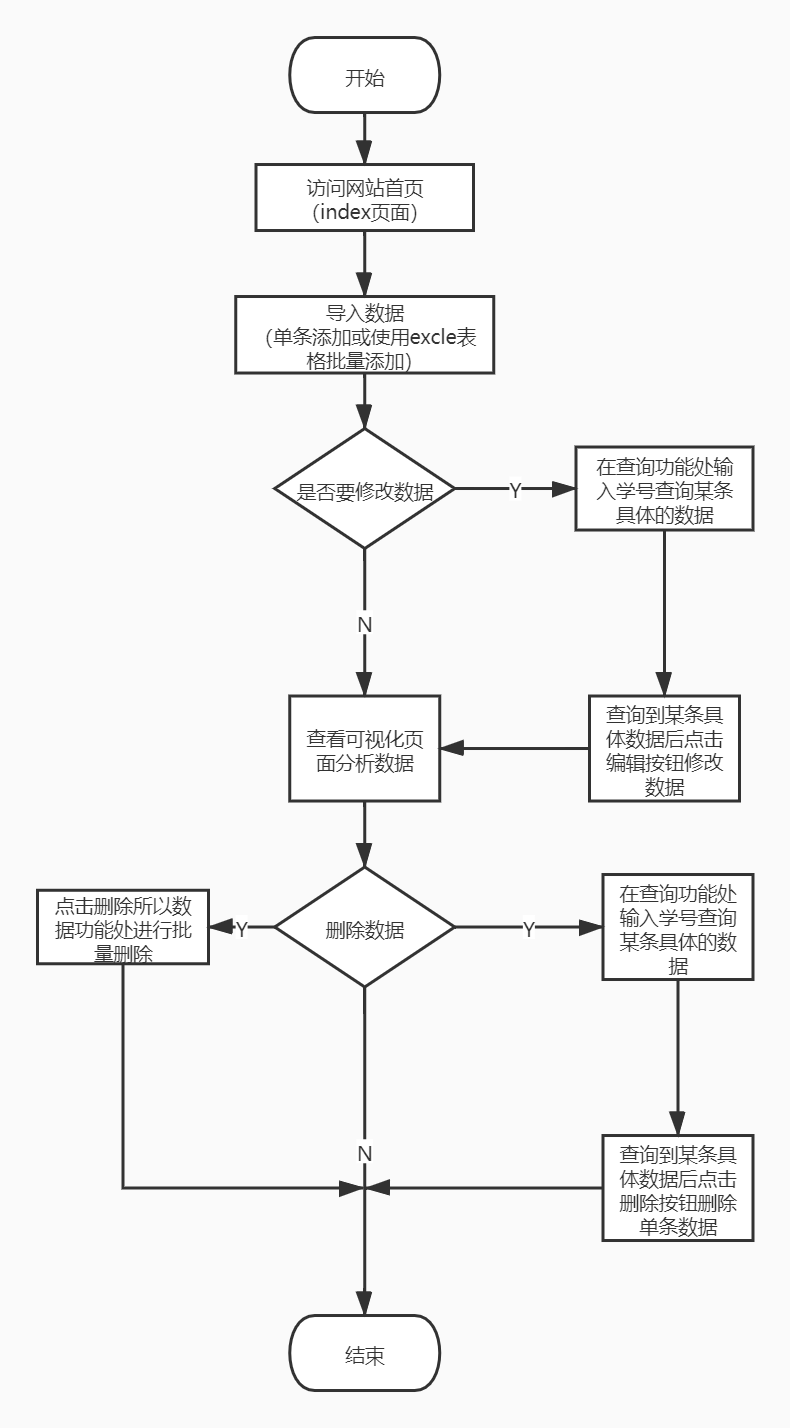
# 一、系统介绍

## 1、系统使用流程图



## python环境

在本系统中使用的python模块及其版本需要在这里介绍一下

1. pymysql

python在python2中连接 MySQL 服务器使用的是mysqldb，但在python3中连接连接 MySQL 服务器使用的是pymysql。本系统使用的python版本是pyhton3故安装pymysql，使用命令pip install pymysql。

