onlineCoding项目安装说明

目录

[项目说明 2](#_Toc18068)

[一、系统软件安装 3](#_Toc14593)

[1. 安装Apache2+PHP 3](#_Toc392)

[2. 安装Python3 3](#_Toc25531)

[3. 安装MinGW 5](#_Toc30776)

[4. 项目代码部署 10](#_Toc32307)

[二、 软件配置 10](#_Toc995)

[1. 配置Python3 10](#_Toc17953)

[2. 配置Apache2 12](#_Toc30446)

[3. 调整项目代码 13](#_Toc27041)

[4. 重启Apache服务器 13](#_Toc32552)

[三、 已知问题 14](#_Toc16096)

项目说明

本项目onlineCoding ( <https://github.com/LuciferStarSS/onlineCoding> )中绝大部分代码源自开源项目codeLauncher ( <https://github.com/dadadel/codelauncher> ) 。

原项目只能在本地运行，运行后监听本地5000端口（http://127.0.0.1:5000），经过修改后，借助Apache2提供WEB服务，可以处理外部的访问请求。

将原项目的WEB页面功能全部剔除，只保留接收POST请求，以供WEB端用户调用。

在原项目的基础上，添加了对用户输入的支持。

一、系统软件安装

1. 安装Apache2+PHP

由Apache2负责提供WEB服务，PHP做简单的服务端控制。

安装配置方法略，网上有大把攻略。

测试版本：Apache 2.4.25.0，PHP8.2.12

安装路径：D:\networks\apache2

D:\networks\php8

1. 安装Python3

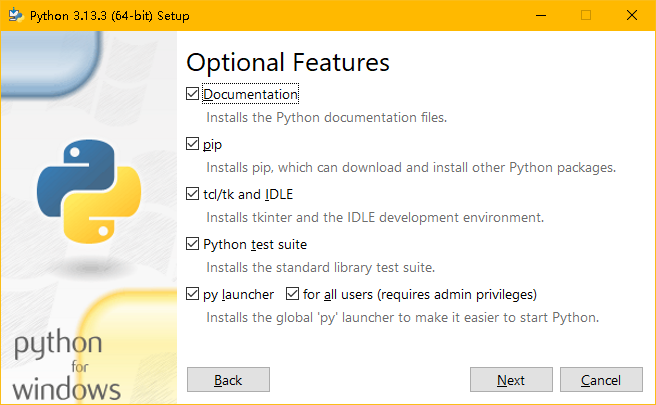
codeLauncher是Python代码，且该平台支持在线编辑和运行Python，所以需要安装。

测试版本：3.13.3（64-bit）

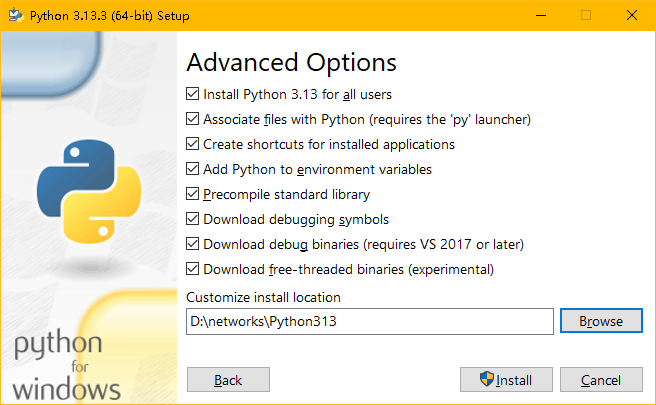
下载地址：https://www.python.org/ftp/python/3.13.3/python-3.13.3-amd64.exe

