什么是requests模块?

之前我们学习过python内置的网络请求库urllib，使用起来流程比较复杂。**requests模块对urllib做了完美封装，几乎包括了所有的urllib功能，而且使用起来简洁方便，目前被广泛使用。**

requests库常用的方法

序号	方法	描述
1	requests.request(url)	构造一个请求，支持以下各种方法
2	requests.get()	发送Get请求
3	requests.post()	发送Post请求
4	requests.head()	获取HTML的头部信息
5	requests.put()	发送Put请求
6	requests.patch()	提交局部修改的请求
7	requests.delete()	提交删除请求

最常用的方法为get()和post()分别用于发送get请求和post请求。

get请求语法结构：

```
requests.get(url, params=None)
```

参数说明：

```
url:需要爬取的网站的网址
params:请求参数
```

该方法的结果为Response对象，包含服务器的响应信息。

response对象的常用属性：

序号	属性或方法	描述
1	response.status_code	响应状态码
2	response.content	把response对象转换为二进制数据
3	response.text	把response对象转换为字符串数据
4	response.encoding	定义response对象的编码
5	response.cookies	获取请求后的cookie
6	response.url	获取请求网址
7	response.json()	内置的JSON解码器
8	Response.headers	以字典对象存储服务器响应头，字典键不区分大小写

post请求语法结构：

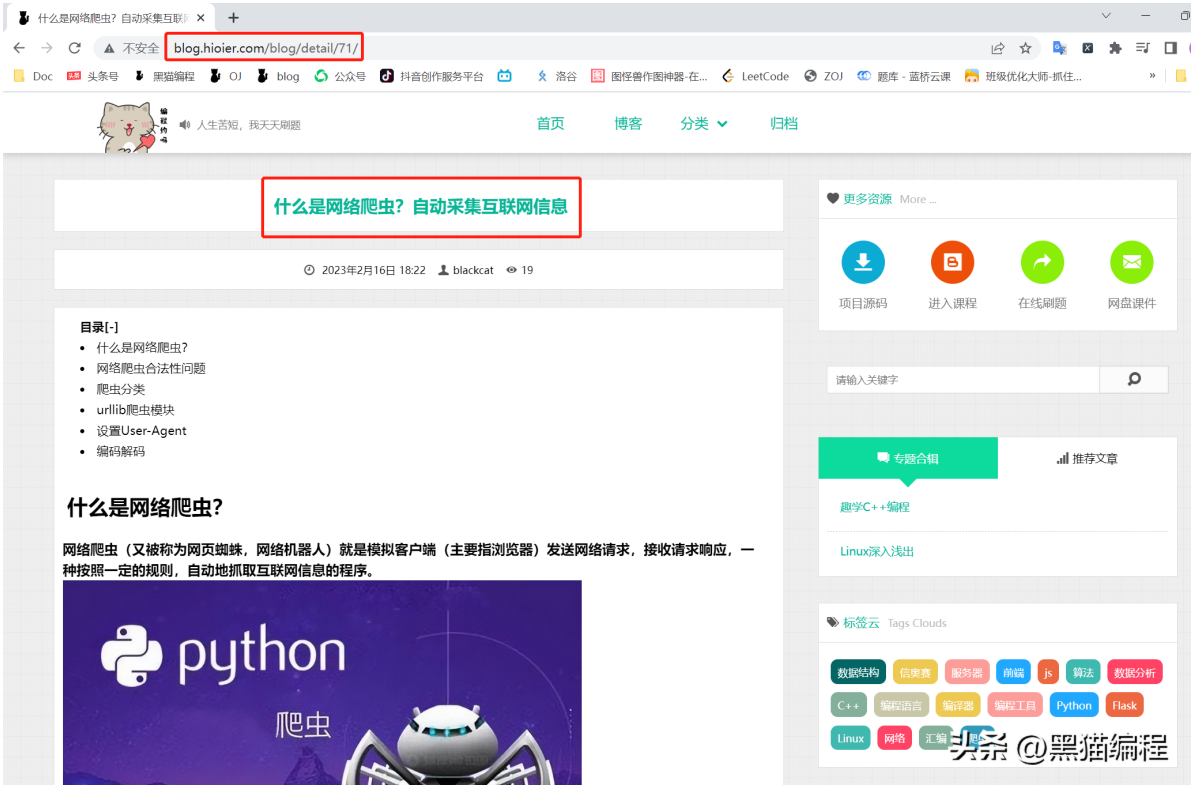
```
requests.post( url, data=None)
```

参数说明：

```
url：需要爬取的网站的网址
data：请求数据
```

正则表达解析数据

项目目标：爬取博客平台所有文章标题。



先分析单独一篇文章，文章页面url是有规律的，目前71号是我当前最后一篇，起始是第1篇。

```python
import requests
import re

url = "http://blog.hioier.com/blog/detail/71/"

response = requests.get(url)

# print(response.text)

title = re.findall("<h3 class=\"arc-title index-title\">(.*)</h3>", response.text)

print(title[0])

# 输出：什么是网络爬虫？自动采集互联网信息
```

