

1. 首先是下载 vs code

按照老师教学视频里的步骤进行下载，建议不添加桌面快捷，直接固定到任务栏就可以。（在不知道作业行驶的时候下载的，忘记截图了）；下载完毕进入 vs code 之后，建议将 welcome 的内容关掉，注册或者不注册都可以正常使用。

2. 安装 git，即代码版本管理软件

Download for Windows

[Click here to download](#) the latest (2.48.1) 64-bit version of Git for Windows. This is the most recent maintained build. It was released 21 days ago, on 2025-02-13.

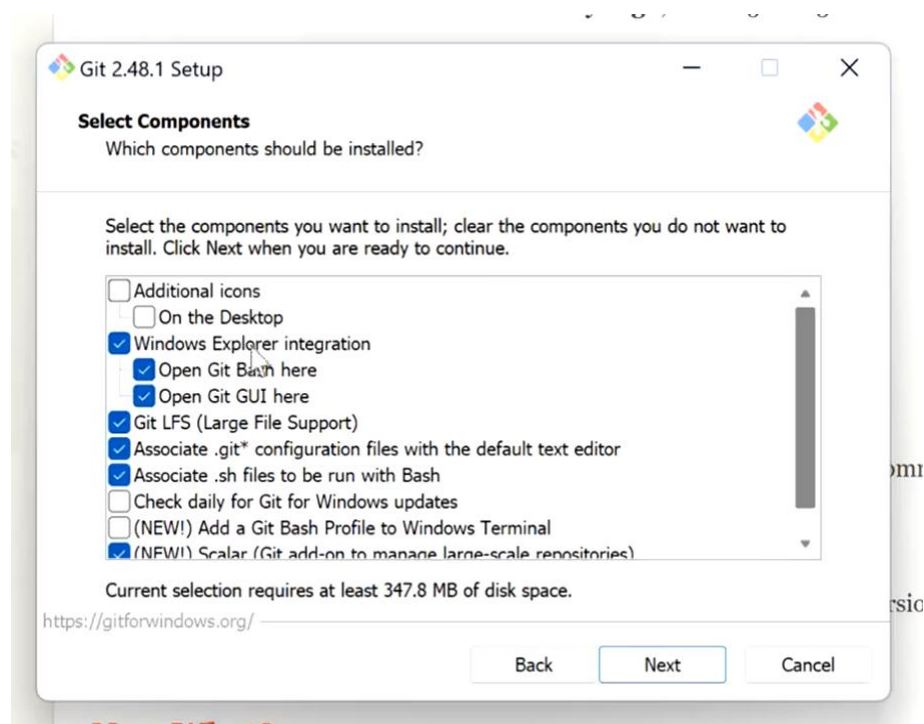
Other Git for Windows downloads

Standalone Installer

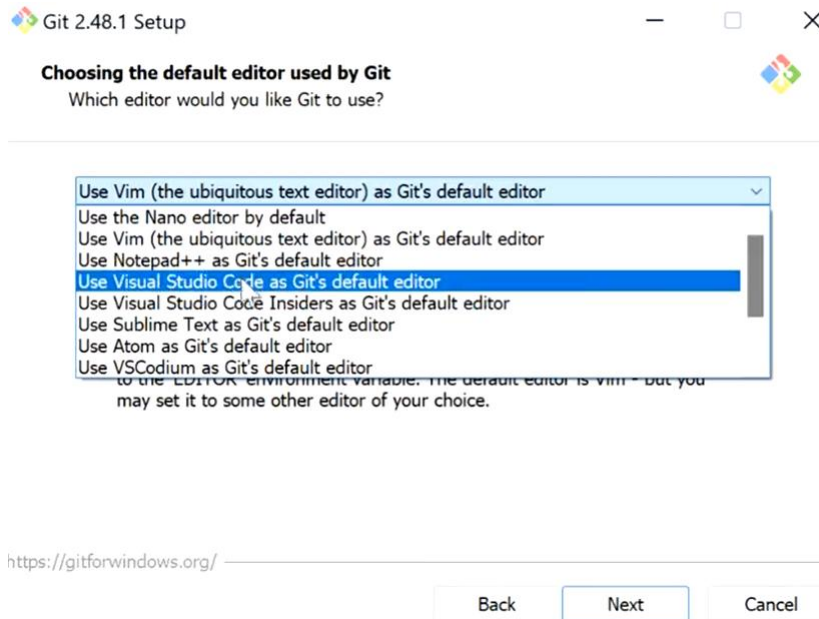
[32-bit Git for Windows Setup.](#)

