

## Selenium之CSS Selector定位详解

By: 授客 QQ: 1033553122 QQ群: 7156436

### CSS selector定位

CSS(Cascading Style Sheets)是一种语言, 它被用来描述 HTML 和 XML 文档的样式。

百度输入框:

```
<input name="wd" id="kw1" style="width: 489px;" type="text" maxLength="100"
autocomplete="off" jQuery110203956990375154798="25"/>
```

“百度一下” 按钮

```
<input class="btn self-btn" id="su1" onmouseout="this.className='btn'"
onmousedown="this.className='btn btn_h'" type="submit" value="百度一下"/>
```

#### ● Type selector

语法: E

语义: E元素

示例:

```
# coding= utf-8
from selenium import webdriver
import time

if __name__ == "__main__":
    driver = webdriver.Ie()
    driver.implicitly_wait(5)
    driver.maximize_window()

    driver.get("http://www.baidu.com/")

    #百度输入框 Type selector 定位(E)
    search_ipt = driver.find_element_by_css_selector('input')
    search_ipt.send_keys(u'授客')

    driver.find_element_by_id('su1').click()

    time.sleep(3)
    driver.quit()
```

注意: 改成火狐驱动运行上述代码, 运行不通过

#### ● child Combinator selector

语法: E>F

语义: E元素的一个子元素F

```
driver.find_element_by_css_selector("body>form>a")
```

## ● ID selector

语法: E#myid

语义: ID值等同myid的元素E

示例:

```
# coding= utf-8
from selenium import webdriver
import time

if __name__ == "__main__":
    driver = webdriver.Ie()
    driver.implicitly_wait(5)
    driver.maximize_window()
    driver.get("http://www.baidu.com/")

    #百度输入框 ID selector 定位(E#element_id)
    search_ipt = driver.find_element_by_css_selector('input#kw1')
    search_ipt.send_keys(u'授客')

    driver.find_element_by_id('su1').click()

    time.sleep(3)
    driver.quit()
```

说明:似乎不加E元素也行driver.find\_element\_by\_css\_selector("#kw1")

## ● Attribute selector

语法: E[attribute]

语义: 带有一个attribute属性的元素E

示例:

```
# coding= utf-8
from selenium import webdriver
import time

if __name__ == "__main__":
    driver = webdriver.Ie()
    driver.implicitly_wait(5)
    driver.maximize_window()
    driver.get("http://www.baidu.com/")

    #百度输入框 Attribute selector定位(E[attribute])
    search_ipt = driver.find_element_by_css_selector('input[name]')
    search_ipt.send_keys(u'授客')

    driver.find_element_by_id('su1').click()
```

```
time.sleep(3)
driver.quit()
```

语法: `E[attribute="value1"]`

语义: 带有一个`attribute`属性, 且属性值恰好等于`value1`的元素`E`

示例:

```
# coding= utf-8
from selenium import webdriver
import time

if __name__ == "__main__":
    driver = webdriver.Ie()
    driver.implicitly_wait(5)
    driver.maximize_window()
    driver.get("http://www.baidu.com/")

    #百度输入框 Attribute selector定位(E[attribute])
    search_ipt =driver.find_element_by_css_selector('input[name="wd"]')
    search_ipt.send_keys(u'授客')

    driver.find_element_by_id('su1').click()

    time.sleep(3)
    driver.quit()
```

语法: `E[attribute~="value1"]`

语义: 带有一个`attribute`属性, 且属性值是有由多个空格隔开, 其中一个值恰好等于`value1`的元素`E`

`driver.find_element_by_css_selector("input[class~='btn']")`

示例:

```
# coding= utf-8
from selenium import webdriver
import time

if __name__ == "__main__":
    driver = webdriver.Ie()
    driver.implicitly_wait(5)
    driver.maximize_window()
    driver.get("http://www.baidu.com/")

    search_ipt =driver.driver.find_element_by_id('kw1')
    search_ipt.send_keys(u'授客')
```

```
# “百度一下”按钮Attribute selector定位(E[attribute~="value1"])
driver.find_element_by_css_selector('input[class~="btn"]').click()

time.sleep(3)
driver.quit()
```

