

## onlineCoding 项目（纯 PHP 版）安装说明

## 目录

项目说明 .....	2
一、系统软件安装 .....	3
1. 安装 Apache2+PHP .....	3
2. 安装 Python3 .....	3
3. 安装 MinGW .....	5
4. 项目代码部署 .....	10
二、 <del>软件配置</del> .....	10
1. <del>配置 Python3</del> .....	10
2. <del>配置 Apache2</del> .....	12
3. <del>调整项目代码</del> .....	13
4. <del>重启 Apache 服务器</del> .....	13
三、已知问题 .....	14

## 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

### 2. 安装 Python3

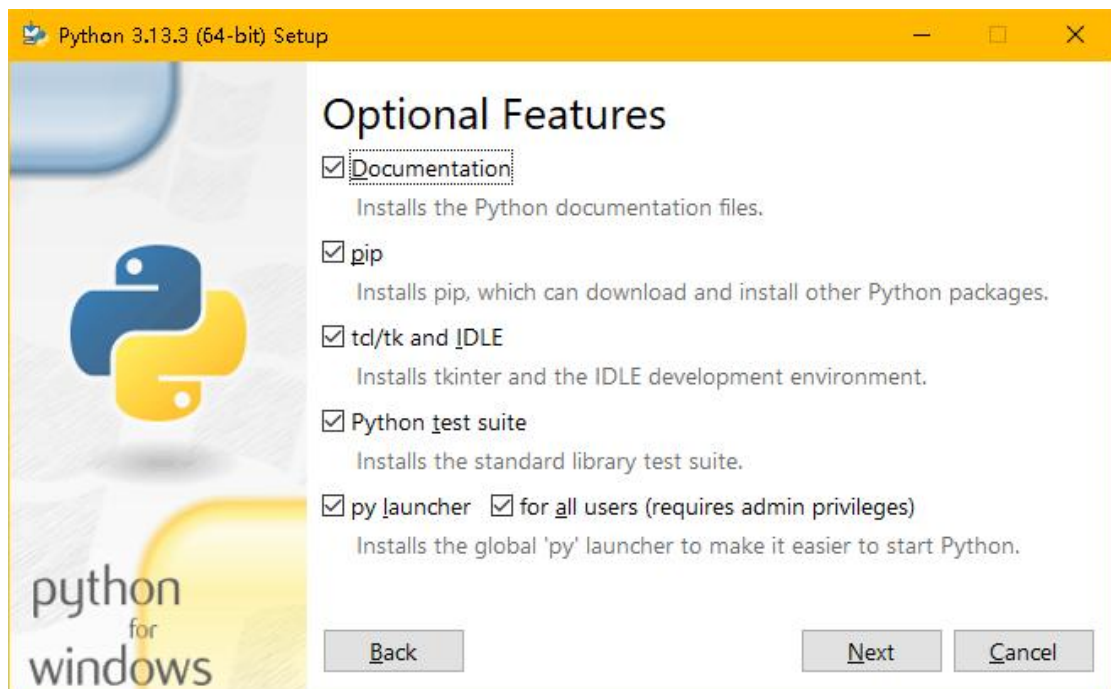
codeLauncher 是 Python 代码，且该平台支持在线编辑和运行 Python，所以需要安装。

测试版本：3.13.3（64-bit）

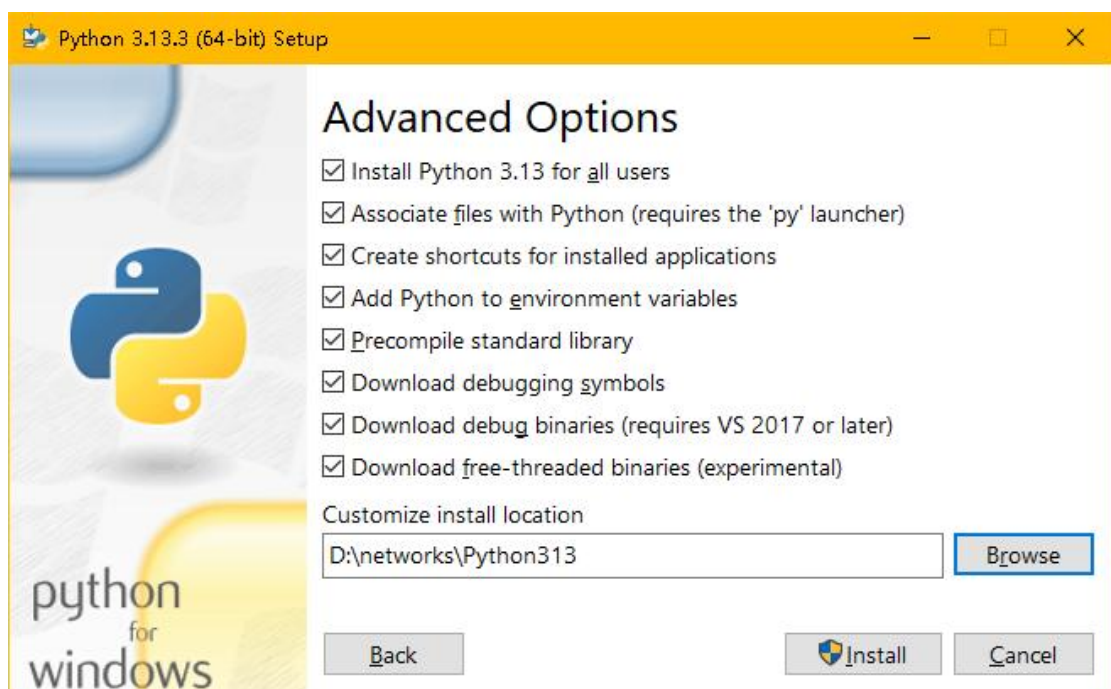
下载地址：<https://www.python.org/ftp/python/3.13.3/python-3.13.3-amd64.exe>