[64-bit Git for Windows Setup.](#)

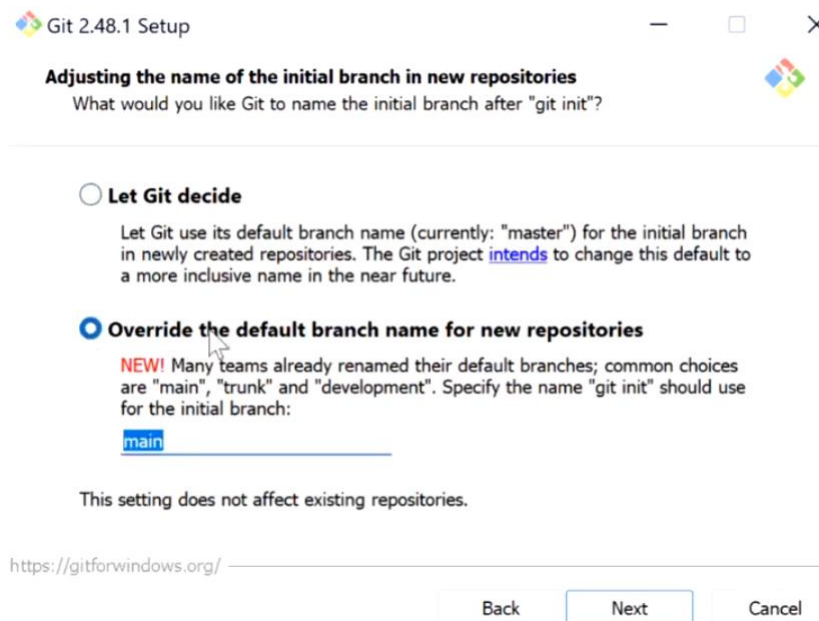
建议选择 standalone installer 下面的 64-bit。开始下载。



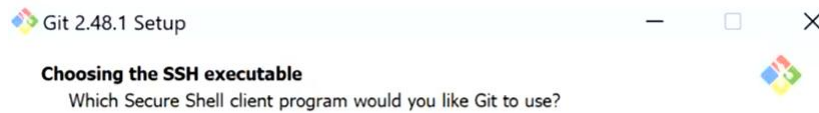
在微软应用商店里安装 terminal 终端，我的已经下载好了。
然后勾选 add a git...terminal



选择图上这个



接着选择这个



https://gitforwindows.org/

Back Next Cancel

选择这个默认的



https://gitforwindows.org/

Back Next Cancel

Git 2.48.1 Setup

— □ ×

Configuring the line ending conversions

How should Git treat line endings in text files?

☒ Checkout Windows-style, commit Unix-style line endings

Git will convert LF to CRLF when checking out text files. When committing text files, CRLF will be converted to LF. For cross-platform projects, this is the recommended setting on Windows ("core.autocrlf" is set to "true").

☐ Checkout as-is, commit Unix-style line endings

Git will not perform any conversion when checking out text files. When committing text files, CRLF will be converted to LF. For cross-platform projects, this is the recommended setting on Unix ("core.autocrlf" is set to "input").

☐ Checkout as-is, commit as-is

Git will not perform any conversions when checking out or committing text files. Choosing this option is not recommended for cross-platform projects ("core.autocrlf" is set to "false").

<https://gitforwindows.org/>

BackNextCancel

Git 2.48.1 Setup

— □ ×

Configuring the terminal emulator to use with Git Bash

Which terminal emulator do you want to use with your Git Bash?