1. xlrd

python操作excel主要用到xlrd和xlwt这两个库，即xlrd是读excel，xlwt是写excel的库,本系统的批量录入学生信息功能是使用excle表格导入的，故我们需要安装xlrd模块。

（3）目前最新的xlrd只支持.xls文件。若是我们.xlsx后缀的excle表格，pandas.read\_excel(‘xxx.xlsx’)会报错：xlrd.biffh.XLRDError: Excel xlsx file; not supported，本系统需要使用的就是1.2.0版本的xlrd，该版本可以正常导入.xlsx后缀的excle表格，安装1.2.0版本的xlrd模块命令为pip install xlrd==1.2.0。同时这里需要注意的是，待上传的表格需要放在网站的根目录下，如：C:\Users\Administrator\Desktop\VSS。

指定安装1.9.0版本的pyecharts：pip3 install pyecharts==1.9.0

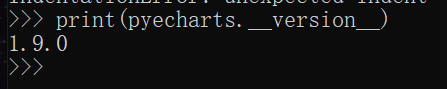


图4.4 pyecharts版本图

以bar图表为例，导包代码为：from pyecharts.charts import Bar

Pyecharts的github地址：<https://github.com/pyecharts/pyecharts>

Ptecharts的官方文档地址：<https://pyecharts.org/#/>

# **二、如何运行系统**

## 1、配置数据库

数据库的配置

先搭建好数据库环境，搭建好数据库环境后我们需要先创建一个我们想要的数据库，以我的为例，我将Python数据可视化的毕业生就业统计管理系统的数据库命名为students,接着我们需要到settings.py中进行配置，如下图所示，在DATABASES={}中需配置数据库的账号密码、主机ip、端口号以及该系统的数据库名等。

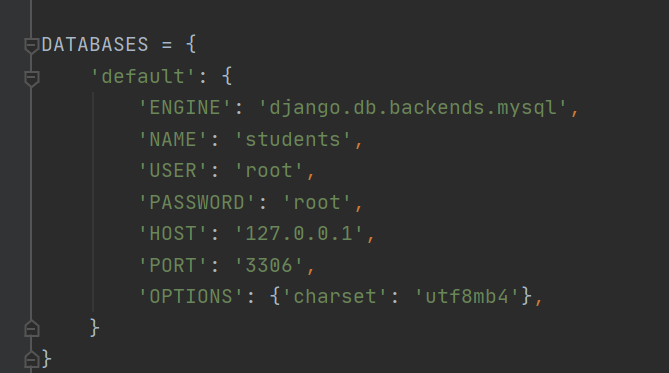


图4.3 settings.py配置图

## 2、配置django

**（1）配置models.py并对数据进行迁移**

models.py是django与数据库之间的桥梁，model封装与定义了信息。它包含你想要存储的字段和操作数据的方法，每个模型都映射了一张数据表，为Django提供了一个访问数据库的API。每个model都是继承于django.db.models.Model 的Python类，model的一个类对应数据库的一个表，一个属性，对应数据表中的一个字段。