语法: E[attribute^="value1"]

语义: 带有一个attribute属性, 且属性值以字符串“value1”开头的元素E

示例:

```
# coding= utf-8
from selenium import webdriver
import time

if __name__ == "__main__":
    driver = webdriver.Ie()
    driver.implicitly_wait(5)
    driver.maximize_window()
    driver.get("http://www.baidu.com/")

search_ip = driver.find_element_by_id('kw1')
search_ip.send_keys(u'授客')

# “百度一下”按钮Attribute selector定位(E[attribute^="value1"])
driver.find_element_by_css_selector('input[class^="btn"]').click()

time.sleep(3)
driver.quit()
```

语法: E[attribute\$="value1"]

语义: 带有一个attribute属性, 且属性值恰好以字符串“value1”结尾的元素E

示例:

```
# coding= utf-8
from selenium import webdriver
import time

if __name__ == "__main__":
    driver = webdriver.Ie()
    driver.implicitly_wait(5)
    driver.maximize_window()
    driver.get("http://www.baidu.com/")

search_ip = driver.find_element_by_id('kw1')
```

```
search_ipt.send_keys(u'授客')

# “百度一下”按钮Attribute selector定位(E[attribute$="value1"])
driver.find_element_by_css_selector('input[class$="btn"]').click()

time.sleep(3)
driver.quit()
```

语法: E[attribute\*="value1"]

语义: 带有一个attribute属性, 且属性值包含字符串“value1”的元素E

示例:

```
# coding= utf-8
from selenium import webdriver
import time

if __name__ == "__main__":
    driver = webdriver.Ie()
    driver.implicitly_wait(5)
    driver.maximize_window()
    driver.get("http://www.baidu.com/")

search_ipt = driver.find_element_by_id('kw1')
search_ipt.send_keys(u'授客')

# “百度一下”按钮Attribute selector定位(E[attribute*="value1"])
driver.find_element_by_css_selector('input[class*="btn"]').click()

time.sleep(3)
driver.quit()
```

有道搜索按钮:

```
<input class="s-btn" id="qb" type="submit" value="搜索"/>
```

语法: E[attribute|="value1"]

语义: 匹配任何属性值以“-”作为分隔符, 而且被分割的第一个部分精确等于“value1”的元素 (也匹配属性只有属性值value1的元素)

示例:

```
# coding= utf-8
from selenium import webdriver
import time

if __name__ == "__main__":
    driver = webdriver.Ie()
```

```
driver.implicitly_wait(5)
driver.maximize_window()

driver.get("http://www.youdao.com/")

#有道搜索输入框Attribute selector定位(E[attribute]="value1"])
search_ipt =
driver.find_element_by_css_selector('input[class="s"]')
search_ipt.send_keys(u'授客')

driver.find_element_by_id('qb').click()

time.sleep(3)
driver.quit()
```

### ● Structural pseudo-classes(结构伪类) selector

有道输入框及“搜索”按钮

```
<html>
  <head></head>
  <body>
    <div id="w">
      <div id="n"></div>
      <div class="sp1"></div>
      <div id="x" class="w">
        <div id="logo"></div>
        <div id="fm">
          <form class="c-fm-w" action="http://www.youdao.com/search" method="get" onsubmit="return doSearchTime();">
            <span class="s-inpt-w">
              <input id="query" class="s-inpt" type="text" name="q" autocomplete="off">
            </span>
            <span class="s-btn-w">
              <input id="qb" class="s-btn" type="submit" value="搜索">
            </span>
            <input type="hidden" value="utf8" name="ue">
            <input type="hidden" value="web.index" name="keyfrom">
          </form>
        </div>
      </div>
      <div class="sp2"></div>
      <div id="sb" class="w"></div>
      <div class="sp3"></div>
      <div id="f" class="w"></div>
    </div>
    <div id="snv" class="clear" style="display: none;"></div>
    <script src="http://shared.ydstatic.com/images/myth/js/index_min.js?201311221" type="text/javascript"></script>
    <script type="text/javascript"></script>
  </body>
</html>
```

语法: E:nth-child(n)

