搭建web自动化环境

1. 安装python解释器
2. 安装pycharm
3. 安装selenium
4. 下载谷歌
5. 下载谷歌驱动

web自动化浏览器操作

新建py文件

1. 引入selenium

打开浏览器

1. driver=Webdriver.chrome()

元素定位

上节课我们知道了怎么用代码操作浏览器，

这节课我们来学习一下怎么定位元素。

为什么要学习元素定位呢？

我们想要往输入框输入或者往按钮进行点击

需要先找到这个元素。才能进行操作。

所以我们要知道怎么定位元素，怎么找元素。

找元素有八种方式。

我们来看下怎么进行元素定位，先来用id进行元素定位

案例：我们在百度上面实现百度自动搜索的功能

怎么做呢？

找元素

# 元素定位有八种方式:  
# # id name class\_name tagName link和partial\_link xpath css

# id定位：是通过id属性定位，前提页面中id属性，没有就不能用，id在页面中一般是唯一的  
  
# 要求：  
# 1.使用id找到百度输入框, 输入秋水  
# 2.使用id找到百度搜索按钮，点击搜索

# search\_input=driver.find\_element\_by\_id('kw')  
# search\_input.send\_keys('秋水')  
# driver.find\_element\_by\_id('kw').send\_keys('秋水')  
# driver.find\_element\_by\_id('su').click()

这节课我们要学等待，

为什么要学等待呢？

首先第一个是这个等待是面试经常要问的问题，自动化有哪几种等待

第二个，就是你认为你定位到了元素，但是缺报错了了。那位什么报错呢

就要考虑到是不是因为

from selenium.webdriver.support import expected\_conditions as EC

on\_input=WebDriverWait(driver, 10).until(EC.visibility\_of\_element\_located((By.NAME, 'wd')))

# xpath定位:靠路径找到元素 在工作中应用的比较多  
# 绝对路径的写法  
# 相对路径的写法  
  
# 绝对路径的写法 ：从最外层到你要定位的元素详细过程  
# driver.find\_element\_by\_xpath('/html/body/div[1]/div[2]/div[5]/div[1]/div/form/span[1]/input').send\_keys('秋水')  
  
# 相对路径的写法 ：匹配任意层级的元素 以//开始 代表整个文档 //\*[@属性=值]  
# driver.find\_element\_by\_xpath('//input[@name="wd"]').send\_keys('秋水')  
  
# css表达式定位输入框 id定位 #id id、class、标签 css比较灵活还能用层级 属性定位  
# driver.find\_element\_by\_css\_selector('#kw').send\_keys('秋水')  
# driver.find\_element\_by\_css\_selector('.s\_ipt').send\_keys('秋水')

# 通过找后代元素 元素 后代元素  
#

# css表达式定位输入框 id定位 #id id、class、标签  
# css3中空格和>的区别  
# A>B 表示选择A元素的所有子B元素。  
#  
# 与A B的区别在于，A B选择所有后代元素，而A>B只选择一代。  
#  
# A+B表示HTML中紧随A的B元素(A与B是同级元素)。  
# driver.find\_element\_by\_id('kw')  
# element=driver.find\_element\_by\_css\_selector('#kw')  
# element.send\_keys('秋水')

elemt=driver.find\_element\_by\_css\_selector('#head\_wrapper #kw')  
# elemt.send\_keys('秋水')  
# elment=driver.find\_element\_by\_css\_selector('#head\_wrapper .s\_btn')  
# elment.click()

from selenium.webdriver import ActionChains

driver=webdriver.Chrome()  
# 最大化  
driver.maximize\_window()  
# 2.访问百度网址 浏览器上进行的  
driver.get('https://www.baidu.com/')  
temp\_element =driver.find\_element\_by\_id('s-usersetting-top')  
ActionChains(driver).move\_to\_element(temp\_element).perform()  
driver.find\_element\_by\_link\_text('搜索设置').click()