# 单一属性定位:

## 注意: 元素定位中元素的值一定要加英文的冒号.

## 注意:

**由于页面元素的属性值值不唯一,如下:**

**find\_element\_by\_XXX : 返回的是一个元素,匹配到对的第一个 (页面有多个,也只返回匹配到的第一个)**

**find\_elements\_by\_XXX :返回页面当中,匹配表达式的所有元素. 结果是一个列表,列表的成员是WebElement对象 (页面有多少个元素,只要和表达式匹配上,哪怕500个也会给你返回,返回的是个列表)**

## Id定位:

浏览器对象.find\_element\_by\_id(id值) -----🡪id定位,页面元素id值要唯一

## Class定位:

浏览器对象.find\_element\_by\_class\_name(页面元素的class值) ----🡪 class定位,页面元素的class值要唯一

find\_elements\_by\_class\_name(页面元素的class值) -🡪返回匹配到的所有元素的webelement对象

## tag\_name定位:

浏览器对象.find\_element\_by\_tag\_name(标签名) ----🡪以页面的标签名来定位元素,如果页面有多个同名的标签名,则是只返回匹配到的第一个标签名

页面标签名: button-🡪(按钮标签名) h1到h6 -🡪(标题标签名(从大到小))

Input --🡪(输入框标签名)

find\_elements\_by\_tag\_name(页面元素的标签名) -🡪返回匹配到的所有元素的webelement对象

## name定位:

浏览器对象.find\_element\_by\_name(页面元素的name属性值) ----🡪 name定位,页面元素的name值要唯一,不唯一只返回匹配到的第一个.

find\_elements\_by\_name(页面元素的name名) -🡪返回匹配到的所有元素的webelement对象

## link\_text定位:

浏览器对象.find\_element\_by\_link\_text(“页面元素的文本内容”) ---🡪 link\_text定位,页面元素的文本值值要唯一,不唯一只返回匹配到的第一个.

find\_elements\_by\_link\_name(页面元素的文本名) -🡪返回匹配到的所有元素的webelement对象

## partial\_link\_text定位:

partial\_link\_text()—定位的元素文本内容可以输入模糊文本,例如 :页面元素的文本内容为: 王者荣耀 我可以输入 王者 去进行定位, 就例如SQL语句里的link模糊查询

浏览器对象.find\_element\_by\_partial\_link\_text(“绝对文本/模糊文本”) ---🡪 link\_text定位,页面元素的文本值值要唯一,不唯一只返回匹配到的第一个.

find\_elements\_by\_link\_name(页面元素的文本名) -🡪返回匹配到的所有元素的webelement对象

## XPath定位:

绝对定位: / 开头 如下:

爷节点/父节点/子节点

从根节点开始,一直到当前节点, 节点顺序/节点位置

缺点:太死板,不灵活,开发一改东西就废了

## 相对定位: 重点:

相对路径定位: // 开头 //

相对路径: 相对参照物, // 相对于前面节点.如果//在开头,那就是相对于整个html

定位原则:

1. 不依靠节点顺序和节点位置
2. 只要能够找得到匹配特征的元素就可以

定位辅助工具(谷歌浏览器,火狐浏览器,其他浏览器自测反正IE不行(喵的,废物)):

