What are valid values for the id attribute in HTML?

Asked 12 years, 2 months ago   Active 3 months ago   Viewed 463k times

▲

2072

▼

🔖

249

🕐

When creating the `id` attributes for HTML elements, what rules are there for the value?

`html`

98   This differs between HTML5 and previous versions of the spec. I explained it here: mathiasbynens.be/notes/html5-id-class – Mathias Bynens Oct 24 '11 at 8:41

5   I noticed SharePoint 2010 assigning a value like this - {8CC7EF38-31D8-4786-8C20-7E6D56E49AE2}-{E60CE5E2-6E64-4350-A884-654B72DA5A53} for a dynamically generated table within a Web Part & a page containing an ID value of that sort did not break in any of the popular browsers. Dealing with such ID values through JavaScript is tricky though - mvark.blogspot.in/2012/07/... – mvark Jul 20 '12 at 18:34

3   HTML4 and HTML5 requirements for `ID` values are very different. Here's a quick and complete rundown of HTML5 `ID` rules: stackoverflow.com/a/31773673/3597276 – Michael Benjamin Aug 3 '15 at 0:32

6   Please note: Doing as some of the answers have said and using a period (`**.**`) with jQuery will run you into quite a bit of trouble, for example, using `<input id="me.name" />` and then `$("#me.name").val()` will cause jQuery to look for a `<me>` tag with the class `.name` , which no one wants really! – Can O' Spam Aug 26 '15 at 15:20 ✏

3   @SamSwift웃 Nope, you just have to escape special characters as docs say. Please check this online demo. – Álvaro González Nov 10 '16 at 9:58

23 Answers

| Active | Oldest | Votes |

▲

1740

▼

✓

🕐

For HTML 4, the answer is technically:

> ID and NAME tokens must begin with a letter ([A-Za-z]) and may be followed by any number of letters, digits ([0-9]), hyphens ("-"), underscores ("_"), colons (":"), and periods (".").

HTML 5 is even more permissive, saying only that an id must contain at least one character and may not contain any space characters.

The id attribute is case sensitive in XHTML.

As a purely practical matter, you may want to avoid certain characters. Periods, colons and '#' have special meaning in CSS selectors, so you will have to escape those characters using a backslash in CSS or a double backslash in a selector string passed to jQuery. Think about how often you will have to escape a character in your stylesheets or code before you go crazy with periods and colons in ids.
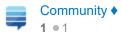
For example, the HTML declaration `<div id="first.name"></div>` is valid. You can select that element in CSS as `#first\.name` and in jQuery like so: `$('#first\\.name')` . But if you forget the backslash,

`$('#first.name')`, you will have a perfectly valid selector looking for an element with id `first` and also having class `name`. This is a bug that is easy to overlook. You might be happier in the long run choosing the id `first-name` (a hyphen rather than a period), instead.

You can simplify your development tasks by strictly sticking to a naming convention. For example, if you limit yourself entirely to lower-case characters and always separate words with either hyphens or underscores (but not both, pick one and never use the other), then you have an easy-to-remember pattern. You will never wonder "was it `firstName` or `FirstName`?" because you will always know that you should type `first_name`. Prefer camel case? Then limit yourself to that, no hyphens or underscores, and always, consistently use either upper-case or lower-case for the first character, don't mix them.

A now very obscure problem was that at least one browser, Netscape 6, incorrectly treated id attribute values as case-sensitive. That meant that if you had typed `id="firstName"` in your HTML (lower-case 'f') and `#FirstName { color: red }` in your CSS (upper-case 'F'), that buggy browser would have failed to set the element's color to red. At the time of this edit, April 2015, I hope you aren't being asked to support Netscape 6. Consider this a historical footnote.

edited Dec 11 '19 at 18:26
Community ♦
1 ● 1

answered Sep 17 '08 at 1:42
dgvid
24.3k ● 4 ● 37 ● 55

79 Note that class and id attributes *are* case-sensitive in XHTML, all other attributes are not. Eric Meyer mentioned this in a CSS workshop I attended. – John Topley Apr 22 '09 at 10:35

33 Also note that if you try to write a CSS rule to target an element by ID, and the ID beings with a number, it won't work. Bummer! – Zack The Human Jan 20 '10 at 0:53

56 As for '.' or ':' in an ID using jQuery, see the jQuery FAQ. It contains a small function that does the necessary escaping. – Wolfram May 6 '10 at 10:18 ✏️

