

1. Fork [第 04 周打卡](#) 仓库至你的名下，然后将你名下的这个仓库 Clone 到你的本地计算机

```
$ cd
(base) 86152@LAPTOP-QOK67RV0 MINGW64 ~
$ cd repo
(base) 86152@LAPTOP-QOK67RV0 MINGW64 ~/repo
$ ls -l
total 16
drwxr-xr-x 1 86152 197609 0 Mar 20 15:52 myproject/
drwxr-xr-x 1 86152 197609 0 Mar 20 14:11 prj1/
drwxr-xr-x 1 86152 197609 0 Mar 14 22:06 week01/
drwxr-xr-x 1 86152 197609 0 Mar 16 20:24 week02/
drwxr-xr-x 1 86152 197609 0 Mar 20 17:49 week03/
(base) 86152@LAPTOP-QOK67RV0 MINGW64 ~/repo
$ git clone https://gitcode.com/fcl22024110158/week04.git
Cloning into 'week04'...
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 5 (delta 0), reused 5 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (5/5), 8.43 KiB | 169.00 KiB/s, done.
(base) 86152@LAPTOP-QOK67RV0 MINGW64 ~/repo
$ cd week04/
(base) 86152@LAPTOP-QOK67RV0 MINGW64 ~/repo/week04 (main)
$
```

查看远程仓库的地址

```
/C:/Users/86152/repo/week04
(base) 86152@LAPTOP-QOK67RV0 MINGW64 ~/repo/week04 (main)
$ git remote show origin
* remote origin
  Fetch URL: https://gitcode.com/fcl22024110158/week04.git
  Push URL: https://gitcode.com/fcl22024110158/week04.git
  HEAD branch: main
  Remote branch:
    main tracked
  Local branch configured for 'git pull':
    main merges with remote main
  Local ref configured for 'git push':
    main pushes to main (up to date)
(base) 86152@LAPTOP-QOK67RV0 MINGW64 ~/repo/week04 (main)
$
```

2. 用 VS Code 打开项目目录，新建一个 `environment.yml` 文件，指定安装 Python 3.12，然后运行 `conda env create` 命令创建 Conda 环境

```
! environment.yml U
! environment.yml
1 name: week04
2 channels:
3   - conda-forge
4 dependencies:
5   - python=3.12
6

-PW-R--R-- 1 86152 197609 87 Apr 7 14:37 environment.yml
(base) 86152@LAPTOP-QOK67RVO MINGW64 ~/repo/week04 (main)
$ cat environment.yml
name: week04
channels:
- conda-forge
dependencies:
- python=3.12
(base) 86152@LAPTOP-QOK67RVO MINGW64 ~/repo/week04 (main)
$ conda env create
D:\Anaconda3_install\Lib\argparse.py:2006: FutureWarning: `remote`
deprecated and will be removed in 25.9. Use `conda env create --
ad.
    action(self, namespace, argument_values, option_string)
Retrieving notices: ...working... done
Channels:
- conda-forge
- https://repo.anaconda.com/pkgs/main
- https://repo.anaconda.com/pkgs/r
- https://repo.anaconda.com/pkgs/msys2
Platform: win-64
```

3.新建一个 `contacts.txt` 文件，每行写一个联系人，每个联系人都包含姓名、性别、邮箱三个字段，用空格分隔，例如

```
郭芙蓉 女 guofurong@126.com
李秀莲 男 lixiulian@163.com
祝无双 女 zhuwushuang@163.com(base) 86152@LAPTOP-QOK67RV0 MINGW64
(main)
$ cat contacts.txt environment.yml
白展堂 男 baizhantang@163.com
佟湘玉 女 tongxiangyu@163.com
吕轻侯 男 lvqinghou@126.com
郭芙蓉 女 guofurong@126.com
李秀莲 男 lixiulian@163.com
祝无双 女 zhuwushuang@163.comname: week04
channels:
- conda-forge
dependencies:
- python=3.12
(base) 86152@LAPTOP-QOK67RV0 MINGW64 ~/repo/week04 (main)
$ |
```

建议活学活用，改换其他例子 (甚至是一些极端情况) 测试程序的稳健性

4、 新建一个 `main.py` 文件，里面写 Python 代码，要求读取 `contacts.txt` 文件的内容，进行数据处理后，输出一个 `emails.txt` 文件，例如

要求输出是先按邮箱域名排序 (`126.com` 排在 `163.com` 之前)，然后再按邮箱用户名排序 (`guofurong` 排在 `lvqinghou` 之前)

5.可以将以上“任务要求”的文本，复制粘贴到大模型 (比如豆包、DeepSeek) 里，请 AI 来帮助编写程序初稿

2. 编写 `main.py` 文件

在 VS Code 中新建 `main.py` 文件，输入以下 Python 代码：

```
python ^

def read_contacts():
    contacts = []
    with open('contacts.txt', 'r', encoding='utf-8') as file:
        for line in file:
            name, gender, email = line.strip().split()
            contacts.append((name, gender, email))
```

图像生成 AI 编程 解题答疑 AI 搜索 帮我写作

```
main.py > read_contacts
1 def read_contacts():
2     contacts = []
3     with open('contacts.txt', 'r', encoding='utf-8') as file:
4         for line in file:
5             name, gender, email = line.strip().split()
6             contacts.append((name, gender, email))
7     return contacts
8
9 def generate_emails(contacts):
10     emails = []
11     for name, gender, email in contacts:
12         title = "先生" if gender == "男" else "小姐"
13         email_content = f"to: <{email}>\nsubject: {title}"
14         emails.append(email_content)
15     return emails
```

6. AI 回复的只是静态代码，而且可能含有错误，所以我们必须在 Conda 环境里运行代码，逐行调试，检查每一行代码的运行都符合我们的期望 (越是初学者越应该慢慢调试、检查、试验，借此学习)

7. 将你学习理解实践这些概念所产生的笔记，以及试验性的代码，都 `add`、`commit`、`push` 到 **GitCode** 平台你名下的仓库里，最后提交 **PR**