

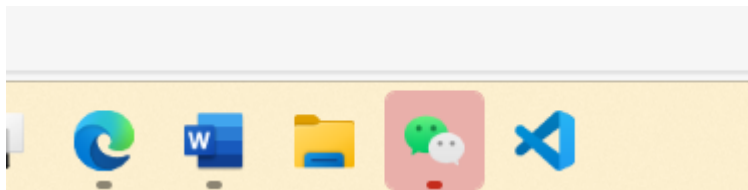
## 第一周学习报告

### ● 收藏网址

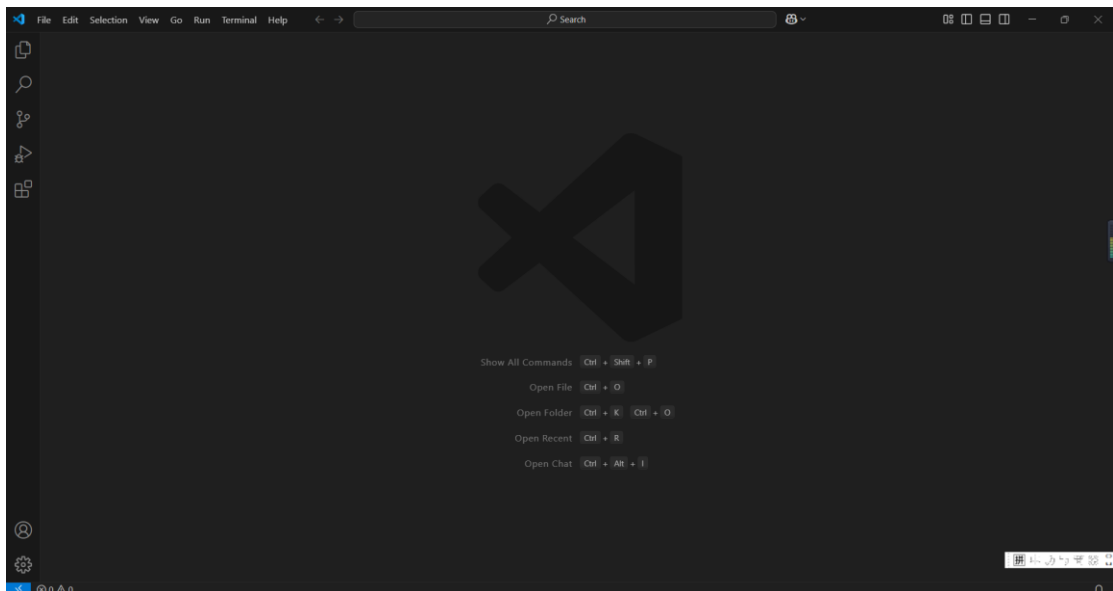


### ● 下载 visual studio code

#### 1. 下载并固定到任务栏

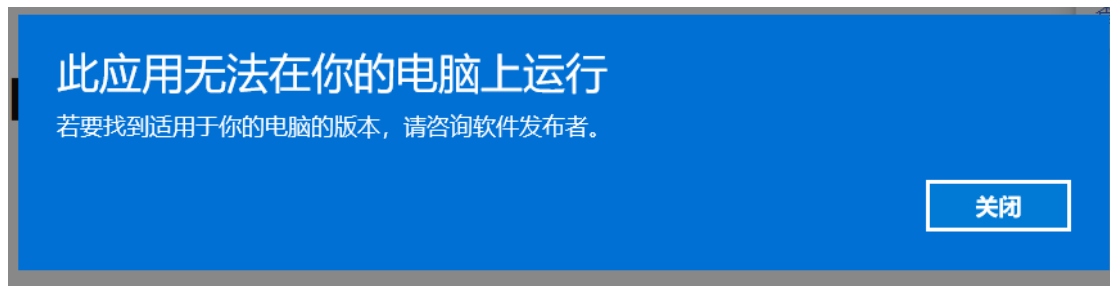


