# Find\_element方法:

**From selenium.webdriver.common.py import By**

**如下几种:**

**Driver.find\_element(By.XPATH,”//div[@id=”u1”]//a[@name=”tj\_login”]”)**

**如下解包使用:**

**Loc = (By.XPATH,”//div[@id=”u1”]//a[@name=”tj\_login”]”)**

**Driver.find\_element(\*Loc).click()**

**Driver.find\_elements(By.XPATH,”//div[@id=”u1”]//a[@name=”tj\_login”]”)**

# webdriver截图方法:

driver.save\_screenshot(登录窗口.png)

# 等待时间方式:

## 强制等待Sleep:

**From time import sleep**

**Sleep(7)**

## 隐性等待:

**每个会话只需要调用一次,他会在每个find…方法自动使用.**

**智能等待:超时时间10秒/20秒/30秒/40秒**

**Driver.implicitly\_wait(10(隐性等待的时间)) # 10代表10秒等待,每0.5秒查询一次.**

## 显性等待:

**显性等待 等待(WebDriverWait类) + 条件(expected\_condition) 有等待和条件来实现**

**引入等待: from selenium.webdriver.support.wait import WebDriverWait**

**传入: driver 会话对象; timeout 最多等待多长时间;**

**Poll\_frequency 每隔几秒查看一次,由于此参数使用默认值参数,不传参数,默认0.5秒查询一次**

**WebDriverWait(driver,超时时间,查看周期=0.5)**

**WebDriverWait(driver,超时时间,查看周期=0.5).until(条件)**

**WebDriverWait(driver,超时时间,查看周期=0.5).not\_until(条件**

**引入条件: from selenium.webdriver.support import expected\_condition as EC**

**代码如下:**

**显性等待:**

**Wait = WebDriverWait(driver,20)**

**元素表达:**

**Loc = (By.XPATH,”//div[@id=”u1”]//a[@name=”tj\_login”]”)**

**条件:**

**Wait.until(EC. Visibility\_of\_element\_located(Loc))**

**)**

**条件最常用:**

**Presence\_of\_element\_located : 元素存在**

**Visibility\_of\_element\_located : 元素可见**

**Element\_to\_be\_clickable : 元素可点击**

# 显性等待条件:

**1、 titleIs( String title) 判断当前页面的title是否精确等于预期**

**2、 titleContains( String title) 判断当前页面的title是否包含预期字符串**

**3、 urlToBe( String url) 判断当前页面的url是否精确等于预期**

**4、 urlContains( String fraction) 判断当前页面的url是否包含预期字符串**

**7、 presenceOfElementsLocated( By locator) 判断元素是否出现，必须所有符合条件的元素都加载出来，才通过。**

**8、 elementIfVisible(WebElement element) 如果给定元素是可见的且具有非零大小，否则为null**

**9、presenceOfAllElementsLocatedBy( By locator) 判断元素是否出现。**

**10、 visibilityOfElementLocated( By locator) 传入类型为：locator**

**判断某个元素是否可见. 可见代表元素非隐藏，并且元素宽和高都不等于 0**

**11、visibilityOfAllElementsLocatedBy( By locator) 判断某组元素是否可见**

**12、visibilityOfAllElements(final List<WebElement> elements)**

**传入类型为：element**

**判断某个元素是否可见. 可见代表元素非隐藏，并且元素宽和高都不等于 0**

**13、textToBePresentInElement( WebElement element, String text)**

**判断某个元素中的text是否包含了预期的字符串；**

**14、textToBePresentInElement(By locator, String text) 判断某个元素中的 text 是否 包含 了预期的字符串**

**15、textToBePresentInElementLocated(final By locator, final String text)**

**判断某个元素中的 text 是否 包含 了预期的字符串**

**16、textToBePresentInElementValue( WebElement element, String text)**

**判断某个元素中的 value 属性是否包含 了预期的字符串**

**17、textToBePresentInElementValue(final By locator, final String text)**

**判断某个元素中的 value 属性是否包含 了预期的字符串**

**18、frameToBeAvailableAndSwitchToIt(final String frameLocator)**

**断该 frame 是否可以 switch进去，如果可以的话，返回 True 并且 switch 进去，否则返回 False**

**19、frameToBeAvailableAndSwitchToIt(final By locator)**

**断该 frame 是否可以 switch进去，如果可以的话，返回 True 并且 switch 进去，否则返回 False**

**20、invisibilityOfElementLocated(final By locator)**

**某个元素中是否不存在于dom树或不可见；**

**21、invisibilityOfElementWithText(final By locator, final String text)**

**判断带有文本的元素要么不可见，要么文本不存在于元素上**

**22、elementToBeClickable(final By locator)**

**判断某个元素中是否可见并且是enable的，这样的话才叫clickable；**

**23、elementToBeClickable(final WebElement element)**

**判断某个元素中是否可见并且是enable的，这样的话才叫clickable；**

**24、stalenessOf(final WebElement element)**

**判断一个元素是否仍在DOM中，传入WebElement对象，可以判断页面是 否刷新了。**

**26、elementToBeSelected(WebElement element)**

**页面元素处于被选中状态**

**27、elementSelectionStateToBe( WebElement element, boolean selected)**

**判断某个元素的选中状态是否符合预期,传入element**

**28、elementToBeSelected(By locator)**

**判断某个元素是否被选中了,一般用在下拉列表；**

**29、elementSelectionStateToBe(final By locator, final boolean selected)**

**判断某个元素的选中状态是否符合预期,传入locator**

# 页面嵌套iframe:

**元素在iframe当中,html当中,内嵌了另外一个html(iframe)**

**3.1 分辨元素是否在iframe中**

**3.2 在代码当中,从当前的HTML,切换到iframe当中的HTML,然后再找元素**.

**切换方式: driver.switch\_to.frame(iframe标识)**

**Iframe表达式: //iframe[@name=”login\_frame\_qq”] 只是用于再页面找到iframe,确定唯一**

**Iframe的识别: index , name属性 , or webelement**

**Iframe切换如下:**

**Driver.switch\_to\_frame(“login\_frame\_qq”) #name属性**

**Driver.switch\_to\_frame(3) #下标选择iframe,可以通过//iframe查看页面iframe,然后确定我们切换的iframe是第几个传入下标值 注意:下标值 :从0开始**

**Driver.switch\_to\_frame(driver.find\_element(By.XPATH,”//iframe[@name=”login\_frame\_qq”]”)) 此种方法是先通过iframe的元素定位定位到,然后将对象传给frame切换方法.**

**注意: 切换iframe后一定要用sleep(1)等待1秒**

## 退出iframe到主html:

**退出iframe : driver.switch\_to.default\_content()**

**固定用法**

## 退到上一级iframe:

**适用于html🡪iframe 🡪 iframe 🡪iframe**

**多层iframe嵌套**

**Driver.switch\_to.parent\_frame()**

# 窗口切换:

1. **动作: 导致新窗口的出现**
2. **获取: 所有窗口的句柄,id\_list = driver.window\_handles**
3. **切换: driver.switch\_to.window(新窗口句柄)**
4. **获取当前窗口句柄: driver.current\_window\_handle**

# Alert&&prompt&&confirm事件弹窗:

1. 动作:导致alert的出现
2. 切换:
3. al = driver.switch\_to.alert

al.accept() 关闭弹窗

al.text 获取文本