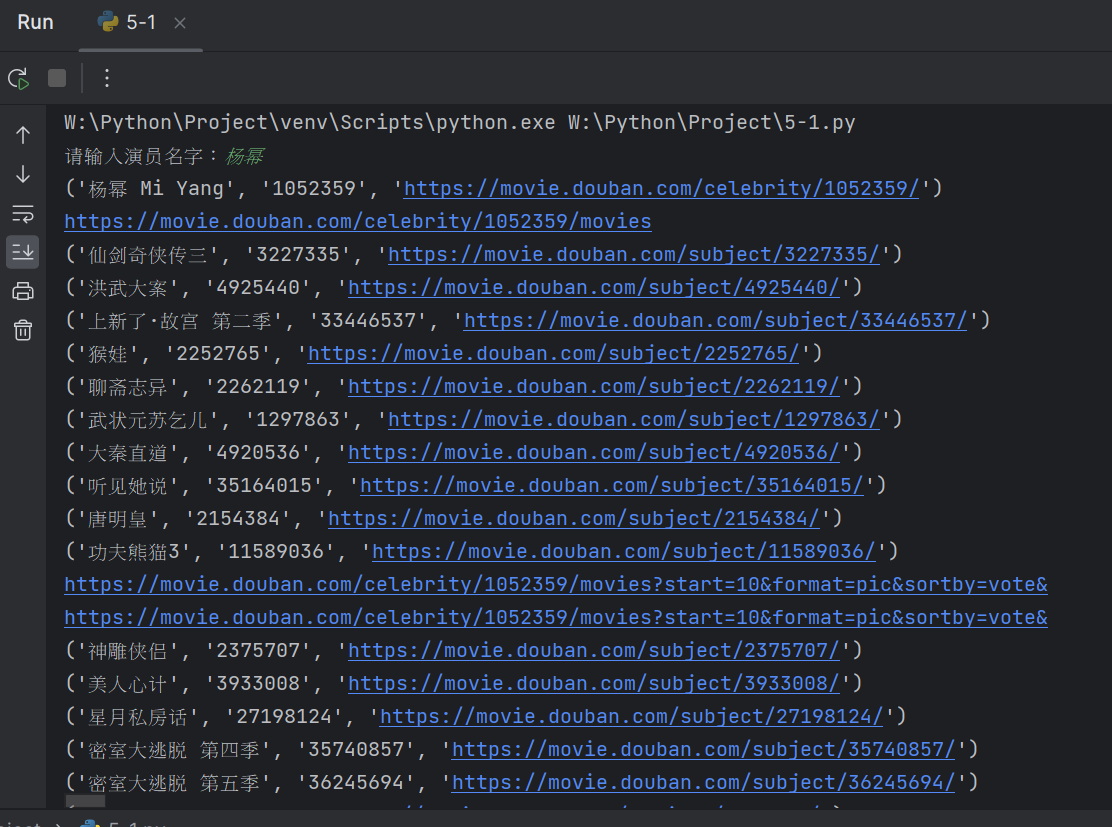
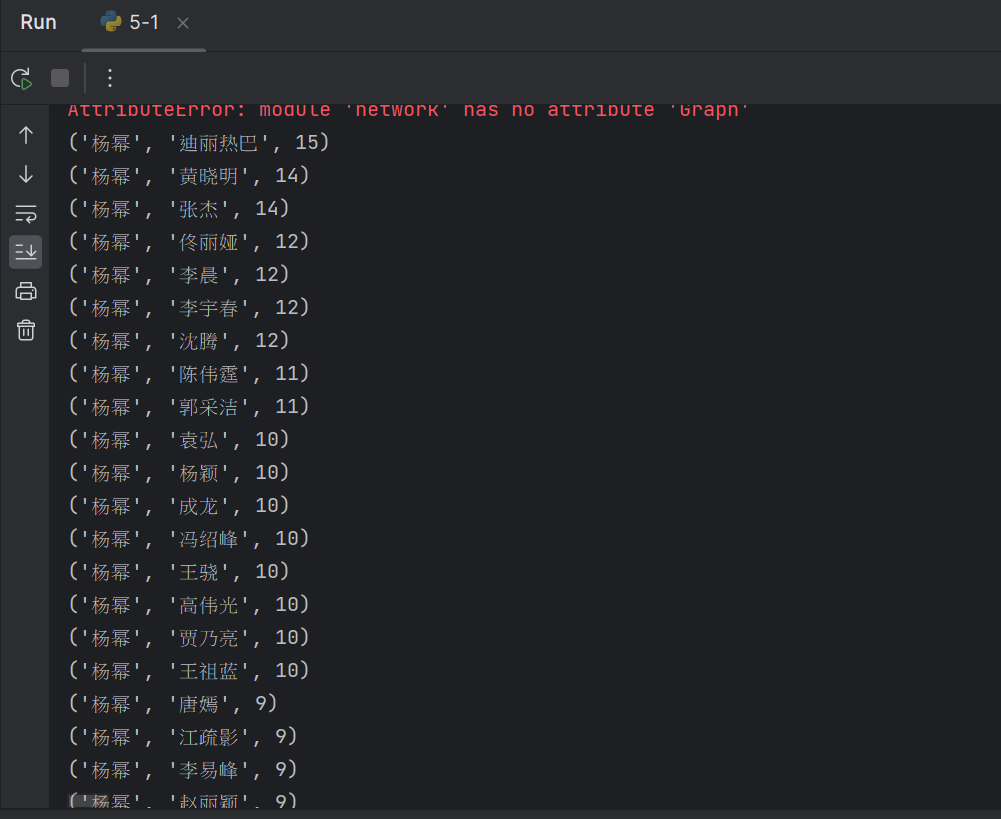
