理财产品爬虫参考

目录

[（一） 运行环境搭建 1](#_Toc453144029)

[（二） 创建爬虫项目 1](#_Toc453144030)

[（三） 编辑items.py文件 1](#_Toc453144031)

[（四） 编辑爬虫主体程序(azhaoshang.py) 2](#_Toc453144032)

[（五） 运行爬虫 4](#_Toc453144033)

[ **附录：** 4](#_Toc453144034)

1. 运行环境搭建

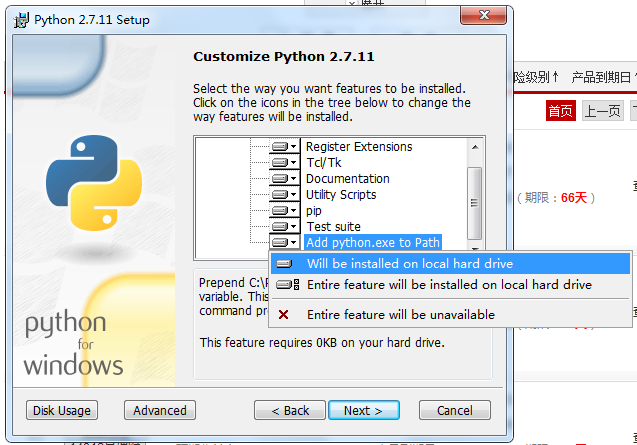
python，Scrapy，Selenium，Phontomjs，Firefox浏览器

1. 安装python2.7

<https://www.python.org/downloads/release/python-2711/>

安装过程注意事项：如下图，

Add python.exe to Path项需要修改为Will be installed on…



1. 升级pip

命令行运行python -m pip install -U pip

（如果没有安装pip，安装指南https://pip.pypa.io/en/stable/installing/）

1. 命令行运行pip install scrapy，
2. 命令行运行pip install selenium，
3. 下载Phontomjs，把phantomjs.exe程序放入python的安装目录(C:\Python27)

下载地址<http://phantomjs.org/download.html>

1. 下载Firefox浏览器 + 安装firebug插件

下载地址：<http://www.firefox.com.cn/download/>

运行环境搭建完成。

1. 创建爬虫项目

以下都以招商银行理财产品的爬虫编写作为例子来说明。

* 运行“创建爬虫.py”程序（运行方法见[附录（一）](#附录（一）)），根据提示输入爬虫项目名称和要爬取网站的域名，例如zhaoshang，cmbchina.com

该程序将在C:\目录下创建一个名为zhaoshang的爬虫项目。

1. 编辑items.py文件

该文件位于C:\zhaoshang\zhaoshang目录。

items文件用来定义需要爬取的字段。

复制下面的代码到items文件；

如果网页上有与这里某个含义对应的字段就需要提取，没有含义对应的字段就用''表示缺失,而这里没有列出的字段都不需要提取；

import scrapy

class ZhaoshangItem(scrapy.Item):

# define the fields for your item here like:

# name = scrapy.Field()

#复制下面的代码

fxyh = scrapy.Field() #发行银行

cpmc = scrapy.Field() #产品名称

gmbz = scrapy.Field() #购买币种

sylx = scrapy.Field() #收益类型

tzfw = scrapy.Field() #投资范围

fxdj = scrapy.Field() #风险等级

tzqx = scrapy.Field() #投资期限

yqsy = scrapy.Field() #预期收益

xsqs = scrapy.Field() #销售起始日期

xsjz = scrapy.Field() #销售截止日期

qsje = scrapy.Field() #起始金额

kfsh = scrapy.Field() #可否赎回

fxdq = scrapy.Field() #发行地区

1. 编辑爬虫主体程序(Qzhaoshang.py)

程序文件位于C:\zhaoshang\zhaoshang\spiders 目录

实际代码文件参见项目示例zhaoshang

# -\*- coding: utf-8 -\*-

import sys

reload(sys)

sys.setdefaultencoding('utf-8')

import scrapy

from selenium import webdriver

import time

#以上代码需要复制

class AzhaoshangSpider(scrapy.Spider):

name = "azhaoshang" #爬虫名称，后面运行爬虫的时候需要输入此名称

allowed\_domains = ["cmbchina.com"] #网站域名

**#以上3行代码不需要复制**

start\_urls = (

http://www.cmbchina.com/cfweb/Personal/Default.aspx',

) #产品列表页的起始网址，需要修改为自己爬取网站的网址

'''

情形一：产品列表页是静态页面（比较少见），也就是火狐浏览器里右键查看网页源文件，能直接看到网页上显示的内容;

情形二：产品列表页是动态页面，火狐浏览器里右键查看网页源文件，里面查看不到网页上显示的内容，网页内容是经过javascript动态生成的;

对于情形一，不需要下面\_\_init\_\_和to\_page\_bottom两个函数，因此不要复制；而情形二需要调用\_\_init\_\_函数，因此要把\_\_init\_\_函数复制过去；