7   The `id` attribute is [w3.org/TR/html4/struct/global.html#adef-id](case sensitive in HTML4) and has to begin with a letter (limited to A to Z). Also note that your example should not make your element's text color red since your CSS refers to an element with class `FirstName` not to your `id`. – Augustus Kling Sep 30 '11 at 7:55

5   The error that Augustus pointed out is still present. In your example, if you have `id="firstName"` in either HTML 4 or HTML 5 and `#FirstName { color: red }` in CSS, *only a buggy browser will* set the element's color to red. – Stephen Booher Jan 26 '12 at 17:15

▲

225

▼

↺

From the HTML 4 specification:

> ID and NAME tokens must begin with a letter ([A-Za-z]) and may be followed by any number of letters, digits ([0-9]), hyphens ("-"), underscores ("_"), colons (":"), and periods (".").

A common mistake is to use an ID that starts with a digit.

answered Sep 16 '08 at 9:09
Peter Hilton
16.6k ● 6 ● 46 ● 75

21 Note that HTML5 allows much more then HTML4 see for example 456bereastreet.com/archive/201011/... and w3.org/TR/html5/elements.html#the-id-attribute – Mr Shark Nov 30 '10 at 8:33

IE6 didn't support ID starting with underscores, but it's dead anyway. – doc_id Mar 3 '12 at 12:00

2   @rahmanisback regarding IE6, one would have thought so, but I'm finishing up a proposal right now for a bank and they insist that any application a vendor develops runs in IE6. This is for 30,000 users. Heck, if we could just get them to update their browsers on all those desktops, it might just help the unemployment rate. – Karl Sep 13 '12 at 13:50

2   @Karl I'm sorry to hear this. Do all of your efforts to warn about IE6 security issues. However IE7 will soon be the new IE6, so yeah it appears to be our fate in this industry to remedy MS past mistakes. – doc_id Sep 14 '12 at 6:25

@MrShark The second link is broken; New link – SWdV Jun 14 '15 at 14:34 ✎

159

▲ ▼

+50

↻

You can technically use colons and periods in id/name attributes, but I would strongly suggest avoiding both.

In CSS (and several JavaScript libraries like jQuery), both the period and the colon have special meaning and you will run into problems if you're not careful. Periods are class selectors and colons are pseudo-selectors (eg., ":hover" for an element when the mouse is over it).

If you give an element the id "my.cool:thing", your CSS selector will look like this:

```
#my.cool:thing { ... /* some rules */ ... }
```

Which is really saying, "the element with an id of 'my', a class of 'cool' and the 'thing' pseudo-selector" in CSS-speak.

Stick to A-Z of any case, numbers, underscores and hyphens. And as said above, make sure your ids are unique.

That should be your first concern.

edited Dec 30 '13 at 15:03   answered Sep 16 '08 at 14:01

Michael Thompson
3,663 ● 1 ● 27 ● 22

22   You can use colons and periods - but you'll need to escape them using double backslashes, eg: $('#my\\.cool\\:thing') or escaping a variable: $('#'+id.replace(/\./,'\\.').replace(/\:/,'\\:')) groups.google.com/group/jquery-en/browse_thread/thread/… – joeformd Dec 3 '09 at 10:41

2   Why not numerals; why just A-Z? Numbers are very useful IDs when referring to elements that are related to data that's keyed with a number, as long as you don't start with the number. – cori May 2 '11 at 16:35

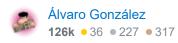5   Just FYI, dashes *are* technically hyphens. Minus sign isn't in ASCII character set. en.wikipedia.org/wiki/Plus_and_minus_signs#Character_codes – Anton Strogonoff Jul 8 '11 at 18:31

4   If you have these characters (`.` , `:`) in ids, and cannot remove them (cough ... Sharepoint), you can get around this in CSS with attribute selectors instead of id selectors, e.g. `[id='my.cool:thing']` , however this selector will have a lower specificity than an id selector, which might cause other problems. – Faust Jun 7 '13 at 7:36

2   Old, I know, but updated to include numbers and backpedal on hyphens – Michael Thompson Dec 30 '13 at 15:03

68

▲

jQuery **does** handle any valid ID name. You just need to escape metacharacters (i.e., dots, semicolons, square brackets...). It's like saying that JavaScript has a problem with quotes only because you can't write

```
var name = 'O'Hara';
```

answered Jan 10 '09 at 20:40

Álvaro González
126k ●36 ●227 ●317

Strictly it should match

67

```
[A-Za-z][-A-Za-z0-9_:.]*
```

But jquery seems to have problems with colons so it might be better to avoid them.