如下图所示为本系统中models.py的配置

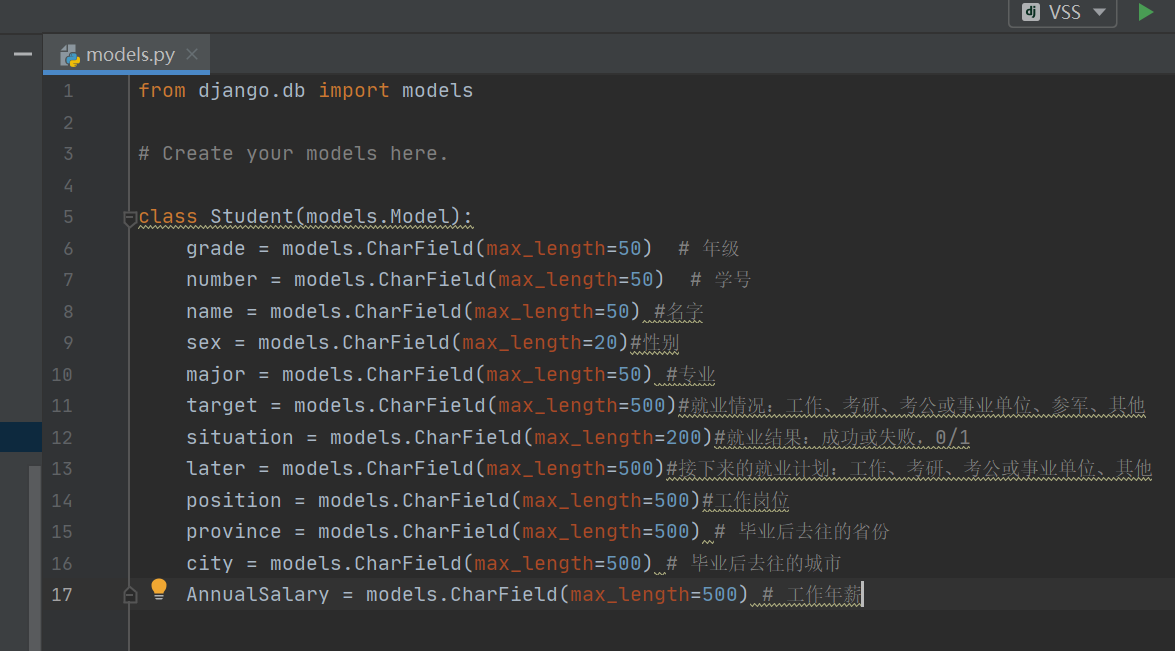


图4.2 models.py配置图

models.py是已经写好的，我们只需要对数据进行迁移，迁移命令如下：

生成迁移文件：（根据模型类生成sql语句）

python manage.py makemigrations

执行迁移：（执行sql语句生成数据表）

python manage.py migrate

**（2）iframe框架的配置**

（代码里我已经配置好了，不用你配置，这里只是说明一下）

运行pyechart代码后会生成可视化的页面或图片，在网站上展示这些可视化页面有多种方法，本系统采用的方式是使用ifram框架。

实现过程我们以条形统计图为例进行解说

调用bar( )函数，其中.render( )将生成的图形化页面指定位置在C:\Users\Administrator\Desktop\VSS\vsapp\templates\visual\bar\_is\_selected.html，然后在barVisval.html页面中使用代码<iframe frameborder="0" width="70%" height="600px" src="/bar\_is\_selected"></iframe>，即可将bar\_is\_selected.html可视化页面嵌套进barVisval.html 页面进行展示，具体流程见下方截图