☒ Use MinTTY (the default terminal of MSYS2)

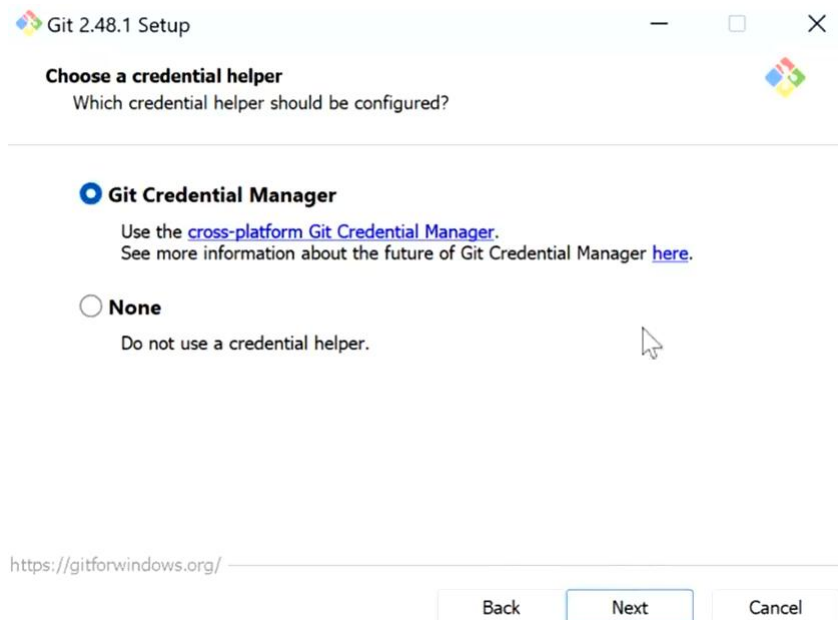
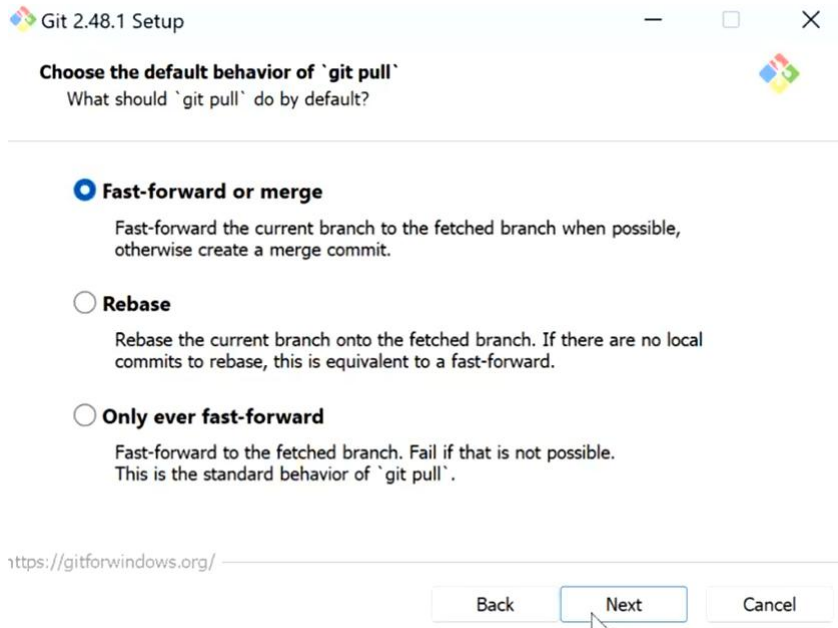
Git Bash will use MinTTY as terminal emulator, which sports a resizable window, non-rectangular selections and a Unicode font. Windows console programs (such as interactive Python) must be launched via 'winpty' to work in MinTTY.

☐ Use Windows' default console window

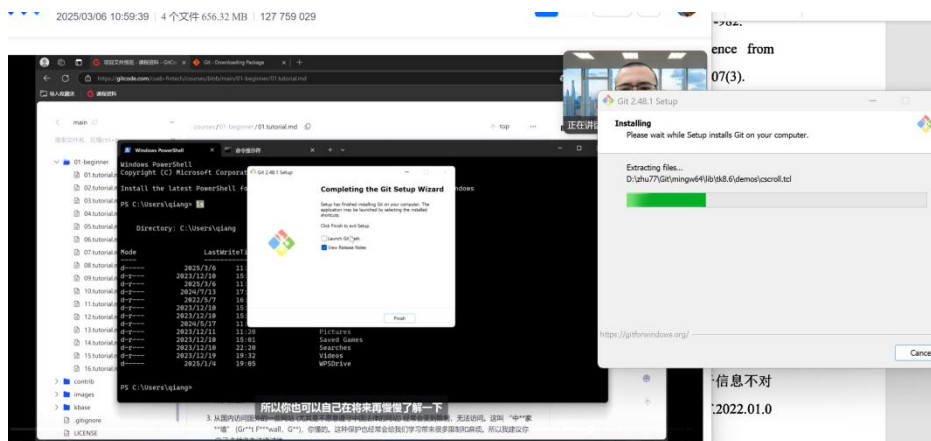
Git will use the default console window of Windows ("cmd.exe"), which works well with Win32 console programs such as interactive Python or node.js, but has a very limited default scroll-back, needs to be configured to use a Unicode font in order to display non-ASCII characters correctly, and prior to Windows 10 its window was not freely resizable and it only allowed rectangular text selections.