安装路径：D:\ProgrammingTools\Python313\

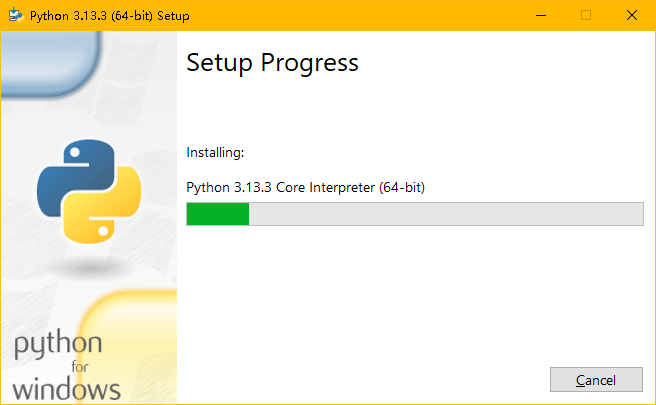




这里配置成了给所有用户使用，但实际没啥用，还是会有权限问题，后面有解决方案。



第一次安装，等待时间太长，就取消了，最终的安装路径有调整。



需要等待比较长的时间

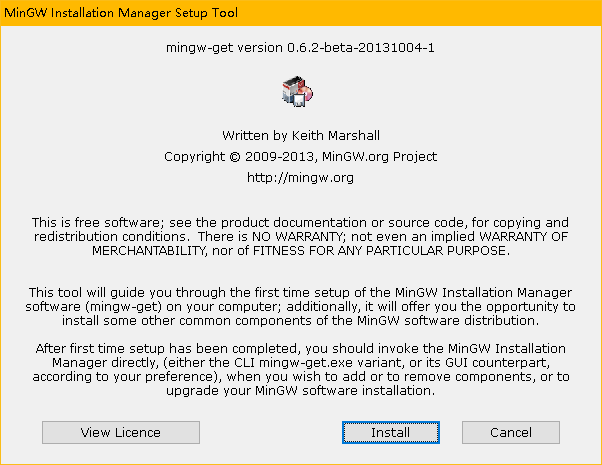
1. 安装MinGW

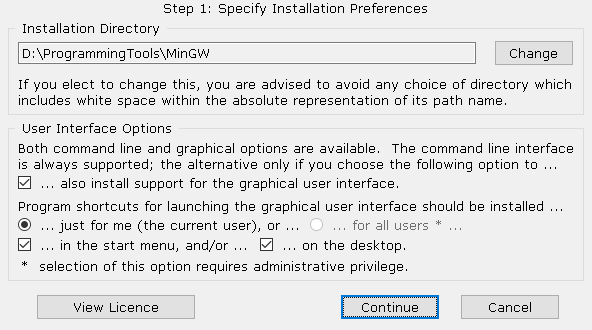
codeLauncher支持编译和运行C/C++，所以需要安装gcc和g++编译器。

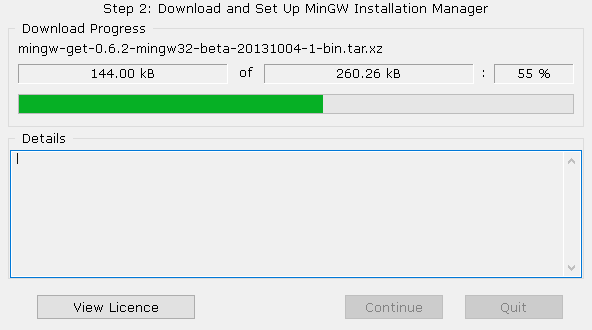
测试版本：MinGW-w64-for 32 and 64 bit Windows

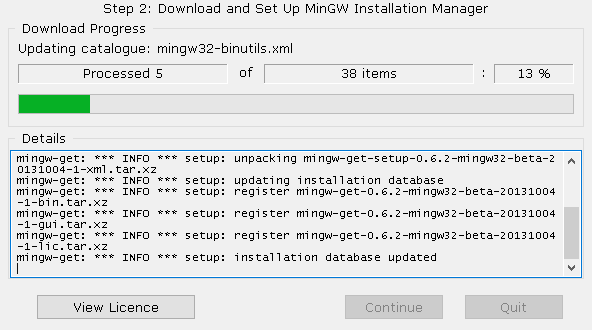
下载地址：https://nchc.dl.sourceforge.net/project/mingw/Installer/mingw-get-setup.exe