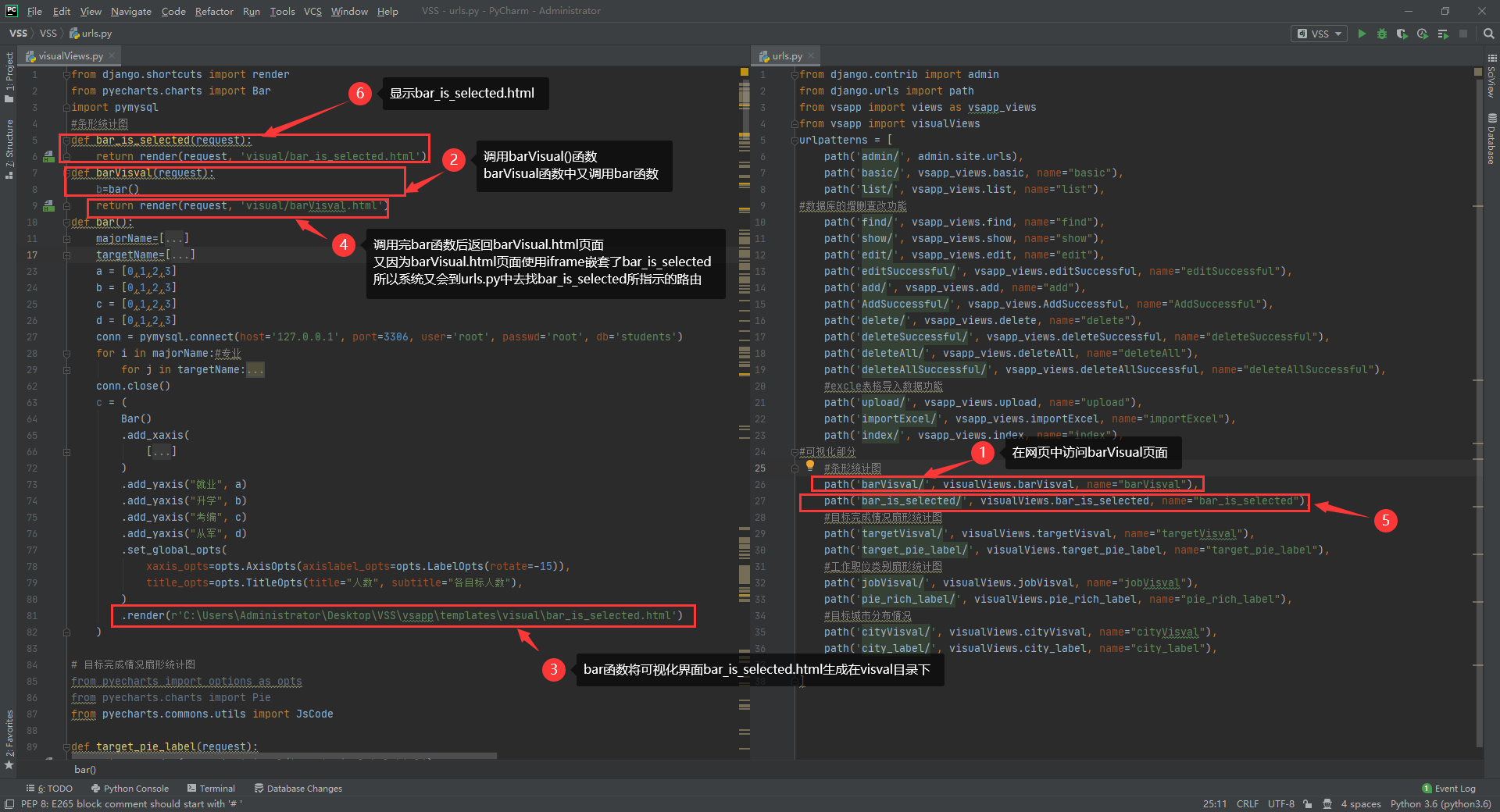


图4.2 models.py配置图

django中默认是不使用iframe标签的，所以我们需要在django中进行配置，如下截图所示在global\_settings.py中要修改代码为将X\_FRAME\_OPTIONS = 'DENY'X-Frame-Options修改为X\_FRAME\_OPTIONS = 'SAMEORIGIN'，只有开启X-Frame-Options，使用django框架开发的网站才能用iframe标签。

