

What is the difference between readonly="true" & readonly="readonly"?

Asked 8 years, 3 months ago Active 1 year, 1 month ago Viewed 106k times

What is the difference between:

62

```
<input name="TextBox1" type="text" id="TextBox1" readonly="true" />
```

and:



11

```
<input name="TextBox1" type="text" id="TextBox1" readonly="readonly" />
```

When i set `readonly` to `true` it works somewhat different from `readonly='readonly'`. W3C standard says `readonly` should be `'readonly'` & not `'true'`. Why most of the browsers allow `readonly='true'` which has somewhat different functionality than `readonly='readonly'` ?

html

edited Aug 14 '14 at 7:16



Ciro Santilli 新疆改造中心996ICU六四事件
169k 36 653 521

asked May 30 '11 at 6:22



TechITek
2,630 12 48 82

What markup DOCTYPE, DTD, and what browser mode? – Keith May 30 '11 at 6:27

- 3 A more global answer for your specific question is that browsers are made to be compatible with all sorts of loose (and sometimes terrible) coding. Figuring out the strictest coding that will work with all browsers is a virtue, because kludges will only work until browser developers get fed up with them :) – Merlyn Morgan-Graham May 30 '11 at 6:38

related stackoverflow.com/questions/7730695/... – Adrien Be Jan 24 '14 at 14:07

possible duplicate of stackoverflow.com/questions/1033944/... because both are boolean attributes (not flagged) – Ciro Santilli 新疆改造中心996ICU六四事件 Jul 5 '14 at 16:37

By using our site, you acknowledge that you have read and understand our Cookie Policy, Privacy Policy, and our Terms of Service.



61



Giving an element the attribute `readonly` will give that element the `readonly` status. It doesn't matter what value you put after it or if you put any value after it, it will still see it as `readonly`. Putting `readonly="false"` won't work.

Suggested is to use the W3C standard, which is `readonly="readonly"`.

edited Jul 26 '18 at 9:43



Jez

12.9k

20

86

160

answered May 30 '11 at 6:29



jerone

10.9k

2

33

51

12 Putting `readonly="false"` won't work in the sense that it won't do the expected thing. It will "work" (have an effect) in the sense that it will make the element `readonly` (in current browsers, even if it's invalid according to the spec). – [ShreevatsaR](#) Oct 31 '16 at 21:07



This is a property setting rather than a valued attribute

55



These *property settings* are values per se and don't need any assignments to them. When they are present, an element has this boolean property set to `true`, when they're absent they're `false`.

```
<input type="text" readonly />
```

It's actually browsers that are liberal toward value assignment to them. If you assign any value to them it will simply get ignored. Browsers will only see the presence of a particular property and ignore the value you're trying to assign to them.

This is of course good, because some frameworks don't have the ability to add such properties without providing their value along with them. Asp.net MVC Html helpers are one of them. jQuery used to be the same until version 1.6 where they added the concept of properties.

There are of course some implications that are related to XHTML as well, because attributes in XML need values in order to be well formed. But that's a different story. Hence browsers have to ignore value assignments.

Anyway. **Never mind the value you're assigning to them** as long as the name is correctly spelled so it will be detected by browsers. But for readability and maintainability it's better to assign meaningful values to them like:

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).

as opposed to

```
readonly="johndoe"
readonly="01/01/2000"
```

that may confuse future developers maintaining your code and may interfere with future specification that may define more strict rules to such property settings.

edited Aug 22 '16 at 15:09



Stonecrusher

102 1 2 13

answered May 30 '11 at 6:38



Robert Koritnik

79.5k 43 247 369

10 Only readonly and readonly="readonly" are valid in HTML. – Tim Down May 30 '11 at 10:01

11 -1 because the *"arguably best choice to use"* is invalid in both HTML 4.01 and HTML5. – user247702 Aug 28 '14 at 12:36 ✎

1 @Stijn: rephrased that part... So you can remove the downvote if you wish to. – Robert Koritnik Aug 28 '14 at 14:05

1 @RobertKoritnik thank you, done. – user247702 Aug 28 '14 at 14:09

readonly="true" is **invalid HTML5**, readonly="readonly" is valid.

30

HTML5 spec:

<http://www.w3.org/TR/html5/forms.html#attr-input-readonly> :

The readonly attribute is a boolean attribute

<http://www.w3.org/TR/html5/infrastructure.html#boolean-attributes> :

The presence of a boolean attribute on an element represents the true value, and the absence of the attribute represents the false value.