安装路径：D:\ProgrammingTools\Python313\

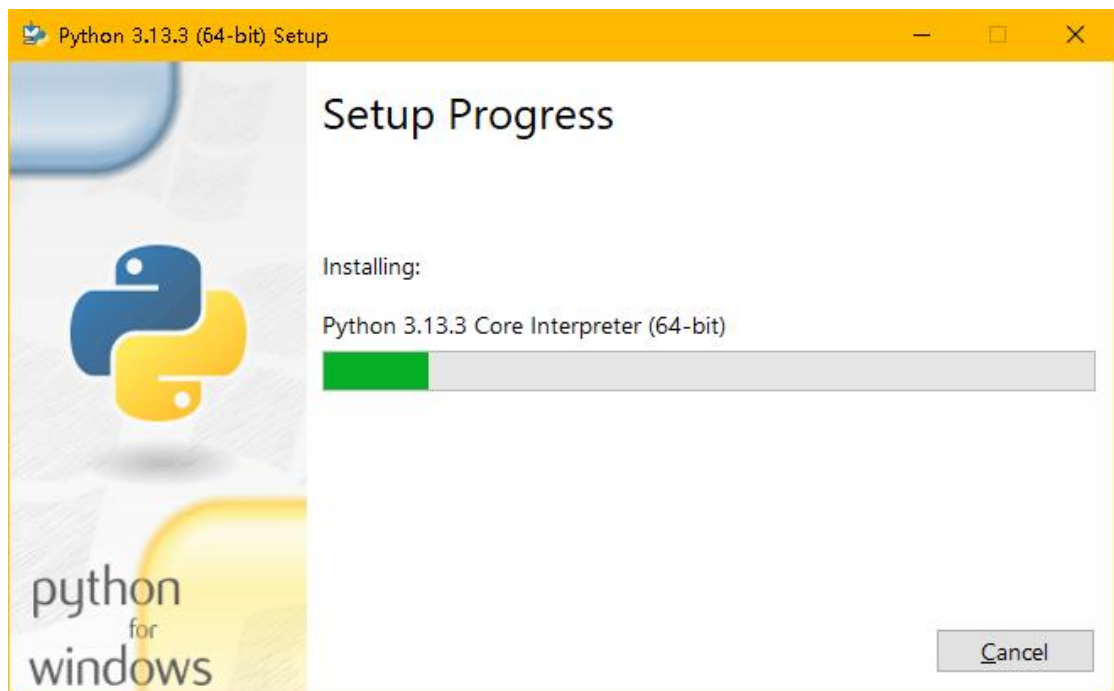




这里配置成了给所有用户使用，但实际没啥用，还是会有权限问题，后面有解决方案。



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

