**Selenium**

1. 在selenium中定位网页元素有几种方式？
2. ID
3. NAME
4. CLASS\_NAME
5. TAG\_NAME
6. XPATH
7. CSS\_SELECTOR
8. LINK\_TEXT
9. PARTIAL\_LINKTEXT
10. 使用find\_elements选择的是符合条件的所有元素，如果没有符合条件的元素，返回空列表。

使用find\_element选择的是符合条件的第一个元素，如果没有符合条件的元素，抛出异常(NoSuchElement)

1. UI自动化三种等待方式
2. 强制等待：sleep(2)
3. 隐式等待：wd.implicitly\_wait(10)

周期性（每隔半秒钟）重新寻找元素，直到该元素找到，或者超出等待时长抛出异常。（有时需要等元素全部加载出来再继续，所以这种需要等待时还需要用sleep）

1. 显式等待：

WebDriverWait(driver,timeout,poll\_frequency=0.5,ignored\_exception=None)

1. 获取元素的文本内容：element.text
2. 获取元素属性：element.get\_attribute(‘class’)
3. 获取输入框里的文字：element.get\_attribute(‘value’)

**CSS\_SELECTOR**

1. wd.find\_element\_by\_css\_selector(#id)/(.classname)/(tagname)
2. 元素2是元素1的直接子元素，最终选择元素2

-》元素1>元素2

例：#container>div

1. 元素2是元素1的后代元素

-》元素1 元素2

-》元素1 元素2 元素3 元素4（最终选择元素4）

1. 根据属性选择元素，用[ ]

例：element=wd.find\_elements\_by\_css\_selector(‘[href=”https://......”]’)

1. 可以在Chrome中F12，ctrl+F搜索CSS表达式验证对不对
2. 满足其中一种条件用逗号隔开

例：.plant, container

1. 指定元素是父元素的第几个子节点

nth-child

例：span:nth-child(2) ->选择的是第二个子元素并且是span类型的子节点

如果不加节点类型限制，直接写：

:nth-child(2)

父元素的倒数第几个子节点：

例：p:nth-last-child(2)

1. 指定选择的元素是父元素的第几个某类型的子节点

例：span:nth-of-type(2) ->选择的是span类型的第二个子元素

例：span:nth-last\_of-type(2) ->span类型的倒数第二个元素

1. 奇数节点和偶数节点

例：p:nth-child(even) ->偶数节点 #t1 :nth-child(even)

P:nth-child(odd) ->奇数节点 #t2 p:nth-child(odd)

1. 相邻兄弟节点

例：h3+span

后续所有兄弟节点选择

例：h3~span

1. 切换到frame中

switch\_to.frame()

返回默认的：

switch\_to.default\_content()

1. 切换窗口

for handle in wd.window\_handles:

wd.switch\_to.window(handle)

if ‘’ in wd.title:

break

保存当前窗口变量：

mainWindow=wd.current\_window\_handle

wd.switch\_to.window(mainWindow)

1. radio单选框

* 先打印当前选中的“老师名字”
* 再选择新的

<div id="s\_radio">

<input type="radio" name="teacher" value="小江老师">小江老师<br>

<input type="radio" name="teacher" value="小雷老师">小雷老师<br>

<input type="radio" name="teacher" value="小凯老师" checked="checked">小凯老师

</div>

element=wd.find\_element\_by\_css\_selector(‘#s\_radio input[checked=”checked”]’)

print(‘当前选中的是：’+element.get\_attribute(‘value’))

wd.find\_element\_by\_css\_selector(‘#s\_radio input[value=”小雷老师”]’).click()

1. checkbox复选框

* 先把已经选中的选项全部点击一下，确保都是未选状态
* 再点击小雷老师

<div id="s\_checkbox">

<input type="checkbox" name="teacher" value="小江老师">小江老师<br>

<input type="checkbox" name="teacher" value="小雷老师">小雷老师<br>

<input type="checkbox" name="teacher" value="小凯老师" checked="checked">小凯老师

</div>

elements=wd.find\_elements\_by\_css\_selector(‘#s\_checkbox input[checked=”checked”]’)

for element in elements:

element.click()

wd.find\_element\_by\_css\_selector(‘#s\_checkbox input[value=”小雷老师”]’).click()

1. select单选框

<option value="foo">Bar</option>

首先创建select对象，再选择

from selenium.webdriver.support.ui import Select

select=Select(wd.find\_element\_by\_id(“ss\_single”))

* select.select\_by\_value('foo’)
* select.select\_by\_index(0)
* select.select\_by\_visible\_text(‘Bar’)
* deselect\_by\_value
* deselect\_by\_index
* deselect\_by\_visible\_text
* deselect\_all

1. select多选框

先创建对象，再清除所有选择，再选择

1. 在浏览器F12中，点击console，输入SetTimeout(function(){debugger},5000)
2. Alert弹出框

print(wd.switch\_to.alert.text)

wd.switch\_to.alert.accept()

Confirm确认框

wd.switch\_to.alert.accept() ->OK

wd.switch\_to.alert.dissmiss() ->Cancel

1. 获取窗口大小

wd.get\_window\_size()

改变窗口大小

wd.set\_window\_size(x,y)

获取当前窗口URL地址

wd.current\_url()

截屏

wd.get\_screenshot\_as\_file(‘1.png’)

1. 选择属性值

包含某个字符串的元素: a[href\*=“miitbeian”]

以某个字符串开头的元素: a[href^=“http”]

以某个字符串结尾的元素: a[href$=“gov.cn”]

**XPath**

1. XPath (XML Path Language)
   1. /html/body/div
   2. 如果xpath路径以正斜杠(/)开始，就表示从根节点开始 ->绝对路径
2. //div 相对路径 找到所有div
   1. //div//p 只要在div内部的p就找出来
   2. //div/p div的直接子节点p
   3. //div/\* 通配符
3. 选择属性值[@属性名=‘属性值’]
   1. 选择 style属性值 包含 color 字符串的 页面元素：//\*[contains(@style,’color’)]
   2. 选择 style属性值 以color字符串开头：//\*[starts-with(@style,'color')]
4. 选择p类型的第2个子元素

//p[2]

1. 选择父节点 /..

//\*[@id=’china’]/../../..

1. 选择某个节点的后续兄弟节点：following-sibling::\*

following-sibling::select

选择某个节点的前面兄弟节点：preceding-sibling::\*

1. 如果从一个元素内部找元素，需要先确定元素创建webelement对象，再从这个对象中找-》(‘.//p’) 前边要加一个.，否则还是从webdriver中找