<https://gitforwindows.org/>

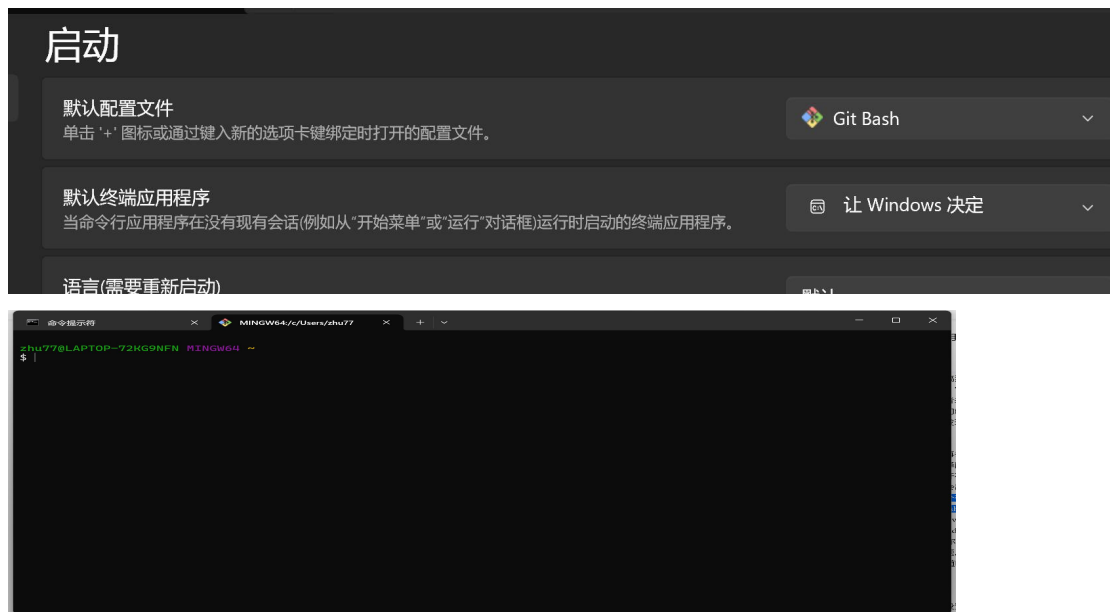
BackNextCancel



然后就下载



打开命令提示符之后，设置默认配置文件如下：

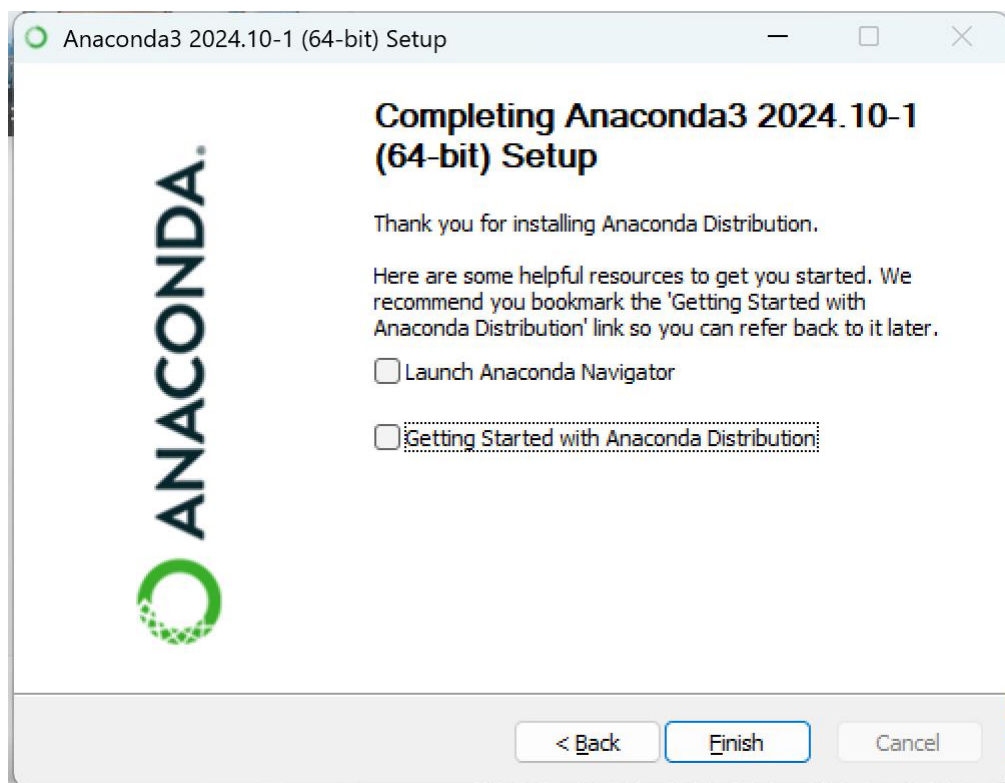


然后将光标改成“块状”，这样便于区分字符。

结束。

3. 下载 Anaconda

选择 windows 下载，然后不用勾选这两个



然后打开命令提示符，输入 conda，按回车


```

MINGW64: c:/Users/zhu77
zhu77@LAPTOP-72KG9NFN MINGW64 ~
$ conda
usage: conda-script.py [-h] [-v] [--no-plugins] [-V] COMMAND ...

conda is a tool for managing and deploying applications, environments and packages.

options:
  -h, --help            Show this help message and exit.
  -v, --verbose         Can be used multiple times. Once for detailed output, twice for INFO logging, four times for TRACE logging.
  --no-plugins          Disable all plugins that are not built into conda.
  -V, --version         Show the conda version number and exit.

commands:
  The following built-in and plugins subcommands are available.

COMMAND                description
activate               Activate a conda environment.
build                  Build conda packages from a conda recipe.
clean                  Remove unused packages and caches.
commands               List all available conda subcommands (including those from plugins). Generate tab-completion.
compare                Compare packages between conda environments.
config                 Modify configuration values in .condarc.
content-trust           Signing and verification tools for Conda
convert                Convert pure Python packages to other platforms (a.k.a., subdirs).
