jQuery get html of container including the container itself

Ask Question

132

28

How do i get the html on '#container' including '#container' and not just what's inside it.

```
<div id="container">
 <div id="one">test 1 </div>
 <div id="two">test 2 </div>
 <div id="three">test 3 </div>
 <div id="four">test 4 </div>
</div>
```

I have this which gets the html inside #container. it does not include the #container element itself. That's what i'm looking to do

```
var x = $('#container').html();
$('#save').val(x);
```

Check http://jsfiddle.net/rzfPP/58/

javascript jquery

edited Jun 23 '11 at 19:14
mc10
8,459 4 28 51

asked Jun 23 '11 at 19:12
Pinkie
4,324 19 68 114

you could put container inside another container and get that containers html... but that seems a little hacky. perhaps if we knew a little more about the problem, we could come up with a workable solution? what are you doing with a text area full of html? – Patricia Jun 23 '11 at 19:18

possible duplicate of Get selected element's outer HTML – Ian Mackinnon Apr 10 '15 at 10:15

10 Answers

141

If you wrap the container in a dummy `P` tag you will get the container HTML also.

All you need to do is

✓

```
var x = $('#container').wrap('<p/>').
```

Check working example at http://jsfiddle.net/rzfPP/68/

To `unwrap()` the `<p>` tag when done, you can add

```
$('#container').unwrap();
```

edited Apr 17 '13 at 8:10
Toby Allen
8,778 8 63 117

answered Jun 23 '11 at 20:07
Hussein
35.4k 20 101 137

1 Very nice solution yet so simple. – Pinkie Jun 23 '11 at 20:18

2 That's still creating an extra element... – mc10 Jun 24 '11 at 0:00

16 @mc10 we can simply use clone() and you will not have to worry about extra elements created. `var x = $('#container').clone().wrap('<p />').parent().html();` . The idea of wrap is great and allot less complicated then most of the solutions provided. – Pinkie Jun 24 '11 at 0:18

5 Why a `<p>` tag? Wouldn't a `<div>` make more sense? – Martin Burch Jun 12 '14 at 23:26

2 This has a much simpler solution. See my answer. – 1.44mb Sep 2 '15 at 10:34

```
var x = $('#container').get(0).outerHT
```

112

UPDATE : This is now supported by Firefox as of FireFox 11 (March 2012)

As others have pointed out, this will not work in FireFox. If you need it to work in FireFox, then you might want to take a look at the answer to this question : [In jQuery, are there any function that similar to html() or text() but return the whole content of matched component?](#)

edited May 23 '17 at 11:54

Community ♦
1 1

answered Jun 23 '11 at 19:15

ShaneBlake
9,454 2 21 41

1 Very handy, thanks! – Jared Eitnier Oct 12 '12 at 14:19

6 This works in Firefox, and it's better than the accepted solution as of now. – luqita Mar 12 '14 at 18:06

this is an elegant way to do that, better then the accepted answer which was just a dirty hack – minhajul Oct 29 '15 at 7:07

64

```
var x = $('#container')[0].outerHTML;
```

answered Jun 23 '11 at 19:16

MikeM
23.8k 4 53 75

1 Firefox issue is out-of-date so I suggest to vote up this answer. – Rob Dec 16 '13 at 11:59

1 Using .clone() works, but this is much cleaner, imo. The accepted answer creates new elements in the DOM = bad. – pete Mar 15 '14 at 20:30

50

I like to use this;

```
$('#container').prop('outerHTML');
```

answered Aug 2 '15 at 17:47

1.44mb
2,458 19 30

3 This seems like the best solution to me, requires no dom manipulation, or pseudo dom manipulation (the wrap methods). And the jquery object already has the property. – Nieminen Jul 26 '17 at 19:47

love this solution – Sameera Kumarasingha Dec 31 '18 at 8:59

12

```
$('#container').clone().wrapAll("<div/
```

Update: outerHTML works on firefox now so use the other answer unless you need to support very old versions of firefox

edited May 26 '16 at 14:23

answered Jun 23 '11 at 19:17

Robert Noack
971 9 20

outerHTML property does not work on Firefox so you need to do it with clone – Robert Noack Jun 23 '11 at 19:20

1 Better solution than the accepted. – Ibraheem Feb 26 '14 at 14:05

1

```
$.fn.outerHtml = function()
{
if (this.length)
{
    var div = $('<div style="display:n
    var clone =
    $(this[0].cloneNode(false)).html(t
    var outer = div.html();
    div.remove();
    return outer;
}
else
    return null;
};
```

from http://forum.jquery.com/topic/jquery-getting-html-and-the-container-element-12-1-2010

edited Jun 24 '11 at 5:53

Brad Mace
20.4k 12 82 132

answered Jun 23 '11 at 19:21

Achu Sreedhar
11 1

It doesn't work. – Pinkie Jun 23 '11 at 19:29

0

```
var x = $($('div').html($('#container'
```

edited Dec 15 '11 at 2:49

animuson ♦
42.4k 22 115 130

answered Jun 23 '11 at 19:15

kei
15.8k 1 25 53

6 You are creating 3 jquery objects in one statement. Although it works, it's an overkill. – Pinkie Jun 23 '11 at 19:27

0

Oldie but goldie...

Since user is asking for jQuery, I'm going to keep it simple:

```
var html = $('#container').clone();
console.log(html);
```

[Fiddle here.](#)

edited Jan 8 '17 at 12:45

Konrad Viltersten
12.5k 31 131 253

answered Feb 3 '16 at 13:24

prinsen
549 4 6

1 This does not help in retrieving the html of the container itself. It does not even return html code. I can see that it may be practical to access the target node via clone to avoid modifying the dom, but then it would be good to mention why you use this method. – AeonOfTime Jul 24 '18 at 7:11

0

[Firefox doesn't support outerHTML](#), so you need to [define a function](#) to help support it:

```
function outerHTML(node) {
    return node.outerHTML || (
        function(n) {
            var div = document.createE
            div.appendChild( n.cloneNo
            var h = div.innerHTML;
            div = null;
            return h;
        }
    )(node);
}
```

Then, you can use outerHTML:

```
var x = outerHTML($('#container').get(
$('#save').val(x);
```

edited May 23 '17 at 11:47

Community ♦
1 1

answered Jun 23 '11 at 19:24

mc10
8,459 4 28 51

-2

Simple solution with an example :

```
<div id="id_div">
  <p>content<p>
</div>
```

Move this DIV to other DIV with id = "other_div_id"

```
$('#other_div_id').prepend( $('#id_di
```

Finish

answered Aug 30 '13 at 13:22

hassane86
1

The OP is requesting the raw HTML represented by a jQuery object, where this method moves a jQuery object from one place to another. – Simon Robb Nov 21 '13 at 1:56