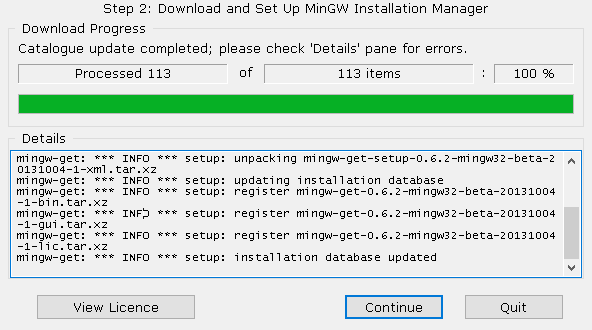
安装路径：D:\ProgrammingTools\MinGW\

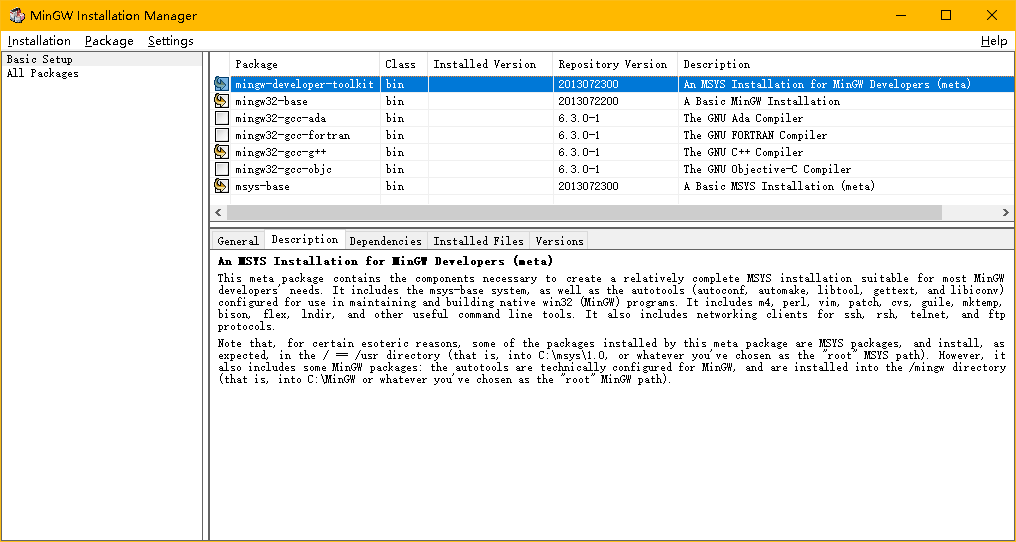


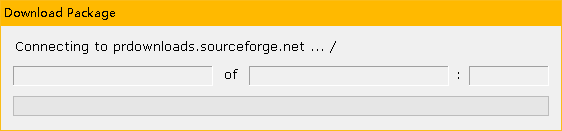


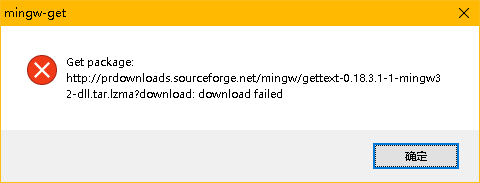


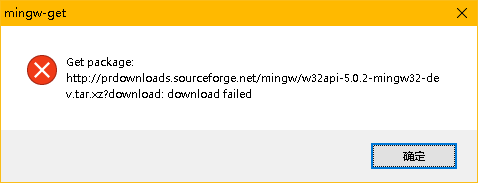


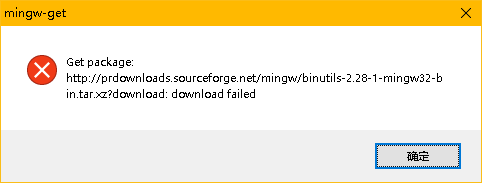




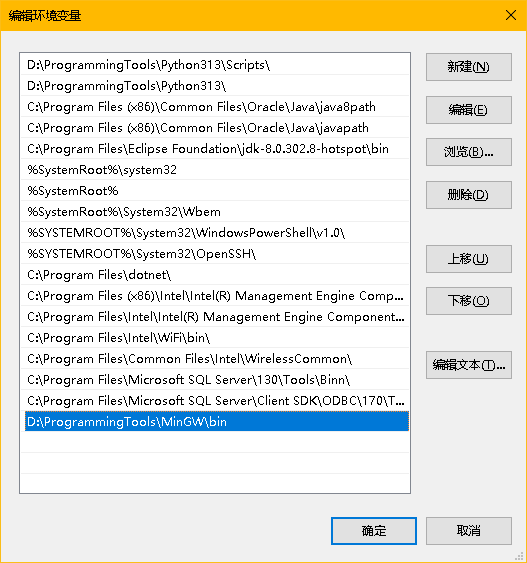








安装过程中会有一些文件下载失败，重复安装几次即可。



安装完毕后，记得给系统环境变量里的Path追加路径，否则会找不到编译器。

1. 项目代码部署

WSGI目录中的是codeLauncher，不要放WEB目录

E:\WSGI\codeLauncher\

PHP目录中的放WEB目录

E:\HTDOCS\class\ceditor\

1. 软件配置
2. 配置Python3

A.以管理员身份运行cmd.exe

用如下命令永久性地修改镜像站点为清华的源，可以加速下载操作：

