第二次课程作业

[redacted]

一. 任务

按照中国人口前 20 的城市给出两室一厅房子出租和购买的最高及最低价。

二. 设计思路

1. 选取目标网站：58 同城
2. 自动生成不同城市 url
3. 使用 python 的 lxml 库定位价格信息
4. 自动获取下一页的 url
5. 最高价最低价获取策略：
   (1) 比较所有页面住房的价格信息确定
   (2) 通过网页自带价格排序功能确定
6. 单位不一致的情况下，需要爬取价格单位以及建筑面积等信息

三. 实验环境

1. 语言：python
2. 编译器：VSCode
3. 使用 python 库：

```
import requests # HTTP 库
import json
from lxml import etree # 解析库
from fake_useragent import UserAgent # 伪装请求头
import time, re, datetime
import pandas as pd
```

四. 实现步骤

1. 找到中国人口前 20 的城市列表（2019 年数据）

```
city20 = ["重庆","上海","北京","成都","天津","广州","深圳","武汉",
          "南阳","临沂","石家庄","哈尔滨","苏州","保定","郑州","西安",
          "赣州","邯郸","温州","潍坊"]
```

2. 找到不同城市对应的 58 同城网址特征

```
https://wh.58.com/ # 武汉 - wh
https://cd.58.com/ # 成都 - cd
```

