

我们检测到你可能使用了 Adblock 或 Adblock Plus，它的部分策略可能会影响到正常功能的使用（如关注）。
你可以设定特殊规则或将知乎加入白名单，以便我们更好地提供服务。（为什么？）



[Passage4]Scrapy中提取数据的机制——Selector

 佐啾
暂退乎，需要Python资源的见个人简介。

如何从获取到的网页文本中进一步提取我们需要的数据？如果你喜欢Xpath，那么你可能会使用lxml，如果你熟悉CSS选择器，那么你可能更倾向于BeautifulSoup或是Pyquery，亦或是你独爱强大灵活的正则表达式(re模块)，我所提到的这些都是解析网页文本流行模块。而在Scrapy中，更好的选择是使用Scrapy自带的Selector。

目录

- Selecotr是什么？
- Selector的基本用法
- 两个重要的数据类型——SelectorList和Selector
- 总结

1.Selecotr是什么？

- Scrapy提供了自己提取数据的机制，它们被称作选择器（Selector，下面用英文表示），它们筛选出由CSS或Xpath制定的特定的HTML部分。
- Scrapy的Selector建立在lxml库上，这意味它们在解析速度和精度上都非常相似。

2.Selector的基本用法

Selector是scrapy.selector下的一个类，我们通过构造这个类的实例才能进一步提取数据，我们看一个基本实例：

```
from scrapy.selector import Selector
body = '<body>hello</body>'
selectors = Selector(text=body)
data = selectors.xpath('//body/text()').extract_first()
print(data)
```

hello

提取body标签下的文本，接着通过extract_first函数提取到了文本内容。

如果不想使用xpath，Selector对象还支持了CSS和正则表达式的方式提取信息，还是上面的例子，比如我们要用CSS选择器，我们只需要修改一行代码：

```
data = selectors.css('body::text').extract_first()
```

使用正则表达式也可以得到同样结果，这里就不演示了。

3.两个重要的数据类型——SelectorList和Selector

我们执行两行代码看一下结果：

```
>>>type(selectors.css('body::text'))
>>>type(selectors.css('body::text')[0])

<class 'scrapy.selector.unified.SelectorList'>
<class 'scrapy.selector.unified.Selector'>
```

- 第一行代码：我们已经知道，凡是要提取数据之前必须都先实例化为Selector对象，然后我们使用这个对象提供的CSS选择器提取我们需要的信息。但是注意到我们得到的是一个SelectorList数据类型。
- 第二行代码：第一行代码实际返回的也是一个只包含一个元素的list，因此我们提取这个元素继续判断，发现它的类型也是Selector。

然而SelectorList和Selector并不是我们希望得到的结果，如何提取它们的内容呢，scrapy提供了两个方法：

- **.extract_first()**：提取SelectorList对象中第一个元素的内容。
- **.extract()**：如果是SelectorList对象使用，则返回包含内容的列表；如果是Selector使用，则返回它的内容。

我们看一下实例：

```
>>>from scrapy.selector import Selector
>>>body = '<body>hello</body><body>world</body>'
>>>selectors = Selector(text=body)
>>>
>>>selectors.xpath('//body/text()').extract() # 1
>>>selectors.xpath('//body/text()').extract_first() # 2
>>>selectors.xpath('//body/text())[0].extract() # 3

['hello', 'world']
'hello'
'hello'
```

你可能会好奇，实例中#2和#3的效果是一样的，那么实际中选哪个呢？答案是最好按照#2的写法，这是由于我们索引列表时可能会发生越界导致程序异常，而.extract_first()方法可以避免这点，当提取对象的内容不存在时这个方法会返回None。另外，你也可以指定返回的内容。我们可以看一个实例：

```
>>>selectors.xpath('//a')
[]
>>>selectors.xpath(''
```



4.总结

- Selector是scrapy中提取数据的机制，它支持xpath，css和正则表达式三种规则。
- 使用上述三种规则提取数据之前，我们需要把待解析的对象实例化为一个Selector对象，这个类位于scrapy.selector下。
- Selector机制下有两个重要的数据类型：SelectorList和Selector，其中前者是后者的集合，前者也是一个列表对象。
- scrapy提供了extract和extract_first这两种方法提取SelectorList和Selector的内容。在提取列表中的某个元素内容时，建议使用extract_first方法而非采用索引。

参考资料

[1] 《Python3 网络爬虫开发实战》 崔庆才

[2] [Selectors - Scrapy 1.5.1 documentation](#)

编辑于 2018-12-14

Python 爬虫 (计算机网络) scrapy

文章被以下专栏收录

 我写Python

进入专栏

推荐阅读



Python有哪些常用的标准库？
Python基础教程，常用标准...

Daly 发表于Pytho...



表达式	说明
article	选取所有article元素的所有子节点
/article	选取根元素article
article/a	选取所有属于article的子元素的a元素
//div	选取所有div子元素(不论出现在文档任何地方)
article//div	选取所有属于article元素的后代的div元素，不管它出现在article之下的任何位置

手把手教你学会Xpath高级用法

地球的外星... 发表于Pytho...



Python爬虫基础练习(十四)Selenium爬取淘宝商品

HDMI




python

谁也别

2 条评论

切换为时间排序

写下你的评论...

 菠菜有料

1 年前

知乎



首发于
我写Python

关



佐咖 (作者) 回复 菠菜有料
多谢朋友指正，已修改。

1 年前

👍 赞

