**练习脚本： /Users/sherryzhang/PycharmProjects/crawler/movie.pym**

目的：爬取豆瓣上的电影列表和影评

**浏览器打开网页的过程**

输入地址 > DNS服务器找到服务器主机 > 发送请求 > 服务器解析 > 发送浏览器结果（包括HTML, JS, CSS等）> 浏览器解析 > 网页

**爬虫的基本流程**

1. 浏览器发送消息给网址所在服务器(HTTP Request)
   * GET - 读取数据 “显示”请求
   * POST - 提交数据（如表单）
2. 获取服务器响应内容(HTTP Response)
3. 解析内容
4. 保存数据

**查看网页的Headers**

1. Developer > Developer Toolbar > Network tab > Headers tab

**BeautifulSoup的常见用法**

soup = BeautifulSoup(html,'lxml') #创建beautifulsoup对象soup,lxml对应的是HTML解析器

print(soup.prettify()) #格式化输出soup对象的内容

print(soup.find\_all('a')) #find\_all()方法搜索当前tag的所有tag子节点,并判断是否符合过滤器的条件

print(soup.title.string) #用.string可以获取标签内部的文字

print(soup.p["class"]) #输出p这个标签里属性名为class的值，返回的是列表

print(link.get('href’)) #用get()方法传入属性的名称作为过滤条件

print(soup.find\_all(attrs={'id': 'list-1'})) #attrs可以传入字典的方式来查找标签，但如果要查找class相关的可以更改attrs={'class\_':'element'}或者soup.find\_all('',{"class":"element})。

print(soup.select('#list-2 .element’)) #通过select()直接传入CSS选择器就可以完成选择,"标签1，标签2" 找到所有的标签1和标签2，“标签1 标签2” 找到标签1内部的所有的标签2

**Mac自带的python安装第在三库**

1. Request库：sudo pip install requests
2. lxml库：sudo pip install lxml

**Python查看函数的使用方法**

1. 例如查看html库里的fromstring()这个方法的用途

            import ltml

            print help(html.fromstring)

**Requests库、Ltml库、Json库的使用**

1. 例如查看html库里的fromstring()这个方法的用途
2. 可以用text()来获取标签中的文本内容