edited Dec 9 '11 at 10:19   answered Sep 16 '08 at 9:11

nasmorn   Mr Shark
2,070 ●1 ●18 ●28   **23.8k** ●5 ●29 ●36

5   Or alternatively: "so it might be better to avoid jquery". ;) – domsson Jan 11 at 13:52

HTML5:

63

gets rid of the additional restrictions on the id attribute see here. The only requirements left (apart from being unique in the document) are:

1. the value must contain at least one character (can't be empty)

2. it can't contain any space characters.

PRE-HTML5:

ID should match:

```
[A-Za-z][-A-Za-z0-9_:.]*
```

1. Must Start with A-Z or a-z characters
2. May contain `-` (hyphen), `_` (underscore), `:` (colon) and `.` (period)

but one should avoid `:` and `.` beacause:

For example, an ID could be labelled "a.b:c" and referenced in the style sheet as #a.b:c but as well as being the id for the element, it could mean id "a", class "b", pseudo-selector "c". Best to avoid the confusion and stay away from using . and : altogether.

edited Jun 20 at 9:12   answered Jan 18 '13 at 7:09

Community ♦   Zaheer Ahmed

HTML5: Permitted Values for ID & Class Attributes

60

As of HTML5, the only restrictions on the value of an ID are:

1. must be unique in the document

2. must not contain any space characters

3. must contain at least one character

Similar rules apply to classes (except for the uniqueness, of course).

So the value can be all digits, just one digit, just punctuation characters, include special characters, whatever. Just no whitespace. This is very different from HTML4.

In HTML 4, ID values must begin with a letter, which can then be followed only by letters, digits, hyphens, underscores, colons and periods.

In HTML5 these are valid:

```
<div id="999"> ... </div>
<div id="#%LV-||"> ... </div>
<div id="____V"> ... </div>
<div id="⌘⌥"> ... </div>
<div id="♥"> ... </div>
<div id="{}"> ... </div>
<div id="©"> ... </div>
<div id="⇧⌘¤☆€~¥"> ... </div>
```

Just bear in mind that using numbers, punctuation or special characters in the value of an ID may cause trouble in other contexts (e.g., CSS, JavaScript, regex).

For example, the following ID is valid in HTML5:

```
<div id="9lions"> ... </div>
```

However, it is invalid in CSS:

From the CSS2.1 spec:

> ## 4.1.3 Characters and case
>
> In CSS, *identifiers* (including element names, classes, and IDs in selectors) can contain only the characters [a-zA-Z0-9] and ISO 10646 characters U+00A0 and higher, plus the hyphen (-) and the underscore (_); **they cannot start with a digit, two hyphens, or a hyphen followed by a digit**.

In most cases you may be able to escape characters in contexts where they have restrictions or special meaning.

W3C References

HTML5

> ### 3.2.5.1 The `id` attribute
>
> The `id` attribute specifies its element's unique identifier (ID).
>
> The value must be unique amongst all the IDs in the element's home subtree and must contain at least one character. The value must not contain any space characters.
>
> Note: There are no other restrictions on what form an ID can take; in particular, IDs can consist of just digits, start with a digit, start with an underscore, consist of just punctuation, etc.
>
> ### 3.2.5.7 The `class` attribute
>
> The attribute, if specified, must have a value that is a set of space-separated tokens representing the various classes that the element belongs to.
>
> The classes that an HTML element has assigned to it consists of all the classes returned when the value of the class attribute is split on spaces. (Duplicates are ignored.)
>
> There are no additional restrictions on the tokens authors can use in the class attribute, but authors are encouraged to use values that describe the nature of the content, rather than values that describe the desired presentation of the content.
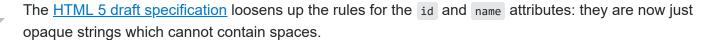
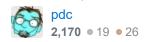edited Dec 5 '16 at 18:29   answered Aug 2 '15 at 15:53

Michael Benjamin
243k ●75 ●420 ●541

33

In practice many sites use `id` attributes starting with numbers, even though this is technically not valid HTML.

The HTML 5 draft specification loosens up the rules for the `id` and `name` attributes: they are now just opaque strings which cannot contain spaces.

answered Sep 16 '08 at 10:04

pdc
2,170 ●19 ●26

31

Hyphens, underscores, periods, colons, numbers and letters are all valid for use with CSS and JQuery. The following should work but it must be unique throughout the page and also must start with a letter [A-Za-z].