pip config set global.index-url https://pypi.tuna.tsinghua.edu.cn/simple

B.进入E:\WSGI\codeLauncher\目录，安装并激活虚拟环境：

pip install virtualenv

virtualenv menv

./menv/Scripts/activate.bat

这个很重要，可以解决WSGI下python的权限问题。

**更新【1】：**这个目录下的数据，换机器后，一定要根据当前python重新生成。

C.安装pathlib：**更新【3】**

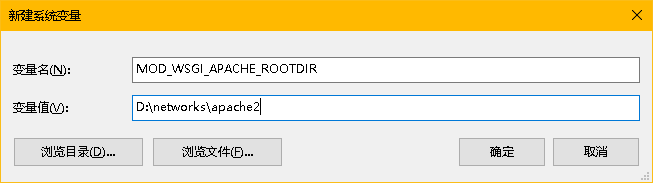
pip install pathlib

D.安装Flask：

pip install flask

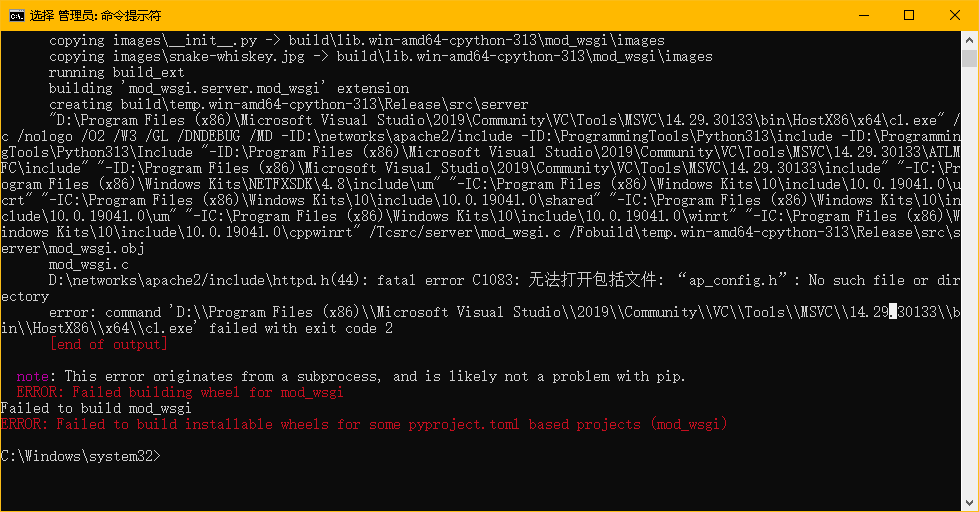
E.安装WSGI

i.先配置一下系统变量



ii.下载安装：

pip install mod\_wsgi



出现上述问题，是因为在编译WSGI时发现少了头文件ap\_config.h。

由于我电脑里已经安装了VC2019，所以安装程序自动调用了VC的编译器，如果没有安装，可能就调用刚安装的gcc编译器了。（没有做测试，实际情况不明。）

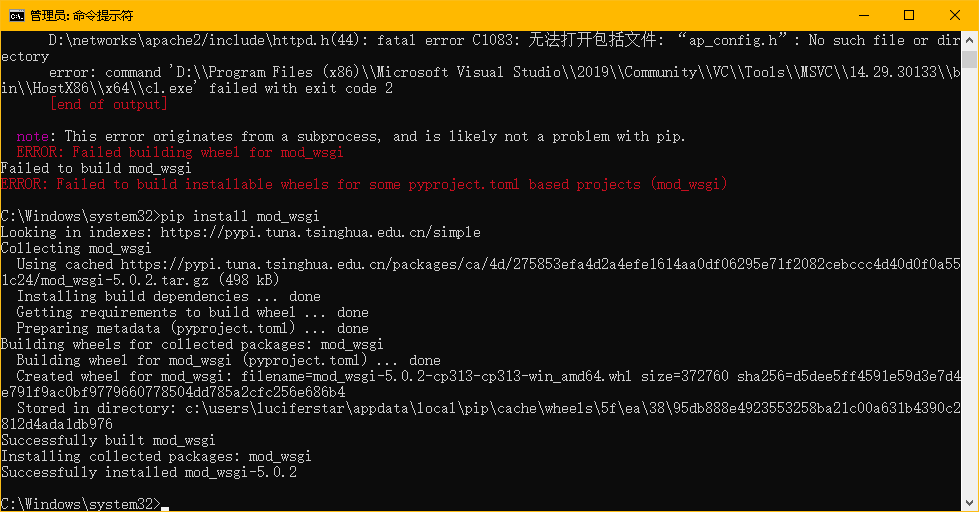