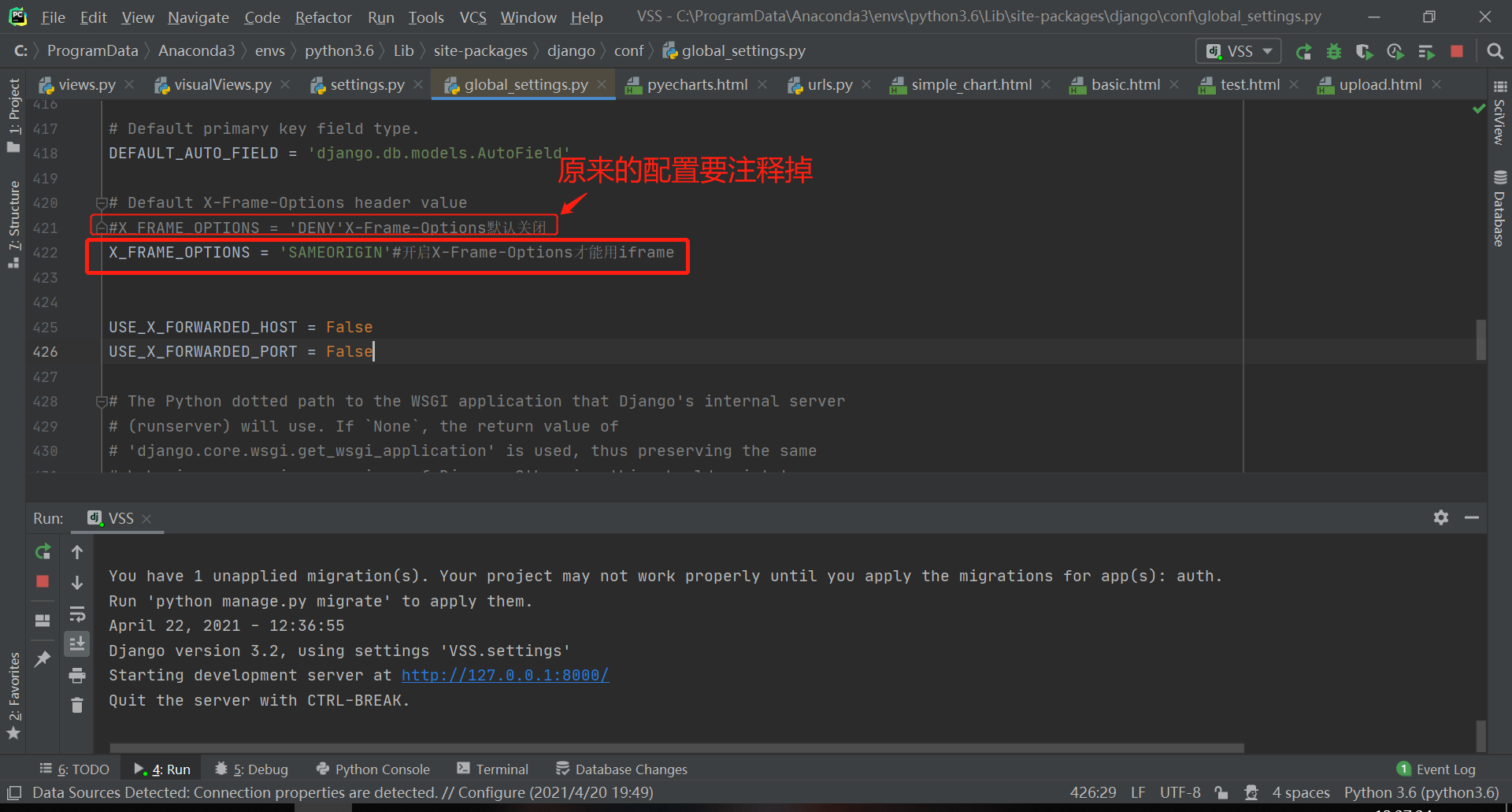