create                 Create a new conda environment from a list of specified packages.
deactivate             Deactivate the current active conda environment.
debug                  Debug the build or test phases of conda recipes.

```

可以看到 conda 所在的路径

```
命令提示符
MINGW64: c:/Users/zhu77
+ v

token          See 'conda token --help'.
update (upgrade) Update conda packages to the latest compatible version.

zhu77@LAPTOP-72KG9NFN MINGW64 ~
$ conda1
bash: conda1: command not found

zhu77@LAPTOP-72KG9NFN MINGW64 ~
$ which conda
/d/zhu77/Anaconda/Scripts/conda

zhu77@LAPTOP-72KG9NFN MINGW64 ~
$ echo $PATH
c:/Users/zhu77/bin:/mingw64/bin:/usr/local/bin:/usr/bin:/bin:/mingw64/bin:/usr/bin:/c:/Users/zhu77/bin:/d/zhu77/python/Scripts:/d/zhu77/python:/c:/Windows/system32:/c:/Windows:/c:/Windows/System32/Wbem:/c:/Windows/System32/WindowsPowerShell/v1.0:/c:/Windows/System32/OpenSSH:/d/bandizip:/d/zhu77/matlab2023a/runtime/win64:/d/zhu77/matlab2023a/bin:/cmd:/mingw64/bin:/usr/bin:/d/zhu77/Anaconda:/d/zhu77/Anaconda/Library/mingw-w64/bin:/d/zhu77/Anaconda/Library/usr/bin:/d/zhu77/Anaconda/Library/bin:/d/zhu77/Anaconda/Scripts:/c:/Users/zhu77/AppData/Local/Microsoft/WindowsApps:/d/zhu77/Microsoft VS Code/bin:/usr/bin/vendor_perl:/usr/bin/core_perl

zhu77@LAPTOP-72KG9NFN MINGW64 ~
$ python
Python 3.13.0 (tags/v3.13.0:60403a5, Oct 7 2024, 09:38:07) [MSC v.1941 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> which python
File "<python-input-0>", line 1
    which python
        ^^^^^^
SyntaxError: invalid syntax
>>> exit()
```