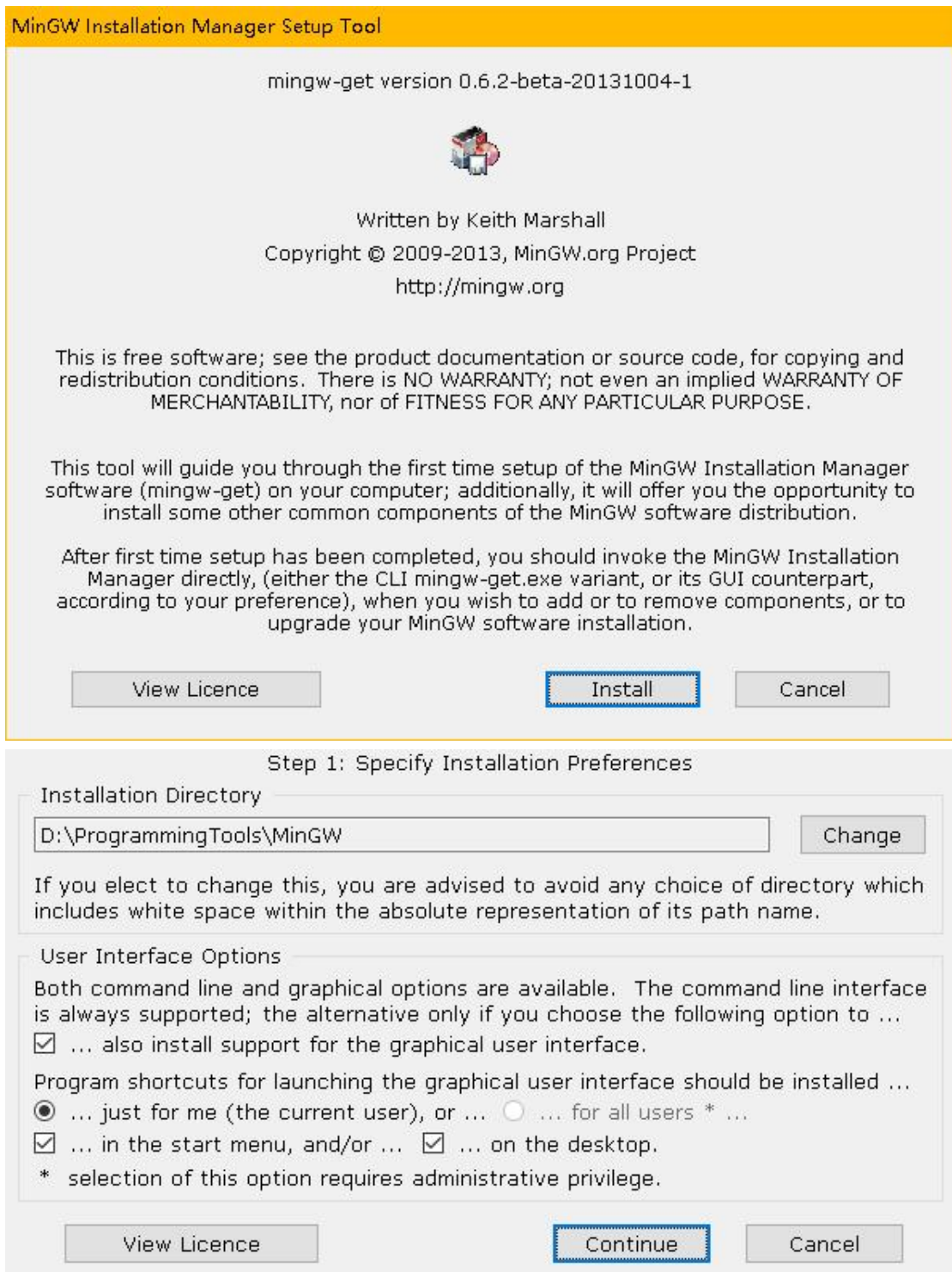
### 3. 安装 MinGW

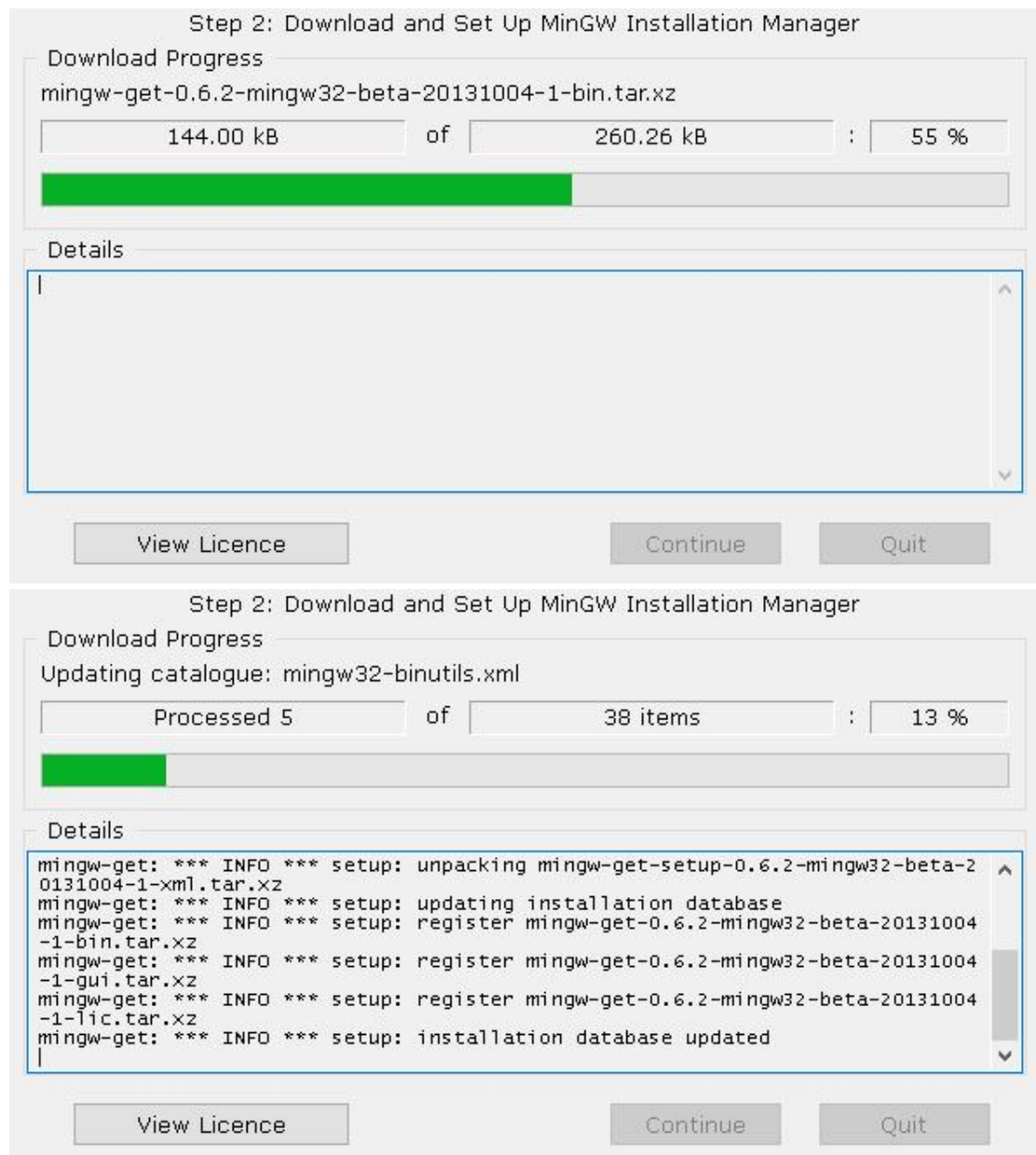
codeLauncher 支持编译和运行 C/C++, 所以需要安装 gcc 和 g++编译器。