进一步地，循环获取所有文章标题：

```python
import requests
import re

titles = []

for i in range(1, 72):

    url = "http://blog.hioier.com/blog/detail/" + str(i);

    response = requests.get(url)

    title = re.findall("<h3 class=\"arc-title index-title\">(.*)</h3>", response.text)
```

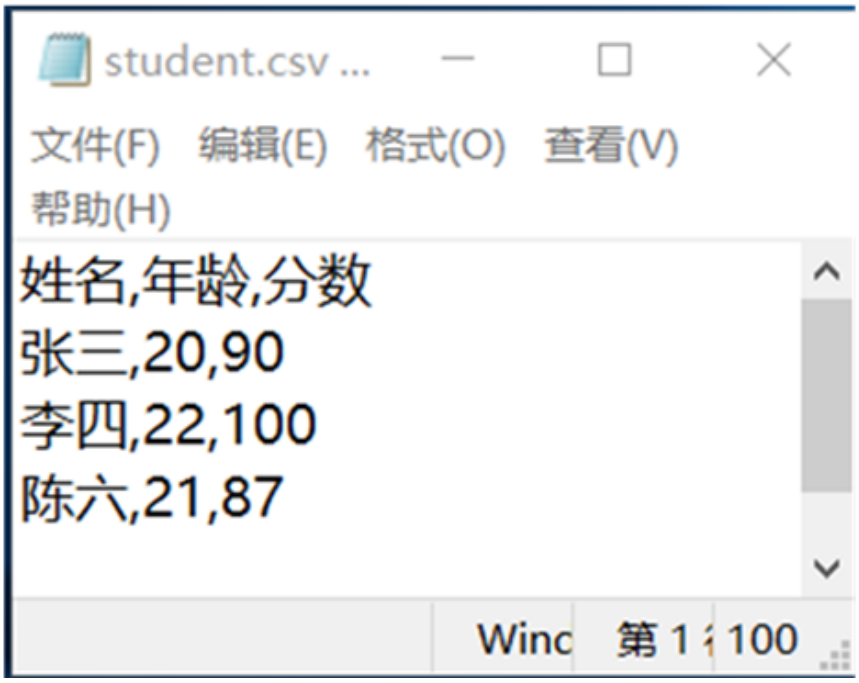
```
13
14      title.insert(0, i)
15      # print(title[0])
16      titles.append(title)
17
18  print(titles)
19  [[1, '栈，先进后出的数据结构 '], [2, 'CCF CSP-J 2022 第二轮认证试题解析 '], [3, 'Nginx简明教程 '], [4, 'Docker轻量级虚拟化，镜像和容器 '], [5, '青岛大学开源OJ在线测评环境搭建 '], [6, 'Nodejs后端运行和javascript刷题 '], [7, '莫比乌斯反演 '], [8, 'Python OpenCV简介和图像灰度处理 '], [9, '什么是C++编程 '], [10, 'C++变量，存储数据的容器 '], [11, 'C++输入输出流 '], [12, 'Dev C++安装配置 '], [13, '计算机基础知识 '], [14, 'C++标准数据类型 '], [15, 'C++设置域宽、保留小数位数和cmath数学库 '], [16, 'C++语法阶段课程总结 2022.11.24 '], [17], [18, 'Linux简介和云服务器配置 '], [19, 'Linux命令行操作 '], [20, 'Vim编辑器之神 '], [21, 'tmux终端复用器 '], [22, 'Linux用户和用户组 '], [23, 'Numpy简介和数据类型 '], [24, 'Numpy数组属性和数组创建 '], [25, 'SSH远程登录 '], [26, 'Linux文件权限设置 '], [27, 'Ubuntu软件安装和卸载 '], [28, 'Opencv图像二值化处理 '], [29, 'C++格式化输入输出 '], [30, 'C++语法系列 '], [31, '趣学C++编程 '], [32, 'Shell脚本和变量 '], [33, 'Numpy切片、索引 广播和迭代 '], [34, 'Opencv图像降噪 '], [35, 'Python Opencv绘制图形和文字 '], [36, 'Python opencv 人脸识别 '], [37, 'Opencv调取摄像头拍照和从多媒体文件读取视频帧 '], [38, 'Shell test命令和条件判断 '], [39, 'Shell循环结构 '], [40, 'Shell函数 '], [41, 'Shell正则表达式 '], [42, 'Shell三剑客之sed '], [43, 'Linux深入浅出 '], [44, 'Shell三剑客之awk '], [45, '计算机网络发展史和网络拓扑结构 '], [46, '什么是计算机网络OSI模型和TCP/IP模型？ '], [47, '计算机网络奈奎斯特定理和香农定理 '], [48, '计算机网络IP地址和子网掩码 '], [49, '计算机网络TCP/UDP协议，三次握手原理 '], [50, '计算机网络应用层体系结构 '], [51, '防火墙是什么墙？ '], [52, '什么是计划任务？让计算机定时执行特定任务 '], [53, 'Linux操作系统进程管理 '], [54, '通过Github，免费搭建自己的博客项目 '], [55, 'Github全球最大的程序员交友网站 '], [56, 'git版本创建与回退 '], [57, 'git分支管理，平行宇宙中的代码合并 '], [58, '一文详解HTML和CSS '], [59, 'Javascript基础入门 '], [60, 'Javascript在线刷题输入输出模板 '], [61, '音乐项目-人脸识别登录 '], [62, 'JavaScript点击按钮控制图片切换 '], [63, 'Docker搭建仓库和数据卷管理 '], [64, 'Javascript获取className属性和slice切片 '], [65, 'Javascript定时器 '], [66, 'DOSBox配置8086CPU汇编语言开发环境 '], [67, 'Javascript对象和选项卡 '], [68, 'Javascript轮播图 '], [69, 'Javascript正则表达式 '], [70, 'Javascript鼠标事件和拖拽原理 '], [71, '什么是网络爬虫？自动采集互联网信息 ']]
```