而我输入 `which python` 之后报错了，问了豆包之后得知，windows 环境中要用 `where python`，`which python` 是 Linux 或 macOS 系统终端里用来查看 Python 命令路径的指令。

我需要输入 `exit ()` 退出 python 交互式环境。

```
命令提示符
MINGW64/c/Users/zhu77
/d/zhu77/Anaconda/Scripts/conda

zhu77@LAPTOP-72KG9NFN MINGW64 ~
$ echo $PATH
/c/Users/zhu77/bin:/mingw64/bin:/usr/local/bin:/usr/bin:/bin:/mingw64/bin:/usr/bin:/c/Users/zhu77/bin:/d/zhu77/python/Scripts:/d/zhu77/python:/c/Windows/system32:/c/Windows:/c/Windows/System32/Wbem:/c/Windows/System32/WindowsPowerShell/v1.0:/c/Windows/System32/OpenSSH:/d/bandizip:/d/zhu77/matlab2023a/runtime/win64:/d/zhu77/matlab2023a/bin:/cmd:/mingw64/bin:/usr/bin:/d/zhu77/Anaconda:/d/zhu77/Anaconda/Library/mingw-w64/bin:/d/zhu77/Anaconda/Library/usr/bin:/d/zhu77/Anaconda/Library/bin:/d/zhu77/Anaconda/Scripts:/c/Users/zhu77/AppData/Local/Microsoft/WindowsApps:/d/zhu77/Microsoft VS Code/bin:/usr/bin/vendor_perl:/usr/bin/core_perl

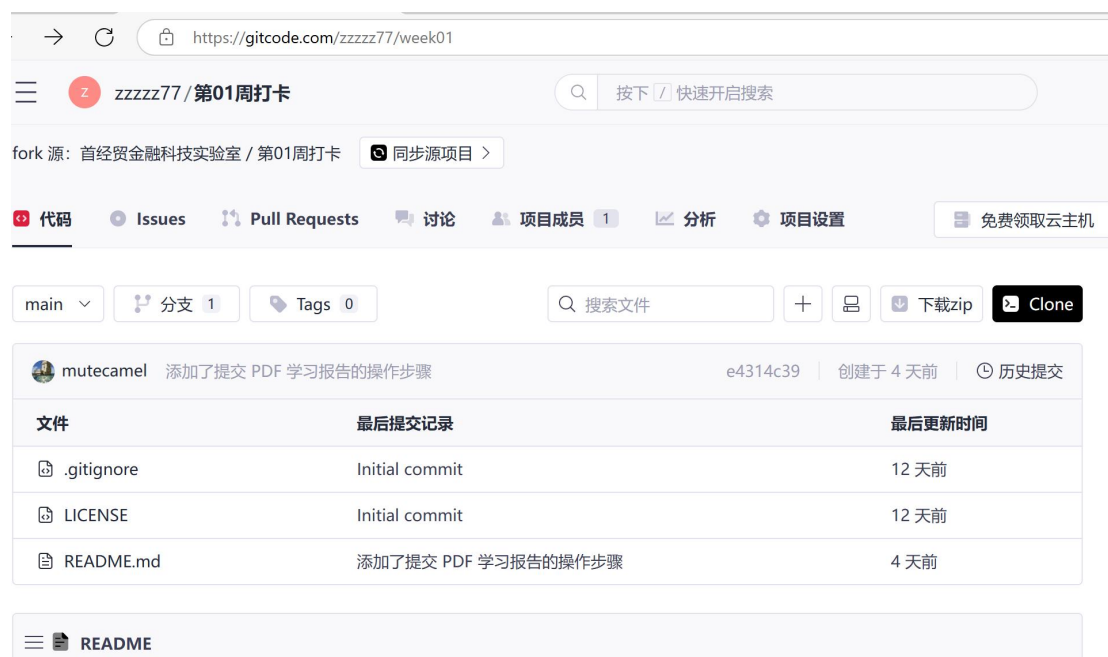
zhu77@LAPTOP-72KG9NFN MINGW64 ~
$ python
Python 3.13.0 (tags/v3.13.0:60403a5, Oct 7 2024, 09:38:07) [MSC v.1914 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> which python
File "<python-input-0>", line 1
      which python
      ^^^^^^^
SyntaxError: invalid syntax
>>> exit()

zhu77@LAPTOP-72KG9NFN MINGW64 ~
$ where python
D:\zhu77\python\python.exe
D:\zhu77\Anaconda\python.exe
C:\Users\zhu77\AppData\Local\Microsoft\WindowsApps\python.exe

zhu77@LAPTOP-72KG9NFN MINGW64 ~
$
```

