*Day02回顾**

爬取网站思路

```
1、找URL规律
2、正则表达式(先分析页面,查看网页源代码)
3、定义程序框架
```

存入csv文件

```
import csv
with open('xxx.csv','w') as f:
    writer = csv.writer(f)
    writer.writerow([])
    writer.writerows([(),(),()])
```

持久化存储之MySQL

```
db = pymysql.connect('IP',...)
cursor = db.cursor()
# cursor.execute('SQL',[])
# cursor.execute('SQL',[(),(),()])
db.commit()
cursor.close()
db.close()
```

持久化存储之MongoDB

```
conn = pymongo.MongoClient('IP',27017)
db = conn['库名']
myset = db['集合名']
# myset.insert_one({})
# myset.insert_many([{},{},{}])
```

requests模块

- get()

```
1、发请求并获取响应对象
2、res = requests.get(url,headers=headers)
```

- 响应对象res属性

```
res.text ： 字符串
res.content ： bytes
res.encoding: 字符编码 res.encoding='utf-8'
res.status_code ： HTTP响应码
res.url ： 实际数据URL地址
```

多级页面数据抓取

```
1、先爬去一级页面,提取链接,继续跟进
2、爬取二级页面,提取数据
3、... ...
```

lxml使用流程

```
from lxml import etree

parse_html = etree.HTML(res.text)
r_list = parse_html.xpath('xpath表达式')
```

xpath

- 匹配规则

```
1、节点对象列表 ：  //div[@class="tiger"]
2、字符串列表:    //div[@class="t"]/@href
                  //div[@class="t"]/text()
3、函数 ： //div[contains(@id,"tiger")]//a/@href
```

- xpath高级

```
1、基准xpath表达式(节点对象列表)
2、for r in [节点对象列表]:
       username = r.xpath('.//')
```

Day03笔记

lxml解析库

- 安装

```
sudo pip3 install lxml
```

- 使用流程

```
1、导模块
   from lxml import etree
2、创建解析对象
   parse_html = etree.HTML(html)
3、解析对象调用xpath
   r_list = parse_html.xpath('xpath表达式')
```

- 练习

```
<div class="wrapper">
    <i class="iconfont icon-back" id="back"></i>
    <a href="/" id="channel">新浪社会</a>
    <ul id="nav">
        <li><a href="http://domestic.firefox.sina.com/" title="国内">国内</a></li>
        <li><a href="http://world.firefox.sina.com/" title="国际">国际</a></li>
        <li><a href="http://mil.firefox.sina.com/" title="军事">军事</a></li>
        <li><a href="http://photo.firefox.sina.com/" title="图片">图片</a></li>
        <li><a href="http://society.firefox.sina.com/" title="社会">社会</a></li>
        <li><a href="http://ent.firefox.sina.com/" title="娱乐">娱乐</a></li>
        <li><a href="http://tech.firefox.sina.com/" title="科技">科技</a></li>
        <li><a href="http://sports.firefox.sina.com/" title="体育">体育</a></li>
        <li><a href="http://finance.firefox.sina.com/" title="财经">财经</a></li>
        <li><a href="http://auto.firefox.sina.com/" title="汽车">汽车</a></li>
    </ul>
    <i class="iconfont icon-liebiao" id="menu"></i>
</div>

1、返回所有 <a> 节点的文本内容
2、提取所有的href的属性值
3、提取所有href的值,不包括 /
4、获取 图片、军事、...,不包括新浪社会
```

猫眼电影（xpath）

- 目标

```
1、地址：猫眼电影 - 榜单 - top100榜
2、目标：电影名称、主演、上映时间
```

- 步骤

```
1、确定是否为静态页面（右键-查看网页源代码，搜索关键字确认）
2、写xpath表达式
3、写程序框架
```

- xpath表达式

```
1、基准xpath：匹配所有电影信息的节点对象列表


2、遍历对象列表，依次获取每个电影信息
    for dd in dd_list:
        电影名称 ：
        电影主演 ：
        上映时间 ：
```

- 代码实现（修改之前urllib库代码）

```
1、将urllib库改为requests模块实现
2、改写parse_page()方法
```

```

```

链家二手房案例（xpath）

- 实现步骤

1. 确定是否为静态

```
打开二手房页面 -> 查看网页源码 -> 搜索关键字
```

2. xpath表达式

```
1、修改方法：右键 -> copy xpath -> 测试修改
2、基准xpath表达式(匹配每个房源信息节点列表)

3、依次遍历后每个房源信息xpath表达式
    * 名称：
    * 总价：
    * 单价：
```

3. 代码实现

```
1
```

百度贴吧图片抓取

- 目标

```
1  抓取指定贴吧所有图片
```

- 思路

```
1  1、获取贴吧主页URL,下一页,找到不同页的URL规律
2  2、获取1页中所有帖子URL地址:[帖子链接1,帖子链接2,...]
3  3、对每个帖子链接发请求,获取图片URL
4  4、向图片的URL发请求,以wb方式写入本地文件
```

- 实现步骤

1. **贴吧URL规律**

```
1  http://tieba.baidu.com/f?kw=??&pn=50
```

2. **xpath表达式**

```
1  1、帖子链接xpath
2
3
4  2、图片链接xpath
5
6
7  3、视频链接xpath
8
9     # 注意:此处视频链接前端对响应内容做了处理,需要查看网页源代码来查看,复制HTML代码在线格式化
```

3. **代码实现**

```
1
```

requests.get()参数

查询参数-params

- 参数类型

```
1  字典,字典中键值对作为查询参数
```

- 使用方法

```
1、res = requests.get(url,params=params,headers=headers)
2、特点:
    * url为基准的url地址，不包含查询参数
    * 该方法会自动对params字典编码,然后和url拼接
```

- 示例

```

```

代理参数-proxies

- 定义

```
1、定义：代替你原来的IP地址去对接网络的IP地址。
2、作用：隐藏自身真实IP,避免被封。
```

- 普通代理

获取代理IP网站

```
西刺代理、快代理、全网代理、代理精灵、... ...
```

参数类型

```
1、语法结构
    proxies = {
        '协议':'协议://IP:端口号'
    }
2、示例
    proxies = {
        'http':'http://IP:端口号',
        'https':'https://IP:端口号'
    }
```

示例

1. 使用免费普通代理IP访问测试网站: http://httpbin.org/get

```

```

2、使用收费普通代理IP访问测试网站: http://httpbin.org/get

```
1、从代理网站上获取购买的普通代理的api链接
2、从api链接中提取出IP
3、随机选择代理IP访问网站进行数据抓取
```

```
1
```

3. 思考: 建立一个自己的代理IP池，随时更新用来抓取网站数据

```
1
```

- 私密代理

语法格式

```
1、语法结构
proxies = {
    '协议':'协议://用户名:密码@IP:端口号'
}

2、示例
proxies = {
    'http':'http://用户名:密码@IP:端口号',
    'https':'https://用户名:密码@IP:端口号'
}
```

示例代码

```
1
```

今日作业

糗事百科（xpath）

```
1、URL地址: https://www.qiushibaike.com/text/
2、目标 ： 用户昵称、段子内容、好笑数量、评论数量
```

电影天堂（xpath）