分析可知不同城市的 58 同城首页为 `https://{城市首字母缩写}.58.com/`

(1) 租房页面：https://wh.58.com/chuzu/ （武汉为例）
(2) 买房页面：https://wh.58.com/xinfang/

3. 获取所有城市的名字及缩写的对应关系

目标页面：`https://www.58.com/changecity.html`

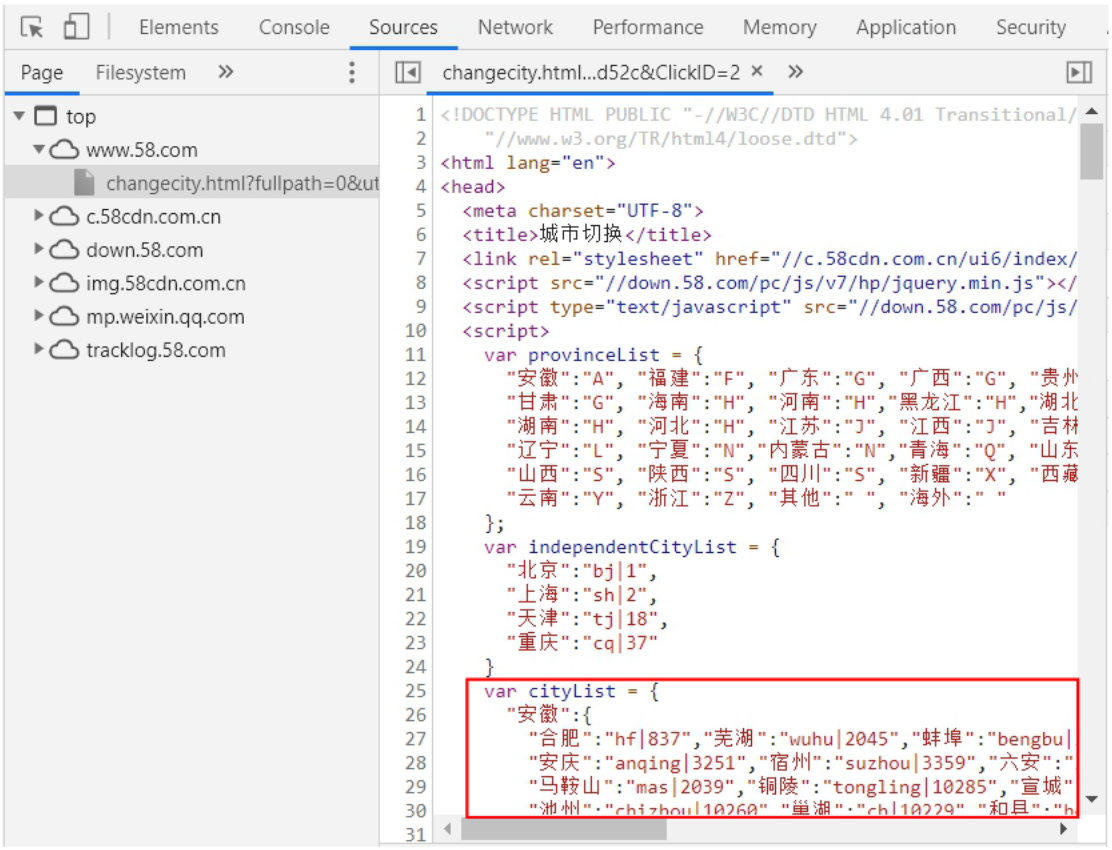
(1) 获取页面源代码

(2) 根据正则 `cityList = (.*?)</script>` 获取相关数据


```
{
    "安徽":{
      "合肥":"hf|837","芜湖":"wuhu|2045","蚌埠":"bengbu|3470","阜阳":"fy|2325","淮南":"hn|2319",
      "安庆":"anqing|3251","宿州":"suzhou|3359","六安":"la|2328","淮北":"huaibei|9357","滁州":"chuzhou|10266",
      "马鞍山":"mas|2039","铜陵":"tongling|10285","宣城":"xuancheng|5633","亳州":"bozhou|2329","黄山":"huangshan|2323",
      "池州":"chizhou|10260","巢湖":"ch|10229","和县":"hexian|10892","雷邱":"hq|11226","桐城":"tongcheng|11296",
      "宁国":"ningguo|5645","天长":"tianchang|10273","东至":"dongzhi|10262","无为":"wuweixian|10232"
    },
    "福建":{
      "福州":"fz|304","厦门":"xm|606","泉州":"qz|291","莆田":"pt|2429","漳州":"zhangzhou|710",
```

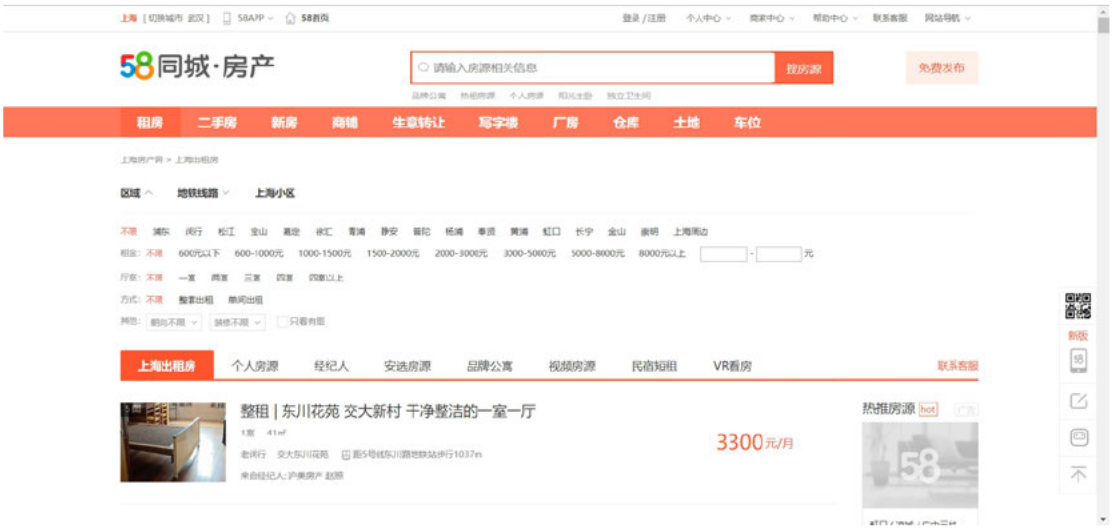

(3) 数据处理得到城市名及其缩写的词典

```
api = "https://www.58.com/changecity.html"

headers = self.session.headers.copy()

response = self.session.get(api, headers=headers)


html = response.text

res = re.findall("cityList = (.*?)</script>", html, re.S)[0]

res = re.sub("\s", "", res)

dic = json.loads(res)

for k, v in dic.items():

    or k1, v1 in v.items():

        dic[k][k1] = v1.split("|")[0]
```