#### 2. 下载完成

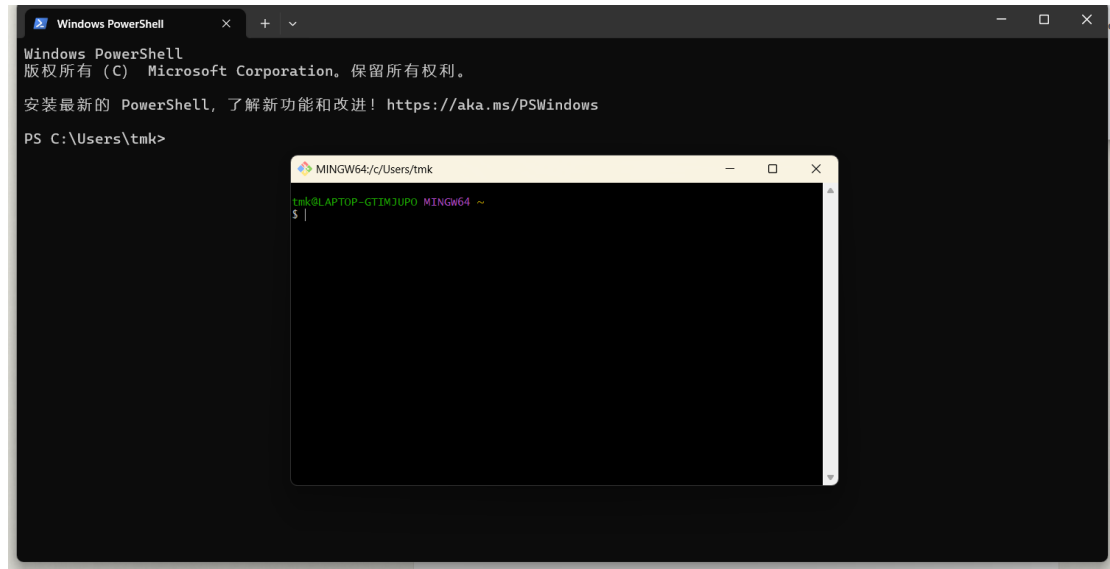


### ● 下载 git

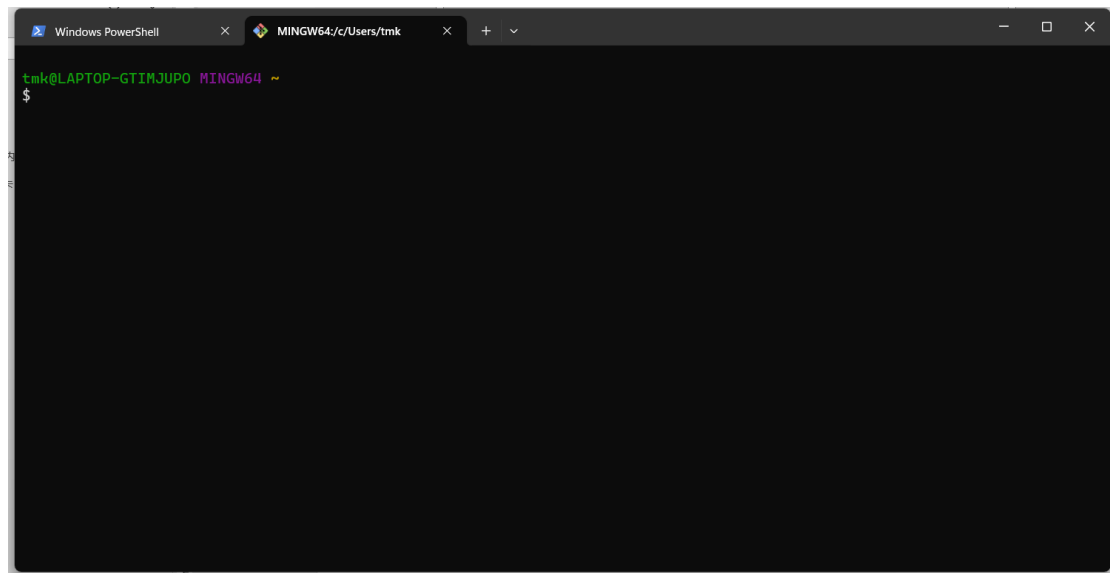
（开始下载完成后电脑提示“无法运行”，之后卸载了杀毒软件重新下载，等了一个小时后终于下载完成了）