语义: 父元素的第n个子元素: E

注意: 所有主流浏览器均支持类似:nth-child() 选择器, 但IE8及更低版本都不支持, 其它主流浏览器, 比如火狐支持

示例:

```
#coding= utf-8
from selenium import webdriver
```

```
import time

if __name__ == "__main__":
    driver = webdriver.Firefox()
    driver.implicitly_wait(5)
    driver.maximize_window()

    driver.get('http://www.youdao.com/')

    div_em = driver.find_element_by_id('x') #定位父级元素

    #有道输入框structural pseudo-classes selector定位(E:nth-child(n))
    search_ipt
    =div_em.find_element_by_css_selector('div:nth-child(2)>form>span>input')
    search_ipt.send_keys(u'授客')

    driver.find_element_by_id('qb').click()

    time.sleep(3)
    driver.quit()
```

语法: E:nth-last-child(n)

语义: 父元素的倒数第n个子元素: E

注意: 实际写法类似这样: n=-n+倒数, n=-2n+倒数

示例:

```
#coding= utf-8
from selenium import webdriver
import time

if __name__ == "__main__":
    driver = webdriver.Firefox()
    driver.implicitly_wait(5)
    driver.maximize_window()

    driver.get('http://www.youdao.com/')

    div_em = driver.find_element_by_id('x') #定位父级元素

    #有道输入框structural pseudo-classes selector定位
    (E:nth-last-child(n))
    search_ipt
    =div_em.find_element_by_css_selector('div:nth-last-child(n-5)>form>span>input')
```

```
search_ipt.send_keys(u'授客')

driver.find_element_by_id('qb').click()

time.sleep(3)
driver.quit()
```

语法: **E:nth-of-type(n)**

语义: 父元素的子元素中, 与元素E同类型的第n个元素

示例:

```
#coding= utf-8
from selenium import webdriver
import time

if __name__ == "__main__":
    driver = webdriver.Firefox()
    driver.implicitly_wait(5)
    driver.maximize_window()

    driver.get('http://www.youdao.com/')

    div_em = driver.find_element_by_id('x') #定位父级元素

    #有道输入框structural pseudo-classes selector定位
(E:nth-of-type(n))
    search_ipt
    =div_em.find_element_by_css_selector('div:nth-of-type(2)>form>
span>input')
    search_ipt.send_keys(u'授客')

    driver.find_element_by_id('qb').click()

    time.sleep(3)
    driver.quit()
```

语法: **E:nth-last-of-type(n)**

语义: 父元素的子元素中, 与元素E同类型的第n个元素, 从倒数算起

示例:

```
#coding= utf-8
from selenium import webdriver
import time

if __name__ == "__main__":
    driver = webdriver.Firefox()
```



```
driver.implicitly_wait(5)
driver.maximize_window()

driver.get('http://www.youdao.com/')

div_em = driver.find_element_by_id('x') #定位父级元素

#有道输入框structural pseudo-classes selector定位
(E:nth-last-of-type(n))
search_ipt =div_em.
div_em.find_element_by_css_selector('div:nth-last-of-type(n-1)
>form>span>input')
search_ipt.send_keys(u'授客')

driver.find_element_by_id('qb').click()

time.sleep(3)
driver.quit()
```

语法: E:first-child

语义: 父元素的第一个子元素: E

示例:

```
#coding= utf-8
from selenium import webdriver
import time

if __name__ == "__main__":
    driver = webdriver.Firefox()
    driver.implicitly_wait(5)
    driver.maximize_window()

    driver.get('http://www.youdao.com/')

    div_em = driver.find_element_by_id('fm') #定位父级元素

    #有道输入框structural pseudo-classes selector定位(E:first-child)
    search_ipt =div_em.
    div_em.find_element_by_css_selector('div:first-child>form>span
    >input')
    search_ipt.send_keys(u'授客')

    driver.find_element_by_id('qb').click()

    time.sleep(3)
```

```
driver.quit()
```

