jQuery: select all elements of a given class, except for a particular Id

Ask Question

135

20

This is probably pretty simple.

I want to select all elements of a given class `thisClass` , except where the id is `thisId` .

i.e. something equivalent to (where -/minus implies remove):

```
$(".thisClass"-"#thisId").doAction();
```

jquery jquery-selectors

edited Mar 21 '16 at 21:44

Jayant Bhawal
642 13 23

asked Mar 31 '10 at 7:43

Ankur
21.5k 97 224 303

6 Answers

Use the [:not selector](#).

Home

PUBLIC

Stack Overflow

Tags

✓

If you have multiple ids or selectors just use the comma delimiter, in addition:

```
(".thisclass:not(#thisid,#thatid)").doAction();
```

edited Oct 25 '16 at 17:37
JonH
27.9k 9 71 129

answered Mar 31 '10 at 7:46
rahul
153k 43 206 248

3 What if you want to exclude more than one classes? THX – SoulMagnet Feb 3 '14 at 13:41

13 From the documentation: `All selectors are accepted inside :not(), for example: :not(div a) and :not(div,a)` so just use the comma delimited selectors to do multiple `(".thisclass:not(#thisid,#thatid)").doAction();` – Chase Mar 11 '14 at 1:22

worked with single quotes like this - $(".thisclass:not('#thisid')").doAction(); – Muhammad Raja Aug 18 '15 at 9:43

or if you want to bind to all child of elements that has a specific class name, except one, you can do like this:$('.thisclass:not(#id) .otherclass').doAction() – dalmate Jan 11 '16 at 15:37

what if i have a standard code like: $('#some').notimportant, $('#another').dosomethingelse and i want to avoid execution over an specific id given dynamically? – Botea Florin Nov 16 '17 at 20:39

https://api.jquery.com/not/

29

```
$(".thisClass").not("#thisId").doAction();
```

edited Sep 21 '15 at 10:52

answered Apr 21 '14 at 19:51

Karl Adler
7,371 3 42 64

3 `.not()` isn't a selector. It's a function. But there is also `:not()` selector like other answer mentions. – Wolle Jul 29 '14 at 10:45

I'll just throw in a JS (ES6) answer, in case someone is looking for it:

6

```
Array.from(document.querySelectorAll(".myClass:not(#myId)"
    doSomething(el);
}
```

Update (this may have been possible when I posted the original answer, but adding this now anyway):

```
document.querySelectorAll(".myClass:not(#myId)").forEach((
    doSomething(el);
});
```

This gets rid of the `Array.from` usage.

other things): https://developer.mozilla.org/en-US/docs/Web/API/NodeList

edited Oct 30 '18 at 11:32

answered Mar 21 '16 at 21:32

Jayant Bhawal

642 13 23

5

```
$(".thisClass[id!='thisId']").doAction();
```

Documentation on selectors: http://api.jquery.com/category/selectors/

edited Dec 1 '14 at 23:30

jenzz

6,385 5 42 66

answered Mar 31 '10 at 7:46

Amy B

14.1k 12 57 76

5 did you mean $(".thisClass[id!='thisId']").doAction(); ? – detay Jun 23 '11 at 11:16

9 Is there a missing ']' in the code sample? – David Oct 26 '12 at 20:36

You could use the **.not** function like the following examples to remove items that have an exact id, id containing a http://www.w3schools.com/jquery/jquery_ref_selectors.asp for more information on jQuery selectors.

Ignore by Exact ID:

```
 $(".thisClass").not('[id="thisId"]').doAction();
```

Ignore ID's that contains the word "Id"

```
$(".thisClass").not('[id*="Id"]').doAction();
```

Ignore ID's that start with "my"

```
$(".thisClass").not('[id^="my"]').doAction();
```

edited Mar 24 '16 at 14:29

answered Mar 21 '16 at 21:17

ScottyG

1,274 1 14 28

2

Using the `.not()` method with selecting an entire element is also an option.

This way could be usefull if you want to do another action with that element directly.

```
$(".thisClass").not($("#thisId")[0].doAnotherAction()).doA
```

2,912 1 11 22