Working with colons and periods needs a bit more work but you can do it as the following example shows.

```html
<html>
<head>
<title>Cake</title>
<style type="text/css">
```

```
        #i\.Really\.Like\.Cake {
            color: green;
        }
        #i\:Really\:Like\:Cake {
            color: blue;
        }
    </style>
    </head>
    <body>
        <div id="i.Really.Like.Cake">Cake</div>
        <div id="testResultPeriod"></div>

        <div id="i:Really:Like:Cake">Cake</div>
        <div id="testResultColon"></div>
        <script type="text/javascript"
src="http://ajax.googleapis.com/ajax/libs/jquery/1.4.2/jquery.min.js"></script>
        <script type="text/javascript">
            $(function() {
                var testPeriod = $("#i\\.Really\\.Like\\.Cake");
                $("#testResultPeriod").html("found " + testPeriod.length + " result.");

                var testColon = $("#i\\:Really\\:Like\\:Cake");
                $("#testResultColon").html("found " + testColon.length + " result.");
            });
        </script>
    </body>
    </html>
```

HTML5

25

Keeping in mind that ID must be unique, ie. there must not be multiple elements in a document that have the same id value.

The rules about ID content in HTML5 are (apart from being unique):

```
This attribute's value must not contain white spaces. [...]
Though this restriction has been lifted in HTML 5,
an ID should start with a letter for compatibility.
```

This is the **W3** spec about ID (från MDN):

```
Any string, with the following restrictions:
must be at least one character long
must not contain any space characters
Previous versions of HTML placed greater restrictions on the content of ID values
(for example, they did not permit ID values to begin with a number).
```

More info:

- **W3** - global attributes (`id`)

- **MDN** atribute (`id`)

18

To reference an id with a period in it you need to use a backslash. Not sure if its the same for hyphens or underscores. For example: HTML

```
<div id="maintenance.instrumentNumber">############0218</div>
```

CSS

```
#maintenance\.instrumentNumber{word-wrap:break-word;}
```

3   Hyphens and underscores don't normally need to be escaped. However, the exception to this is if the hyphen appears at the start of the identifier and is followed by another hyphen (eg. `\--abc`) or a digit (eg. `\-123`). – MrWhite Nov 14 '13 at 1:25

16

From the HTML 4 spec...

ID and NAME tokens must begin with a letter ([A-Za-z]) and may be followed by any number of letters, digits ([0-9]), hyphens ("-"), underscores ("_"), colons (":"), and periods (".").

EDIT: d'oh! Beaten to the button, again!

16

Also, never forget that an ID is unique. Once used, the ID value may not appear again anywhere in the document.

You may have many ID's, but all must have a unique value.

On the other hand, there is the class-element. Just like ID, it can appear many times, but the value may be used over and over again.

A unique identifier for the element.

12 There must not be multiple elements in a document that have the same id value.

Any string, with the following restrictions:

1. must be at least one character long

2. must not contain any space characters:

   - U+0020 SPACE
   - U+0009 CHARACTER TABULATION (tab)
   - U+000A LINE FEED (LF)
   - U+000C FORM FEED (FF)
   - U+000D CARRIAGE RETURN (CR)

Using characters except `ASCII letters and digits, '_', '-' and '.'` may cause compatibility problems, as they weren't allowed in `HTML 4`. Though this restriction has been lifted in `HTML 5`, an ID should start with a letter for compatibility.

edited Oct 6 '16 at 13:55
The_Lone_Devil
476 ● 10 ● 15

answered Jul 15 '16 at 13:02
Bhavin Solanki
1,266 ● 10 ● 25

---

11 It appears that although colons (:) and periods (.) are valid in the HTML spec, they are invalid as id selectors in [CSS](#) so probably best avoided if you intend to use them for that purpose.

answered Apr 22 '09 at 10:18
lstg
127 ● 1 ● 4

> @MathiasBynens The link is broken. Now it's [mothereff.in/css-escapes#0foo%23bar.baz%3Aqux](#) – Oriol Nov 8 '14 at 19:52 ✎

> They aren't invalid if you escape them correctly. See [mothereff.in/css-escapes#0foo%23bar.baz%3Aqux](#) – Mathias Bynens Nov 10 '14 at 7:09

---

10 for `HTML5`

> The value must be unique amongst all the IDs in the element's home subtree and must contain at least one character. The value must not contain any space characters.