而to\_page\_bottom需要视情况采用。

'''

def \_\_init\_\_(self, \*args, \*\*kwargs):

super(AzhaoshangSpider, self).\_\_init\_\_(\*args, \*\*kwargs) #此处AzhaoshangSpider需要修改为自己的名称

self.browser = webdriver.PhantomJS() # 如果把PhantomJS替换为Firefox,爬虫会启动一个Firefox浏览器，能看到爬虫运行时浏览器的动态变化

self.browser.get(self.start\_urls[0])

#self.browser2.refresh() #刷新网页，个别网站（建行）需要刷新一次才能显示完整内容

self.browser.implicitly\_wait(10) #等待超时时间

time.sleep(5)

self.to\_page\_bottom()

self.prod\_urls = list() #用来保存所有产品详情页的网址

def to\_page\_bottom(self): # 该函数的作用是滚动到网页底部，因为招行网站需要滚动到底部来加载更多产品信息，视情况采用。

for \_ in range(6):

self.browser.execute\_script("window.scrollTo(0, document.body.scrollHeight);")

time.sleep(1)

def parse(self, response):

'''

函数作用：解析动态产品列表页

重点:

（1）self.browser.find\_elements\_by\_xpath('//div/span[@class="inlineBriefName"]/a')，

获得列表页中所有产品详情页的网址(href属性)所在的elements（通常是a元素），需要根据爬取的网站具体情况修改。

其中xpath函数内是网页中需要获取的元素，获取方式见：《附录（一）》

（2）next\_page = self.browser.find\_element\_by\_xpath('//div[@id="pager"]/div/a[contains(text(), "下一页")]')，

定位翻页按钮，需要根据爬取的网站具体情况修改。

'''

self.log('开始运行爬虫')

for a in self.browser.find\_elements\_by\_xpath('//div/span[@class="inlineBriefName"]/a'): #需要修改

prod\_url = a.get\_attribute('href')

self.prod\_urls.append(prod\_url)

#以下翻页代码针对动态产品列表网页，如果是静态网页，需要重写！如果所有产品在一页上，不需要翻页，则不需要复制

self.log('----------------翻页------------------')

next\_page = self.browser.find\_element\_by\_xpath(' /html/body/form/div[4]/div[2]/div[10]/div[5]/div/div/a[5]') #需要修改

next\_page.click()

time.sleep(5)

self.to\_page\_bottom ()

body = self.browser.page\_source

yield scrapy.Request(url=self.browser.current\_url, body=body, callback=self.parse)

for u in self.prod\_urls:

yield scrapy.Request(u,callback=self.prod\_parse)

'''

prod\_parse函数用来解析理财产品详情页

下面有两个prod\_parse函数，其中第一个用来演示详情页是静态网页的情形；

第二个演示详情页是动态网页的情形；

根据实际网页类型复制一个就行。

'''

#（一）静态网页

def prod\_parse(self, response):

'''

解析产品详情页（http://www.cmbchina.com/cfweb/personal/productdetail.aspx?code=1200052）的函数;

ZhaoshangItem已经预先在items.py中定义，其中fxyh,cpmc,... 每个字段都有特定含义，

如果网页上有与这里某个含义对应的字段就需要提取，没有含义对应的字段就用''表示缺失,而这里没有列出的字段都不需要提取；

xpath函数内的内容都需要修改

例如下面的语句：

product['cpmc'] = ''.join(response.xpath('//div[@style="padding: 0px 3px;"]/text()').extract()).strip().encode('mbcs')

其中'cpmc'表示产品名, xpath('//div[@style="padding: 0px 3px;"]/text()')语句可以提取出产品名称,

.extract()).strip().encode('mbcs')不需要修改,encode('mbcs')的作用是处理中文乱码。

'''

sys.path.append('c:\\zhaoshang') #爬虫项目目录，需要修改

from zhaoshang.items import ZhaoshangItem

product = ZhaoshangItem()

product['fxyh'] = '招商银行'.encode('mbcs') #爬取的银行名称，需要修改

product['cpmc'] = ''.join(response.xpath('//div[@style="padding: 0px 3px;"]/text()').extract()).strip().encode('mbcs')

product['gmbz'] = ''.join(response.xpath('//li[9]/span/text()').extract()).strip().encode('mbcs')

product['sylx'] = ''.join(response.xpath('//li[6]/span/text()').extract()).strip().encode('mbcs')

product['tzfw'] = '' #详情页上没有对应字段，赋值为空字符串，表示缺失值

product['fxdj'] = ''.join(response.xpath('//li[7]/span/text()').extract()).strip().encode('mbcs')

product['tzqx'] = ''.join(response.xpath('//div[@id="ctl00\_content\_panel"]//tr[4]//

p[@style="line-height:125%;"]/span[1]/text()').extract()).strip().encode('mbcs')

product['yqsy'] = ''.join(response.xpath('//li[5]/span/text()').extract()).strip().encode('mbcs')

product['xsqs'] = ''.join(response.xpath('//li[2]/span/text()').extract()).strip().encode('mbcs')

product['xsjz'] = ''.join(response.xpath('//li[3]/span/text()').extract()).strip().encode('mbcs')

product['qsje'] = ''

product['kfsh'] = ''

product['fxdq'] = ''

return product

#（二）动态网页

def prod\_parse(self, response):