4. 找到爬取租房和买房信息的关键页面

目的	条件	url
买房	base_url	https://{ }.58.com/chuzu/
	筛选‘二室’条件后	https://{ }.58.com/chuzu/j2/
租房	base_url	https://{ }.58.com/xinfang/loupan/all/
	筛选‘二室’条件后	https://{ }.58.com/xinfang/loupan/all/h2
	选择价格升序	https://{ }.58.com/xinfang/loupan/all/h2_s2
	选择价格降序	https://{ }.58.com/xinfang/loupan/all/h2_s1




```
<!-- 筛选条件 -->
<!-- 租金 -->
▶<dl id="secitem-rent" class="secitem">…</dl>
<!-- 厅室 -->
▼<dl class="secitem">
  <dt>厅室：</dt>
  ▼<dd id="secitem-room">
    <a rel="nofollow" href="https://sh.58.com/chuzu/" class="select"
    onclick="clickLog('from=fcpc_list_sh_Tingshi_')">不限</a>
    <a rel="nofollow" href="https://sh.58.com/chuzu/j1/" onclick=
    "clickLog('from=fcpc_list_sh_Tingshi_j1')">一室</a>
    <a rel="nofollow" href="https://sh.58.com/chuzu/j2/" onclick=
    "clickLog('from=fcpc_list_sh_Tingshi_j2')">两室</a>
    <a rel="nofollow" href="https://sh.58.com/chuzu/j3/" onclick=
    "clickLog('from=fcpc_list_sh_Tingshi_j3')">三室</a>
    <a rel="nofollow" href="https://sh.58.com/chuzu/j4/" onclick=
    "clickLog('from=fcpc_list_sh_Tingshi_j4')">四室</a>
    <a rel="nofollow" href="https://sh.58.com/chuzu/j5/" onclick=
    "clickLog('from=fcpc_list_sh_Tingshi_j5')">四室以上</a>
  </dd>
</dl>
```


```
<!-- 户型-->
▼<div class="filter-item">
  <label class="item-title">户型：</label>
  ▼<div class="item-mod">
    <span>全部</span>
    ▶<a class="multi-item" data-id="1" data-type="h" data-url="https://
    sh.58.com/xinfang/loupan/all/h1/" rel="nofollow" href="https://
    sh.58.com/xinfang/loupan/all/h1/">…</a>
    ▶<a class="multi-item" data-id="2" data-type="h" data-url="https://
    sh.58.com/xinfang/loupan/all/h2/" rel="nofollow" href="https://
    sh.58.com/xinfang/loupan/all/h2/">…</a>
    ▶<a class="multi-item" data-id="3" data-type="h" data-url="https://
    sh.58.com/xinfang/loupan/all/h3/" rel="nofollow" href="https://
    sh.58.com/xinfang/loupan/all/h3/">…</a>
    ▶<a class="multi-item" data-id="4" data-type="h" data-url="https://
    sh.58.com/xinfang/loupan/all/h4/" rel="nofollow" href="https://
    sh.58.com/xinfang/loupan/all/h4/">…</a>
    ▶<a class="multi-item" data-id="5" data-type="h" data-url="https://
    sh.58.com/xinfang/loupan/all/h5/" rel="nofollow" href="https://
    sh.58.com/xinfang/loupan/all/h5/">…</a>
    </div>
  </div>
```

```
▼<a class href="https://sh.58.com/xinfang/loupan/all/h2_s2/" rel=
"nofollow">
    <span>价格</span>
    <i class="list-ico defult-down"></i>
  </a>
```

5. 构造城市租房 URL

```
''爬取租房信息的爬虫方法'''
assert self.all_city_dict is not None, "获取所有城市信息失败"
print("---all_city_dict---")
format_city = self.all_city_dict.pop(city, None)
print("format_city:", format_city)
```

```
assert format_city is not None, "{}该城市不在爬取城市之内".format(city)


'''构造该城市租房页面 url，获取所需数据'''
self.city = city
start_url = "https://{}.58.com/chuzu/j2/".format(format_city)
```