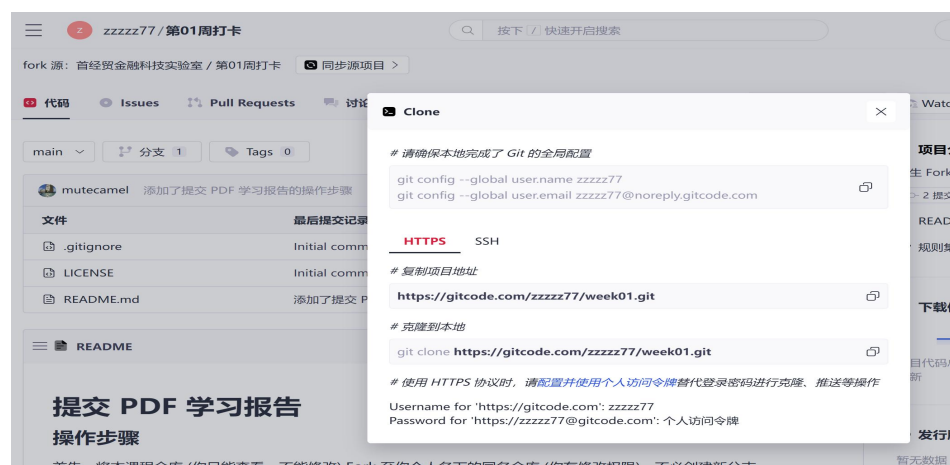
以后使用 python，就直接打开 vs code，然后输入 python，回车，就可以了。

4. fork



The screenshot shows a web browser displaying the GitCode repository page for 'zzzzz77/第01周打卡'. The page includes a search bar, a navigation menu with '代码' (Code), 'Issues', 'Pull Requests', '讨论' (Discussion), '项目成员' (Project Members), '分析' (Analysis), and '项目设置' (Project Settings). Below the navigation menu, there are buttons for 'main', '分支' (Branches), 'Tags', and a search bar. A table lists the repository's files: .gitignore, LICENSE, and README.md, along with their commit history and last update times. A 'Clone' button is visible, and a 'Clone' dialog box is open, showing the repository URL and instructions for cloning using HTTPS or SSH.