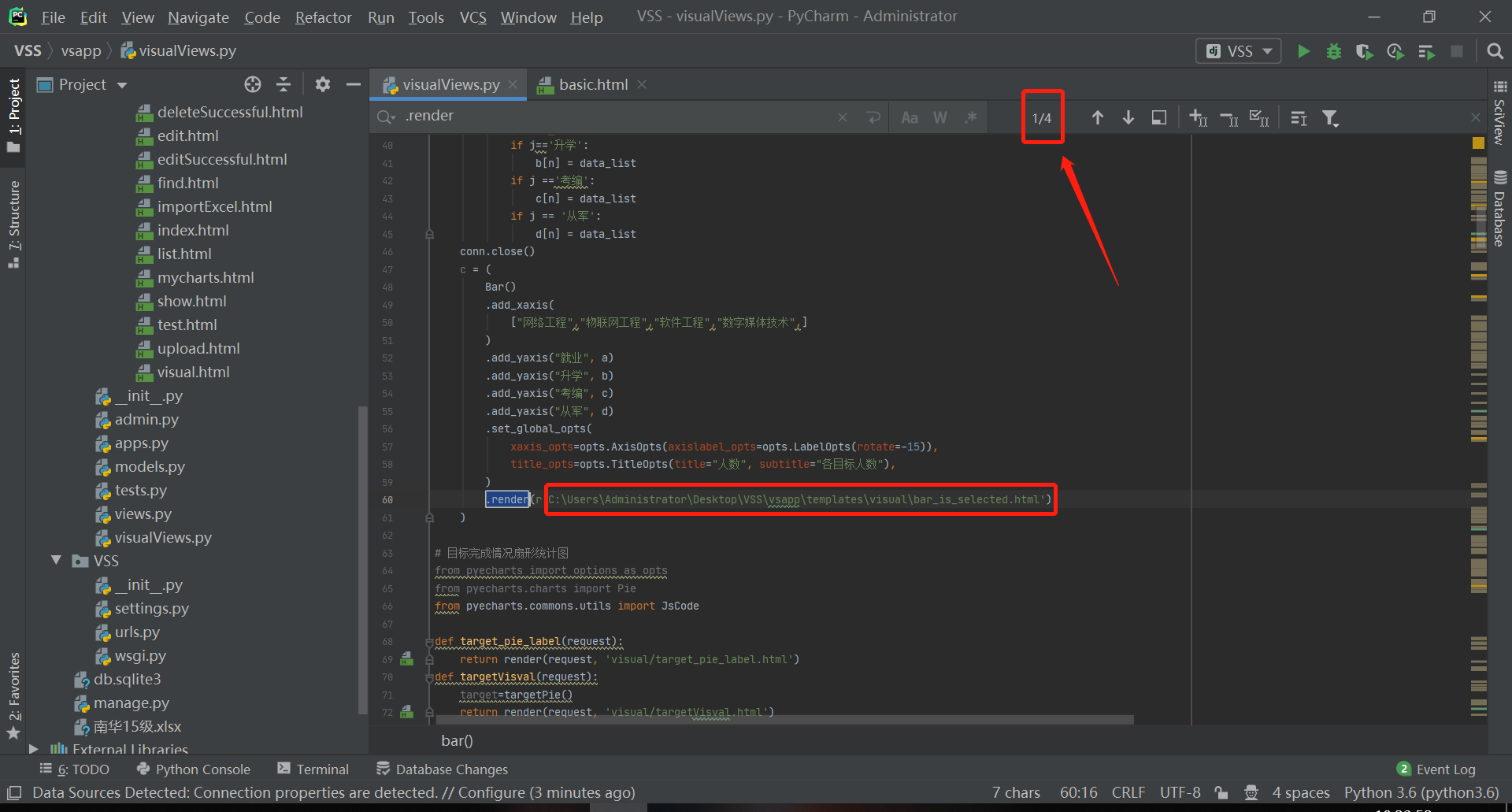