## 1. 下载 terminal 和 git



## 2. 默认配置 git bash





```
MINGW64:/c/Users/tmk
tmk@LAPTOP-GTIMJUPO MINGW64 ~
$ |
```

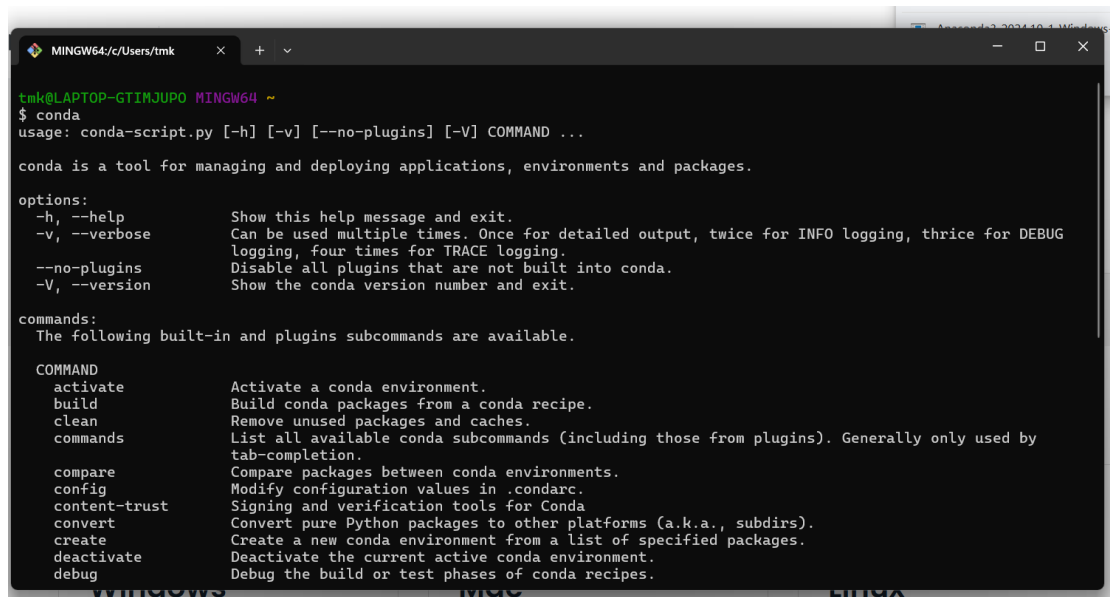
### 3.设置光标



```
MINGW64:/c/Users/tmk
tmk@LAPTOP-GTIMJUPO MINGW64 ~
$ █
```

## ● 下载 anaconda

### 1.下载完成并在终端打开



```
MINGW64:/c/Users/tmk
tmk@LAPTOP-GTIMJUPO MINGW64 ~
$ conda
usage: conda-script.py [-h] [-v] [--no-plugins] [-V] COMMAND ...

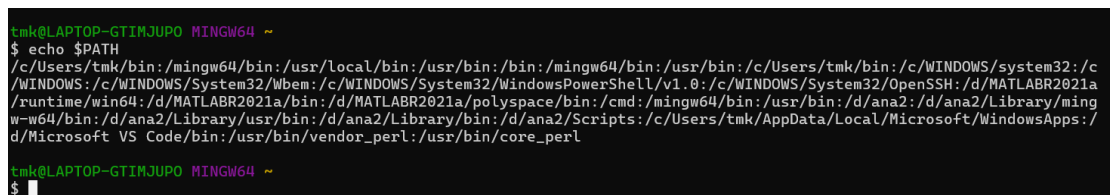
conda is a tool for managing and deploying applications, environments and packages.

options:
  -h, --help            Show this help message and exit.
  -v, --verbose          Can be used multiple times. Once for detailed output, twice for INFO logging, 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.
```