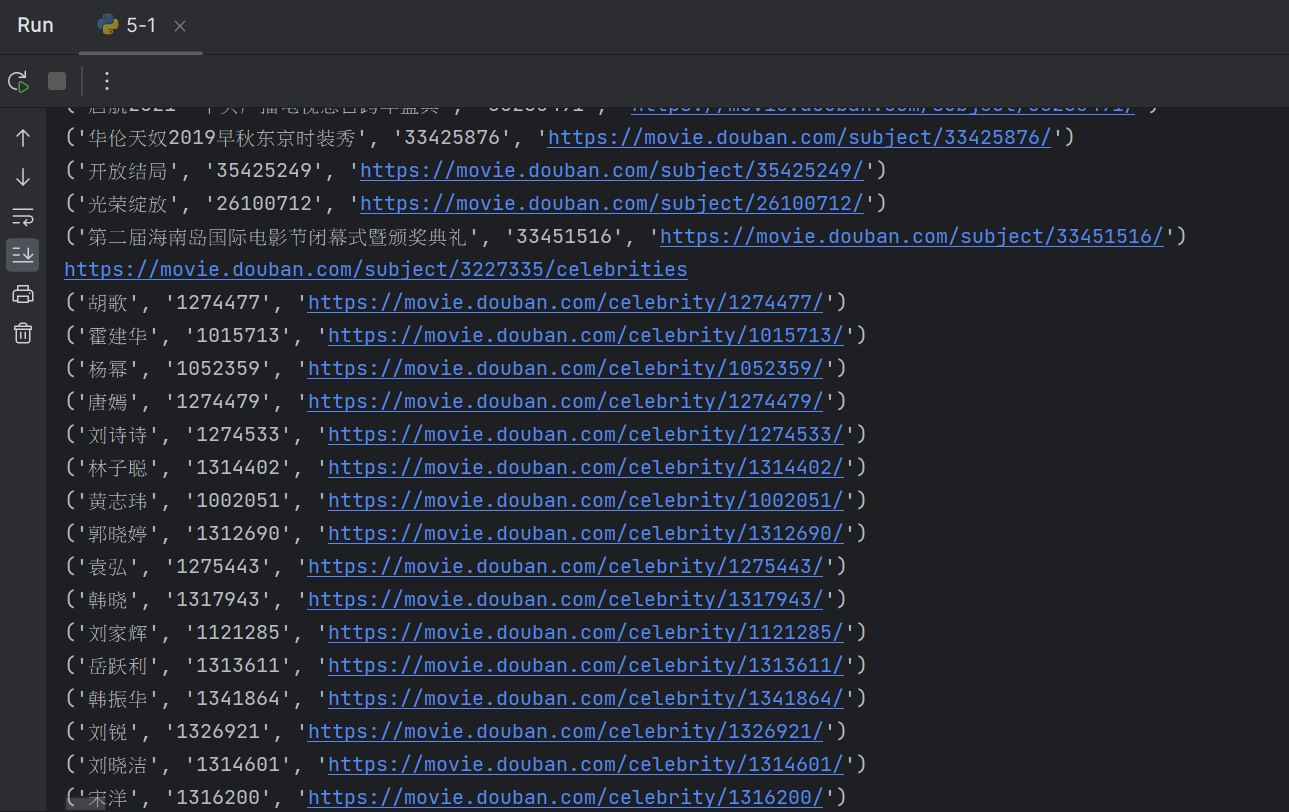
1.

