Requests数据抓取

1. Requests简介与安装

Requests是python的一个HTTP客户端库，几乎可以解决我们遇到的任何爬虫问题，其强大简洁的API足以让人体会到python的优雅。

Windows下安装通过pip install requests

之后可以在CMD下运行python



2.要使用requests，首先推荐阅读requests的官方文档

<https://2.python-requests.org//zh_CN/latest/user/quickstart.html>

HTTP请求中我们通常只会用GET和POST，requests对于区分了两种不同的请求方式。分别是带参数和不带参数，下边给出实例：

#不带参数

<https://www.baidu.com>

#带参数

<https://www.baidu.com/s?wd=python>

判断URL是否带有参数，可以通过对符号？的判断。一般网站URL后边带有？说明带有参数，参数的规则如下

1）？后接参数

2）参数之间用&连接

Requests实现GET请求，对可带参数的URL有两种方式：

两种方法都可以，实际开发中建议第一种，因为更加简洁优雅，体现了python的语法。

POST方法就是我们日常生活中提交表单的方法，比如登录验证之类的。Requests实现的POST需要传递参数data，可以使字典或者json结构，或者元组，列表等。

需要注意的是，GET请求传递的是params，POST传递的是data，不能混淆二者。

此外，服务器返回的信息里包含了许多我们需要的数据，比如

Html.status\_code：相应状态码

Html.raw:原始响应体

Html.content:字节响应体，需要解码

Html.text：字符串的响应方式

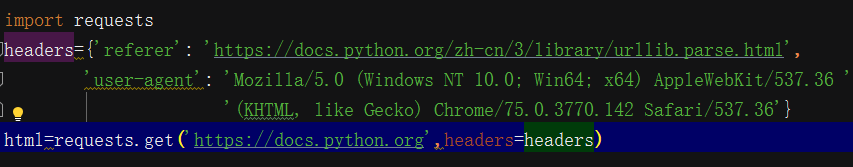
Html.headers：服务器响应头

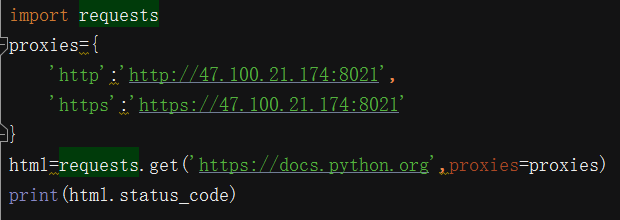
Html.josn()：requests内置的json解码器

Html.cookies：获取请求后的cookies

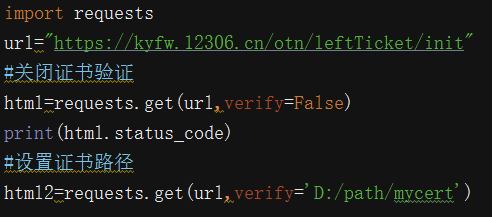
Html.encoding：获取编码格式

3.复杂的请求方式包括请求头，代理，证书验证和cookies验证。Requests对此做了简化，将这些功能封装在了requests参数中。

1. 添加请求头：请求头是字典格式
2. 使用IP代理



1. 证书验证：通常是用来关闭的，默认是True，如果需要设置证书文件，则需要传递证书路径



1. 超时设置，有时候因为各种因素，我们请求一个网站，在得到响应之前，会等待一段时间。如果不想让程序等待那么长时间，可以自己设置一个等待时间，超时的话会引发一个异常。



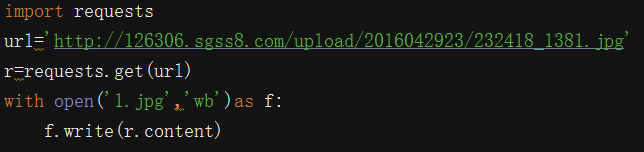
1. 使用cookies，cookies是用来识别用户的，在requests中以字典形式存在。获取方式主要是通过服务器的设置，我们也可以自己伪造。

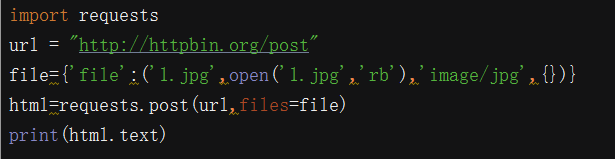
以上是我们自己传递cookies，那么如何获取服务器的cookies呢？第一种办法是打开浏览器的开发者工具，在network里边寻找；第二种方法是通过requests获取



4.上传与下载

下载文件就是从服务器请求文件，然后保存在本地，以下载图片为例，代码如下：

文件下载本身得到的内容是字节流，以字节方式写入才能实现下载。

文件的上传更为复杂一些，是将本地的文件以字节流形式上传到服务器，由服务器接收后做出相应。

文件上传的file是字段名，‘1.jpg’是上传的文件名，之后是本地路径，然后是上传格式，字典里是其他参数。

5.urllib常用参数补充

具体参见<https://docs.python.org/zh-cn/3/library/urllib.parse.html#module-urllib.parse>

Urllib.parse.urlparse：分解URL

Urllib.parse.urljoin：将两个URL结合在一起

Urllib.parse.quote：对字符串转义

Urllib.parse.urlsplit：分割URL