At least one character, no spaces.

This opens the door for valid use cases such as using accented characters. It also gives us plenty of more ammo to shoot ourselves in the foot with, since you can now use id values that will cause problems with both CSS and JavaScript unless you're really careful.

answered Aug 21 '13 at 10:57

9

1. IDs are best suited for naming parts of your layout so should not give same name for ID and class

2. ID allows alphanumeric and special characters

3. but avoid using of `# : . * !` symbols

4. not allowed spaces

5. not started with numbers or a hyphen followed by a digit

6. case sensitive

7. using ID selectors is faster than using class selectors

8. use hyphen "-" (underscore "_" can also use but not good for seo) for long CSS class or Id rule names

9. If a rule has an ID selector as its key selector, don't add the tag name to the rule. Since IDs are unique, adding a tag name would slow down the matching process needlessly.

10. In HTML5, the id attribute can be used on any HTML element and In HTML 4.01, the id attribute cannot be used with: `<base>, <head>, <html>, <meta>, <param>, <script>, <style>, and <title>.`

edited May 2 '13 at 6:51   answered May 2 '13 at 6:28

Web Designer cum
Promoter
1,285 ● 14 ● 25

"underscore can also use but not good for seo": Since when did search engines take any interest in CSS class or id names? – Nick Rice May 7 '16 at 16:55

9

Any **Alpha-numeric value** and "**-**" and "**_**" is valid. But, you should *start the id name* with any character between **A-Z** or **a-z**.

answered Nov 12 '16 at 13:52

Tazwar Utshas
731 ● 1 ● 14 ● 25

6

No spaces, must begin with at least a char from a to z and 0 to 9.

answered May 9 '17 at 4:22

Wembo Mulumba
699 ● 5 ● 8

5

values can be : [a-z],[A-Z],[0-9],[* _ : -]

it is use for HTML5 ...

we can add id with any tag.

In HTML

4 **ID** should start with **{A-Z}** or **{a-z}** you can Add **Digits, period, hyphen, underscore, colons.**

For example:

```
<span id="testID2"></span>
<span id="test-ID2"></span>
<span id="test_ID2"></span>
<span id="test:ID2"></span>
<span id="test.ID2"></span>
```

But Even Though You can Make ID with Colons(:) or period(.) It is hard for **CSS** to use these ID as Selector. Mainly when you want to Use Pseudo elements (:before,:after).

Also in **JS** it is Hard to select these ID's. So you should use first four ID's As preferred by many developer around and if it's necessary than you can use last two also.

Since *ES2015* we can as well use -almost- all **unicode characters** for ID's, if the document charset is set to UTF8.

2 Test out here: https://mothereff.in/js-variables

Enter a variable name: permalink

ಠ_ಠ

That's a valid identifier according to ECMAScript 6 / Unicode 8.0.0.

Read about: https://mathiasbynens.be/notes/javascript-identifiers-es6

> In ES2015, identifiers must start with $, _, or any symbol with the Unicode derived core property ID_Start.
>
> The rest of the identifier can contain $, _, U+200C zero width non-joiner, U+200D zero width joiner, or any symbol with the Unicode derived core property ID_Continue.

```
const target = document.querySelector("div").id
console.log(
    target
)
document.getElementById(target).style.backgroundColor = "black"
```

```css
div {
  border: 1px black solid;
  width: 100%;
  height: 200px
}
```

```html
<div id="HECOM-ETW">
```

[▷ Run code snippet]   [↗ Expand snippet]

Should you use it? Probably not a good idea!

Read about: https://stackoverflow.com/a/52799593/2494754

edited May 20 at 15:12   answered Sep 7 '19 at 21:36

                                          NVRM
                                          5,371 ●1 ●41 ●54

▲

-4

▼

alphabets-> caps & small
digits-> 0-9
special chars-> ':', '-', '_', '.'

↺   the format should be either starting from '.' or an alphabet, followed by either of the special chars of more
alphabets or numbers. the value of the id field must not end at an '_'.
Also, spaces are not allowed, if provided, they are treated as different values, which is not valid in case of the
id attributes.

                                          answered Sep 22 '12 at 18:27

                                          Shashank N. Pandey
                                          11 ●1

🔥 **Highly active question**. Earn 10 reputation in order to answer this question. The reputation requirement helps protect this
    question from spam and non-answer activity.