**关于元素定位中字符切割的参考字符设置**

1. **场景**

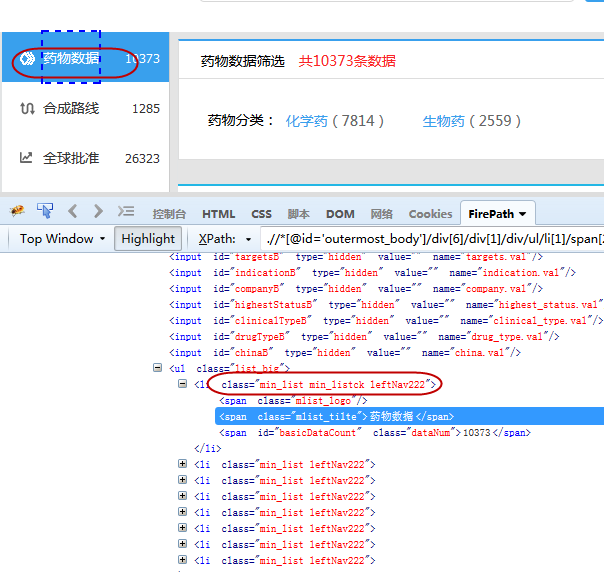
在设计python+selenium自动化测试框架的过程中，关于元素定位表达式切割参考字符修改了三次，从最开始的 = 到 , 到现在的 ==

**由于python中不能直接定义的一个webelement元素**，不像java，可以定义一个空的web element元素，然后 element.click()就可以点击这个元素。Python中不是这样，python中明确需要先指定用八种方法之一去找，所以在实际框架中写页面元素都需要写成这样

Element\_localtor = “xpath,//\*/div[@id=’searchinput’]”

框架中利用方法去把双引号内部切割两部分，part1 = xpath， part2=xpath表达式，这里逗号就是切割参考字符。

最早开始是采用等号（=）来做切割参考字符，但是元素定位大部分采用xpath，而且xpath很多地方需要写@class=’xx’，由于localtor字符串里面有两个或者多个=，造成切割不精确，从而导致元素查找失败。最后发现逗号也不适合做参考字符，原因如下：

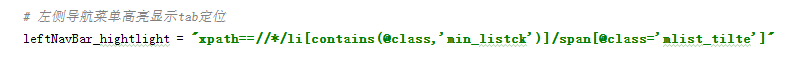


需求： 需要判断当前左侧导航中高亮部分是不是和当前title相符，两个都指向同一个子库。由于八个元素都差不多，元素不好定位，唯一区别就是，当高亮元素被点击，class中有一个关键字 min\_listck，当哪个tab高亮显示，就在哪个tab对应的class中显示min\_listck，唯一遗憾的就是，这个min\_listck出现位置不统一，例如当前药物数据对应class=min\_list min\_listck leftNav222

看看其他子库高亮：



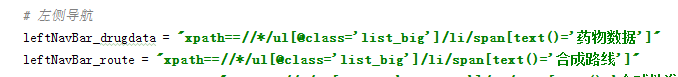
所以class这个value不能写死，只能通过xpath中函数contains来定位这个高亮部分。



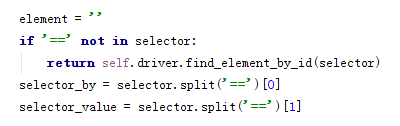
那么问题就来了，这个contains中包含一个,（逗号），字符切割出问题，造成元素定位不准确。

1. **解决办法**

页面元素写localtor方法，xpath和表达式直接用双等号（==）隔开



**在写元素定位中切割元素部分，参考字符就用==，双等号前半部分就是告诉selenium通过哪种方法去定位元素，后半部分就是该方法的表达式。**



所以，你网上搜索到基于python封装selenium方法中元素切割大部分是等号或者逗号隔开。 逗号隔开比等号要好，只有遇到contains的xpath表达式的时候逗号才不好用，建议采用==