**XXXX大学**

**XXXX学年度第X学期**

Python网络爬虫

实训报告

学 院：

专业班级：

姓　　名：

学　　号：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 一级指标 | 二级指标 | 三级指标及分值 | | 分值 | 评分及成绩 |
| 得分 |
| 代码  （占60%） | 1.达到基本要求，功能点完整（55%） | 网页抓取 | 10 | 55 |  |
| 页面解析 | 10 |
| 数据意义 | 15 |
| 数据存储 | 20 |
| 4.运行测试（5%） | | | 5 |  |
| 项目报告（20%） | 1.内容与结构完整（10%） | | | 10 |  |
| 2.规格满足要求，文字描述规范，思路描述清晰（10%） | | | 10 |  |
| 答辩成绩 （20%） | 1.对实训设计技术和结构熟悉，能进行合理说明（10%） | | | 10 |  |
| 2.能较好的回答答辩所提问题，解释合理清晰（10%） | | | 10 |  |
| 合计（百分制） | | | | |  |

**目录**

[1. 爬取主体说明 1](#_Toc90415066)

[1.1 背景 1](#_Toc90415067)

[1.2 数据来源 1](#_Toc90415068)

[1.3 数据项 1](#_Toc90415069)

[2. 技术可行性分析 2](#_Toc90415070)

[2.1 技术方案 2](#_Toc90415071)

[2.2 相关依赖包说明 2](#_Toc90415072)

[3. 详细设计及实现 3](#_Toc90415073)

[3.1 抓取部分 3](#_Toc90415074)

[3.2 解析部分 5](#_Toc90415075)

[3.3 存储部分 5](#_Toc90415076)

[4. 运行测试 7](#_Toc90415077)

[5. 总结 9](#_Toc90415078)

# 1. 爬取主题说明

## 1.1 背景

在这个互联网的时代下，信息的传播不再困难，人们可以通过上网获取各种各样的信息，同时娱乐方式也在不断丰富，通过互联网，人们可以随时随地观看动画、电影、电视剧、漫画、小说，聆听音乐，玩游戏等等。在此背景下，ACG文化不断发展壮大，且备受年轻人的喜爱，各种各样的动画、漫画、游戏层出不穷，一个人看过的动画、漫画和玩过游戏，少则几部多则成百上千部，在此情况下，对于一部分喜欢记录的人来说，由其希望能有一个用于记录自己看过的动画、漫画和玩过的游戏等的工具，而Bangumi番组计划正是能实现该需求的一个网站。

## 1.2 数据来源

Bangumi官网介绍是：“一个专注于动漫、音乐、游戏领域，帮助你分享、发现与结识同好的ACG网络。”我们可以在此记录自己看过的动画和书籍、玩过的游戏、听过的音乐等等，其独特的收视进度管理功能是该网站的一大特点，我们还可以在此与同好一起交流讨论，分享各自的评论，寻觅趣味相投的伙伴。

本人使用此网站已有三年之多，记录了自己看过的所有动漫，本次Python课程设计要求写一个网络爬虫程序爬取一个网站，我选择的网站是Bangumi-我看过的动画：<https://bangumi.tv/anime/list/430090/collect>，共有七个网页。

## 1.3 数据项

网络爬虫是一种按照一定的规则，自动地抓取万维网信息的程序或脚本。爬虫最重要的是解析爬取到的数据，从中提取有用的信息，本程序将所有我看过的动画一部部的爬取出来，并从爬取到的数据中解析出了12种有用信息，即12个数据项，并存放至MySQL数据库和Excel表格中。这12个数据项分别是：id、chinese\_name中文名、original\_name原名、episodes话数、broadcast\_time放送时间、people制作人员、star\_time收藏时间、score个人评分、tag个人标签、URL网页网址、low\_image\_URL缩略封面图网址、high\_image\_URL高清封面图网址，此外还将缩略封面图和高清封面图爬取下载到本地，并将其文件路径一并保存下来，所有总计有14个数据项。

