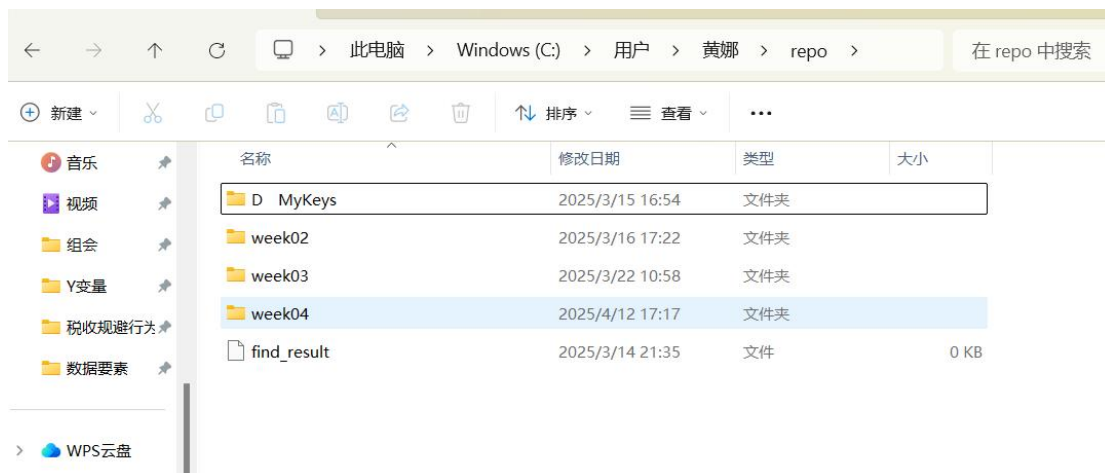


将本课程仓库 Fork 至个人名下的同名仓库



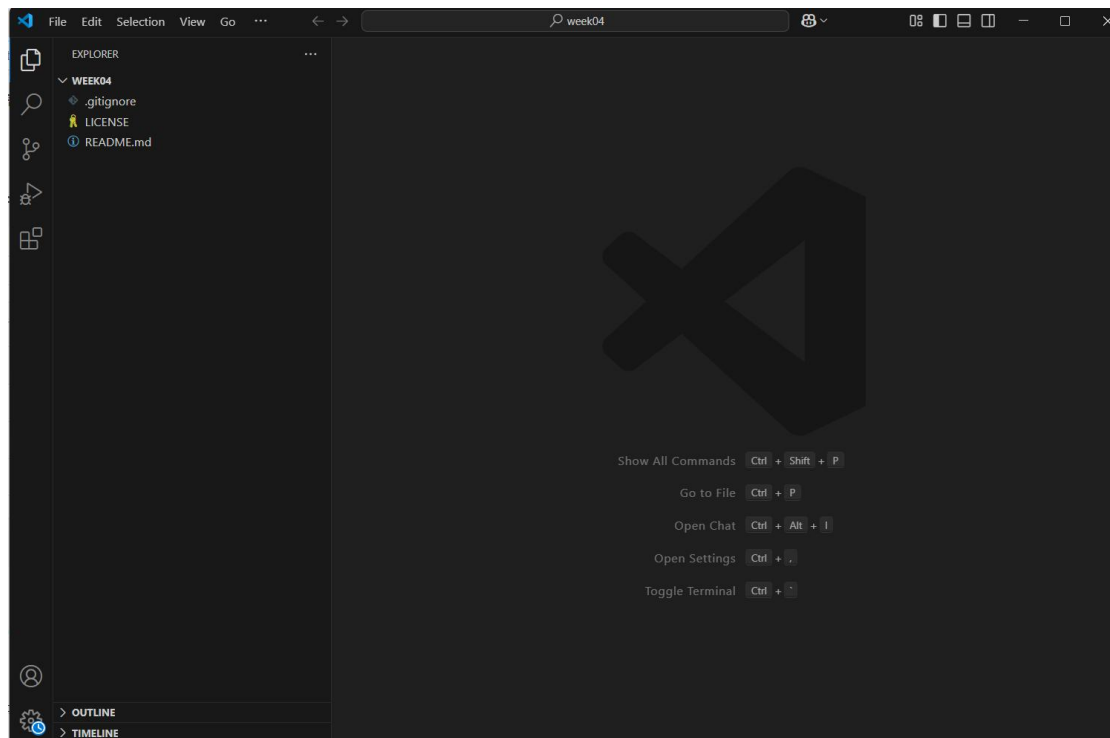
```
(base) huangna@LAPTOP-ANR0AIJ2 MINGW64 ~/repo
$ cd week04/
(base) huangna@LAPTOP-ANR0AIJ2 MINGW64 ~/repo/week04 (main)
$ pwd
/c/Users/huangna/repo/week04
(base) huangna@LAPTOP-ANR0AIJ2 MINGW64 ~/repo/week04 (main)
$ git remote show origin
* remote origin
Fetch URL: https://gitcode.com/superqiqi/week04.git
Push URL: https://gitcode.com/superqiqi/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) huangna@LAPTOP-ANR0AIJ2 MINGW64 ~/repo/week04 (main)
$
```

