frame嵌套页面元素定位

案例：在Frame.html文件种定位搜狗搜索页面，进行搜索操作。

**from** selenium **import** webdriver

**from** time **import** sleep

driver=webdriver.Firefox()

*#设置网页文件路径，r代表路径转义*

file\_path=r'E:\Python\_script\Webdriver\Frame.html'

*#路径转义另一种写法*

*# file\_path='E:\\Python\_script\\Webdriver\\Frame.html'*

driver.get(file\_path)

*#切换到frame页面内*

driver.switch\_to.frame("search")

*#定位到搜索框按钮输入关键词*

driver.find\_element\_by\_css\_selector("#query").send\_keys("Python")

sleep(3)

driver.find\_element\_by\_css\_selector("#stb").click()

sleep(3)

driver.quit()

多窗口切换操作

**案例：打开我要自学网Selenium课程主页，然后打开2-1课程详情页面，再回到课程主页打开3-1课程详情页面**

**from** selenium import webdriver

**from** time import sleep

driver=webdriver.Firefox()

**#打开Selenium课程页面**

driver.get("http://www.51zxw.net/list.aspx?cid=615")

**# 获取课程主页的窗口句柄**

selenium\_index=driver.current\_window\_handle

sleep(2)

**#点击2-1课程链接。进入课程详情页面**

driver.find\_element\_by\_partial\_link\_text('2-1').click()

sleep(4)

**#跳转到课程主页窗口，点击3-1课程**

driver.switch\_to.window(selenium\_index)

sleep(3)

driver.find\_element\_by\_partial\_link\_text('3-1').click()

sleep(3)

driver.quit()

警告弹窗处理

#### 案例：点击百度首页设置按钮，然后进入搜索设置页面，点击“保存设置”或“恢复默认”按钮，处理警告弹窗窗口

from selenium import webdriver

from **time** import **sleep**

driver=webdriver.Firefox()

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

driver.find\_element\_by\_link\_text('设置').click()

**sleep**(2)

driver.find\_element\_by\_link\_text('搜索设置').click()

**sleep**(3)

driver.find\_element\_by\_link\_text('保存设置').click()

*# driver.find\_element\_by\_link\_text('恢复默认').click()*

**sleep**(3)

*#处理警告窗口*

alert=driver.switch\_to\_alert()

alert.**accept**()

**sleep**(2)

**sleep**(2)

driver.quit()

上传文件

案例：在百度搜索上传本地图片进行搜索。

**from** selenium **import** webdriver

**from** time **import** sleep

driver=webdriver.Firefox()

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

driver.find\_element\_by\_css\_selector(".soutu-btn").click()

sleep(3)

driver.find\_element\_by\_css\_selector(".upload-pic").send\_keys(r"E:\Python\_script\Webdriver\shuiyin.png")

sleep(3)

*# driver.quit()*

滚动条控制操作

案例：打开我要自学网页面，然后将滚动条拖到最底部，然后再拖到顶部

from selenium import webdriver

from time import sleep

driver=webdriver.**Firefox**()

driver.**get**("http://www.51zxw.net/")

**sleep**(2)

#将滚动调拖到最底部

js="var action=document.documentElement.scrollTop=10000"

driver.**execute\_script**(js)

**sleep**(2)

#将滚动条拖到最顶部

js="var action=document.documentElement.scrollTop=0"

driver.**execute\_script**(js)

**sleep**(3)

driver.**quit**()

网页截图操作

案例：分别打开我要自学网页面和百度页面，然后进行截图

**from** selenium **import** webdriver

**from** time **import** sleep

*#加载浏览器驱动*

driver=webdriver.Firefox()

*#打开自学网页面并截图*

driver.get("http://www.51zxw.net")

driver.get\_screenshot\_as\_file(r"E:\Python\_script\51zxw.jpg")

*#打开百度页面并截图*

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

driver.get\_screenshot\_as\_file(r"E:\Python\_script\baidu.png")

sleep(2)

driver.quit()

Cookie处理

### 什么是Cookie

Cookie是储存在用户本地终端上的数据，实际上是一小段的文本信息。

### Cookie作用

帮助 Web 站点保存有关访问者的信息，方便用户的访问。如记住用户名密码实现自动登录。

案例：查看访问我要自学网时的Cookie内容

from selenium import webdriver

driver=webdriver.**Firefox**()

driver.**get**("http://www.51zxw.net/")

#获取cookie全部内容

cookie=driver.**get\_cookies**()

#打印全部cookile信息

**print**(cookie)

#打印cookie第一组信息

**print**(cookie[0])

#添加cookie

driver.**add\_cookie**({'name':'51zxw','value':'www.51zxw.net'})

**for** cookie **in** driver.**get\_cookies**():

**print**("%s --- %s" %(cookie['name'],cookie['value']))

driver.**quit**()

自动化测试验证码问题

### 验证码作用

不少网站在用户登录、用户提交信息等登录和输入的页面上使用了验证码技术。验证码技术可以有效防止恶意用户对网站的滥用，使得网站可以有效避免用户信息失窃、保证网站稳定安全性。

但是验证码给自动化测试带来一些不便，使脚本无法正常运行覆盖功能模块。

### 如何解决

#### 1.去掉验证码

这是最简单的方法，对于开发人员来说，只是把验证码的相关代码注释掉即可，如果是在测试环境，这样做可省去了测试人员不少麻烦，如果自动化脚本是要在正式环境跑，这样就给系统带来了一定的风险。

#### 2.设置万能码

去掉验证码的主要是安全问题，为了应对在线系统的安全性威胁，可以在修改程序时不取消验证码，而是程序中留一个“后门”---设置一个“万能验证码”，只要用户输入这个“万能验证码”，程序就认为验证通过，否则按照原先的验证方式进行验证。

#### 3.验证码识别技术（OCR）

例如可以通过Python-tesseract 来识别图片验证码，Python-tesseract是光学字符识别Tesseract OCR引擎的Python封装类。能够读取任何常规的图片文件(JPG, GIF ,PNG , TIFF等)。不过，目前市面上的验证码形式繁多，目前任何一种验证码识别技术，识别率都不是100% 。

#### 4.记录cookie

通过向浏览器中添加cookie 可以绕过登录的验证码。

### 基于Cookie绕过验证码自动登录

案例：使用Cookie绕过百度验证码自动登录账户。

from selenium import webdriver

from time import sleep

driver=webdriver.**Firefox**()

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

#手动添加cookie

driver.**add\_cookie**({'name':'BAIDUID','value':'9E4BF1D44014…(根据实际获取值填写)'})

driver.**add\_cookie**({'name':'BDUSS','value':'根据实际抓包获取值填写'})

**sleep**(2)

driver.**refresh**()

**sleep**(3)

driver.**quit**()

参考文档：

http://www.cnblogs.com/fnng/