'''

解析产品详情页（http://finance.ccb.com/cn/finance/ProductDetails.html?PRODUCT\_ID=ZHQYBB20160400042）的函数;

JianhangItem已经预先在items.py中定义，其中fxyh,cpmc,... 每个字段都有特定含义，

如果网页上有与这里某个含义对应的字段就需要提取，没有含义对应的字段就用''表示缺失,而这里没有列出的字段都不需要提取；

xpath函数内的内容都需要修改

例如下面的语句：

product['cpmc'] = self.browser2.find\_element\_by\_xpath('//table[@class="cell"]//td[@id="name2"]').text.encode('mbcs')

其中'cpmc'表示产品名, find\_element\_by\_xpath('//table[@class="cell"]//td[@id="name2"]').text 语句可以提取出产品名称,

.text.strip().encode('mbcs')不需要修改,encode('mbcs')的作用是处理中文乱码。

'''

sys.path.append('c:\\jianhang')

from jianhang.items import JianhangItem

product = JianhangItem()

self.browser2 = webdriver.PhantomJS()

self.browser2.get(response.url)

time.sleep(5)

#self.browser2.refresh() #刷新网页

product['fxyh'] = '建设银行'.encode('mbcs')

product['cpmc'] = self.browser2.find\_element\_by\_xpath('//table[@class="cell"]//

td[@id="name2"]').text.strip().encode('mbcs')

product['gmbz'] = self.browser2.find\_element\_by\_xpath('//\*[@id="currencyType"]').text.strip().encode('mbcs')

product['cplx'] = '' #详情页上没有对应字段，赋值为空字符串，表示缺失值

product['sylx'] = self.browser2.find\_element\_by\_xpath('//\*[@id="yieldSpec"]').text.strip().encode('mbcs')

product['tzfw'] = ''

product['fxdj'] = self.browser2.find\_element\_by\_xpath('//\*[@id="riskLevel"]').text.strip().encode('mbcs')

product['tzqx'] = self.browser2.find\_element\_by\_xpath('//\*[@id="investPeriod2"]').text.strip().encode('mbcs')

product['yqsy'] = self.browser2.find\_element\_by\_xpath('//td[@id="yieldRate2"]').text.strip().encode('mbcs')

product['xsqs'] = self.browser2.find\_element\_by\_xpath('//\*[@id="collBgnDate2"]').text.strip().encode('mbcs')

product['xsjz'] = ''

product['qsje'] = self.browser2.find\_element\_by\_xpath('//\*[@id="purFloorAmt2"]').text.strip().encode('mbcs')

product['kfsh'] = self.browser2.find\_element\_by\_xpath('//\*[@id="proMode"]').text.strip().encode('mbcs')

product['fxdq'] = self.browser2.find\_element\_by\_xpath('//\*[@id="saleCitys"]').text.strip().encode('mbcs')

self.browser2.close()

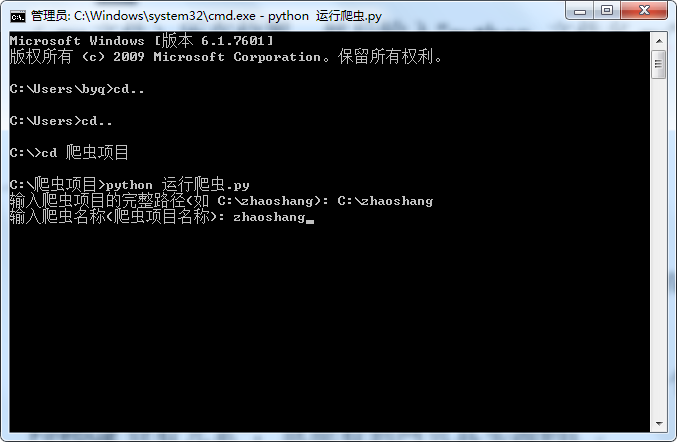
return product

1. 运行爬虫

运行“运行爬虫.py”程序，根据提示输入爬虫项目所在文件夹路径(例如C:\zhaoshang)和爬虫名称(例如zhaoshang)，回车确认后爬虫将开始运行，爬虫的输出结果保存在”爬虫名称.csv”文件（例如azhaoshang.csv）里，csv文件位于爬虫项目所在文件目录(C:\zhaoshang)下。

* **附录：**
* **（一）python程序运行方法（Windows为例）**

按 + R 键，弹出“运行”窗口，输入cmd，打开命令行窗口，切换到python程序（.py文件）所在位置，然后输入”python 文件名.py”，回车即可运行。



* **（二）XPATH知识**

1**）**XPath教程：http://www.w3school.com.cn/xpath/index.asp

2）XPath获取

用Firefox浏览器打开理财产品的网页，在网页某个元素上点击鼠标右键，选择“使用Firebug查看元素”，就能看到网页真实源码：