# 2. 技术可行性分析

## 2.1 技术方案

本程序的数据抓取部分使用了urllib模块；数据解析部分使用了BeautifulSoup和re模块；数据存储部分使用了pymysql和xlwt模块。

## 2.2 相关依赖包说明

1. urllib模块：urllib是Python标准库中最为常用的Python网页访问模块，它可以使用户像访问本地文本文件一样读取网页的内容，其提供了一个访问网页的简单易懂的API接口，我们可以使用它操作网页URL，并对网页的内容进行抓取处理。
2. BeautifulSoup模块：BeautifulSoup是Python中一个可以处理HTML/XML的函数库，是一个网页分析工具，使用它可以从HTML或XML文件中提取数据，用来快速转换被抓取的网页，它产生一个转换后的DOM树，其内容尽可能和原文档内容含义一致，我们可以使用它来解析抓取到的数据。
3. re模块：re即正则表达式，re模块是Python内置的一个用于处理正则表达式的模块，它使Python语言拥有了全部的正则表达式功能，使用此模块我们可以从爬取到的字符串数据中方便的检查一个字符串是否与某种模式匹配，并将其有用信息提取出来。
4. pymysql模块：pymysql是Python中用于操作MySQL数据库的一个模块，我们可以使用它来对MySQL数据库中的数据进行增删改查等各种操作。
5. xlwt模块：xlwt是Python中用于操作Excel表的一个模块，但其仅能用来写xls格式的文件，使用此模块我们可以十分方便地将数据保存至Excel表中。

# 3. 详细设计及实现

本程序可分为三大部分：抓取部分、解析部分、存储部分，分别完成数据的抓取、解析和存储，共编写了8个函数，如下：

1. main函数：程序的入口。
2. getHTML函数：用于获取HTML页面内容，即获取网页的HTML代码。
3. saveHTML函数：用于将HTML页面保存至本地。
4. getPageCount函数：用于获取要爬取的页面总数。
5. changeLink函数：用于将网页中的链接由相对路径改为绝对路径，或者修改不符合要求的链接。
6. getData函数：用于解析爬取到的网页数据。
7. saveDataToDatabase函数：用于将数据保存至MySQL数据库。
8. saveDataToExcel函数：用于将数据保存至Excel表。

## 3.1 抓取部分

此部分用于完成数据抓取，主要涉及的函数有getHTML、saveHTML、getPageCount、changeLink、getData。

首先完成准备工作，调用getPageCount函数获取要爬取的页面总数，该函数通过分析网页的HTML代码的相关部分，从而解析出共有多少个页面需要爬取，关键代码如图3-1。

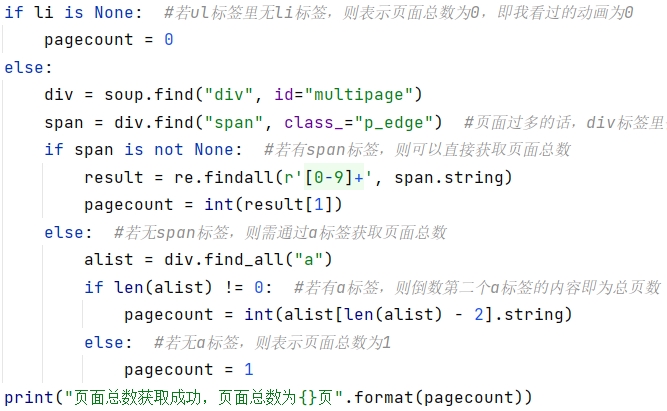


图3-1 爬取部分-getPageCount

接着在getData函数里，根据上述获取的页面总数，一个一个的爬取网页，获取每个HTML页面的内容，并修改相关链接，再保存HTML页面，不同页面的链接即是在URL地址后方加上?page=参数，关键代码如图3-2。

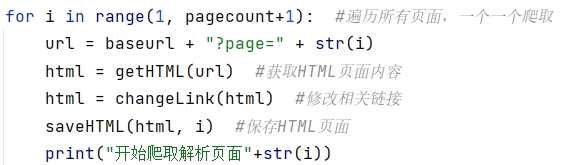


图3-2 爬取部分-getData

其中getHTML函数用于获取HTML页面内容，使用到了User-Agent即用户代理，用来模拟浏览器头部信息，以防止网站的反爬虫导致本程序无法爬取数据，本函数首先创建Request对象，并传入headers，再使用urlopen打开网站获取返回的response对象，该对象中存放了网站的所有信息，接着调用response对象的read方法获取其中的信息，并解码保存起来，代码如图3-3。

