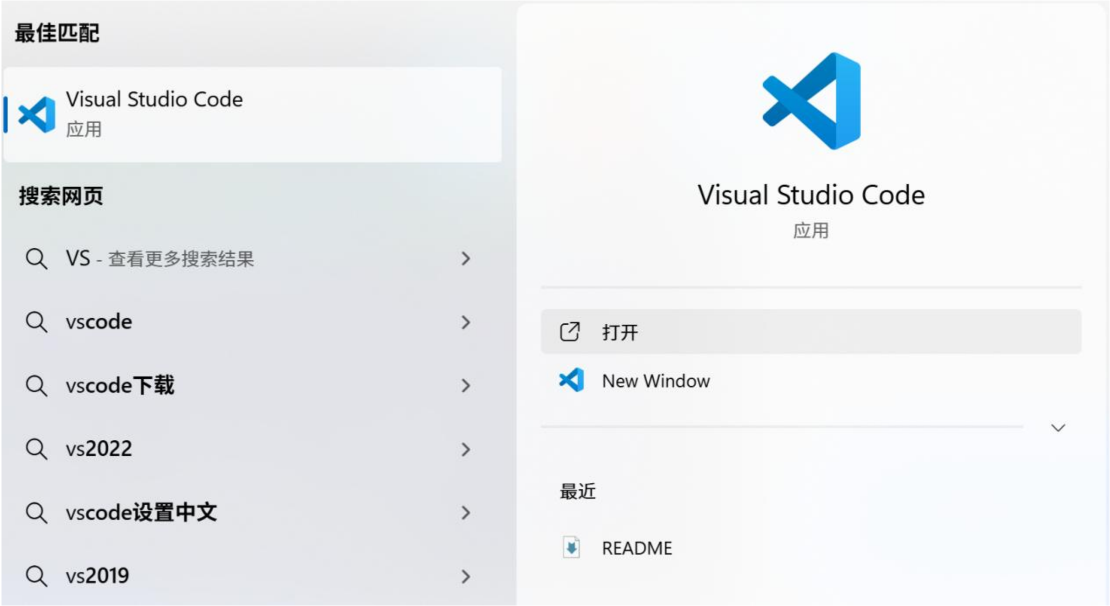
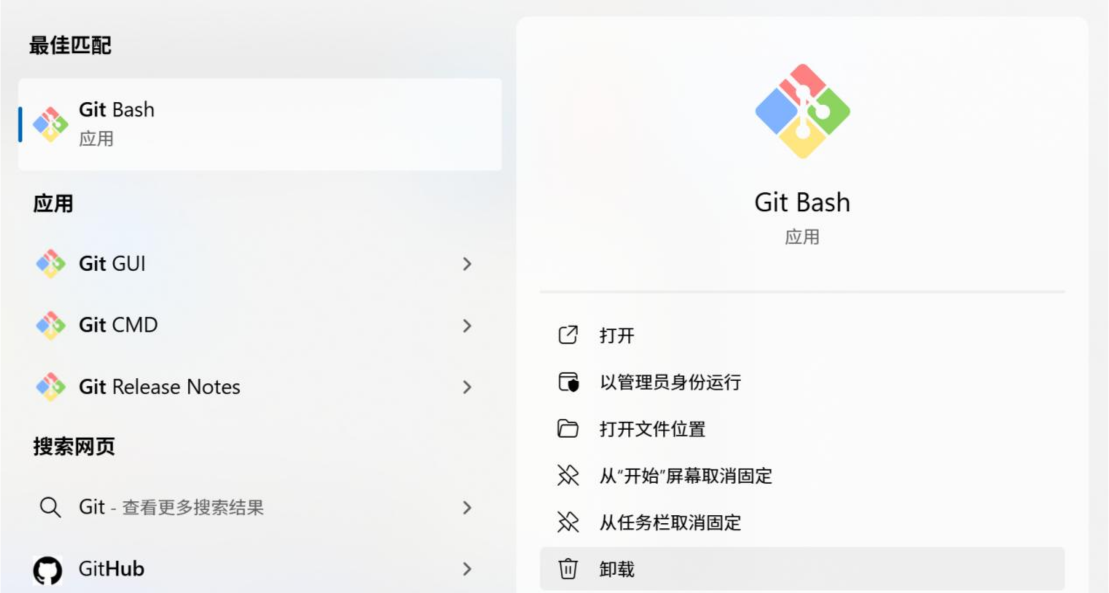
Week01 学习报告