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

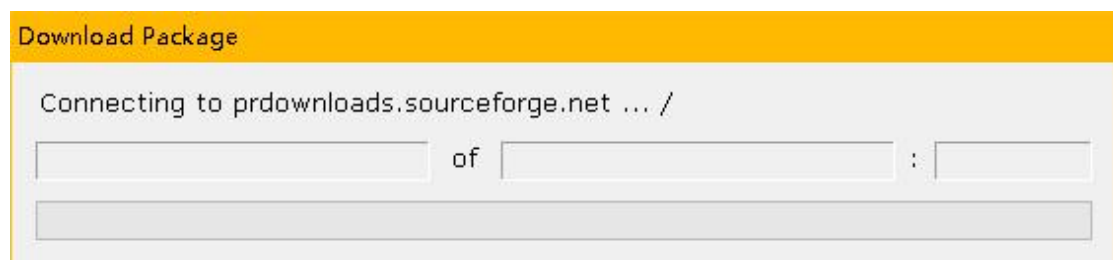
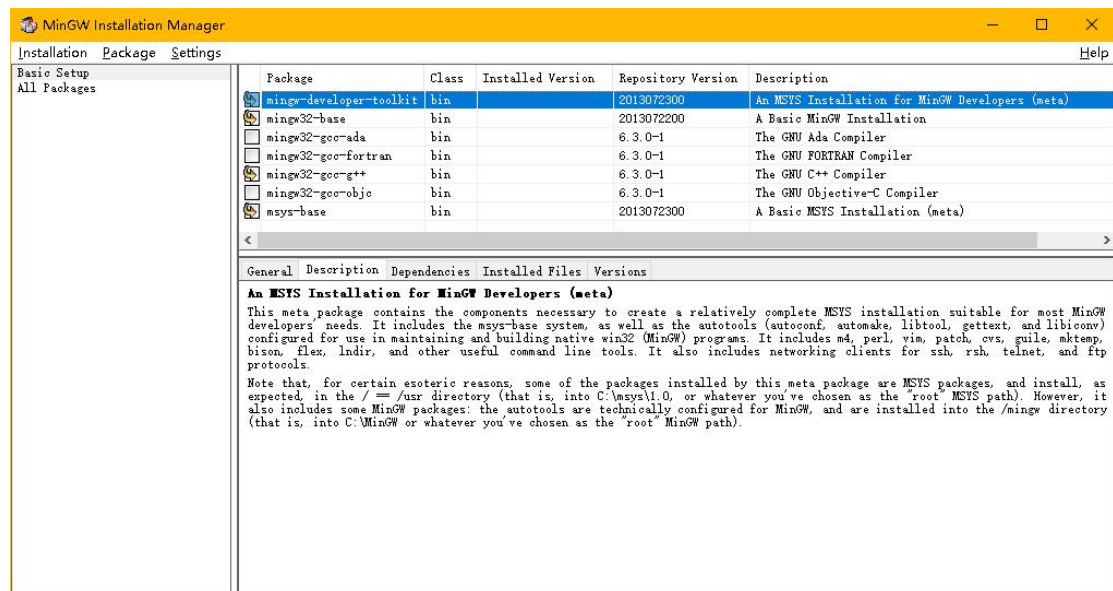
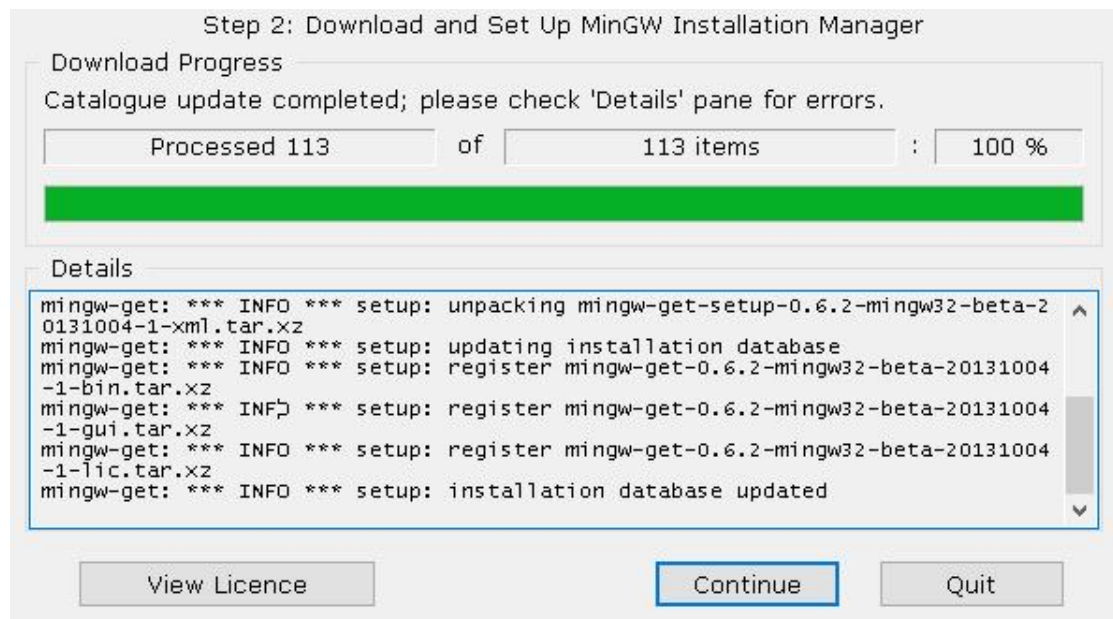
下载地址: <https://nchc.dl.sourceforge.net/project/mingw/Installer/mingw-get-setup.exe>