import json  
import time  
from bs4 import BeautifulSoup  
import requests  
import urllib.parse  
from selenium import webdriver  
import networkx as nx  
import matplotlib.pyplot as plt  
  
  
  
  
  
def search\_person\_by\_name(name):  
 url = 'https://search.douban.com/movie/subject\_search?search\_text=' + urllib.parse.quote(name)  
 # user\_agent='Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/105.0.0.0 Safari/537.36'  
 browser = webdriver.Chrome()  
 browser.get(url)  
 soup = BeautifulSoup(browser.page\_source, 'html.parser')  
 results = soup.select('#wrapper .detail .title a')[0]  
 name = results.text  
 id = results['href'].strip().split('/')[-2]  
 web = results['href']  
 person = (name, id, web)  
 print(person)  
 time.sleep(2)  
 return person  
  
  
def search\_movies\_by\_person(person):  
 list = []  
 url = f'https://movie.douban.com/celebrity/{person[1]}/movies'  
 while True:  
 print(url)  
 browser = webdriver.Chrome()  
 browser.get(url)  
 soup = BeautifulSoup(browser.page\_source, 'html.parser')  
 movies = soup.select('#wrapper .article dd h6 a')  
 for item in movies:  
 name = item.text.strip()  
 id = item['href'].strip().split('/')[-2]  
 web = item['href']  
 moive = (name, id, web)  
 list.append(moive)  
 print(moive)  
 time.sleep(1)  
 if len(soup.select('.next a')) == 0:  
 break  
 url = f'https://movie.douban.com/celebrity/{person[1]}/movies' + soup.select('.next a')[0]['href']  
 print(url)  
 return list  
  
  
def search\_persons\_by\_moive(moive):  
 list = []  
 url = f'https://movie.douban.com/subject/{moive[1]}/celebrities'  
 print(url)  
 browser = webdriver.Chrome()  
 browser.get(url)  
 soup = BeautifulSoup(browser.page\_source, 'html.parser')  
 results = soup.select('.list-wrapper')  
 for item in results:  
 if (item.select('h2')[0].text.strip() == '演员 Cast'):  
 members = item.select('li .name a')  
 for member in members:  
 name = member.text.strip().split(' ')[0]  
 id = member['href'].strip().split('/')[-2]  
 web = member['href']  
 number = (name, id, web)  
 list.append(number)  
 print(number)  
 time.sleep(0.5)  
 return list  
  
  
# 获取演员合作次数字典  
def draw(name):  
 dic = {}  
 person = search\_person\_by\_name(name)  
 movies = search\_movies\_by\_person(person)  
 for item in movies:  
 numbers = search\_persons\_by\_moive(item)  
 for number in numbers:  
 if number[0] in dic:  
 dic[number[0]] += 1  
 else:  
 dic[number[0]] = 1  
 print(dic)  
 return dic  
  
  
def draw\_graph(person\_name, num\_dict, tp='spring', max\_node=50):  
 *"""  
 rela\_dict: 节点关系列表，比如：[('张国立', '王刚', 30), ('张国立', '张铁林', 5), ('张国立', '黄晓明', 2)]  
 num\_dict: 每个演员参与合作的电影数量比如：{'张国立':33,'王刚':30,'张铁林':5,'黄晓明':2}  
 person\_name: 演员名称  
 tp: 图形分布类型（circular/spring/shell/spectral）  
 max\_node：待显示的节点数量，默认是50个  
 """* rela\_list = [(person\_name, person, num\_dict[person]) for person in num\_dict if person != person\_name]  
  
 plt.figure(figsize=(50, 50))  
  
 if len(rela\_list) > max\_node:  
 rela\_list = sorted(rela\_list, key=lambda x: x[2], reverse=True)  
 rela\_list = rela\_list[0:max\_node]  
 for node in rela\_list:  
 print(node)  
  
 G = nx.Graph()  
 for u, v, w in rela\_list:  
 G.add\_edge(u, v, weight=w)  
  
 layout = tp + "\_layout"  
  
 attr = getattr(nx, layout)  
 if not attr:  
 print("can not find layout", tp)  
 attr = nx.circular\_layout  
 pos = attr(G)  
 if tp == 'circular':  
 pos = nx.circular\_layout(G) # 节点圆环分布  
 elif tp == 'spring':  
 pos = nx.spring\_layout(G) # 节点放射分布  
 elif tp == 'shell':  
 pos = nx.shell\_layout(G) # 节点同心圆分布，当节点较少时等同于圆环分布  
 elif tp == 'spectral':  
 pos = (nx.  
  