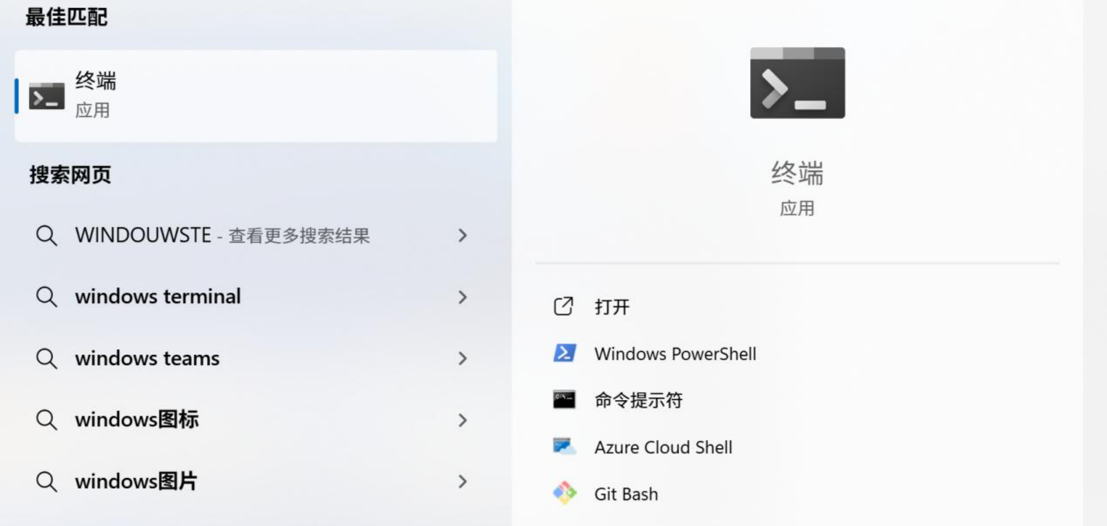
1、安装 VS Code 并把它固定在任务栏



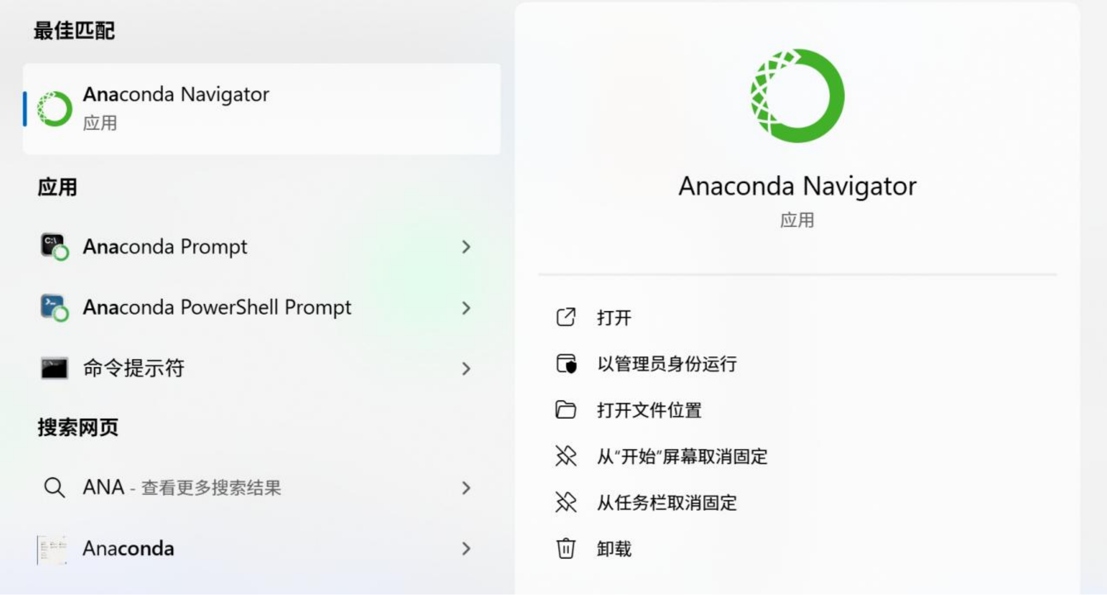
2、安装 Git 并把它固定在任务栏



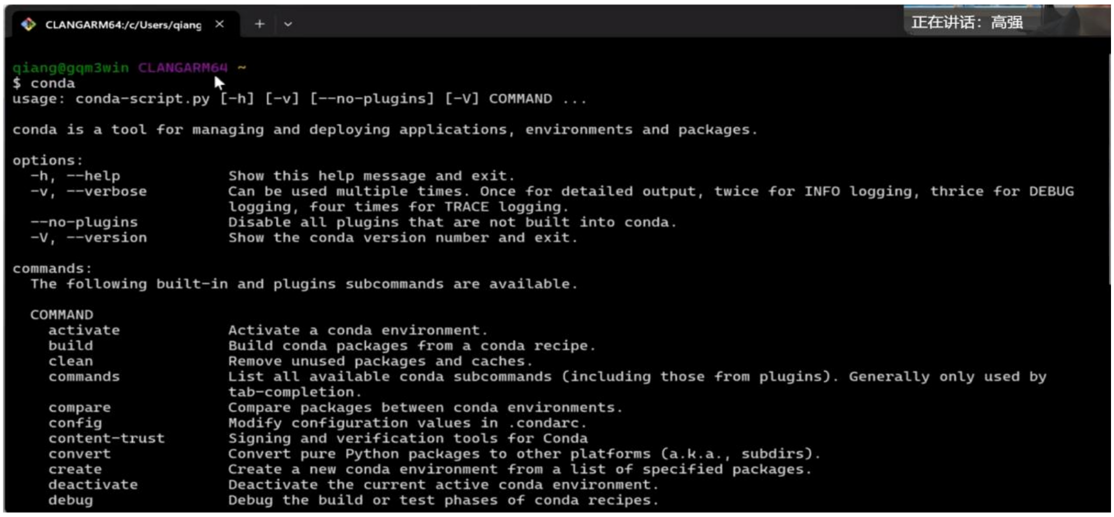
3、安装 Windows Terminal 并把它固定在任务栏



4、安装 Anaconda 并把它固定到任务栏（Python 解释器，Windows 客户）

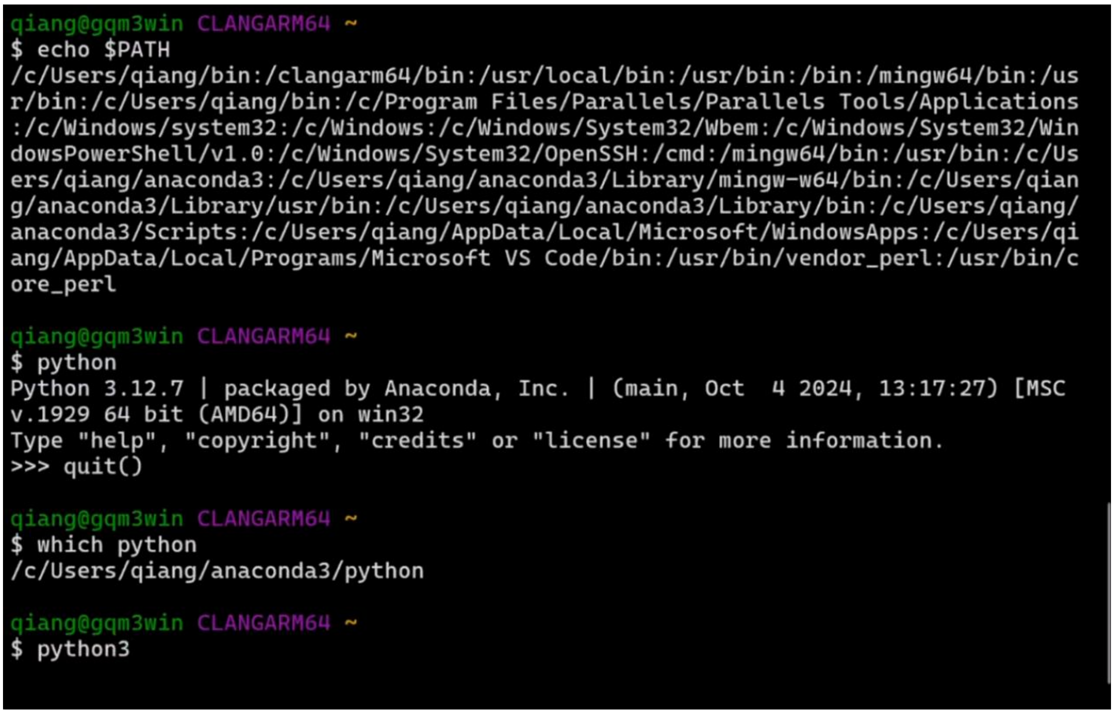


选择 Pass 环境变量可以在 Git 终端里使用 Anaconda。



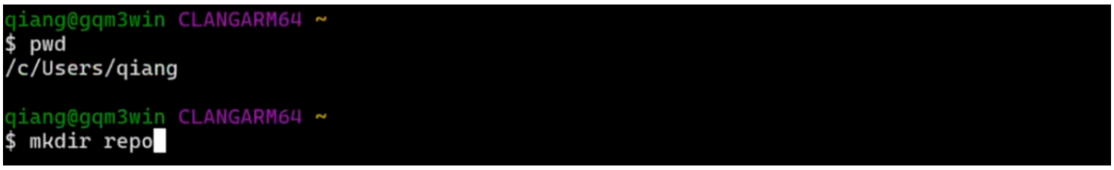
```
qiang@gqm3win CLANGARM64 ~
$ conda
usage: conda-script.py [-h] [-v] [--no-plugins] [-V] COMMAND ...

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, thrice for DEBUG
                        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
