## 设计目的

大学生最关心的莫过于就业，高校的数字化建设日渐成熟，大数据在就业方面的应用正处在积极探索阶段。可以说数据采集是大数据应用开发的一个重要环节，如果没有数据，处理分析也就无从谈起。本次设计目标就是爬取百姓网的招聘信息。

## 系统的需求及其基本功能

就业信息主要来自网络和学校招聘，主要的岗位数据源于网络，需要有一套采集软件也就是爬虫。定制基于python的百姓网招聘信息爬虫系统。列出爬虫抓取的起始页。可以通过添加后续链接的URL至爬取队列，使得爬虫按照要求一步步循环下去。然后分析爬取回来的数据，得到招聘信息，并保存至本地。

## 总体设计方案

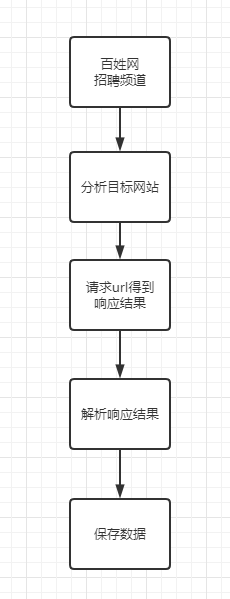
### 分析目标网站

### 生成网站请求链接队列

### 提取每条招聘信息的标题、工资、工作内容、福利等

### 把得到数据保存到excel表中

### 流程图



## 详细设计项目源程序代码

import requests

from lxml import etree

import openpyxl

import time

import pymysql

headers = {

# 'cookie': '\_\_trackId=162393371346049; \_\_city=lanzhou; \_auth\_redirect=https%3A%2F%2Flanzhou.baixing.com%2Fgongzuo%2F%3Fpage%3D1',

'referer': 'https://lanzhou.baixing.com/gongzuo/?page=1',

'user-agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/90.0.4430.72 Safari/537.3',

}

url = 'https://lanzhou.baixing.com/gongzuo/?page={}'

data = []

db = pymysql.connect(user="root", password="123456", host="localhost", database="spider")

cursor = db.cursor()

# 内容提取

def parser\_itme(li):

title = li.xpath('./div[contains(@class,"table-view-body job-list")]/div/a[@class="ad-title"]/text()')[0]

salary = li.xpath('./div[contains(@class,"table-view-body job-list")]/span/text()')[0]

job\_list = \

li.xpath('./div[contains(@class,"table-view-body -small")]/div[contains(@class,"table-view-cap job-list")]')[0]

job = job\_list.xpath("string(.)")

wellfares\_list = li.xpath(

'./div[contains(@class,"table-view-body -small")]/div[contains(@class,"wellfares-badges")]/label/text()')

wellfares = '/'.join(wellfares\_list)

print(title)

cursor.execute(

"INSERT INTO `baixing` (`title`, `salary`, `job`, `wellfares`) VALUES ('%s', '%s', '%s', '%s');" % (

title, salary, job, wellfares))

db.commit()

# 索引页分析

def parser(url):

response = requests.get(url)

html = etree.HTML(response.text)

li\_list = html.xpath('//li[contains(@class,"listing-ad table-view-item apply-item item-regular")]')

for li in li\_list:

parser\_itme(li)

# 保存表格

def save():

book = openpyxl.Workbook()

sheet = book.active

sheet.append(['序号', '标题', '工资', '工作内容', '福利'])

cursor.execute('select \* from baixing')

results = cursor.fetchall()

for li in results:

sheet.append(li)

book.save('百姓网.xlsx')

# 主程序

def main():

for n in range(1, 51):

parser(url.format(n))

print(n)

time.sleep(30)

save()

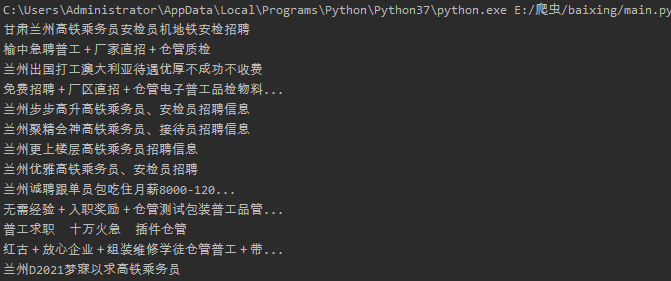
db.close()

