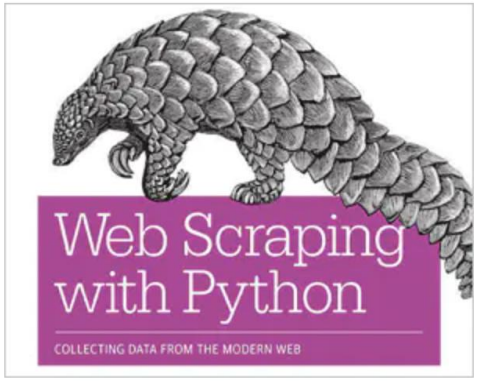
数据可视化 实践课06


Web Scraping
with Python
COLLECTING DATA FROM THE MODERN WEB

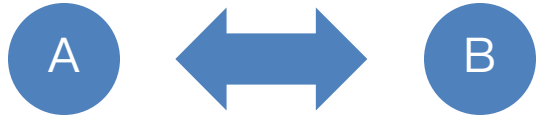
1

爬虫基本操作

浏览器访问网站流程

假设A有一个网络服务器。而B想用一个台式机通过浏览器访问A的服务器上运行的某个网站。整个访问过程归纳如下：

- B输入访问网站的地址后，B的电脑传输一段二进制的数据，这些数据包含**数据头**和**数据内容**
  - **数据头**包含发送方B的mac地址和目的地的ip地址
  - **数据内容**包含了针对A网络服务器的请求，例如，获得某个网页页面。
- B的本地网络路由器将数据打包传输到A的ip地址
- B的数据最后通过物理电缆进行传输
- A的服务器接受到了B的数据包
- A的服务器识别存于**数据头**的端口号，发现是80，意味着这是一个网页请求，于是调用网页服务器相关的程序
- 网页服务器程序接受到如下信息：
  - This is a GET request
  - The following file is requested: index.html
- 网页服务器程序载入正确的HTML 文件，并打包通过本地路由发送给B的电脑



网络服务器　　台式机

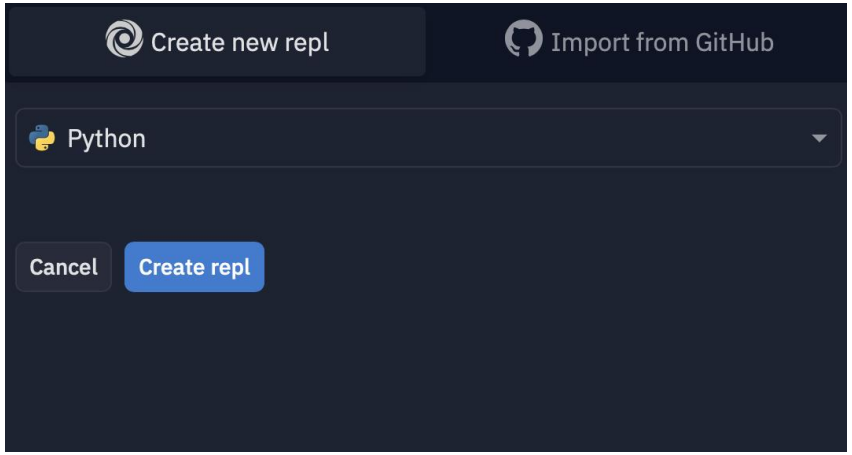
Python模拟页面访问

练习1–1: 请用Python3执行以下代码，获得页面内容

```
from urllib.request import urlopen
strhtml = urlopen("http://pythonscraping.com/pages/page1.html")
print(strhtml.read())
```

如果电脑里没有Python3环境，也可以使用在线编辑器。使用前须创建一个repl使用前需要注册登陆，要求特殊的网络环境

https://repl.it/languages/python3



BeautifulSoup

1. BeautifulSoup将HTML的内容组织成了Python可以识别的对象格式
2. 因为BeautifulSoup不是Python默认的库，需要手动安装（或者通过Anaconda安装）
3. 手动安装方式：pip install beautifulsoup4

可通过**国内源**加快安装速度： pip install beautifulsoup4 **-i https://pypi.mirrors.ustc.edu.cn/simple/**

练习1–2: 获得网页页面某字段的内容

```
from urllib.request import urlopen
from bs4 import BeautifulSoup
html = urlopen("http://www.pythonscraping.com/exercises/exercise1.html")
bsObj = BeautifulSoup(html.read(), "html.parser");
print(bsObj.h1)
```

这段代码解析了exercise1.html这个HTML文件，并输出了h1这个字段的内容：

```
<h1>An Interesting Title<h1>
```

练习1–3: 获得网页页面某字段的内容

```
from urllib.request import urlopen
from bs4 import BeautifulSoup
html = urlopen("https://www.ecnu.edu.cn/info/1094/59213.htm")
bsObj = BeautifulSoup(html.read(), "html.parser");
rs = bsObj.find_all(attrs={"name": "description"});
print ( rs[0]['content'] );
print ( rs[1]['content'] );
```

2

词频统计

练习2–1

1. 使用爬虫工具获得以下网页5条新闻的文本内容：
http://chenhui.li/courses/infovis2025/04–EcnuNews.html
2. 5条新闻的文字内容合并后，使用Jieba库进行词频统计
Python的Jieba库安装：**pip install jieba -i https://pypi.mirrors.ustc.edu.cn/simple/**

注：批量获得链接的参考代码如下：

```
from urllib.request import urlopen
from bs4 import BeautifulSoup
html = urlopen("http://chenhui.li/courses/infovis2025/04–EcnuNews.html")
bsObj = BeautifulSoup(html.read(), "html.parser");
table = bsObj.table
#print(table)
for a in table.find_all('a', href=True, text=True):
        link_text = a['href']
        print(link_text)
```