重新下载Apache2文件，提取里面的include文件：

<https://archive.apache.org/dist/httpd/httpd-2.4.25.tar.bz2>

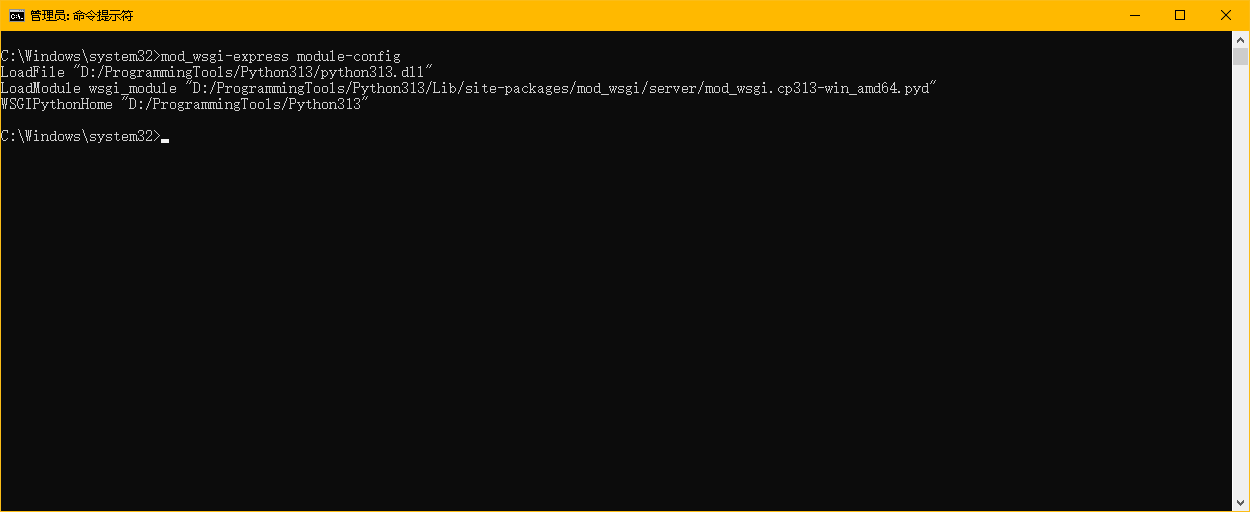
或者直接从github下载一个ap\_config.h:

<https://github.com/omnigroup/Apache/blob/master/httpd/include/ap_config.h>

将该文件保存到D:\networks\apache2\include\后，重新执行wsgi的安装命令。



iii.获得WSGI配置数据（LoadFile那条与LoadModule应该是一样的功能，据猜测。）：



1. 配置Apache2

A.打开文档apache2/conf/httpd.conf

B.将WSGI配置数据添加到httpd.conf文件中：

LoadModule wsgi\_module "D:/ProgrammingTools/Python313/Lib/site-packages/mod\_wsgi/server/mod\_wsgi.cp313-win\_amd64.pyd"

WSGIPythonHome "D:/ProgrammingTools/Python313"