语法: **E:last-child**

语义: 父元素的最后一个子元素: **E**

示例:

```
#coding= utf-8
from selenium import webdriver
import time

if __name__ == "__main__":
    driver = webdriver.Firefox()
    driver.implicitly_wait(5)
    driver.maximize_window()

    driver.get('http://www.youdao.com/')

    div_em = driver.find_element_by_id('x') #定位父级元素

    #有道输入框structural pseudo-classes selector定位(E:last-child)
    search_ipt =div_em.
    div_em.find_element_by_css_selector('div:last-child>form>span>
    input')
    search_ipt.send_keys(u'授客')

    driver.find_element_by_id('qb').click()

    time.sleep(3)
    driver.quit()
```

语法: **E:first-of-type**

语义: 父元素的父元素的子元素中, 第一个类型与**E**元素相同的子元素

示例:

```
#coding= utf-8
from selenium import webdriver
import time

if __name__ == "__main__":
    driver = webdriver.Firefox()
    driver.implicitly_wait(5)
    driver.maximize_window()

    driver.get('http://www.youdao.com/')

    div_em = driver.find_element_by_id('fm') #定位父级元素
```

```
#有道输入框structural pseudo-classes selector定位(E:first-of-type)
search_ipt =div_em.
div_em.find_element_by_css_selector('form>span:first-of-type>input')
search_ipt.send_keys(u'授客')

driver.find_element_by_id('qb').click()

time.sleep(3)
driver.quit()
```

语法: **E:last-of-type**

语义: 父元素的父元素的子元素中, 最后一个类型与E元素相同的子元素

示例:

```
#coding= utf-8
from selenium import webdriver
import time

if __name__ == "__main__":
    driver = webdriver.Firefox()
    driver.implicitly_wait(5)
    driver.maximize_window()

    driver.get('http://www.youdao.com/')

    div_em = driver.find_element_by_id('x')#定位父级元素

    #有道输入框structural pseudo-classes selector定位(E:last-of-type)
    search_ipt =div_em.
    div_em.find_element_by_css_selector('div:last-of-type>form>span>input')
    search_ipt.send_keys(u'授客')

    driver.find_element_by_id('qb').click()

    time.sleep(3)
    driver.quit()
```

语法: **E:only-child**

语义: 父元素唯一的子元素

示例:

```
#coding= utf-8
from selenium import webdriver
```

```
import time

if __name__ == "__main__":
    driver = webdriver.Firefox()
    driver.implicitly_wait(5)
    driver.maximize_window()

    driver.get('http://www.youdao.com/')

    div_em = driver.find_element_by_id('fm')#定位父级元素

    #有道输入框structural pseudo-classes selector定位(E:only-child)
    search_ipt =div_em.
    div_em.find_element_by_css_selector(':only-child>span>input')
    search_ipt.send_keys(u'授客')

    driver.find_element_by_id('qb').click()

    time.sleep(3)
    driver.quit()
```

注意:

```
div_em.find_element_by_css_selector('form:only-child>span>input')
```

这样也行

语法: **E:only-of-type**

语义: 父元素子元素中, 类型与给定元素E相同的唯一子元素

示例:

```
#coding= utf-8
from selenium import webdriver
import time

if __name__ == "__main__":
    driver = webdriver.Firefox()
    driver.implicitly_wait(5)
    driver.maximize_window()

    driver.get('http://www.youdao.com/')

    div_em = driver.find_element_by_id('fm')#定位父级元素

    #有道输入框structural pseudo-classes selector定位(E:only-of-type)
    search_ipt =div_em.
    div_em.find_element_by_css_selector('form:only-of-type>span>in
```

```
put')
search_ipt.send_keys(u'授客')

driver.find_element_by_id('qb').click()

time.sleep(3)
driver.quit()
```

语法: E:empty

语义: 没有子元素的元素E

示例:

```
#coding= utf-8
from selenium import webdriver
import time

if __name__ == "__main__":
    driver = webdriver.Firefox()
    driver.implicitly_wait(5)
    driver.maximize_window()

    driver.get('http://www.youdao.com/')

    div_em = driver.find_element_by_id('fm')#定位父级元素

    #有道输入框structural pseudo-classes selector定位(E:only-of-type)
    search_ipt =div_em.
    div_em.find_element_by_css_selector('form>span>input:empty')
    search_ipt.send_keys(u'授客')

    driver.find_element_by_id('qb').click()

    time.sleep(3)
    driver.quit()
```