词频统计

Jieba分词并进行词频统计的代码：

```
Building prefix dict from the default dictionary ...
Loading model from cache /var/folders/kv/lv7qczln14n8v7qr4h0bfdkr0000gn/T/jieba.
cache
Loading model cost 0.505 seconds.
Prefix dict has been built successfully.
{'我': 1, '的': 2, '数据': 2, '在': 2, '这里': 1, ', ': 1, '你': 1, '哪里': 1}
```

```
import jieba

# 词频统计函数
def index(word):
    dict = {}
    for item in word:
        dict[item] = dict.get(item,0) + 1
    return dict

str = "我的数据在这里，你的数据在哪里"
rs = jieba.lcut(str)
rs2 = index(rs)
print(rs2)
```

练习2-2

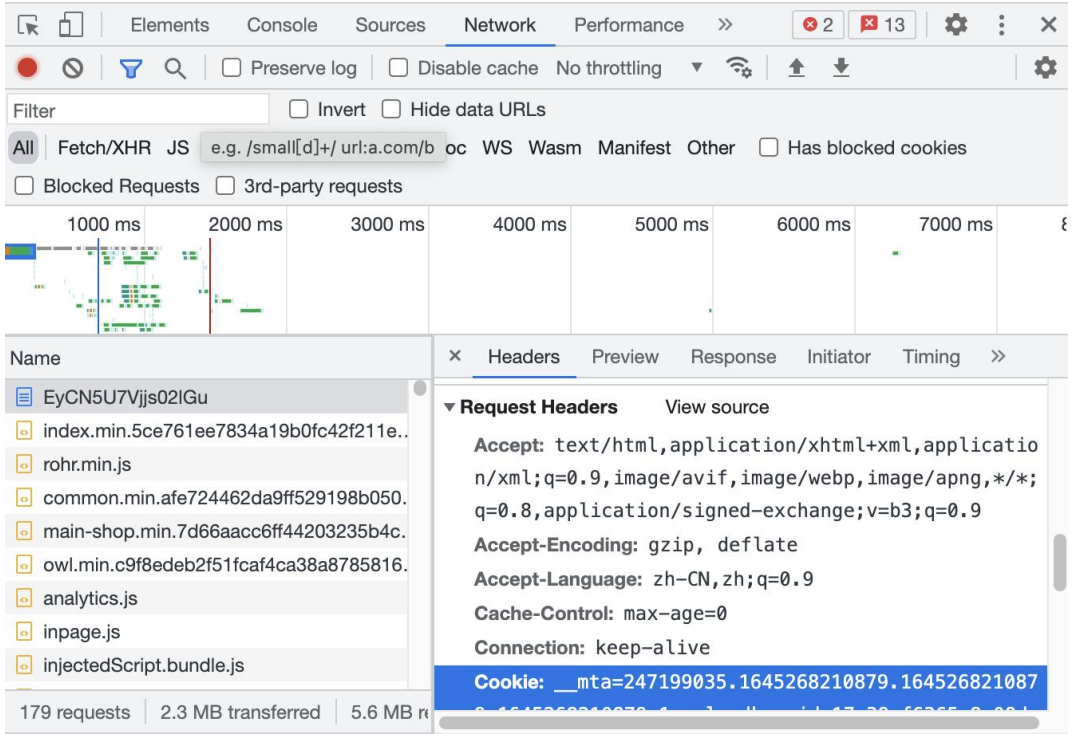
使用EChart的折线图可视化练习2-1结果中词频大于3的词(x为词，y为词频)

3

课后拓展
了解Cookie

Cookie

- 浏览器登录账号后获取Cookie信息



- 示例代码

```
import requests
#要抓取的目标页码地址
url = "https://www.qcc.com/web/search/risk?key=东方财富证券股份有限公司"
#抓取页码内容，返回响应对象
headers = {“User_Agent”:“Mozilla/5.0 (Windows NT 10.0; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/64.0.3282.186 Safari/537.36”, “Cookie”：“该值从浏览器获取”, “Host”: "www.qcc.com"
}
response = requests.get(url, headers = headers )
with open("test.html", "w", encoding="utf-8") as f:
    f.write(response.content.decode());
```

4

课后拓展

了解Selenium

动态网页数据爬取

基本思路：

- 使用Selenium模拟浏览器行为

参考资料：爬虫系列(十二) selenium的基本使用

基本步骤：

- 下载Chrome驱动器，放到Python目录

https://sites.google.com/a/chromium.org/chromedriver/home

注意：下载的版本需与浏览器版本一致，Chrome中输入chrome://settings/help 确定版本

示例代码：

```
from selenium import webdriver
browser = webdriver.Chrome()
browser.get('https://www.ecnu.edu.cn’)
rs = browser.find_element_by_id(‘top-nav')
```

5

课后拓展

了解Playwright

https://github.com/microsoft/playwright

其他可以尝试爬取的数据

- 教职工信息： http://www.cs.ecnu.edu.cn/jzgml/list.htm
- 空气污染数据： https://aqicn.org/city/shanghai/
- 猫眼实时票房： http://piaofang.maoyan.com/dashboard
- 豆瓣电影影评： https://movie.douban.com/
- 拉勾网（招聘要求）：https://www.lagou.com/
- 企查查（公司信息）： https://www.qcc.com/
- 东方财富（上市公司）： https://www.eastmoney.com/
- **大众点评（餐饮消费）**： https://www.dianping.com/