Scrapy爬虫中实战经验总结

# 在Python中引入命令行

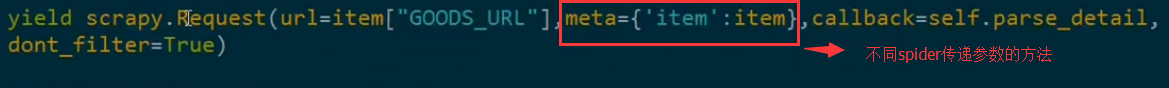
在**工程目录下 BaiduTieba目录**下，创建一个main.py 文件，执行该文件，就相当于在命令行的BaiduTieba下，执行scrapy crawl tieba

from scrapy import cmdline  
cmdline.execute('scrapy list'.split())  
cmdline.execute("scrapy crawl tieba".split())

这是利用**PyCharm的调试功能**也是可以的。

# 同一个spider中，不同parse函数如何传递数据？

将数据如item通过meta参数传递到下一次的请求中，然后利用Response的meta获取该数据。





# 怎么将item数据转化为json？

### 方法：

import json

json.dumps(**dict(item)**,**ensure\_ascii=False**)

### 解释：

json.dumps在默认情况下，对于**非ascii字符**生成的是**相对应的字符编码**，**而非原始字符**，例如：

>>> import json  
>>> **js = json.loads('{"haha": "哈哈"}')**  
>>> print json.dumps(js)  
{"haha": "\u54c8\u54c8"}

解决办法很简单:

>>> print json.dumps(js, **ensure\_ascii=False**)     
{"haha": "哈哈"}

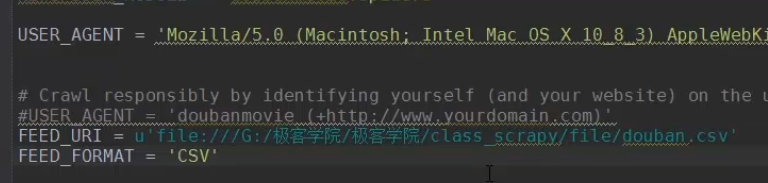
# 怎么将log日志输出到文件中？

在**settings.py**文件中，添加**LOG\_FILE = “E://log/log.log”**

# 在settings.py 中可以直接定义USER-AGENT

还可以设定生成的文件的类型及路径。

**FEED\_URL 、FEED\_FORMAT**



# 错误异常：[解决ValueError('Missing scheme in request url: %s' % self.\_url)](http://blog.csdn.net/lcyong_/article/details/72858453) 利用ImagesPipeline时出现错误。

**原因分析：指定的IMAGES\_URLS\_FIELD的类型必须是一个string的list类型，不能简单的传递url字符串，应该是[url]。如果是单个图片，一个url，就是一个元素的list，当然如果多个url，可以利用append方法添加。**

**如：**

item[**'IMAGES\_URLS'**] = [**"http:"**+li.xpath(**'div[@class="gl-i-wrap"]/div[@class="p-img"]/a[@target="\_blank"]**[**/img/@src'**).extract()[0](mailto:/img/@src').extract()%5b0)]];

item[**'IMAGES\_URLS'**] = “http://www.baidu.com/1.jpg”;//错误

item[**'IMAGES\_URLS'**] = [“<http://www.baidu.com/1.jpg>”];//错误