用 VS Code 打开项目目录，新建一个 environment.yml 文件，指定安装 Python3.12.然后运行 condaenv create 命令创建 conda 环境

```
$ pwd
/c/Users/huangna/repo/week04
(base) huangna@LAPTOP-ANR0AIJ2 MINGW64 ~/repo/week04 (main)
$ code.
bash: code.: command not found
(base) huangna@LAPTOP-ANR0AIJ2 MINGW64 ~/repo/week04 (main)
$ code .
node:internal/modules/cjs/loader:1235
  throw err;
  ^

Error: Cannot find module 'D:\a\Library\d\作业\Microsoft VS Code\resources\app\out\cli.js'
    at Module._resolveFilename (node:internal/modules/cjs/loader:1232:15)
    at Module._load (node:internal/modules/cjs/loader:1058:27)
    at c._load (node:electron/js2c/node_init:2:16955)
    at Function.executeUserEntryPoint [as runMain] (node:internal/modules/run_main:188:12)
    at node:internal/main/run_main_module:28:49 {
  code: 'MODULE_NOT_FOUND',
  requireStack: []
}

Node.js v20.18.3
(base) huangna@LAPTOP-ANR0AIJ2 MINGW64 ~/repo/week04 (main)
$
```



```
(base) huangna@LAPTOP-ANR0AIJ2 MINGW64 ~/repo/week04 (main)
$ pwd
/c/Users/huangna/repo/week04
(base) huangna@LAPTOP-ANR0AIJ2 MINGW64 ~/repo/week04 (main)
$ ls -l
total 24
-rw-r--r-- 1 huangna 197121 18805 Apr 12 17:17 LICENSE
-rw-r--r-- 1 huangna 197121 2239 Apr 12 17:17 README.md
-rw-r--r-- 1 huangna 197121 0 Apr 12 17:35 environment.yml
(base) huangna@LAPTOP-ANR0AIJ2 MINGW64 ~/repo/week04 (main)
$ cat environment.yml
(base) huangna@LAPTOP-ANR0AIJ2 MINGW64 ~/repo/week04 (main)
$ ^C
(base) huangna@LAPTOP-ANR0AIJ2 MINGW64 ~/repo/week04 (main)
$ cat environment.yml
name:week04
channels:
  -conda-forge
dependencies:
  -python=3.12
(base) huangna@LAPTOP-ANR0AIJ2 MINGW64 ~/repo/week04 (main)
$ |
```

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

```
(base) huangna@LAPTOP-ANR0AIJ2 MINGW64 ~/repo/week04 (main)
$ cat contacts.txt
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.com
(base)
```

```
(base) huangna@LAPTOP-ANR0AIJ2 MINGW64 ~/repo/week04 (main)
$ cat contacts.txt
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.com
$ 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
```

新建一个 main.py 文件，里面写 python 代码，要求读取 contact.txt 文件的内容，进行数据处理后，输出一个 emails.txt 文件

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

The screenshot shows a code editor with a file named `main.py` open. The code reads from `contacts.txt` and writes to `emails.txt`. The `contacts.txt` file contains contact information for several people, including Baizhantang, Tongxiangyu, Lvqinghou, Guofurong, Lixiulian, and Zhuwushuang. The `main.py` script uses a try-except block to handle file operations and includes a comment in Chinese explaining the task.

```
1 def generate_emails():
2     try:
3         with open('contacts.txt', 'r', encoding='utf-8') as file:
4             contacts = []
5             for line in file:
6                 name, gender, email = line.strip().split()
7                 contacts.append((name, gender, email))
8
9             contacts.sort(key=lambda x: (x[2].split('@')[1], x[2].split('@')[0]))
10
11         with open('emails.txt', 'w', encoding='utf-8') as output_file:
12             for name, gender, email in contacts:
13                 title = '先生' if gender == '男' else '女士'
14                 output_file.write(f'to: <{email}>\n')
15                 output_file.write(f'尊敬的{name}{{title}}, 您的会员资格即将到期, 请及时续费.\n')
16                 output_file.write('---\n')
17     except FileNotFoundError:
18         print("错误: 未找到 'contacts.txt' 文件.")
19     except Exception as e:
20         print(f"错误: 发生了一个未知错误: {e}")
21
22
23 if __name__ == "__main__":
24     generate_emails()
25
```

On the left side of the editor, there is a chat window with the following text:

你可以把上述代码保存为 main.py 文件，将示例联系人信息保存为 contacts.txt 文件，二者放在同一目录下运行，就会生成 emails.txt 文件。

这段代码的执行结果是什么？ →

如何在生成的邮件提醒中添加联系人的电话号码？ →

除了Python，还有哪些编程语言可以实现这个功能？ →

At the bottom, there are buttons for "帮我写作" (Help me write), "图像生成" (Image generation), "AI 搜索" (AI search), "AI 阅读" (AI reading), and "更多" (More).