6. 获取页面信息

- 通过 time.sleep(2) 来避免过于频繁的访问
- 伪装请求头来发出请求

```
ua = UserAgent()
self.session = requests.Session()
self.session.headers = {
    "user-agent": ua.random
}


def __get_html_source(self, url, params=None):
    '''通过 get 方式获取到网页的源码'''
    time.sleep(2)
    headers = self.session.headers.copy()
    try:
        if not params:
            params = {}
        response = self.session.get(url=url, headers=headers, params=params)
        return response
    except Exception as e:
        with open("./url_log_error.txt", "a", encoding="utf-8") as f:
            f.write(str(datetime.datetime.now()) + "\n")
            f.write(str(e) + "\n")
            f.write("error_url>>:{}".format(url) + "\n")
```

7. 从开始页面提取租房价格信息

(1) 使用 etree 和 xpath 定位所需信息

```
▼<ul class="house-list">
  <!--房源列表信息-->
 ▼<li class="house-cell " logr="gz_1_9616604916999_45417405943562_sortid:
 668167483@postdate:1616814001238" sortid="1616814001238">
  ▶<div class="img-list">…</div>
  ▶<div class="des">…</div>
  ▼<div class="list-li-right">
     <div class="send-time">                              </div>
    ▼<div class="money">
       <b class="strongbox">3000</b>
       "元/月                "
     </div>
   </div>
   <div class="listline"></div>
 </li>
```

(2) 通过比较确定当前页最高价以及最低价

- xpath(“string(.)”)以字符串的形式取出当前标签下的值

- re.sub()去除字符串中的空白字符
- int()将字符转换为整数型
- 如果价格高于当前最高价或低于最低价，则更新最高和最低价

```
def __get_price(self, response):


    html = response.text
    # 开始从页面中提取出想要的数据
    xml = etree.HTML(html)
    xpath_list = xml.xpath("//div[@class='money']/b[@class='strongbox']")
    for price_info_list in xpath_list:
        house_price = re.sub("\s", "", price_info_list.xpath(
            "string(.)"))
        house_price = int(house_price)
        # print(house_price)
        if house_price > self.highest:
            self.highest = house_price
        if house_price and house_price < self.lowest:
            self.lowest = house_price
```

8. 获取下一页页面信息

查看当前页面的切换页面栏，查看元素，可知<a class="next" href="下一页网址">

```
<!--页码 -->
▼<li id="pager_wrap">
  ▼<div class="pager">
    ▶<strong>…</strong>
    ▶<a href="https://sh.58.com/chuzu/j2/pn2/">…</a>
    ▶<a href="https://sh.58.com/chuzu/j2/pn3/">…</a>
      " . . . "
    ▶<a href="https://sh.58.com/chuzu/j2/pn70/">…</a>
    ▼<a class="next" href="https://sh.58.com/chuzu/j2/pn2/">
       <span>下一页</span>
     </a>
   </div>
 </li>
```

```
def __is_exist_next_page(self, response):
    '''判断是否存在下一页,存在拿到下一页的链接，不存在返回 False'''
    xml = self.__response_to_xml(response)
    try:
        next_page_url = xml.xpath("//a[@class='next']/@href")[0]
        # print(next_page_url)
        return next_page_url
    except IndexError:
        return False
```

9. 从开始页面提取新房价格信息

- 新房页面有价格排序选项，分别从价格升序和降序页面提取第一个房源价格即可
- 售价的单位不一致，有房屋总价，也有单位面积价格
- 有的房屋暂无售价，但给出了周边参考价
- 需要额外爬取面积信息和单位信息


```
def __get_xinfang_info(self, url, params):
    response = self.__get_html_source(url, params)
    html = response.text

    # 开始从页面中提取出想要的数据
    xml = etree.HTML(html)
    # 可以定位到总价或周边定价
    xpath_list = xml.xpath("//p[@class='price']/span|//p[@class='favor-tag around-price']/span
")[0]

    house_price = re.sub("\s", "", xpath_list.xpath("string(.)"))
    # <span>145672</span>“万元/套起”
    # 正则匹配单位
    unit = re.sub("<.*>", "", etree.tostring(xpath_list, encoding='UTF-8').decode()).strip()
    xpath_list = xml.xpath("//span[@class='building-area']")[0]
    area = re.sub("\s", "", xpath_list.xpath("string(.)")).split(': ')[1]
    return house_price, unit, area
```