### 2.打开 PATH 环境变量



```
tmk@LAPTOP-GTIMJUPO MINGW64 ~
$ echo $PATH
/c/Users/tmk/bin:/mingw64/bin:/usr/local/bin:/usr/bin:/bin:/mingw64/bin:/usr/bin:/c/Users/tmk/bin:/c/WINDOWS/system32:/c/WINDOWS:/c/WINDOWS/System32/Wbem:/c/WINDOWS/System32/WindowsPowerShell/v1.0:/c/WINDOWS/System32/OpenSSH:/d/MATLABR2021a/runtime/win64:/d/MATLABR2021a/bin:/d/MATLABR2021a/polyspace/bin:/cmd:/mingw64/bin:/usr/bin:/d/ana2:/d/ana2/Library/mingw-w64/bin:/d/ana2/Library/usr/bin:/d/ana2/Library/bin:/d/ana2/Scripts:/c/Users/tmk/AppData/Local/Microsoft/WindowsApps:/d/Microsoft VS Code/bin:/usr/bin/vendor_perl:/usr/bin/core_perl
tmk@LAPTOP-GTIMJUPO MINGW64 ~
$ █
```

### 3.程序演示

```
mingw64: c:/Users/tmk
tmk@LAPTOP-GTIMJUPO MINGW64 ~
$ 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()
File "<stdin>", line 1
quit()
^
SyntaxError: invalid character ' (' (U+FF08)
>>> quit()

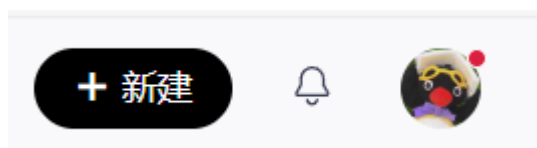
tmk@LAPTOP-GTIMJUPO MINGW64 ~
$ ipython
Python 3.12.7 | packaged by Anaconda, Inc. | (main, Oct 4 2024, 13:17:27) [MSC v.1929 64 bit (AMD64)]
Type 'copyright', 'credits' or 'license' for more information
IPython 8.27.0 -- An enhanced Interactive Python. Type '?' for help.

In [1]: print('hello')
hello

In [2]:
Do you really want to exit ([y]/n)? y

tmk@LAPTOP-GTIMJUPO MINGW64 ~
$
```