安装路径: D:\ProgrammingTools\MinGW\

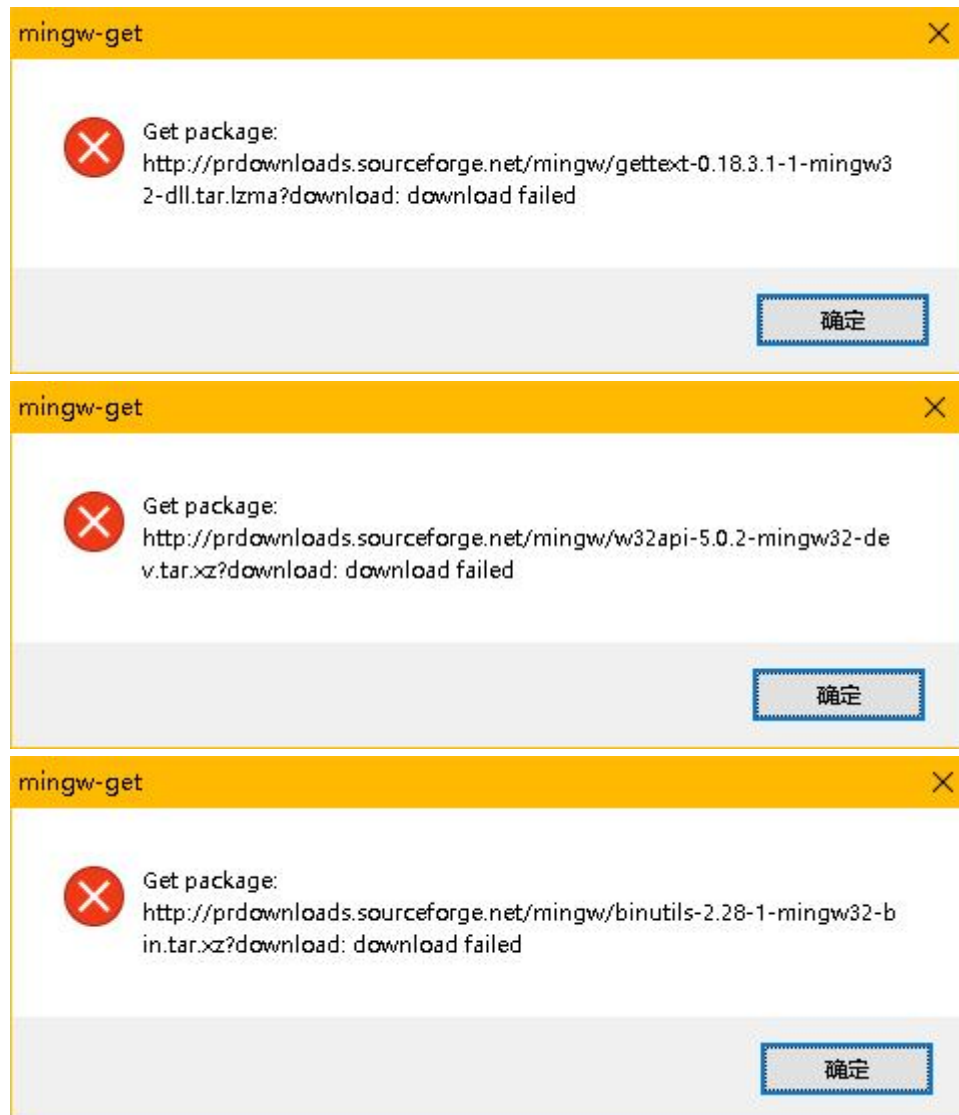




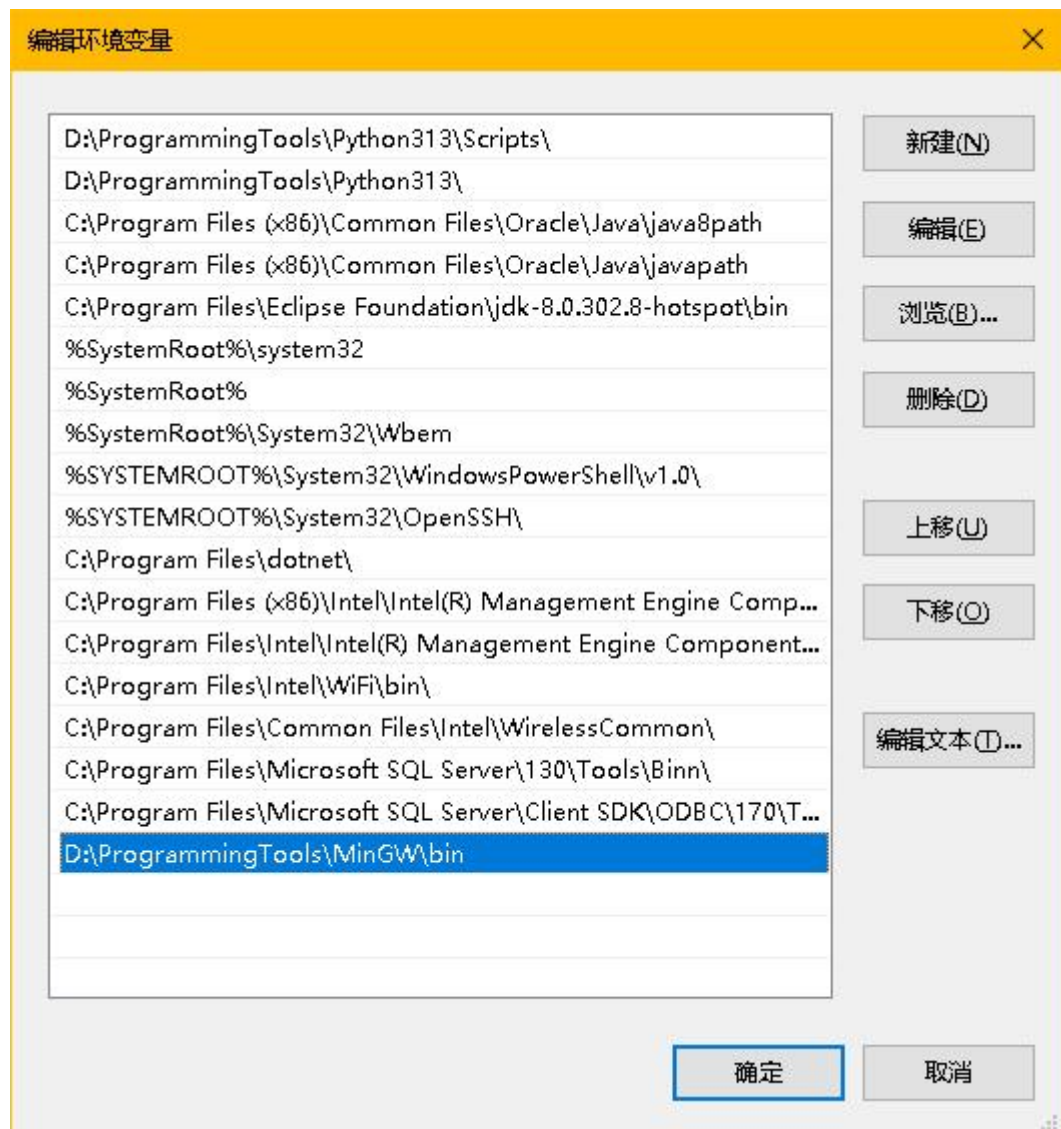








安装过程中会有一些文件下载失败，重复安装几次即可。



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。

#### 4. 项目代码部署

WSGI 目录中的是 codeLauncher，不要放 WEB 目录