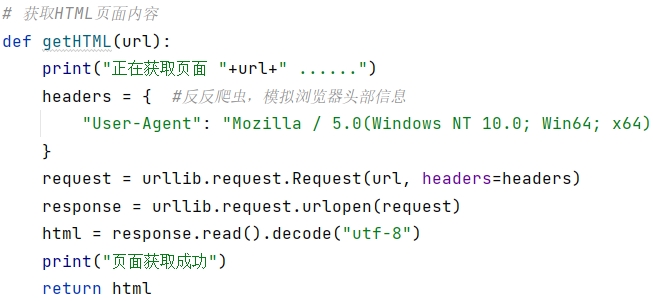


图3-3 爬取部分-getHTML

changeLink函数用于修改不符合要求的链接，由于网站的大部分链接使用相对路径，而相对路径在爬取并保存下来的HTML页面中无法打开，故需用此函数将相对路径修改为绝对路径，如图3-4中对a标签的链接即href属性进行修改。

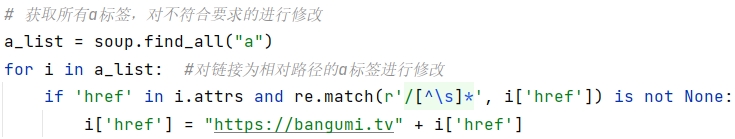


图3-4 爬取部分-changeLink

saveHTML函数用于保存HTML页面至本地，使用open函数打开/建立html文件，再使用write函数将前面获取的HTML页面内容即代码写入文件中，代码如图3-5。



图3-5 爬取部分-saveHTML

## 3.2 解析部分

此部分用于完成数据解析，主要在getData函数中进行。

首先用前面获得并修改过的html内容来创建BeautifulSoup对象，然后分析网站的HTML源代码，并找出其中特定的规律，并按此规律逐一解析BeautifulSoup对象中的数据，获取想要的信息。部分代码如图3-6和图3-7，此部分内容较多且复杂，详细请见程序代码及注释。

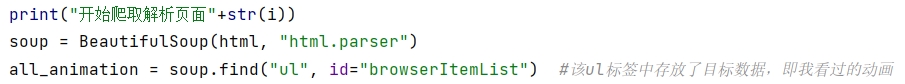


图3-6 解析部分

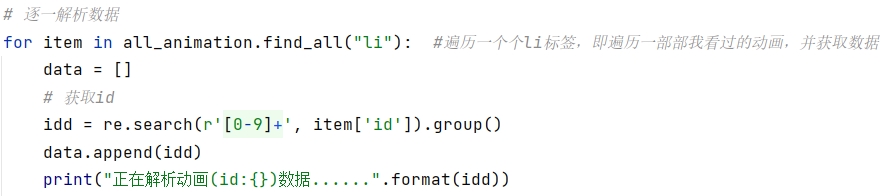


图3-7 解析部分

## 3.3 存储部分

此部分用于完成数据存储，主要涉及的函数有saveDataToDatabase和saveDataToExcel。

前面解析部分中获取到的有用信息存储在一个二维列表datalist中，将此datalist分别传入saveDataToDatabase函数和saveDataToExcel函数中，以将爬取到的有用信息保存至MySQL数据库和Excel表中，关键代码如图3-8。



图3-8 存储部分

saveDataToDatabase函数代码如图3-9。



图3-9 存储部分-MySQL数据库

saveDataToExcel函数代码如图3-10。



图3-10 存储部分-Excel表格

# 4. 运行测试

在程序的每个关键步骤中都插入了一条print语句，以便于在运行程序时能准确的知道运行至哪里，此外还便于排查bug，程序运行输出如图4-1，从运行结果中可以知道每一步都正常运行，下面展示程序运行后所得到的各种结果。