10. 保存成文件

```
    city_58 = Info_58()
    zufang_list = []
    xinfang_list = []
    for city in city20:
        # city_58.highest = 0
```

```
        # city_58.lowest = float('inf')
        # city_58.info_zufang(city)
        # zufang_list.append([city, city_58.highest, city_58.lowest])
        city_58.info_xinfang(city)
        xinfang_list.append([city,city_58.highest+'('+city_58.unit_h+')',city_58.area_h,
                             city_58.lowest+'('+city_58.unit_l+')', city_58.area_l])

    columns_z = ["城市", "最高价", "最低价"]
    columns_x = ["城市", "最高价(单位)", "面积(最高)","最低价(单位)","面积(最低)"]
    dt = pd.DataFrame(xinfang_list, columns=columns_x)
    dt.to_csv("buy_csv.csv", mode='a', index=0)
```

五．实验结果

网页有动态更新，每次爬取结果会存在差别，一下为单次爬取结果

1. 租房信息（元/月）：

城市	最高价	最低价
重庆	12500	280
上海	30000	600
北京	65000	500
成都	38000	300
天津	8000	390
广州	28800	200
深圳	60000	500
武汉	8000	300
南阳	17000	108
临沂	140000	150
石家庄	12000	350
哈尔滨	20000	180
苏州	24000	300
保定	20000	300
郑州	7800	270
西安	14000	350
赣州	5300	200
邯郸	20500	150
温州	45000	300
潍坊	100000	200

2. 买房信息

城市	最高价(单位)	面积(最高)	最低价(单位)	面积(最低)
重庆	42000(元/㎡)	91-340㎡	75(万元/套起)	64-158㎡
上海	145534(元/㎡)	76-99㎡	142(万元/套起)	75-138㎡
北京	168000(元/㎡)	245-443.76㎡	350(万元/套起)	81.05-131㎡
成都	50000(元/㎡起)	61-192㎡	52(万元/套起)	35-48㎡
天津	85000(元/㎡起)	149.78-277.65㎡	155(万元/套起)	79-165㎡
广州	107000(元/㎡)	70-125㎡	560(万元/套起)	88-131㎡
深圳	180000(元/㎡)	79-152㎡	1000(万元/套起)	120-250㎡
武汉	55000(元/㎡)	207-300㎡	60(万元/套起)	33-57㎡
南阳	15000(元/㎡)	90-140㎡	3300(元/㎡)	90-152㎡
临沂	27000(元/㎡起)	95-194㎡	5600(元/㎡)	66.11-113.27㎡
石家庄	35000(元/㎡)	99.53-330㎡	13090(元/㎡)	114.84-145.9㎡
哈尔滨	370(万元/套起)	88-251.39㎡	45(万元/套起)	42.47-73.39㎡
苏州	55000(元/㎡)	113-380㎡	44(万元/套起)	36-69㎡
保定	24000(元/㎡)	49.38-327.2㎡	80(万元/套起)	80-116㎡
郑州	30000(元/㎡)	67.83-136.96㎡	350(万元/套起)	134㎡
西安	37000(元/㎡)	39.13-132.17㎡	5200(元/㎡)	90-150㎡
赣州	13513(元/㎡)	95-160㎡	4900(元/㎡)	93-142㎡
邯郸	16800(元/㎡)	176-288㎡	1500(万元/套)	66.8-381㎡
温州	44000(元/㎡)	45-260㎡	60(万元/套起)	57-128㎡
潍坊	16000(元/㎡)	101-213㎡	350(万元/套起)	103-498.85㎡