C.添加监听端口：

Listen 80 #80端口负责用户页面

Listen 81 #81端口负责WSGI处理

D.启用vhosts（去掉数据前的“#”注释标记即可）：

Include conf/extra/httpd-vhosts.conf

E.修改apache2/conf/extra/httpd-vhosts.conf,添加VirtualHost：

#用户界面

**<VirtualHost \_default\_:80>**

**DocumentRoot "e:/HTDOCS"**

**<Directory "e:/HTDOCS">**

**Require all granted**

**</Directory>**

**</VirtualHost>**

#WSGI服务（注释掉的内容是给非Windows用的）

**<VirtualHost \*:81>**

**#WSGIDaemonProcess flaskapp user=www-data group=www-data threads=5**

**WSGIScriptAlias / "e:/WSGI/codeLauncher/main.wsgi"**

**<Directory "e:/WSGI/codeLauncher">**

**#WSGIProcessGroup flaskapp**

**#WSGIApplicationGroup %{GLOBAL}**

**Require all granted**

**</Directory>**

**</VirtualHost>**

1. 调整项目代码

本项目还在雏形阶段，内部用了不少硬代码，需要根据实际情况进行调整。

A.文件：WSGI/codeLauncher/main.wsgi

行：4，5，7

根据实际情况调整路径。

经过测试，在正确配置了正确的python路径后（下一则调整），第5，7，8，9行代码就不需要了。

chdir要保留，这样runcode.py就能用相对路径了。

B.文件：WSGI/codeLauncher/runcode/runcode.py

行：111

根据实际情况调整python在虚拟环境下的实际路径：

E:\WSGI\codeLauncher\menv\Scripts\python.exe

能否正确地运行python代码，关键就是这个地方。

更新【2】：此处可以使用相对路径：

./menv/Scripts/python.exe

C.文件：PHP/ceditor/index.php

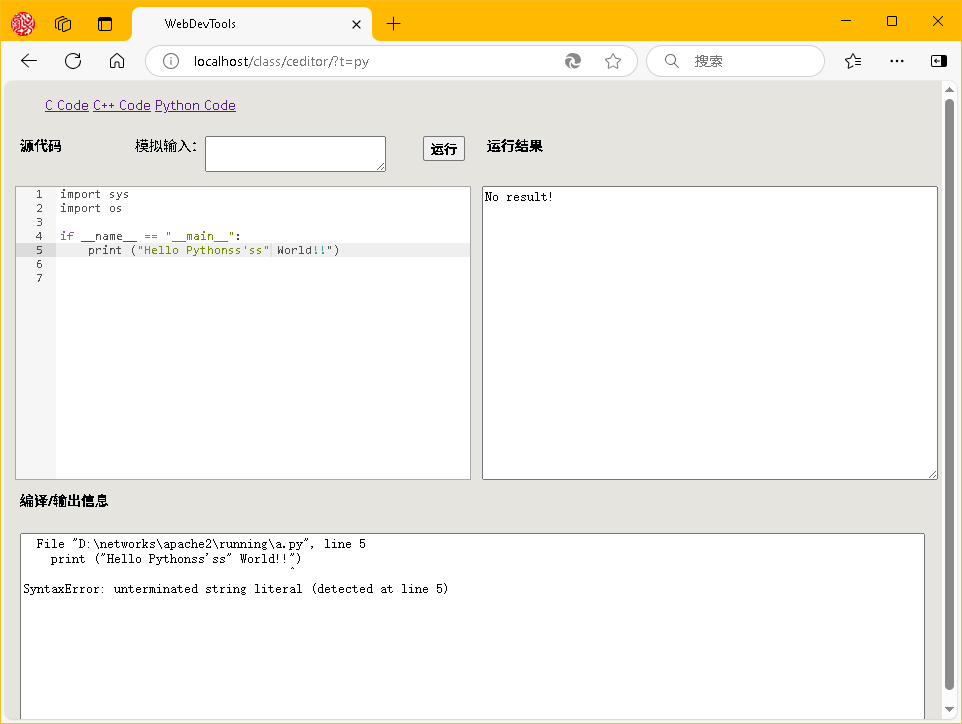
行：120

<http://192.168.1.103:81/>

改成自己的实际IP和端口号。

1. 重启Apache服务器

完成上述配置后，重启一下Apache服务器，就可以通过浏览器访问此平台了。



1. 已知问题

暂不支持中文，可能是字符集问题。

更新【1】：经过测试，menv文件夹内数据，一定要根据当前系统安装的python版本来生成。不同版本可能存在不能在新平台运行的情况（动态链接库的依赖问题）。

更新【2】：在main.wsgi中chdir后，runcode.py里可以使用相对路径。

更新【3】：添加pathlib库。