E:\WSGI\codeLauncher\

just\_PHP 目录中的放 WEB 目录

E:\HTDOCS\class\ceditor\

### 二、软件配置

#### 1. 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】**

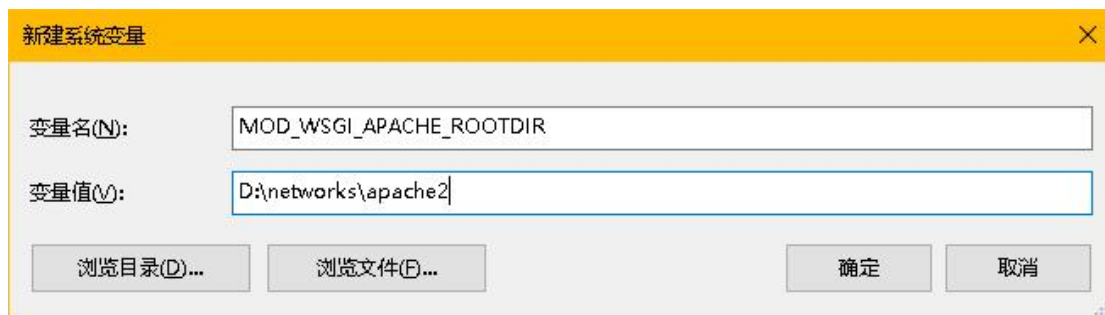
```
pip install pathlib
```