## ● 注册登录



## ● 生成并新建保存 SSH 密钥

```
mingw64: c:/Users/tmk/.ssh
o  + =.. |
. . 0.. |
+ = o.*..o+ |
oBo*=B.o..o=. |
----[SHA256]-----+

mk@LAPTOP-GTIMJUPO MINGW64 ~/repo
cd

mk@LAPTOP-GTIMJUPO MINGW64 ~
pwd
c:/Users/tmk

mk@LAPTOP-GTIMJUPO MINGW64 ~
cd .ssh
ash: cd.ssh: command not found

mk@LAPTOP-GTIMJUPO MINGW64 ~
cd .ssh

mk@LAPTOP-GTIMJUPO MINGW64 ~/.ssh
ls
d_ed25519 id_ed25519.pub

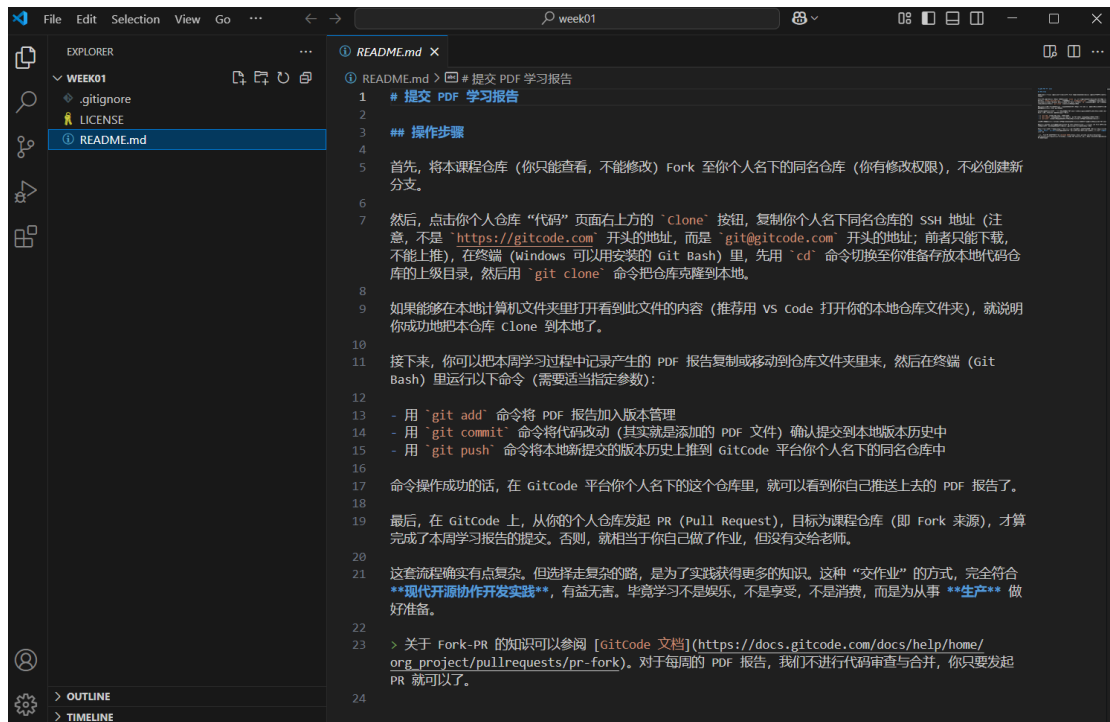
mk@LAPTOP-GTIMJUPO MINGW64 ~/.ssh
cat id_ed25519.pub
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIOhbXe/ImRpvwbsXJc+PhE2lBBWhaCf55KphdHj6ZkPF 2043724248@qq.com

mk@LAPTOP-GTIMJUPO MINGW64 ~/.ssh
```

## SSH 公钥

|              |  |
|--------------|--|
| 👤 tangmingke |  |
| SSH公钥:       | ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIOhbXe/ImRpvwbsXJc+PhE2lBBWhaCf55KphdHj6ZkPF 2043724248@qq.com |
| 过期时间:        | 永不过期   |
| 使用情况:        | 从未被使用  |
| 创建时间:        | 刚刚   |

## ● 完成克隆



```
1 # 提交 PDF 学习报告
2
3 ## 操作步骤
4
5 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。
6
7 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。
8
9 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。
10
11 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）：
12
13 - 用“git add”命令将 PDF 报告加入版本管理
14 - 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
15 - 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
16
17 命令操作成功的话，在 GitCode 平台你个人名下的这个仓库里，就可以看到你自己推送上去的 PDF 报告了。
18
19 最后，在 GitCode 上，从你的个人仓库发起 PR（Pull Request），目标为课程仓库（即 Fork 来源），才算完成了本周学习报告的提交。否则，就相当于你自己做了作业，但没有交给老师。
20
21 这套流程确实有点复杂。但选择走复杂的路，是为了实践获得更多的知识。这种“交作业”的方式，完全符合**现代开源协作开发实践**，有益无害。毕竟学习不是娱乐，不是享受，不是消费，而是为从事**生产**做好准备。
22
23 > 关于 Fork-PR 的知识可以参阅【GitCode 文档】(https://docs.gitcode.com/docs/help/home/org_project/pullrequests/pr-fork)。对于每周的 PDF 报告，我们不进行代码审查与合并，你只要发起 PR 就可以了。
24
```