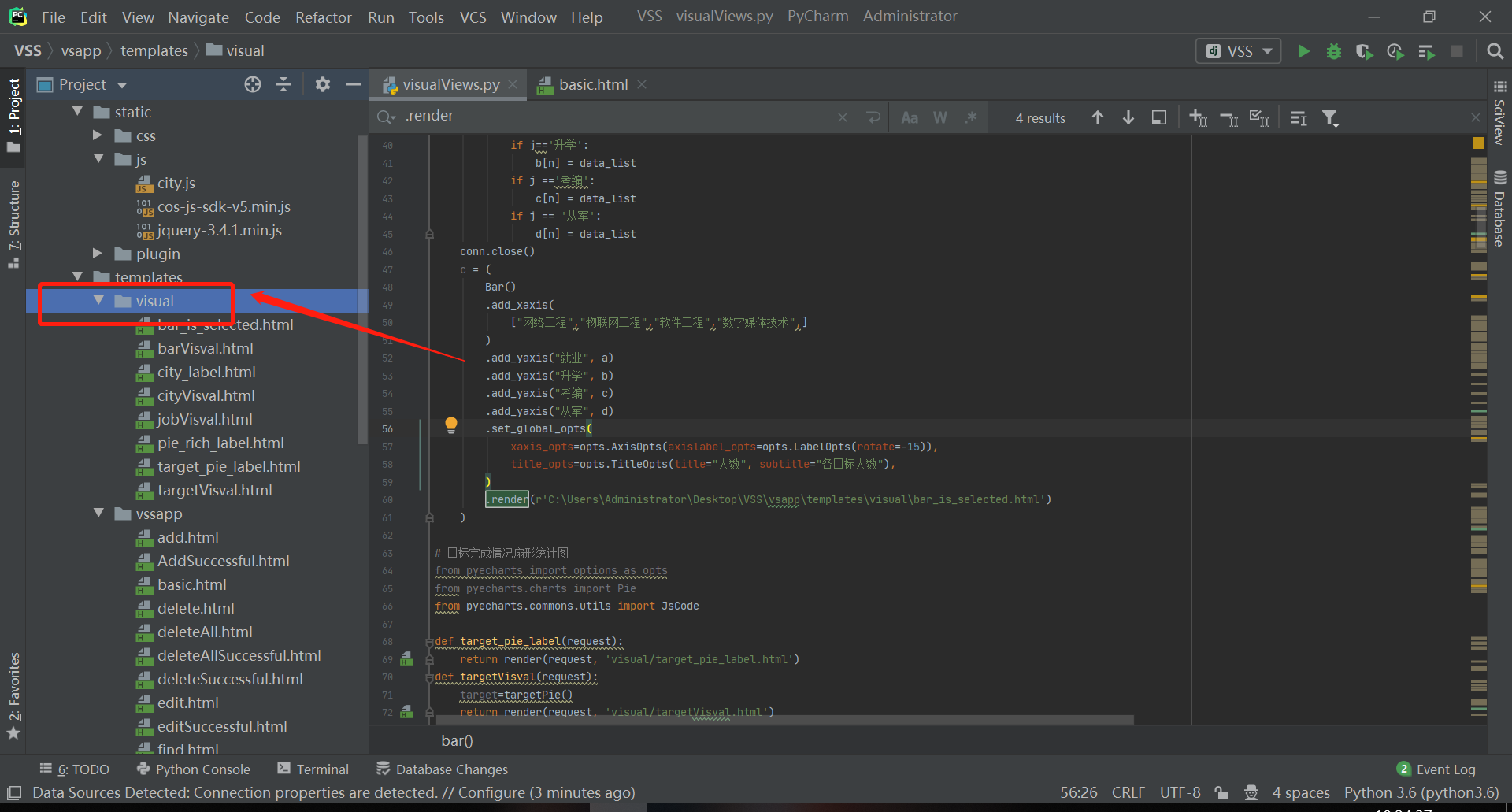
图4.17批量删除数据成功提示页面图

1. **修改.render()中指定的生成可视化页面的地址**

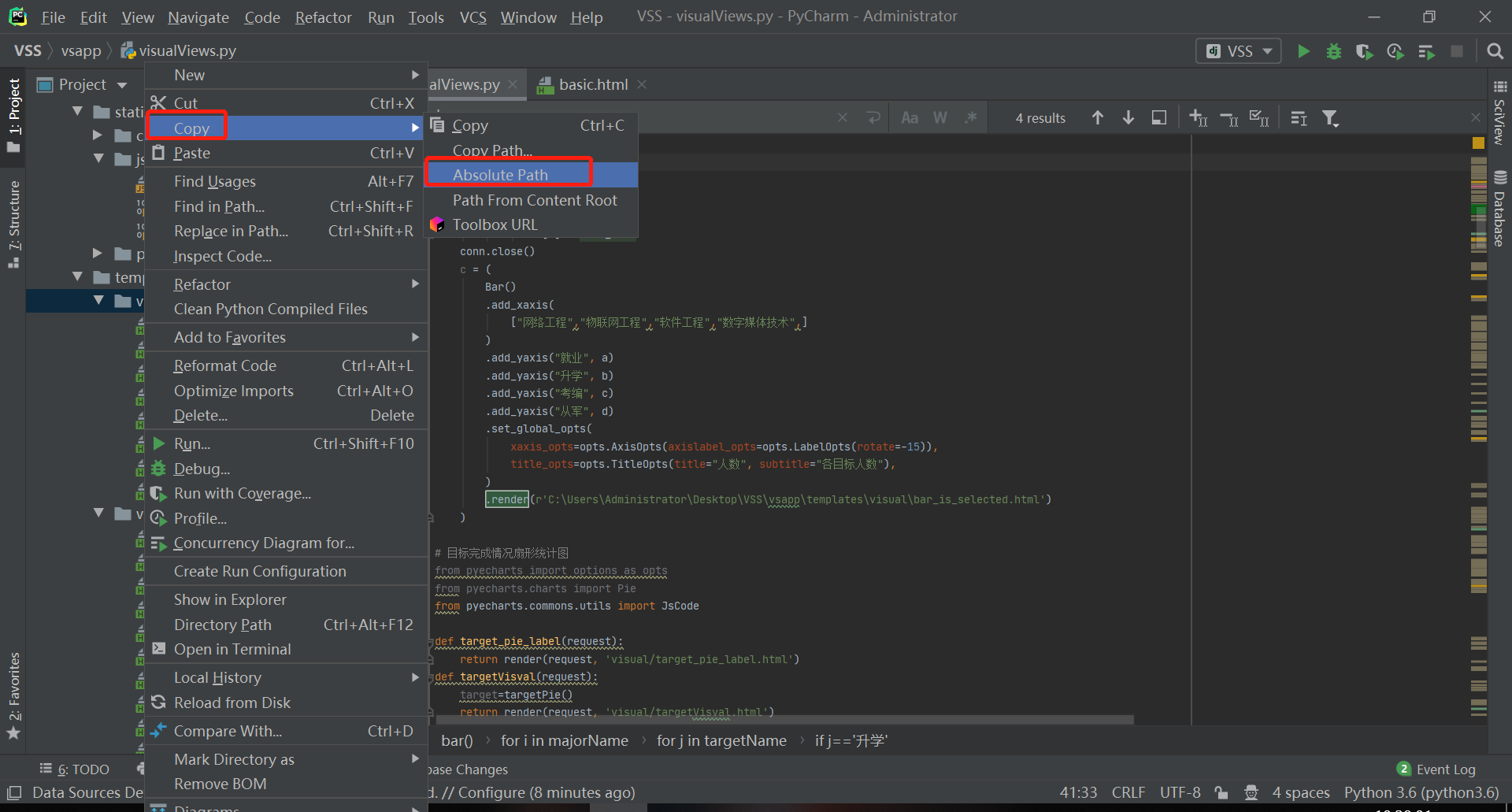
每台电脑的目录不同，需要更换的地址的位置有四个



把上方截图中我圈起来的的.render中的地址换成第二张截图中我箭头所指的那个目录地址（你电脑的目录和我的目录不一样）



可以在右键点击visual目录然后直接复制Absolute Path



**（4）运行系统**

运行Django项目，默认端口8000：python manage.py runserver ip:port

以python manage.py runserver 127.0.0.1:8000为例，则访问页面127.0.0.1:8000/index即可看到我们的网站页面

# 三、参考资料

Dango模板参考文章

<https://www.runoob.com/django/django-template.html>

<https://code.ziqiangxuetang.com/django/django-tutorial.html>

前端参考文章

使用的全局样式：<https://v3.bootcss.com/css/#forms>

使用js实现城市二级联动：https://blog.csdn.net/nihaox1/article/details/7439811

文件上传功能处使用的xlrd参考文章

功能的实现：<https://www.cnblogs.com/puresoul/p/7520198.html>

报错遇到问题时：<https://blog.csdn.net/weixin_44073728/article/details/111054157>

Iframe标签问题参考文章

在django中配置iframe：https://blog.csdn.net/cn\_newer/article/details/103866410

Iframe的边框参考文章：<https://blog.csdn.net/weixin_41574213/article/details/78953323>

Python与mysql交互参考文章

实现增删查改：https://www.jb51.net/article/186416.htm

从sql语句的执行结果中取数据：<https://www.cnblogs.com/z-x-y/p/9055509.html>

展示学生信息列表页面的实现：https://blog.csdn.net/zhouzhiwengang/article/details/89302354

Pyecharts的使用参考：

<http://pyecharts.herokuapp.com/geo>

https://github.com/pyecharts/pyecharts