**开发工程师测试题**

1. **编程题**

https://iftp.chinamoney.com.cn/english/bdInfo/

用程序实现：

1. 从链接页面获取公开数据

2. 需要获取数据的条件: Bond Type=Treasury Bond, Issue Year=2023

3. 解析返回表格数据，列名包括ISIN, Bond Code, Issuer, Bond Type, Issue Date, Latest Rating

4. 保存成有效csv文件

代码：

import requests  
import csv  
f=open('第一题.csv',mode='a',encoding='utf-8',newline='')  
csv\_writer=csv.DictWriter(f,fieldnames=['ISIN', 'Bond\_Code','Issuer','Bond\_Type',' Issue Date','Latest Rating'])  
csv\_writer.writeheader()  
cookies = {  
 'AlteonP10': 'CN2NZiw/F6ySRs5eVXISBA$$',  
 'apache': '4a63b086221745dd13be58c2f7de0338',  
 'ags': 'b168c5dd63e5c0bebdd4fb78b2b4704a',  
 '\_ulta\_id.ECM-Prod.ccc4': '98d17bfdfc8400d4',  
 '\_ulta\_ses.ECM-Prod.ccc4': 'eae47994ee1dac26',  
}  
  
headers = {  
 'Accept': 'application/json, text/javascript, \*/\*; q=0.01',  
 'Accept-Language': 'zh-CN,zh;q=0.9',  
 'Cache-Control': 'no-cache',  
 'Connection': 'keep-alive',  
 'Content-Type': 'application/x-www-form-urlencoded; charset=UTF-8',  
 # 'Cookie': 'AlteonP10=CN2NZiw/F6ySRs5eVXISBA$$; apache=4a63b086221745dd13be58c2f7de0338; ags=b168c5dd63e5c0bebdd4fb78b2b4704a; \_ulta\_id.ECM-Prod.ccc4=98d17bfdfc8400d4; \_ulta\_ses.ECM-Prod.ccc4=eae47994ee1dac26',  
 'Origin': 'https://iftp.chinamoney.com.cn',  
 'Pragma': 'no-cache',  
 'Referer': 'https://iftp.chinamoney.com.cn/english/bdInfo/',  
 'Sec-Fetch-Dest': 'empty',  
 'Sec-Fetch-Mode': 'cors',  
 'Sec-Fetch-Site': 'same-origin',  
 'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/109.0.0.0 Safari/537.36 SLBrowser/9.0.3.5211 SLBChan/11',  
 'X-Requested-With': 'XMLHttpRequest',  
 'sec-ch-ua': '"Chromium";v="9", "Not?A\_Brand";v="8"',  
 'sec-ch-ua-mobile': '?0',  
 'sec-ch-ua-platform': '"Windows"',  
}  
  
data = {  
 'pageNo': '1',  
 'pageSize': '15',  
 'isin': '',  
 'bondCode': '',  
 'issueEnty': '',  
 'bondType': '100001',  
 'couponType': '',  
 'issueYear': '2023',  
 'rtngShrt': '',  
 'bondSpclPrjctVrty': '',  
}  
  
response = requests.post(  
 'https://iftp.chinamoney.com.cn/ags/ms/cm-u-bond-md/BondMarketInfoListEN',  
 cookies=cookies,  
 headers=headers,  
 data=data,  
).json()  
for i in range(0,15):  
 ISIN = response['data']['resultList'][i]['isin']  
 BondCode = response['data']['resultList'][i]['bondCode']  
 Issuer = response['data']['resultList'][i]['entyFullName']  
 Bond\_Type = response['data']['resultList'][i]['bondType']  
 Issue\_Date = response['data']['resultList'][i]['issueEndDate']  
 Latest\_Rating = response['data']['resultList'][i]['debtRtng']  
 # print(ISIN, '\n', BondCode, '\n', Issuer, Bond\_Type, Issue\_Date, Latest\_Rating)  
 data = {  
 'ISIN': ISIN,  
 'BondCode': BondCode,  
 'Issuer': Issuer,  
 'Bond\_Type': Bond\_Type,  
 'Issue\_Date': Issue\_Date,  
 'Latest\_Rating': Latest\_Rating,  
 }  
 csv\_writer.writerow(data)

**二、编程题**

实现自定义正则匹配函数 reg\_search，要求：

输入参数：

1. text (需要正则匹配的文本内容)

2. regex\_list (正则表达式列表)

输出参数：

匹配到的结果列表

例如：

text = ’‘’

标的证券：本期发行的证券为可交换为发行人所持中国长江电力股份

有限公司股票（股票代码：600900.SH，股票简称：长江电力）的可交换公司债券。

换股期限：本期可交换公司债券换股期限自可交换公司债券发行结束

之日满 12 个月后的第一个交易日起至可交换债券到期日止，即 2023 年 6 月 2

日至 2027 年 6 月 1 日止。

‘’‘

regex\_list=[{

'标的证券': '\*自定义\*',

'换股期限': '\*自定义\*'

}]

返回结果：

[{

'标的证券': '600900.SH',

'换股期限': ['2023-06-02', '2027-06-01']

}]

代码：

import re  
  
def reg\_search(text, regex\_list):  
 results = []  
  
 for regex\_dict in regex\_list:  
 results=[]  
 for key, pattern in regex\_dict.items():  
 if key == '标的证券':  
 # 匹配股票代码的正则表达式  
 match = re.search(pattern,text)  
 if match:  
 results.append({key: match.group(1)})  
 if key == '换股期限':  
 # 匹配日期的正则表达式  
 match = re.findall(pattern,text)  
 if match:  
 list=[]  
 for mat in match:  
 mat=mat.replace(' 年','-').replace(' 月','-').replace(' 日','')  
 mat=mat.replace(' ','0')  
 list.append(mat)  
 results[0]['换股期限'] =list  
  
 return results  
text = '''  
标的证券：本期发行的证券为可交换为发行人所持中国长江电力股份有限公司股票（股票代码：600900.SH，股票简称：长江电力）的可交换公司债券。  
换股期限：本期可交换公司债券换股期限自可交换债券发行结束之日满 12 个月后的第一个交易日起至可交换债券到期日止，即 2023 年 6 月 2 日至 2027 年 6 月 1 日止。  
'''  
regex\_list = [  
 {  
 '标的证券': r'股票代码：(\w+\.\w+)',  
 '换股期限': r' (\d{4} 年 \d{1,2} 月 \d{1,2} 日)'  
 }  
]  
  
print(reg\_search(text, regex\_list))

**注意：**

测试限时2个小时，完成后，请把代码以及相关说明文档提交到github或gitee，并发邮件recruitment@orbitfin.ai

邮件标题：**个人姓名**

邮件内容包括：

**1、个人简历；**

**2、GitHub、gitee或者gitlab网址，账号权限设置为公开；**

**3、到岗时间；**

**4、测试时间：例13：00——14：45.**

谢谢您参与测试，辛苦啦，测试结果待技术审核后反馈。