    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). Generally only used by
                        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.
```

Anaconda 和 Python 解释路径

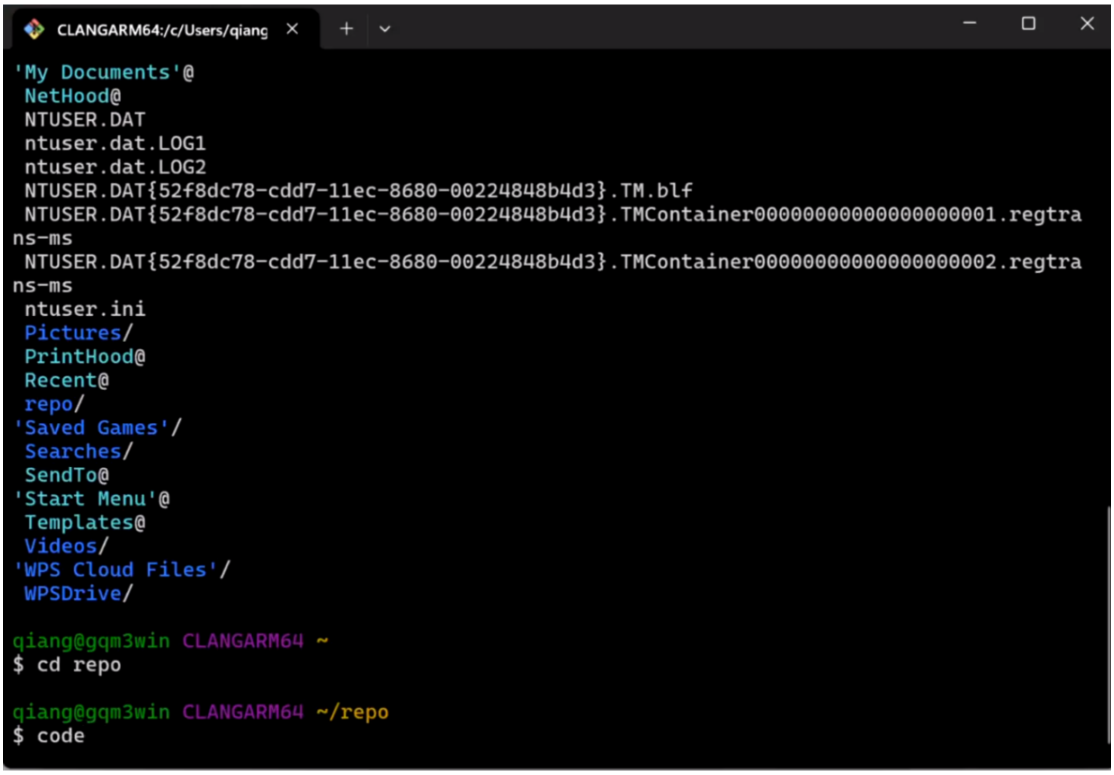


```
qiang@gqm3win CLANGARM64 ~
$ echo $PATH
/c/Users/qiang/bin:/clangarm64/bin:/usr/local/bin:/usr/bin:/bin:/mingw64/bin:/us
r/bin:/c/Users/qiang/bin:/c/Program Files/Parallels/Parallels Tools/Applications
:/c/Windows/system32:/c/Windows:/c/Windows/System32/Wbem:/c/Windows/System32/Win
dowsPowerShell/v1.0:/c/Windows/System32/OpenSSH:/cmd:/mingw64/bin:/usr/bin:/c/Us
ers/qiang/anaconda3:/c/Users/qiang/anaconda3/Library/mingw-w64/bin:/c/Users/qian
g/anaconda3/Library/usr/bin:/c/Users/qiang/anaconda3/Library/bin:/c/Users/qiang/
anaconda3/Scripts:/c/Users/qiang/AppData/Local/Microsoft/WindowsApps:/c/Users/qi
ang/AppData/Local/Programs/Microsoft VS Code/bin:/usr/bin/vendor_perl:/usr/bin/c
ore_perl

qiang@gqm3win CLANGARM64 ~
$ python
Python 3.12.7 | packaged by Anaconda, Inc. | (main, Oct  4 2024, 13:17:27) [MSC
v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> quit()

qiang@gqm3win CLANGARM64 ~
$ which python
/c/Users/qiang/anaconda3/python

qiang@gqm3win CLANGARM64 ~
$ python3
```

