崔庆才丨静觅
2018年05月07日 阅读 288

Scrapy框架的使用之Spider的用法

在Scrapy中，要抓取网站的链接配置、抓取逻辑、解析逻辑里其实都是在Spider中配置的。在前一节实例中，我们发现抓取逻辑也是在Spider中完成的。本节我们就来专门了解一下Spider的基本用法。

1. Spider运行流程

在实现Scrapy爬虫项目时，最核心的类便是 `Spider` 类了，它定义了如何爬取某个网站的流程和解析方式。简单来讲， `Spider` 要做的事就是如下两件：

- 定义爬取网站的动作；
- 分析爬取下来的网页。

对于 `Spider` 类来说，整个爬取循环过程如下所述：

- 以初始的URL初始化Request，并设置回调函数。当该Request成功请求并返回时，Response生成并作为参数传给该回调函数。
- 在回调函数内分析返回的网页内容。返回结果有两种形式。一种是解析到的有效结果返回字典或Item对象，它们可以经过处理后（或直接）保存。另一种是解析得到下一个（如下一页）链接，可以利用此链接构造Request并设置新的回调函数，返回Request等待后续调度。
- 如果返回的是字典或Item对象，我们可通过Feed Exports等组件将返回结果存入到文件。如果设置了Pipeline的话，我们可以使用Pipeline处理（如过滤、修正等）并保存。
- 如果返回的是Reqeust，那么Request执行成功得到Response之后，Response会被传递给Request中定义的回调函数，在回调函数中我们可以再次使用选择器来分析新得到的网页内容，并根据分析的数据生成Item。

通过以上几步循环往复进行，我们完成了站点的爬取。

在上一节的例子中，我们定义的 `Spider` 是继承自 `scrapy.spiders.Spider` 。
`scrapy.spiders.Spider` 这个类是最简单最基本的Spider类，其他Spider必须继承这个类。还有后面一些特殊 `Spider` 类也都是继承自它。

`scrapy.spiders.Spider` 这个类提供了 `start_requests()` 方法的默认实现，读取并请求 `start_urls` 属性，并根据返回的结果调用 `parse()` 方法解析结果。它还有如下一些基础属性：

- `name` 。爬虫名称，是定义Spider名字的字符串。Spider的名字定义了Scrapy如何定位并初始化Spider，它必须是唯一的。不过我们可以生成多个相同的Spider实例，数量没有限制。 `name` 是Spider最重要的属性。如果Spider爬取单个网站，一个常见的做法是以该网站的域名名称来命名Spider。例如，Spider爬取mywebsite.com，该Spider通常会被命名为mywebsite。

- `allowed_domains` 。允许爬取的域名，是可选配置，不在此范围的链接不会被跟进爬取。

- `start_urls` 。它是起始URL列表，当我们没有实现 `start_requests()` 方法时，默认会从这个列表开始抓取。

- `custom_settings` 。它是一个字典，是专属于本Spider的配置，此设置会覆盖项目全局的设置。此设置必须在初始化前被更新，必须定义成类变量。

- `crawler` 。它是由 `from_crawler()` 方法设置的，代表的是本Spider类对应的Crawler对象。Crawler对象包含了很多项目组件，利用它我们可以获取项目的一些配置信息，如最常见的获取项目的设置信息，即Settings。

- `settings` 。它是一个Settings对象，利用它我们可以直接获取项目的全局设置变量。

除了基础属性，Spider还有一些常用的方法：

- `start_requests()` 。此方法用于生成初始请求，它必须返回一个可迭代对象。此方法会默认使用 `start_urls` 里面的URL来构造Request，而且Request是GET请求方式。如果我们想在启动时以POST方式访问某个站点，可以直接重写这个方法，发送POST请求时使用 `FormRequest` 即可。

- `parse()` 。当Response没有指定回调函数时，该方法会默认被调用。它负责处理Response，处理返回结果，并从中提取出想要的数据和下一步的请求，然后返回。该方法需要返回一个包含Request或Item的可迭代对象。

- `closed()` 。当Spider关闭时，该方法会被调用，在这里一般会定义释放资源的一些操作或其他收尾操作。

以上内容可能不太好理解。不过不用担心，后面会有很多使用这些属性和方法的实例。通过这些实例，我们慢慢熟练掌握它们。

本资源首发于崔庆才的个人博客静觅： Python3网络爬虫开发实战教程 | 静觅

如想了解更多爬虫资讯，请关注我的个人微信公众号：进击的Coder

weixin.qq.com/r/5zsjOyvEZ... (二维码自动识别)

关注下面的标签，发现更多相似文章

爬虫 Scrapy 后端 微信

评论

输入评论...

相关推荐

专栏 · 张少林同学 · 6小时前 · 后端
Dubbo 自定义异常，你是怎么处理的?
3

超人汪小建 · 1天前 · 机器学习 / 后端
2018汇总机器学习篇
25