按照老师给的步骤，点击 clone



The screenshot shows the 'Clone' dialog box in the GitCode repository page. The dialog box contains the following text:

```
# 请确保本地完成了 Git 的全局配置
git config --global user.name zzzzz77
git config --global user.email zzzzz77@noreply.gitcode.com

# 复制项目地址
https://gitcode.com/zzzzz77/week01.git

# 克隆到本地
git clone https://gitcode.com/zzzzz77/week01.git

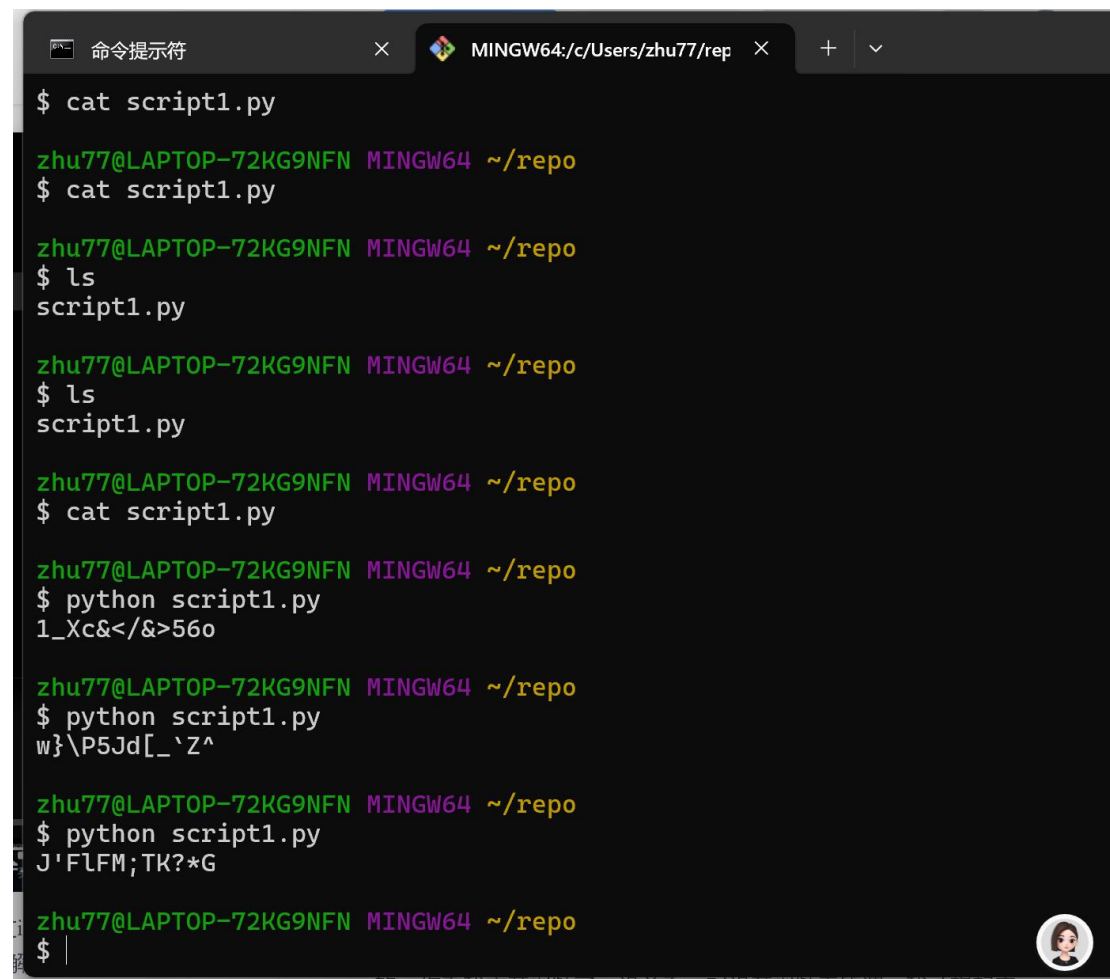
# 使用 HTTPS 协议时，请配置并使用个人访问令牌替代登录密码进行克隆、推送等操作
Username for 'https://gitcode.com': zzzzz77
Password for 'https://zzzzz77@gitcode.com': 个人访问令牌
```


然后复制上面两行代码，打开 vs code，右击粘贴，回车。

接着查看全部配置，用下面的命令，注意：空格不能少，不然会报错。

```
zhu77@LAPTOP-72KG9NFN MINGW64 ~  
$ git config --list --global  
core.editor="D:\zhu77\Microsoft VS Code\bin\code" --wait  
user.name=zzzzz77  
user.email=zzzzz77@noreply.gitcode.com  
  
zhu77@LAPTOP-72KG9NFN MINGW64 ~  
$ |
```

用 python 生成随机密码：



```
命令提示符  MINGW64:/c/Users/zhu77/rep  
$ cat script1.py  
  
zhu77@LAPTOP-72KG9NFN MINGW64 ~/repo  
$ cat script1.py  
  
zhu77@LAPTOP-72KG9NFN MINGW64 ~/repo  
$ ls  
script1.py  
  
zhu77@LAPTOP-72KG9NFN MINGW64 ~/repo  
$ ls  
script1.py  
  
zhu77@LAPTOP-72KG9NFN MINGW64 ~/repo  
$ cat script1.py  
  
zhu77@LAPTOP-72KG9NFN MINGW64 ~/repo  
$ python script1.py  
1_Xc&</&>56o  
  
zhu77@LAPTOP-72KG9NFN MINGW64 ~/repo  
$ python script1.py  
w}\P5Jd[_`Z^  
  
zhu77@LAPTOP-72KG9NFN MINGW64 ~/repo  
$ python script1.py  
J'FLFM;TK?*G  
  
zhu77@LAPTOP-72KG9NFN MINGW64 ~/repo  
$ |
```