4、新建目录指令

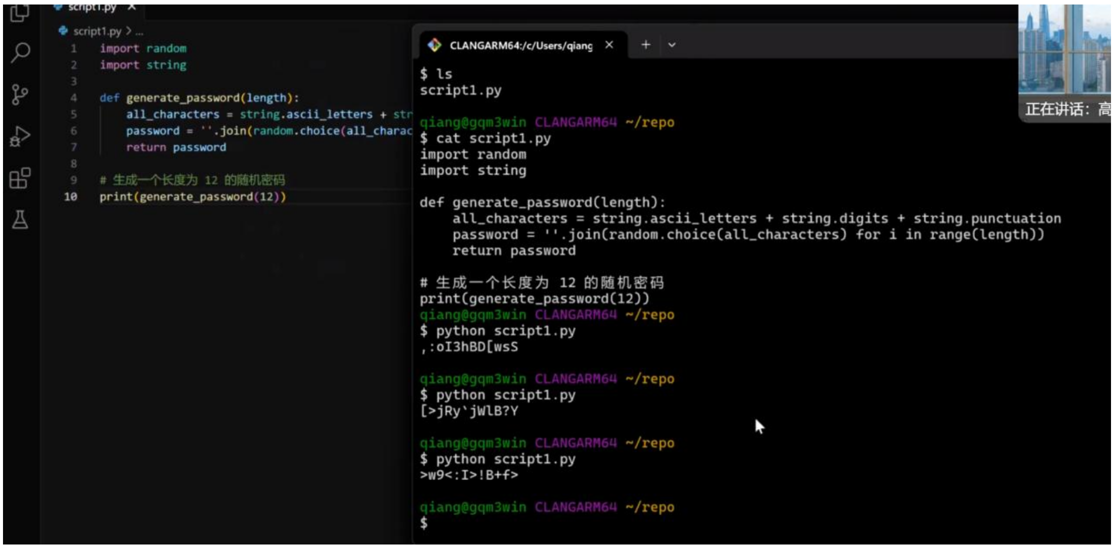


```
qiang@gqm3win CLANGARM64 ~
$ pwd
/c/Users/qiang

qiang@gqm3win CLANGARM64 ~
$ mkdir repo
```



```
'My Documents'@
 NetHood@
 NTUSER.DAT
 ntuser.dat.LOG1
 ntuser.dat.LOG2
 NTUSER.DAT{52f8dc78-cdd7-11ec-8680-00224848b4d3}.TM.blf
 NTUSER.DAT{52f8dc78-cdd7-11ec-8680-00224848b4d3}.TMContainer00000000000000000001.regtra
ns-ms
 NTUSER.DAT{52f8dc78-cdd7-11ec-8680-00224848b4d3}.TMContainer00000000000000000002.regtra
ns-ms
 ntuser.ini
 Pictures/
 PrintHood@
 Recent@
 repo/
'Saved Games'/
 Searches/
 SendTo@
'Start Menu'@
 Templates@
 Videos/
'WPS Cloud Files'/
 WPSDrive/

qiang@gqm3win CLANGARM64 ~
$ cd repo

qiang@gqm3win CLANGARM64 ~/repo
$ code
```

