用Python定时爬取微博热搜

一、应用场景

当今的互联网时代，大部分人的生活都离开不了‘两微一抖’，在平时的新媒体运营工作中，会经常收集网上的重点新闻来作为素材，但因为网页的新闻更新频率较高，因此在工作中会花比较多的时间去收集素材。

接下来我们使用Python，以微博热搜为例子，每隔两小时自动的爬取并存储热搜榜的新闻信息，减少重复性的收集素材的操作。

项目代码

```python
import requests,datetime,openpyxl, time,schedule
from bs4 import BeautifulSoup

# 爬取微博热榜新闻标题，链接，热度
def weiboSpider():
    destnation_url = 'https://s.weibo.com/top/summary'
    destnation = requests.get (destnation_url)
    soup = BeautifulSoup(destnation.text,'html.parser')
    data_list = soup.find('div', class_='data').find_all('td',class_='td-02')
    content_lists = []
    for data in data_list:
        title = data.find('a').text.strip()
        title_href = data.find('a')['href']
        href = 'https://s.weibo.com' + title_href
        try:
            heat = data.find('span').text

        except AttributeError:
            heat = '暂无'
        content_lists.append([title, href, heat])
    return content_lists

# 写入Excel文件
def writeExcel(content_lists):
```

```
    date = datetime.datetime.now().strftime('%m-%d')    # 获取当天时间并转化为“月-日”格式
    # 打开Excel文件，如果不存在则新建Excel文件
    try:
        wb = openpyxl.load_workbook("微博热榜.xlsx")
        sheets = wb.sheetnames
        # 以当天日期为工作表名称，判断工作表名称是否存在，不存在则新建
        if date in sheets:
            sheet = wb[date]    # 工作表存在，读取该工作表
        else:
            sheet = wb.create_sheet(date,0) # 工作表不存在，则新建，0表示插入工作表在第0个位置
            sheet.append(['标题', '链接', '热度'])

    except:
        wb = openpyxl.Workbook()
        sheet = wb.active
        sheet.title = date
        sheet.append(['标题', '链接', '热度'])
    # 遍历爬取数据，并写入Excel文件
    for content_list in content_lists:
        sheet.append(content_list)
    wb.save("微博热榜.xlsx")

# 调用爬虫和写入Excel文件
def main():
    content_lists = weiboSpider()
    writeExcel(content_lists)
    print('成功爬取微博热榜，并写入Excel！ ')

# 执行定时爬取
schedule.every(2).hour.do(main)  # 每隔2小时执行一次，修改数字可更改定时时间
while True:
    schedule.run_pending()      # 检测并运行设定好的所有定时任务
    time.sleep(1)
```

三、项目操作及解析

该代码在运行前安装好 requests模块，BeautifulSoup4模块，openpyxl模块, schedule模块。在运行代码之后，计算机要处于运行状态，才能执行定时爬取任务。

3.1 爬取热搜新闻信息

```
# 该代码块不能单独运行
for data in data_list:
    title = data.find('a').text.strip()
    title_href = data.find('a')['href']
    href = 'https://s.weibo.com' + title_href
    try:
        heat = data.find('span').text

    except AttributeError:
        heat = '暂无'
    content_lists.append([title, href, heat])
```

代码解析：

1. 第5行代码与第8行代码，主要是用来捕获没有热度信息的新闻。根据观察，热搜榜中，唯一没有热度信息的是第一条新闻，而其它新闻都有热度信息。
2. 第10行代码，主要是通过content_lists来存放爬取下来的所有新闻信息，用于后续的excel文件的写入。

3.2 写入Excel文件

```
# 该代码块不能单独运行
date = datetime.datetime.now().strftime('%m-%d')
try:
    wb = openpyxl.load_workbook("微博热榜.xlsx")
    sheets = wb.sheetnames
    # 以当天日期为工作表名称，判断工作表名称是否存在，不存在则新建
    if date in sheets:
        sheet = wb[date]    # 工作表存在，读取该工作表
    else:
        sheet = wb.create_sheet(date,0) # 工作表不存在，则新建，0表示插入工作表在第0个位置
        sheet.append(['标题', '链接', '热度'])
except:
    wb = openpyxl.Workbook()
    sheet = wb.active
    sheet.title = date
    sheet.append(['标题', '链接', '热度'])
    # 遍历爬取数据，并写入Excel文件
```

```
    for content_list in content_lists:
        sheet.append(content_list)
wb.save("微博热榜.xlsx")
```

代码解析：

1. 第2行代码与第8行代码中，try...except语句主要是检测是否已经存在Excel文件。
2. 第6行代码与第8行代码中，if....else语句主要是检测是否已经存在当天日期的工作表，当存在时，会直接将爬取的热搜信心写入到对应的工作表中。若不存在时，会自动生成一个新的工作表，并已当天的日期为表名，再将爬取的热搜信息写入到新的工作表中。

3.3 定时执行任务

```
# 该代码块不能单独运行
schedule.every(2).hour.do(main)  # 每隔2小时执行一次，修改数字可更改定时时间
while True:
    schedule.run_pending()      # 检测并运行设定好的所有定时任务
    time.sleep(1)
```

代码解析：

1. 第2行代码，主要的作用是规定每2个小时，执行一遍main()函数。
2. 第4行代码，主要作用是启动定时任务。