Python: CSS Selector to use inside lxml.cssselect

I am trying to parse the given below html code using `lxml.html` and using `CSSSelector` instead of `XPath`.

```
link = doc.cssselect('html body div.results dl dt a)
```

the above code is giving me `content-1` and `content-2` as output but my desired output is `link 1 link 2`. So I replaced my code with

```
link = doc.cssselect('html body div.results dl dt a[href]')
```

but still am getting the same output. So my question is what's the proper CSS selector to get href attribute.

```
            <div class = "results">
                    <div> some tags here </div>
                       <dl>
                             <dt title = "My Title 1" style = "background: transparent
url('/img/accept.png') no-repeat right center">
                             <a href = "/link 1"> content-1</a>
                             </dt>
                      </dl>

                     <dl>
                            <dt title = "My Title 2" style = "background: transparent
url('/img/accept.png') no-repeat right center">
                            <a href = "/link 2">content-2</a>
                            </dt>
                    </dl>
            </div>
```

python css css-selectors lxml

asked Dec 28 '11 at 13:45

RanRag
11.5k • 9 • 49 • 91

3 Answers

I *believe* you cannot get the attribute value through CSS selectors. You should get the elements...

```
>>> elements = doc.cssselect('div.results dl dt a')
```

...and then get the attributes from them:

```
>>> for element in elements:
...     print element.get('href')
...
/link 1
/link 2
```

Of course, list comprehensions are your friends:

```
>>> [element.get('href') for element in elements]
['/link 1', '/link 2']
```

Since you cannot update properties of attributes in CSS, I believe there is no sense on getting them through CSS selectors. You can "mention" attributes in CSS selectors to retrieve only to match their elements. ~~However, is is just cogitation and I may be wrong; if I am, please someone correct me :)~~ Well, @Tim Diggs confirms my hypothesis below :)

edited Dec 28 '11 at 14:05

answered Dec 28 '11 at 13:53

brandizzi
12k • 2 • 46 • 82

I just figured it out myself. Anyways thanks for the answer – RanRag Dec 28 '11 at 13:56

@brandizzi, you're right - you can only select elements in css, not attributes -- the brackets are for filtering which elements to select (but not a bad idea to select only <a> tags without href attributes (which is what a[href] does). – Tim Diggins Dec 28 '11 at 13:59

@RanRag, you should tick brandizzi's answer as correct even if you didn't need it in the end. – Tim Diggins Dec 28 '11 at 13:59

i was going to tick it but you can only accept an answer after a certain period of time(i believe its arnd 5 mins) – RanRag Dec 28 '11 at 14:02

You need to get the attribute on the result of cssselect (it always returns the element, never an attribute):

firstly, I'm not sure about doc.cssselect (but maybe this is your own function?)

lxml.cssselect is normally used:

```
from lxml.cssselect import CSSSelector
sel = CSSSelector('html body div.results dl dt a[href]')
```

then, assuming you've already got a doc

```
links = []
for a_href in sel(doc):
    links.append(a_href.get('href'))
```

or the more succinct:

```
links = [a_href.get('href') for a_href in doc.cssselect('html body div.results dl dt
a[href]')]
```

answered Dec 28 '11 at 13:55

Tim Diggins
1,908 ● 1 ● 14 ● 28

1 basically doc is equivalent to `doc=lxml.html.fromstring(content)` where content is my html data from `urllib and read` functions – RanRag Dec 28 '11 at 13:59

I have successfully used

```
#element-id ::attr(value)
```

To get the "value" attribute for HTML elements.

answered Jan 20 '14 at 10:55

Drarok
842 ● 9 ● 28