert("Yup, it's not an element")
});
```

Or, as it looks in CoffeeScript:

```
$('#notAnElement').each ->
  alert "Wrong, it is an element"; return
.orElse ->
  alert "Yup, it's not an element"
```

edited Sep 17 '12 at 17:05

answered Aug 6 '12 at 20:10

nilskp

2,324 1 24 30

This is in the JQuery documentation:

6

http://learn.jquery.com/using-jquery-core/faq/how-do-i-test-whether-an-element-exists/

```
alert( $( "#notAnElement" ).length ? 'Not null' : 'Null' );
```

answered May 13 '14 at 21:29

Daniel De León
9,202 4 62 54

-1

You may want to do this all the time by default. I've been struggling to wrap the jquery function or jquery.fn.init method to do this without error, but you can make a simple change to the jquery source to do this. Included are some surrounding lines you can search for. I recommend searching jquery source for `The jQuery object is actually just the init constructor 'enhanced'`

```
var
  version = "3.3.1",

  // Define a local copy of jQuery
  jQuery = function( selector, context ) {

    // The jQuery object is actually just the init constructor 'enhan
    // Need init if jQuery is called (just allow error to be thrown i
    var result = new jQuery.fn.init( selector, context );
    if ( result.length === 0 ) {
      if (window.console && console.warn && context !== 'failsafe') {
        if (selector != null) {
          console.warn(
            new Error('$(\''+selector+'\') selected nothing. Do $(sel
silence warning. Context:'+context)
          );
        }
      }
    }
    return result;
  },

  // Support: Android <=4.0 only
  // Make sure we trim BOM and NBSP
  rtrim = /^[\s\uFEFF\xA0]+|[\s\uFEFF\xA0]+$/g;

jQuery.fn = jQuery.prototype = {
```

Last but not least, you can get the uncompressed jquery source code here: http://code.jquery.com/

answered Nov 29 '18 at 1:27

Devin G Rhode

14.6k 5 27 43