XPath的使用

一、初识XPath

1、XPath 概览：

XPath 的全称是 XML Path Language，最初是用来搜寻 XML 文档的，但是如今同样可以应用于 HTML 文档。

XPath 提供了非常简洁明了的路径选择表达式，及诸多内置函数用于字符串、数值、时间的匹配及节点、序列的处理。几乎所有我们想要定位的节点，都可以通过 XPath 来进行选择。

2、XPath 的使用规则：

下表给出了 XPath 的几个常用规则：

表达式	描述
nodename	选取此节点的所有子节点
/	从根节点开始选取，绝对定位；也代表从当前节点选取直接子节点
//	从符号条件的节点开始选取，不必考虑它们的位置，相对定位
.	选取当前节点
..	选取当前节点的父节点
@	选取属性

几个简单的例子：

- 选取所有节点：`result = html.xpath('//*')`
- 选取所有名为 li 的节点：`result = html.xpath('//li')`
- 选取所有 li 节点的所有直接子节点 a：`result = html.xpath('//li/a')`
- 选取所有 ul 节点的子孙节点 a：`result = html.xpath('//ul//a')`
- 获取其父节点的 class 属性：
  - 方法一（使用 `..`）：`result = html.xpath('//a[@href="link4.html"]/../@class')`
  - 方法二（使用 `parent::`）`result = html.xpath('//a[@href="link4.html"]/parent::*/@class')`

二、XPath 的常用规则

1、属性唯一: 通过元素属性，快速定位

```
result = html.xpath('//*[@href="link3.html"]')
```

2、没有属性: 属性与层级的结合

```
result = html.xpath('//li[@class="item-0"]/a')
```

3、多个属性重名: 属性与逻辑结合

```
result = html.xpath('//li[contains(@class, "li") and
@name="item"]/a/text()')
```

4、某一属性具有多个值: contains 方法, (参数名称, 参数值)

```
result = html.xpath('//li[contains(@class, "li")]/a/text()')
```

5、按序选择: 通过索引实现

```
result1 = html.xpath('//li[1]/a/text()')                # 选取第一个 li 节
点的直接子节点 a
result2 = html.xpath('//li[last()]/a/text()')           # 选取最后一个 li
节点(last())
result3 = html.xpath('//li[position()<3]/a/text()')     # 选择了位置小于 3
的节点
result4 = html.xpath('//li[last()-2]/a/text()')         # 选择了倒数第 3 个
节点
print(result1, result2, result3, result4)
```

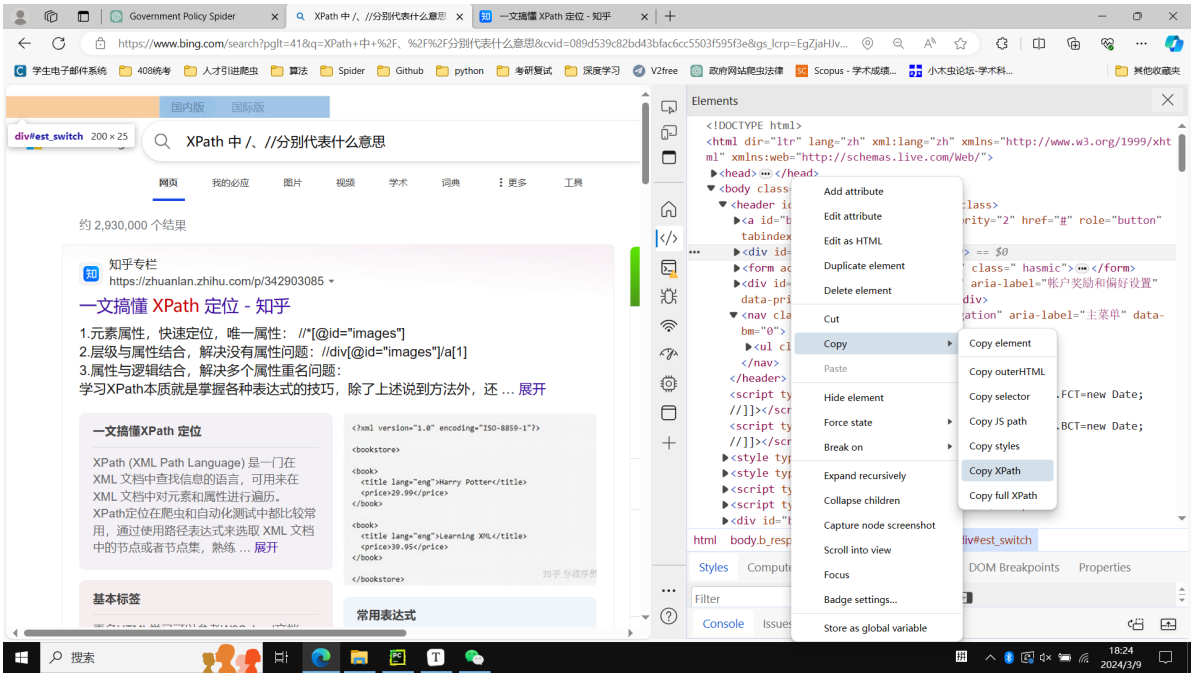
三、XPath 的轴的选取

轴名称	结果
child	当前节点的所有子节点
descendant	当前节点的所有后代节点
descendant-or-self	当前节点**及**所有后代节点
parent	当前节点的父节点
ancestor	当前节点的所有祖先节点
ancestor-or-self	当前节点**及**所有祖先节点
self	当前节点
attribute	当前节点的**所有属性**
following	当前节点的**结束标签之后**的所有结点
namespace	当前节点的所有命名空间节点
preceding	当前节点的**开始标签之前**的所有结点
preceding-sibling	当前节点之前的所有**同级**节点

一般的使用格式为：`轴名称` + `::` + `其他限制条件`

```
result = html.xpath('//li[1]/ancestor::*')                # 第一个 li 节点的所有祖先节点
print(result)
result = html.xpath('//li[1]/ancestor::div')              # 第一个 li 节点的 div 祖先节点
print(result)
result = html.xpath('//li[1]/attribute::*')               # 获取第一个 li 节点的所有属性值
print(result)
result = html.xpath('//li[1]/child::a[@href="link1.html"]')    # 获取第一个 li 节点的 href 属性为 link1.html 的 a 节点
print(result)
result = html.xpath('//li[1]/descendant::span')                # 获取第一个 li 节点的 span 子孙节点
print(result)
result = html.xpath('//li[1]/following::*[2]/@href')           # 获取第一个 li 节点的后续所有节点，有索引限制，故只获取第二个 li 节点
print(result)
result = html.xpath('//li[1]/following-sibling::*/a/text()')   # 获取第一个 li 节点之后的所有同级节点
print(result)
```

四、如何获取 XPath



五、关于 HTML 的基本标签：

标题：`<h1>、<h2>、<h3>、<h4>、<h5>、<h6>、<title>`

段落：<p>

链接：<a>

图像：<img>

样式：<style>
ans
列表：`无序列表<ul>、有序列表<ol>、列表项<li>`

块：`<div>、<span>`

脚本：<script>

注释：<!--注释-->