百度页面连接

[新闻](#) [hao123](#) [地图](#) [视频](#) [贴吧](#) [登录](#) [设置](#)

 [百度一下](#)

```
<div id="ul" style="display: block;">
  <a class="mnav" name="tj_trnews" href="http://news.baidu.com"></a>
  <a class="mnav" name="tj_trhao123" href="http://www.hao123.com"></a>
  <a class="mnav" name="tj_trmap" href="http://map.baidu.com"></a>
  <a class="mnav" name="tj_trvideo" href="http://v.baidu.com"></a>
  <a class="mnav" name="tj_trtieba" href="http://tieba.baidu.com"></a>
  <a id="lb" onclick="return false;" name="tj_login" href="https://passport.baidu.com/v/
  <a id="pf" name="tj_settingicon" href="//www.baidu.com/gaoji/preferences.html"></a>
```

#### ● link pseudo-classes selector

语法: E:link

语义: 还未被访问的超链接元素E

示例:

```
#coding= utf-8
from selenium import webdriver
import time

if __name__ == "__main__":
    driver = webdriver.Firefox()
    driver.implicitly_wait(5)
    driver.maximize_window()

    driver.get('http://www.baidu.com/')

    div_em = driver.find_element_by_id('ul') #定位父级元素

    #百度页面, 新闻连接link pseudo-classes selector定位(E:link)
    news_link = div_em.find_element_by_css_selector('a:link')
    news_link.click()

    time.sleep(3)
    driver.quit()
```

#### ● link pseudo-classes selector

语法: E:visited

语义: 已经访问的超链接元素E

#### ● the user action pseudo-classes selector

语法: E:hover

语义: 选择鼠标指针浮动在其上的元素

示例:

```
#coding= utf-8
from selenium import webdriver
import time

if __name__ == "__main__":
    driver = webdriver.Firefox()
```

```
driver.implicitly_wait(5)
driver.maximize_window()

driver.get('http://www.baidu.com/')

div_em = driver.find_element_by_id('u1') #定位父级元素

#百度页面link pseudo-classes selector定位(E:hao123)
hao123_link = div_em.find_element_by_css_selector('a: hover')
hao123_link.click()

time.sleep(3)
driver.quit()
```

说明: : 打开页面后把鼠标光标移动到对应的连接上方, 自动点击对应元素

#### ● the user action pseudo-classes selector

语法: E:focus

语义: 选择鼠标光标所在输入框元素

示例:

#百度输入框定位user action pseudo-classes selector定位(E:focus)

示例:

```
#coding= utf-8
from selenium import webdriver
import time

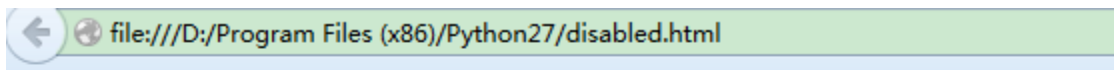
if __name__ == "__main__":
    driver = webdriver.Firefox()
    driver.implicitly_wait(5)
    driver.maximize_window()

    driver.get('http://www.baidu.com/')

    div_em = driver.find_element_by_id('u1') #定位父级元素

    #百度页面link pseudo-classes selector定位(E:focus)
    search_ipt = div_em.find_element_by_css_selector('a: focus')
    #说明: 打开页面后把鼠标点击搜索框, 会定位到该输入框
    search_ipt.send_keys(u'授客')

    driver.find_element_by_id('su1').click()
    time.sleep(3)
    driver.quit()
```



用户:

密码:

请注意, 当您在密码域中键入字符时, 浏览器将使用项目符号来代替这些字符。



#### ● UI element states pseudo-classes selector

语法: E:enabled

语义: 选择状态为enabled的元素

语法: E:disabled

语义: 选择状态为disabled的元素(通常是灰色, 点击无反应的元素)