VS Code 使用豆包辅助生成代码



```python
import random
import string

def generate_password(length):
    all_characters = string.ascii_letters + str
    password = ''.join(random.choice(all_charac
    return password

# 生成一个长度为 12 的随机密码
print(generate_password(12))
```

```
$ ls
script1.py

qiang@gqm3win CLANGARM64 ~/repo
$ cat script1.py
import random
import string

def generate_password(length):
    all_characters = string.ascii_letters + string.digits + string.punctuation
    password = ''.join(random.choice(all_characters) for i in range(length))
    return password

# 生成一个长度为 12 的随机密码
print(generate_password(12))
qiang@gqm3win CLANGARM64 ~/repo
$ python script1.py
,:oI3hBD[wsS

qiang@gqm3win CLANGARM64 ~/repo
$ python script1.py
[>jRy`jWlB?Y

qiang@gqm3win CLANGARM64 ~/repo
$ python script1.py
>w9<:I>!B+f>

qiang@gqm3win CLANGARM64 ~/repo
$
```

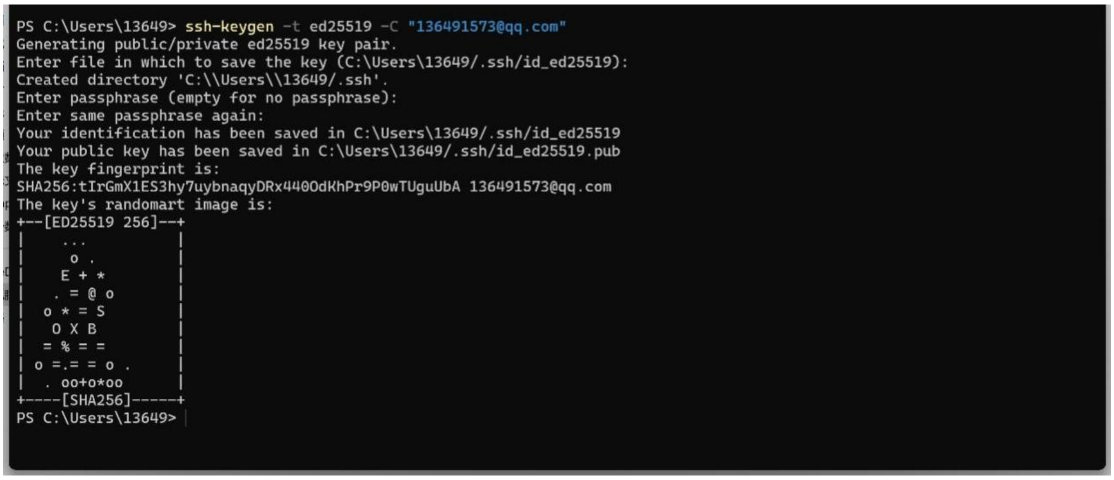
5、设置 SHH 密钥，可以与 Git Code 进行代码推送拉取，安全性高。

1. 打开终端
   - 在 Linux/macOS 上，打开系统自带的终端
   - 在 Windows 上，可以使用 Cmd、Power Shell 或 Git Bash
2. 输入生成密钥的命令

   运行以下命令，将 `your_email@example.com` 替换为您的邮箱：

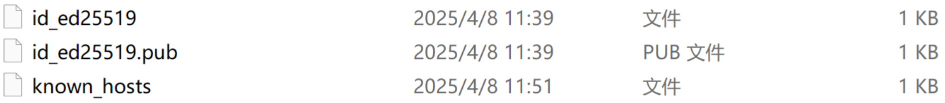
   ```
   ssh-keygen -t ed25519 -C "your_email@example.com"
   ```

   - **`-t ed25519`**：指定密钥类型为 ED25519。
   - **`-C "your_email@example.com"`**：添加注释，通常使用您的邮箱地址，便于识别密钥。