F12 🡪 elments 🡪 Ctrl+F

## 相对定位表达式:

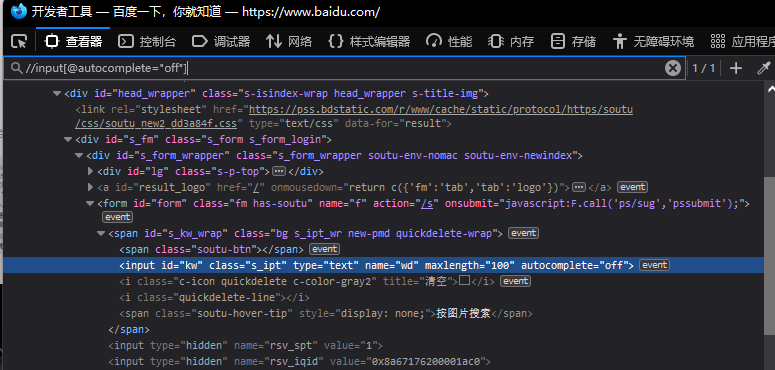
**重点**

1. **//标签名[@属性名=值] 如下:**

**注意:此方法是用元素的属性去定位**

**定位表达式: //input[@autocomplete="off"]**

**通过百度的输入框的此属性autocomplete="off" 来进行相对定位输入框**

****

1. **//标签名[text()=值] 如下:**

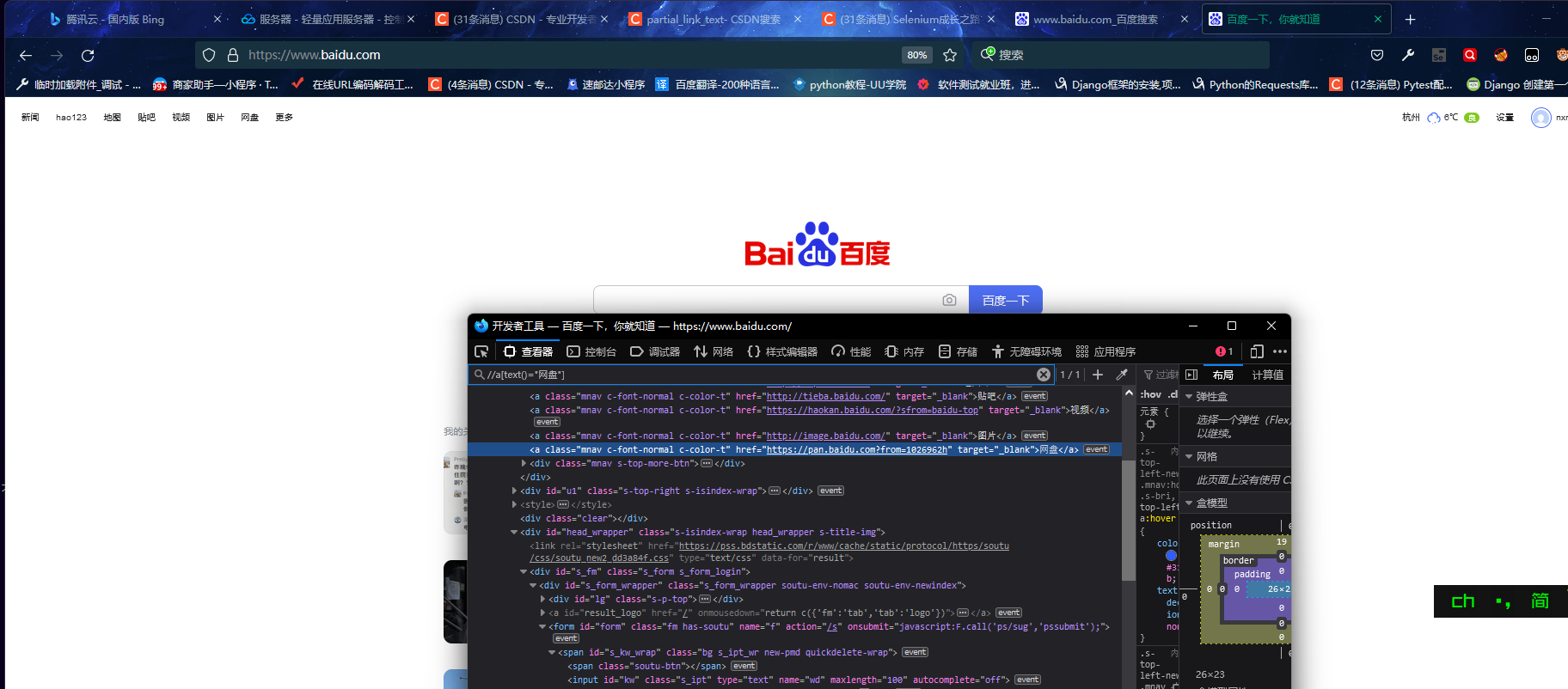
**注意: 此方法用来定位有文本内容的元素**

**<a href="https://pan.baidu.com?from=1026962h" target="\_blank" class="mnav c-font-normal c-color-t">网盘</a>**

**上面的标签为百度主页的网盘按钮**

**定位表达式:** **//a[text()="网盘"]**

**通过上边的定位表达式来定位网盘按钮**

****

1. **//标签名[contains(@属性名,属性值)] 如下:**

**定位表达式: //a[contains(@href, “?from=1026962h”)]**

**定位表达式://a[@href="https://pan.baidu.com?from=1026962h"]**

**两个表达式效果一样,contains (效果类似于模糊查询)**

**这2个表达式都是用来定位百度首页的网盘按钮**

1. **//标签名[contains(text(),值)] 如下:**

**此方法用于查找文本内容,因为加入了text()函数**

**定位表达式: //a[contains(text(),”hao”)]**

**定位表达式:** **//a[text()="hao123"]**

**两个表达式效果一样,contains (效果类似于模糊查询)**

**这2个表达式都是用来定位百度首页的hao123按钮**

1. **//标签名[contains(text(),值)] 如下:**

**所有标签名都可以用\*号表示,只要属性值唯一即可**

**\* 代表所有 如下:**

**定位表达式: //\*[contains(@href, “?from=1026962h”)]**

**定位表达式: //\*[contains(@\*, “?from=1026962h”)]**

**定位表达式: //\*[contains(@\*, “\*”)]🡨 这种写法一般被人看见就说你不会**

## 多重属性定位:

**当单一条件,无法唯一定位到元素.**

**组合条件:**

**逻辑: and or**

**如下:**

示例: <a href="https://pan.baidu.com?from=1026962h" target="\_blank" class="mnav c-font-normal c-color-t">网盘</a>