下面，我们将得到的数据保存到本地文件。

csv文件

csv是一种逗号分隔值文件格式。

	A	B	C
	姓名	年龄	分数
	张三	20	90
	李四	22	100
	陈六	21	87



向CSV文件中写入数据：

- 引入csv模块
- 使用open()函数创建 csv文件
- 借助csv.write()函数创建writer对象
- 调用writer对象的writerow()方法写入一行数据
- 调用writer对象的writerows()方法写入多行数据

```
import csv

with open('data.csv', 'w', newline='') as f:

    writer = csv.writer(f)

    writer.writerow([1, "100块如何花一周"])
    writer.writerow([2, "如何上班时间高效摸鱼"])

    li = [
        [3, "如何一天赚到100万"],
        [4, "从易经到股市，我的财富自由之路"],
        [5, "跟黑猫一起学编程"]
    ]

```

```
    writer.writerows(li)
```

	A	B	C	D
1	1	100块如何花一周		
2	2	如何上班时间高效摸鱼		
3	3	如何一天赚到100万		
4	4	从易经到股市，我的财富自由之路		
5	5	跟黑猫一起编程		
6				

从CSV文件中读取数据：

- 引入csv模块
- 使用open()函数打开CSV文件
- 借助csv.reader()函数创建reader对象
- 读到的每一行都是一个列表(list)

```
import csv

with open('data.csv', 'r') as f:

    reader = csv.reader(f)

    # print(reader)
    for row in reader:
        print(row)
```

数据保存

将获取到的博客所有文章标题直接存储到csv文件：

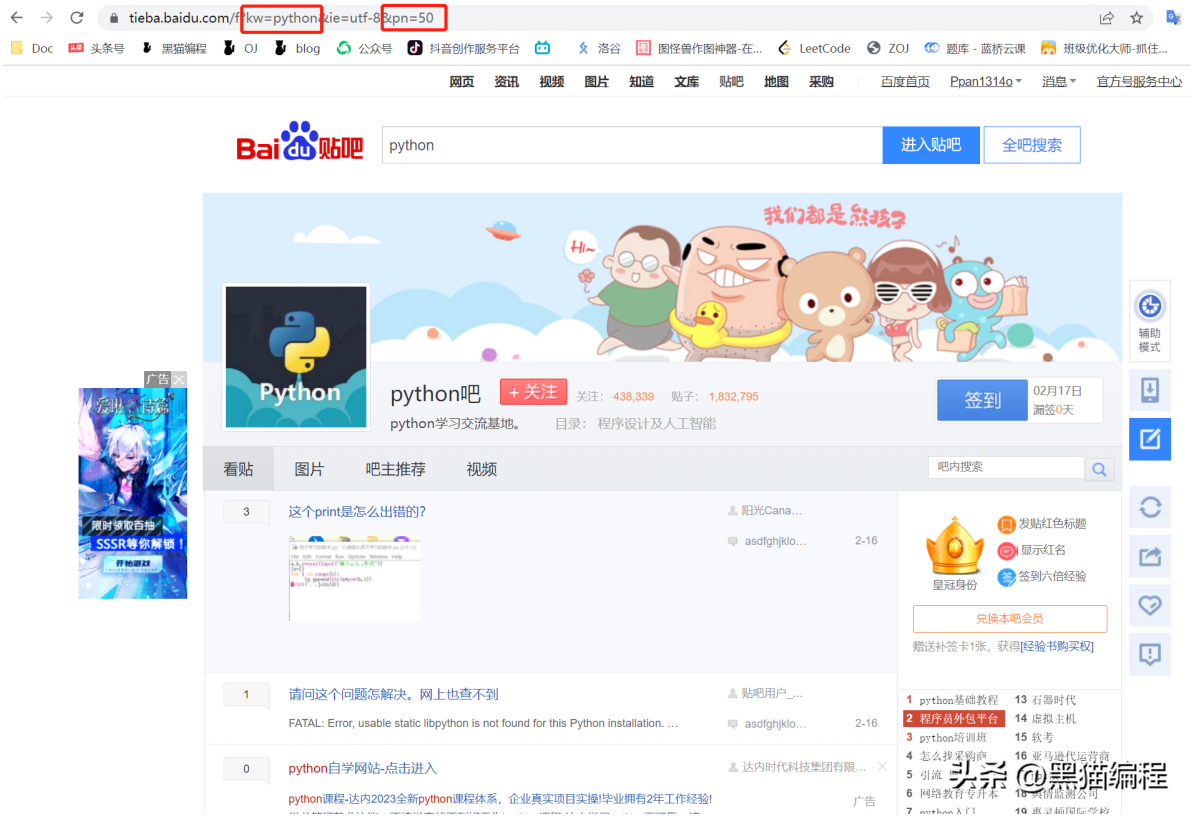
```
with open('data.csv', 'w', newline='') as f:

    writer = csv.writer(f)
    writer.writerows(titles)
```