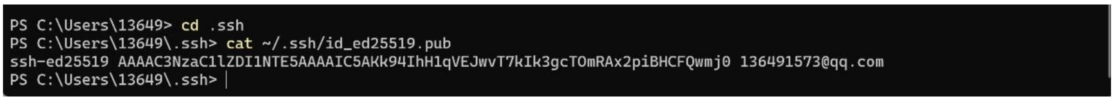


```
PS C:\Users\13649> ssh-keygen -t ed25519 -C "136491573@qq.com"
Generating public/private ed25519 key pair.
Enter file in which to save the key (C:\Users\13649/.ssh/id_ed25519):
Created directory 'C:\\Users\\13649/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in C:\Users\13649/.ssh/id_ed25519
Your public key has been saved in C:\Users\13649/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:tIrGmX1ES3hy7uybnaqyDRx440OdKhPr9P0wTUguUbA 136491573@qq.com
The key's randomart image is:
+--[ED25519 256]--+
|     ...         |
|      o .        |
|     E + *       |
|    . = @ o      |
|   o * = S       |
|    O X B        |
|   = % = =       |
|  o =.= = o .    |
|   . oo+o*oo     |
+----[SHA256]-----+
PS C:\Users\13649>
```

生成自己的密钥

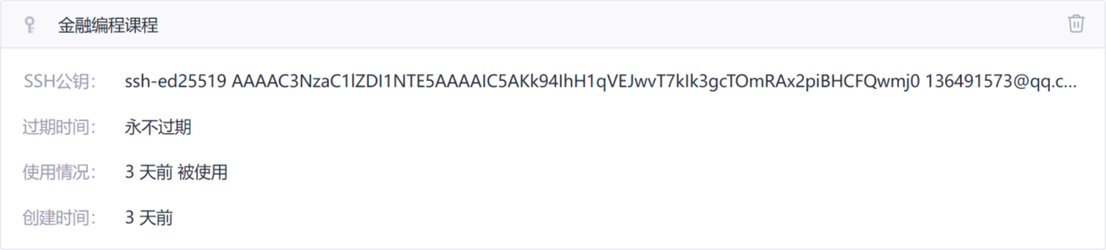


id_ed25519	2025/4/8 11:39	文件	1 KB
id_ed25519.pub	2025/4/8 11:39	PUB 文件	1 KB
known_hosts	2025/4/8 11:51	文件	1 KB



```
PS C:\Users\13649> cd .ssh
PS C:\Users\13649\.ssh> cat ~/.ssh/id_ed25519.pub
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIC5AKk94IhH1qVEJwvT7kIk3gcTOmRAx2piBHCFQwmj0 136491573@qq.com
PS C:\Users\13649\.ssh>
```

复制公钥，添加到 Git Code。



金融编程课程

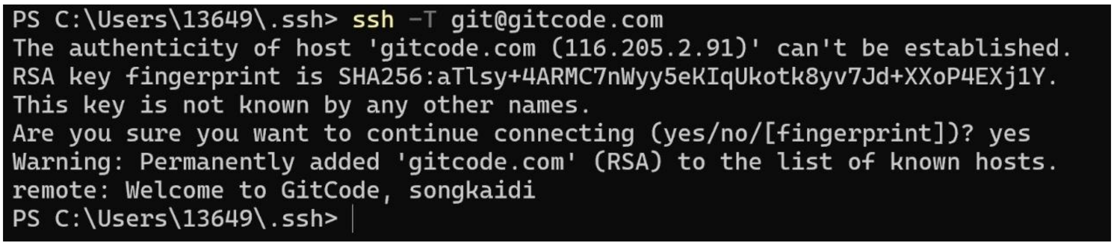
SSH公钥: ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIC5AKk94IhH1qVEJwvT7kIk3gcTOmRAx2piBHCFQwmj0 136491573@qq.c...