### D. 安装 Flask:

```
pip install flask
```

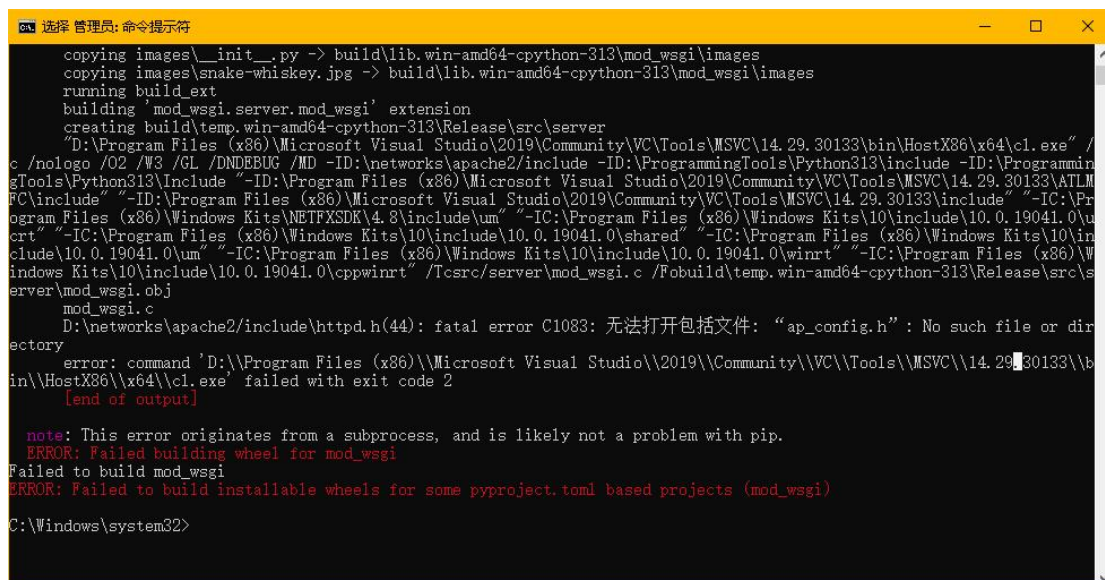
### E. 安装 WSGI

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](https://github.com/omnigroup/Apache/blob/master/httpd/include/ap_config.h) 下载一个 `ap_config.h`:

[https://github.com/omnigroup/Apache/blob/master/httpd/include/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。

```

管理员: 命令提示符
D:\networks\apache2\include\httpd.h(44): fatal error C1083: 无法打开包括文件: "ap_config.h": No such file or directory
error: command 'D:\Program Files (x86)\Microsoft Visual Studio\2019\Community\VC\Tools\MSVC\14.29.30133\bin\HostX86\x64\cl.exe' failed with exit code 2
[end of output]

note: This error originates from a subprocess, and is likely not a problem with pip.
ERROR: Failed building wheel for mod_wsgi
Failed to build mod_wsgi
ERROR: Failed to build installable wheels for some pyproject.toml based projects (mod_wsgi)

C:\Windows\system32>pip install mod_wsgi
Looking in indexes: https://pypi.tuna.tsinghua.edu.cn/simple
Collecting mod_wsgi
  Using cached https://pypi.tuna.tsinghua.edu.cn/packages/ca/4d/275853efa4d2a4efe1614aa0df06295e71f2082ceebccc4d40d0f0a551c24/mod_wsgi-5.0.2.tar.gz (498 kB)
    Installing build dependencies ... done
    Getting requirements to build wheel ... done
    Preparing metadata (pyproject.toml) ... done
Building wheels for collected packages: mod_wsgi
  Building wheel for mod_wsgi (pyproject.toml) ... done
    Created wheel for mod_wsgi: filename=mod_wsgi-5.0.2-cp313-cp313-win_amd64.whl size=372760 sha256=d5dee5ff4591e59d3e7d4e791f9ac0bf9779660778504dd785a2cfc256e686b4
    Stored in directory: c:\users\luciferstar\appdata\local\pip\cache\wheels\5f\ea\38\95db888e4923553258ba21c00a631b4390c2812d4adadb976
Successfully built mod_wsgi
Installing collected packages: mod_wsgi
Successfully installed mod_wsgi-5.0.2

C:\Windows\system32>
  
```

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

```

C:\Windows\system32>mod_wsgi-express module-config
LoadFile "D:/ProgrammingTools/Python313/python313.dll"
LoadModule wsgi_module "D:/ProgrammingTools/Python313/Lib/site-packages/mod_wsgi/server/mod_wsgi.cp313-win_amd64.pyd"
WSGIProxyHome "D:/ProgrammingTools/Python313"

C:\Windows\system32>
  
```

## 2. 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"
WSGIProxyHome "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>
```

### 3. 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\menu\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:

`./menu/Scripts/python.exe`

C. 文件: `PHP/ceditor/index.php`

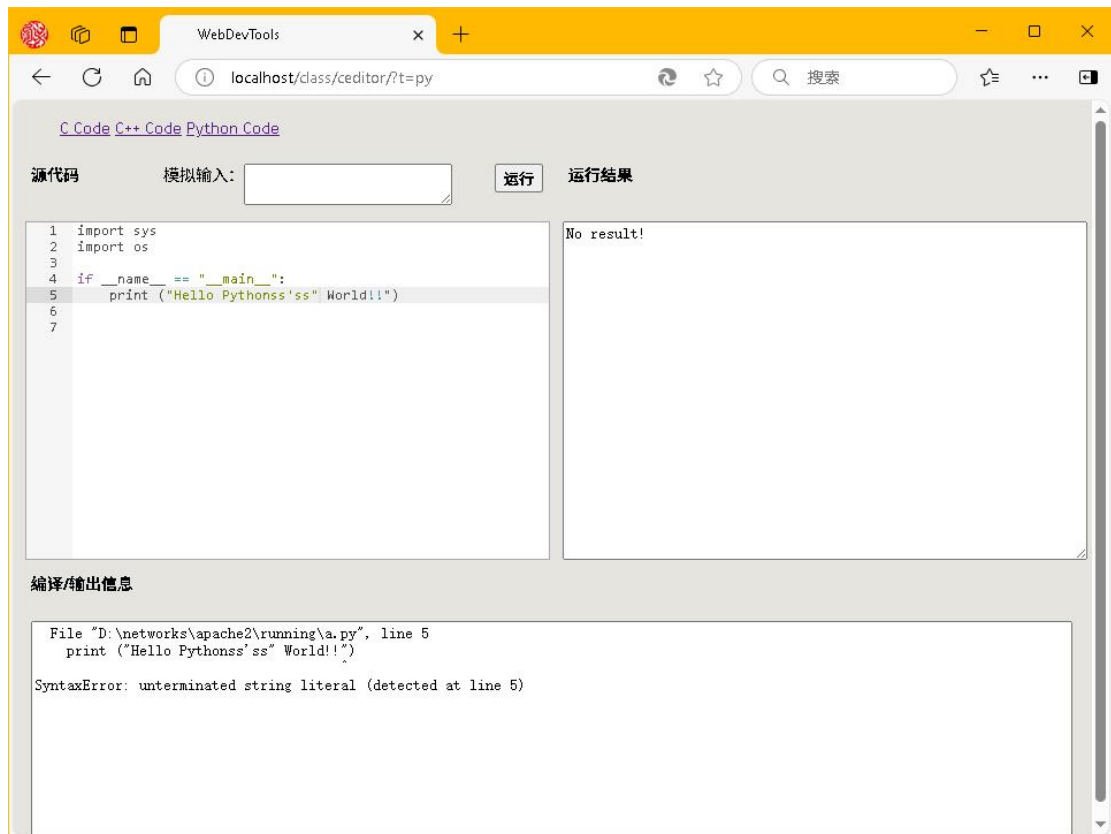
行: 120

<http://192.168.1.103:81/>

改成自己的实际 IP 和端口号。

### 4. 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。



### 三、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。