1	1	栈，先进后出的数据结构
2	2	CCF CSP-J 2022 第二轮认证试题解析
3	3	Nginx简明教程
4	4	Docker轻量级虚拟化，镜像和容器
5	5	青岛大学开源OJ在线测评环境搭建
6	6	Nodejs后端运行和javascript刷题
7	7	莫比乌斯反演
8	8	Python OpenCV简介和图像灰度处理
9	9	什么是C++编程
10	10	C++变量，存储数据的容器
11	11	C++输入输出流
12	12	Dev C++安装配置
13	13	计算机基础知识
14	14	C++标准数据类型
15	15	C++设置域宽、保留小数位数和cmath数学库
16	16	C++语法阶段课程总结 2022.11.24
17	17	
18	18	Linux简介和云服务器配置
19	19	Linux命令行操作
20	20	Vim编辑器之神
21	21	tmux终端复用器
22	22	Linux用户和用户组
23	23	Numpy简介和数据类型
24	24	Numpy数组属性和数组创建
25	25	SSH远程登录
26	26	Linux文件权限设置
27	27	Ubuntu软件安装和卸载
28	28	Opencv图像二值化处理
29	29	C++格式化输入输出
30	30	C++语法系列
31	31	趣学C++编程
32	32	Shell脚本和变量
33	33	Numpy切片、索引 广播和迭代
34	34	Opencv图像降噪
35	35	Python Opencv绘制图形和文字

这样，就实现了数据爬取、解析和存储的流程。

百度贴吧

在贴吧中输入要搜索的信息，比如“python”，就会出现很多python吧，在最下方有翻页按钮，点击第2页，观察url，主要信息是“kw=python”和“pn=50”，第3页“pn=100”



面向对象的解决方案：下载某贴吧a→b页的html页面到本地目录。

```python
import requests

class TiebaSpider:

    def __init__(self, name, start_page, end_page):
        self.name = name
        self.start_page = start_page
        self.end_page = end_page

        self.headers = self.headers = {
            "User-Agent": "Mozilla/5.0 (Macintosh; Intel Mac OS X 10.6; rv2.0.1) Gecko/20100101 Firefox/4.0.1"
        }

        self.rawURL = "https://tieba.baidu.com/f?kw=%s&ie=utf-8&pn={}" % self.name
        # print(self.rawURL)

    def load_page(self, page_num):
        fullURL = self.rawURL.format((page_num - 1) * 50)
        response = requests.get(fullURL, headers=self.headers)
        return response.text

```

```python
    def save_page(self, page_num):
        file_name = "{}吧 第{}页.html".format(self.name, page_num)

        html = self.load_page(page_num)

        with open(file_name, "w", encoding="utf-8") as f:
            f.write(html)


    def run(self):
        for i in range(self.start_page, self.end_page + 1):
            self.save_page(i)

if __name__ == "__main__":

    keyword = input("请输入你的关键词：")
    a, b = [int(i) for i in input("请输入贴吧页数 a b（空格分隔）：").split()]
    # print(keyword, a, b)

    tieba_spider = TiebaSpider(keyword, a, b)
    tieba_spider.run()
```