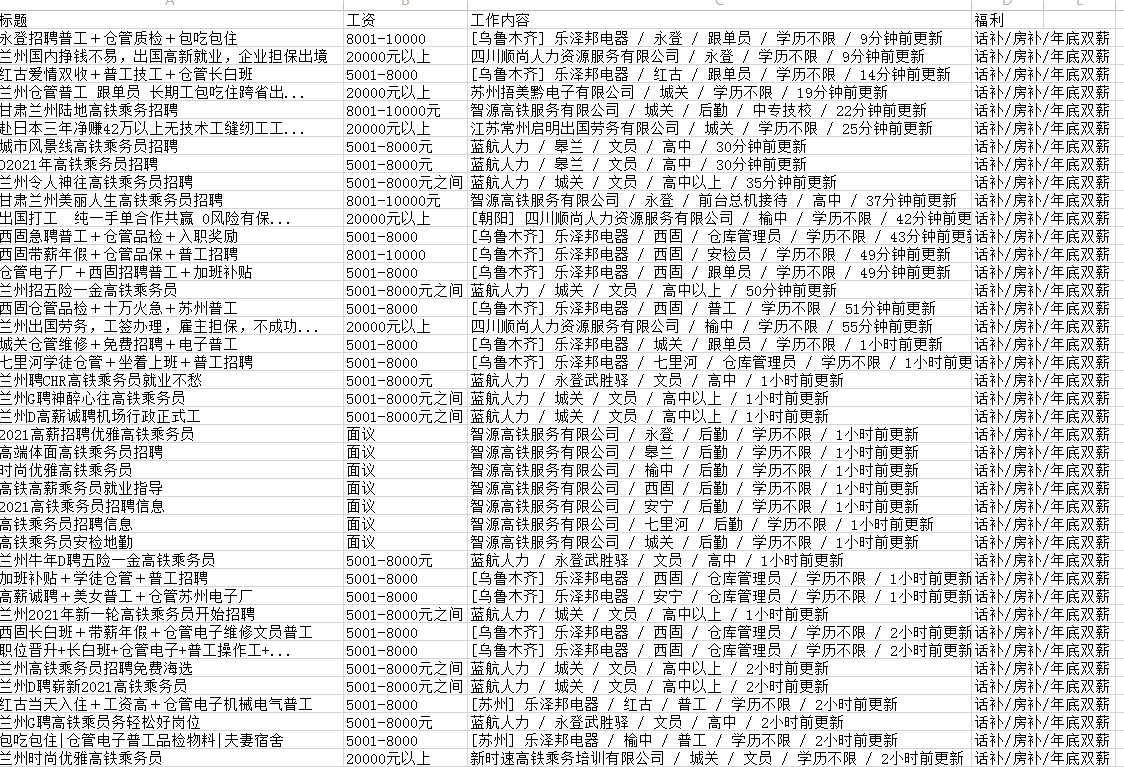
if \_\_name\_\_ == "\_\_main\_\_":

main()

## 运行与测试

运行结果如下：





## 成员分工

## 总结与收获

此次课程设计的题目为网页爬虫，通过课程设计，进一步巩固、加深我们所学专业课程《数据可视化技术》的基本理论知识，理论联系实际，进一步培养我们的综合分析问题，解决问题的能力；全面考核我们所掌握的基本理论知识及其实际业务能力，从而达到提高学生素质的最终目的；利用所学知识，开发小型应用系统，掌握运用python语言编写调试应用系统程序，训练独立开发应用系统，进行数据处理的综合能力；对于给定的设计题目，如何进行分析，理清思路，并给出相应的数学模型；熟练掌握python语言的设计、线程技术，灵活运用各种类库，为今后从事实际工作打下坚实的基础。通过该课程设计，全面系统的理解了python语言的一般原理和基本实现方法。把死板的课本知识变得生动有趣，激发了学习的积极性。把学过的python的知识强化，能够把课堂上学的知识通过自己设计的程序表示出来，加深了对理论知识的理解。

## 指导教师评语