过期时间: 永不过期

使用情况: 3 天前 被使用

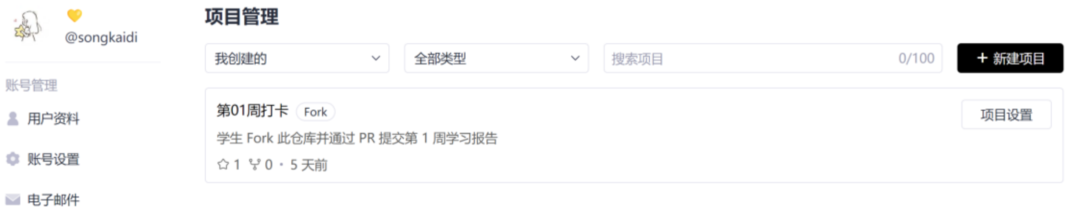
创建时间: 3 天前

在终端测试 SHH



```
PS C:\Users\13649\.ssh> ssh -T git@gitcode.com
The authenticity of host 'gitcode.com (116.205.2.91)' can't be established.
RSA key fingerprint is SHA256:aTlsy+4ARMC7nWyy5eKIqUkotk8yv7Jd+XXoP4EXj1Y.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'gitcode.com' (RSA) to the list of known hosts.
remote: Welcome to GitCode, songkaidi
PS C:\Users\13649\.ssh>
```

6、Fork

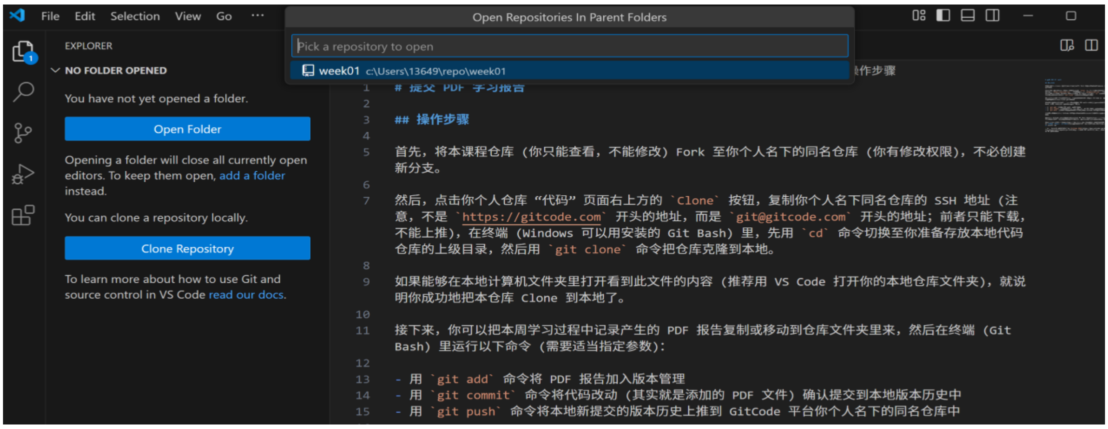


第01周打卡 Fork

学生 Fork 此仓库并通过 PR 提交第 1 周学习报告

☆ 1 ⑂ 0 · 5 天前

7、用 VS Code 打开你的本地仓库文件，克隆成功



Open Repositories In Parent Folders

Pick a repository to open

week01 c:\Users\13649\repo\week01

EXPLORER

NO FOLDER OPENED

You have not yet opened a folder.

Opening a folder will close all currently open editors. To keep them open, add a folder instead.

You can clone a repository locally.

To learn more about how to use Git and source control in VS Code read our docs.

```
# 提交 PDF 学习报告

## 操作步骤

首先，将本课程仓库（你只能查看，不能修改）Fork 至你个人名下的同名仓库（你有修改权限），不必创建新分支。

然后，点击你个人仓库“代码”页面右上方的 `Clone` 按钮，复制你个人名下同名仓库的 SSH 地址（注意，不是 `https://gitcode.com` 开头的地址，而是 `git@gitcode.com` 开头的地址；前者只能下载，不能上推），在终端（Windows 可以用安装的 Git Bash）里，先用 `cd` 命令切换至你准备存放本地代码仓库的上级目录，然后用 `git clone` 命令把仓库克隆到本地。

如果能够在本地计算机文件夹里打开看到此文件的内容（推荐用 VS Code 打开你的本地仓库文件夹），就说明你成功地把本仓库 Clone 到本地了。

接下来，你可以把本周学习过程中记录产生的 PDF 报告复制或移动到仓库文件夹里来，然后在终端（Git Bash）里运行以下命令（需要适当指定参数）:

- 用 `git add` 命令将 PDF 报告加入版本管理
- 用 `git commit` 命令将代码改动（其实就是添加的 PDF 文件）确认提交到本地版本历史中
- 用 `git push` 命令将本地新提交的版本历史上推到 GitCode 平台你个人名下的同名仓库中
```