图4-1 运行结果

程序运行后可得到所有的HTML页面，如图4-2。



图4-2 所有HTML页面

可得到所有我看过的动画封面图，如图4-3。

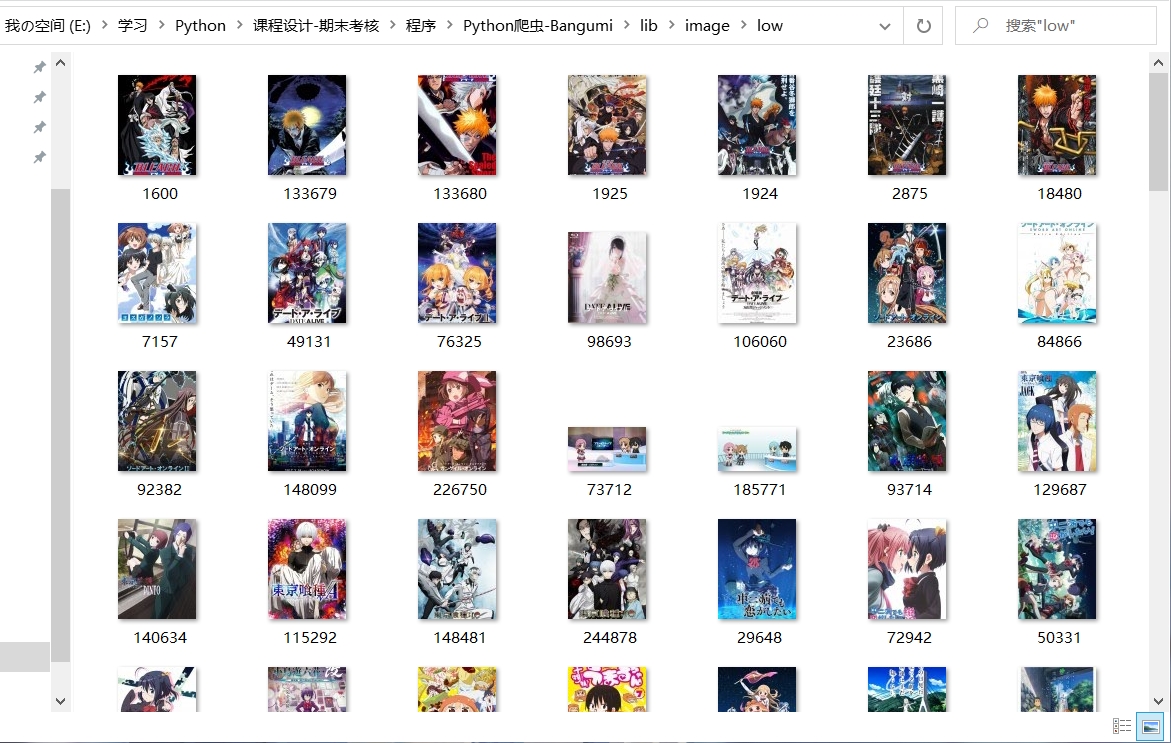


图4-3 所有封面图

可得到所有我看过的动画数据并保存在MySQL数据库中，如图4-4。



图4-4 所有数据-MySQL数据库

可得到所有我看过的动画数据并保存在Excel表中，如图4-5。



图4-5 所有数据-Excel表

此外本人还编写了一个Java Web页面用于直观的展示爬取到的所有数据，如图4-6。



图4-6 直观展示数据

# 5. 总结

本学期学习完Python后，期末课程设计要求写一个网络爬虫程序爬取一个网站，在此之前我只听说过爬虫，但是并未深入了解过，也从没写过爬虫程序。通过本次课程设计，我了解了Python爬虫，掌握了使用Python编写爬虫程序的相关知识，同时也被爬虫这神奇的技术深深吸引。

为了编写出此程序，翻查了课本、查阅了各种网上资料，一步一步的弄懂了所有内容。一个爬虫程序一般可分为三步：爬取、解析、存储，爬取主要是将网页的HTML代码抓取下来，解析主要是对抓取下来的HTML代码进行分析，从中提取有用的、自己想要的信息，存储主要是将获取的信息保存起来。按照这三个步骤一步步来编写程序就很简单了。

在编写程序的过程中也遇到了许多的问题，例如：如何从多且繁杂的HTML代码中准确提取自己想要的信息等等，这些问题在查阅资料以及自己的思考下，一个个的都解决了，这也是学习的一个过程。

本次Python课程设计，本人收获良多，既巩固了Python知识也掌握了爬虫等新知识，程序编写完成并成功运行后，内心的自豪感及满足感油然而生。。。