If the attribute is present, its value must either be the empty string or a value that is an ASCII case-insensitive match for the attribute's canonical name, with no leading or trailing whitespace.

By using our site, you acknowledge that you have read and understand our Cookie Policy, Privacy Policy, and our Terms of Service.

The following are **valid, equivalent and true**:

```
<input type="text" readonly />
<input type="text" readonly="" />
<input type="text" readonly="readonly" />
<input type="text" readonly="ReAdOnLy" />
```

The following are **invalid**:

```
<input type="text" readonly="0" />
<input type="text" readonly="1" />
<input type="text" readonly="false" />
<input type="text" readonly="true" />
```

The absence of the attribute is the only valid syntax for **false**:

```
<input type="text"/>
```

Recommendation

If you care about writing valid XHTML, use `readonly="readonly"`, since `<input readonly>` is invalid and other alternatives are less readable. Else, just use `<input readonly>` as it is shorter.

edited Aug 14 '14 at 7:21

answered Jul 5 '14 at 16:32



Ciro Santilli 新疆改造中心996ICU六四事件

169k 36 653 521

2 +1, but note that the same counts for HTML 4.01 according to [the spec](#). – user247702 Aug 28 '14 at 12:40



19

`readonly="readonly"` is **xhtml** syntax. In xhtml boolean attributes are written this way. In xhtml 'attribute minimization' (`<input type="checkbox" checked>`) isn't allowed, so this is the valid way to include boolean attributes in xhtml. See [this page](#) for more information.

If your [document type](#) is xhtml transitional or strict and you want to validate it, use `readonly="readonly"` otherwise `readonly` is sufficient.

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).



1 ... so this is the **well formed** way to include boolean attributes in XHTML. – Robert Koritnik May 30 '11 at 6:47

@Robert: yep, that the terminology. – Kooilnc May 30 '11 at 6:50

2

I'm not sure how they're functionally different. My current batch of OS X browsers don't show any difference.

I would assume they are all functionally the same due to legacy HTML attribute handling. Back in the day, any flag (Boolean) attribute need only be present, sans value, eg

```
<input readonly>
<option selected>
```

When XHTML came along, this syntax wasn't valid and values were required. Whilst the W3 specified using the attribute name as the value, I'm guessing most browser vendors decided to simply check for attribute existence.

answered May 30 '11 at 6:29



Phil
105k 13 157 179

This has nothing to do with XML being valid, but rather **well formed**. Validity is related to correct naming and valuing of attributes, their nesting etc. Validity has to do with **schema**. – Robert Koritnik May 30 '11 at 6:44

@Robert I'm referring to XML **syntax** though. Valid XML attribute syntax is `attribute_name="value"` – Phil May 30 '11 at 6:54

@Phil: No that's a **well formed** attribute. A valid attribute is a `int_attribute="1"` or `bool_attribute="false"` ... They're of course well formed as well. – Robert Koritnik May 30 '11 at 6:58

@Robert I'm not talking about validating documents. Perhaps I should use the word "legal" in place of "valid" or any other word that defines the accepted or correct use of characters (ie, syntax) – Phil May 30 '11 at 7:06

@Phil. Well formed XML document is one that passes XML formatting validation (ie. single root element, closing element tags **and** attributes have values (any) assigned to them). But valid XML document is one that is valid against some schema (ie. certain attributes must have correct value type assigned to them, or certain elements can only be nested in particular elements etc.) In your case you're talking about well formed XML attributes (so they have some value assigned to them). – Robert Koritnik May 30 '11 at 7:13

By using our site, you acknowledge that you have read and understand our Cookie Policy, Privacy Policy, and our Terms of Service.



According to HTML standards, the use of

1

```
<input name="TextBox1" type="text" id="TextBox1" readonly/>
```



is enough to make the input element readonly. But, XHTML standard says that the usage listed above is invalid due to attribute minimization . You can refer to the links below:

<https://www.w3.org/TR/xhtml1/diffs.html#h-4.5>

http://www.w3schools.com/tags/att_input_readonly.asp

edited May 11 '16 at 4:58

answered May 30 '11 at 6:38



[James Jithin](#)

7,633

3

24

48

readonly="readonly" Important if loading via XHTML or similar XML assumptions. – [Sql Surfer](#) Oct 31 '17 at 14:08

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).