**//标签名[text()=值 and @属性名=值 and contains(@属性名,值)]**

**下面语句用来定位上面的示例:**

**//a[text()="网盘" and @class="mnav c-font-normal c-color-t" and contains(@href,"1026962h")]**

**and表示 与 上面的定位语句意思是:上面3三个条件必须全部满足**

**//标签名[text()=值 or @属性名=值 or contains(@属性名,值)]**

**下面语句用来定位上面的示例:**

**//a[text()="网盘" or @class="mnav c-font-normal c-color-t" or contains(@href,"1026962h")]**

**or表示 或 上面的定位语句意思是:上面3三个条件满足其一**

## 注意:

如果依靠上面方法都无法定位,(仅靠元素本身的特征,已经无法唯一定位)只能借助外部定位: CSS选择器定位 层级定位 轴定位

# 层级定位:

## 层级定位:

**//祖先节点//要找的节点**

**//div[@id="u"]//a[@name="tj\_login"]**

**如百度首页的登录,如果仅靠name属性定位定位到2个,这个时候可以去找我们要的登录按钮元素的父级或爷爷级或祖爷爷级,先定位到登录的祖先级再去定位登录.**

# 轴定位:

## 轴定位:

元素的兄弟姐妹,父母

**轴运算：**

**ancestor：祖先结点 包括父**

**parent:父结点**

**商家助手打单列表的搜索按钮定位:**

**//span[text()="搜索 "] /parent::button[@type="button"]**

**preceding-sibling: 当前元素节点标签之前的所有兄弟结点**

**//li[text()=”文本”]/preceding-sibling::\*[@属性名=”属性值”]**

**following-sibling：当前元素节点标签之后的所有兄弟结点**

**//p[@title=”03008”]/following-sibling::\*[@属性名=”属性值”]**

**preceding: 当前元素节点标签之前的所有结点。（html页面先后顺序）**

**following: 当前元素节点标签之后的所有结点。（html页面先后顺序）**

**超复杂定位:**

//a[contains(text(),”2020-0810 – pytest笔记!”)]/ancestor::div[@class=”work-new-l”]/ preceding-sibling::div[@class=”work-new-r fr”]//a[@href=”/Review/index/homeworkid/MDAwMDAwMDAwMLOGtZeHubewhLVyoQ.html?type=wj” not @class=”red”]

**Python自动化 74集 1.03.12分钟时的写法,**