 spectral\_layout(G)) # 拉普拉斯特征向量分布  
 else:  
 pos = nx.circular\_layout(G) # 节点圆环分布  
  
 # 画边  
 nx.draw\_networkx\_edges(G, pos, width=[d['weight'] for (u, v, d) in G.edges(data=True)], edge\_color='green')  
 # 画点  
 if num\_dict:  
 nx.draw\_networkx\_nodes(G, pos, node\_size=[(num\_dict[node]) \* 800 for node in G.nodes()], node\_color='red')  
 nx.draw\_networkx\_nodes(G, pos, node\_size= [(num\_dict[node]) \* 200 for node in G.nodes()], with\_label=True, node\_color='red')  
 # 标记  
 nx.draw\_networkx\_labels(G, pos, font\_size=40, font\_color='black', font\_family='simhei')  
 # 画图  
 plt.rcParams['figure.figsize'] = (80.0, 40.0)  
 plt.savefig(person\_name + ".png")  
  
  
def info(name):  
 num\_dict = draw(name)  
 draw\_graph(name, num\_dict, 'spring')  
  
  
name = input('请输入演员名字：')  
info(name)





2.

import json  
import time  
from bs4 import BeautifulSoup  
import requests  
import jieba.analyse  
import jieba  
import wordcloud  
  
def get\_dm\_list\_by\_url(dm\_url\_list):  
 time.sleep(1.0)  
 dm\_list = []  
 for cid in dm\_url\_list:  
 url = f"https://api.bilibili.com/x/v1/dm/list.so?oid={cid}"  
 response = requests.get(url)  
 response.encoding = "utf8"  
 soup = BeautifulSoup(response.text, "xml")  
 for dm in soup.select("d"):  
 dm\_list.append(dm.text.strip())  
 time.sleep(1.0)  
 print(dm\_list)  
  
  
def get\_dm\_list\_by\_movie\_id(movie\_id):  
 user\_agent = ('Mozilla/5.0 (Windows NT 10.0; Win64; x64) '  
 'AppleWebKit/537.36 (KHTML, like Gecko) '  
 'Chrome/116.0.0.0 Safari/537.36 Edg/116.0.1938.69')  
 url = f"https://api.bilibili.com/x/web-interface/view/detail?byid={movie\_id}"  
 headers = {  
 'User-Agent': user\_agent  
 }  
 response = requests.get(url=url, headers=headers)  
 data = json.loads(response.text)  
  
 if "data" in data and "View" in data["data"] and "pages" in data["data"]["View"]:  
 dm\_url\_list = [page['cid'] for page in data["data"]["View"]["pages"]]  
 print(dm\_url\_list)  
 return get\_dm\_list\_by\_url(dm\_url\_list)  
 else:  
 print("無效的JSON結構")  
  
if \_\_name\_\_ == "\_\_main\_\_":  
 get\_dm\_list\_by\_movie\_id("BV1Z8411q7gy")  
  
def get\_keyword\_by\_dm(dm\_list):  
 txt = ''.join(dm\_list)  
 jieba.setLogLevel(20)  
 jieba.initialize()  
 kw\_list = jieba.analyse.extract\_tags(txt, topK=100)  
 print(kw\_list)  
 return kw\_list  
  
def gen\_wordcloud(kw\_list, png\_name):  
 string = ' '.join(kw\_list)  
 print(string)  
 wc = wordcloud.WordCloud(  
 width=1000,  
 height=700,  
 background\_color='white',  
 font\_path='msyh.ttc',  
 scale=15,  
 contour\_width=5,  
 contour\_color='red',  
 )  
 wc.generate(string)  
 wc.to\_file(png\_name)

3.

import requests  
import time  
from bs4 import BeautifulSoup  
from selenium import webdriver  
  
browser = webdriver.Chrome()  
url = "https://bj.lianjia.com/ershoufang/rs回龙观/"  
  
while True:  
 time.sleep(1.0)  
 browser.get(url)  
 soup = BeautifulSoup(browser.page\_source, "html.parser")  
  
 sell\_list = soup.select("ul.sellListContent li div.info")  
 for sell in sell\_list:  
 print(sell)  
  
 page\_list = soup.select("div.house-st-page-box a")  
 for page in page\_list:  
 print(page.text, page['href'])  
 if page.text.strip() == "下一页":  
 url = "https://bj.lianjia.com" + page['href']  
 break  
 else:  
 break