示例:

```
#coding= utf-8
from selenium import webdriver
import time
import os

if __name__ == '__main__':
    driver = webdriver.Firefox()
    file_path = os.path.abspath('disabled.html')
    driver.get(file_path)
    driver.implicitly_wait(5)

#账户输入框定位UI elementstates pseudo-classes selector定位(E:disabled)
```



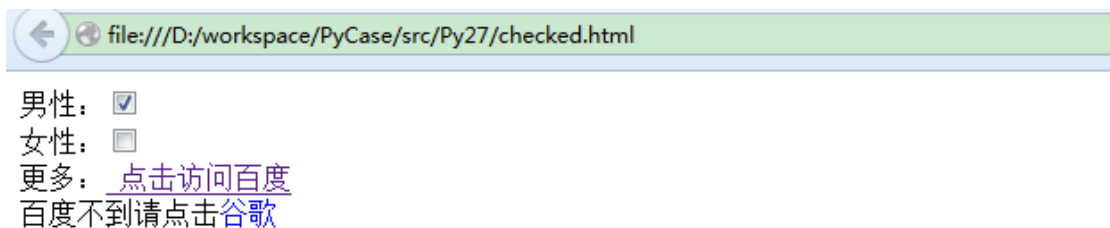
```
driver.find_element_by_css_selector('body>form>input:disabled').click()

#密码输入框定位UI element states pseudo-classes selector定位(E:enabled)
driver.find_element_by_css_selector('body>form>input:enabled').send_keys("test")

time.sleep(3)
driver.quit()
```

注意:

- 1.记得把disabled.html放到src目录下
- 2.一次仅定位一个输入框,也就是说前面的已经定位到了某个输入框,那么后面的输入框不再被定位



当用户点击一个单选按钮时,该按钮会变为选中状态,其他所有按钮会变为非选中状态。



语法: E:checked

语义: 选择状态为checked的元素(通常是灰色,点击无反应的元素)

示例:

```
# coding= utf-8
from selenium import webdriver
import time
import os
```

```
if __name__ == '__main__':
    driver = webdriver.Firefox()
    driver.maximize_window()
    file_path = os.path.abspath('checked.html')
    driver.get(file_path)

    time.sleep(2)

    #复选框定位element stats pseudo-classes selector定位(E:checked)
    driver.find_element_by_css_selector('body>form>input:checked')
    .click()

time.sleep(3)
driver.quit()
```

注意: 记得把checked.html放到src目录下

#### ● Negation pseudo-selector

语法: E:not(s)

语义: 选择与选择器(selector): s,不匹配的元素E

示例:

```
# coding= utf-8
from selenium import webdriver
import time
import os

if __name__ == '__main__':
    driver = webdriver.Firefox()
    driver.maximize_window()
    file_path = os.path.abspath('checked.html')
    driver.get(file_path)

    time.sleep(2)

    #百度超链接定位Negation pseudo-class selector定位(E:not(s))
    driver.find_element_by_css_selector('body>form>a:not(input)').
    click()

time.sleep(3)
driver.quit()
```

注意: 记得把checked.html放到src目录下

```
driver.find_element_by_css_selector('body>form>a:not(input)')
```

- Descendant combinator selector

语法: E F

语义: E元素的一个派生子元素F(子元素, 及子元素的子元素……, 即父元素>派生子元素>派生子子元素……)

示例:

```
# coding= utf-8
from selenium import webdriver
import time
import os

if __name__ == '__main__':
    driver = webdriver.Firefox()
    file_path = os.path.abspath('checked.html')
    driver.get(file_path)

    driver.maximize_window()
    time.sleep(3)

    #百度超链接定位Descendant combinator定位(E F)
    driver.find_element_by_css_selector('body>form a').click()

    time.sleep(3)
    driver.quit()
```

注意:

2. 记得把checked.html放到src目录下

2. 和E>F的区别, E>F 如果F是E元素下的第一个类型与F相同的子元素则不必指定序号, 否则要用类似nth-child(n)之类的指定, 而E F不管是否指定序号, 都可以,

- 其它极少部分选择器及更多关于选择器信息, 参照下文链接

扩展阅读: <http://www.w3.org/TR/css3-selectors/#changesFromCSS2>