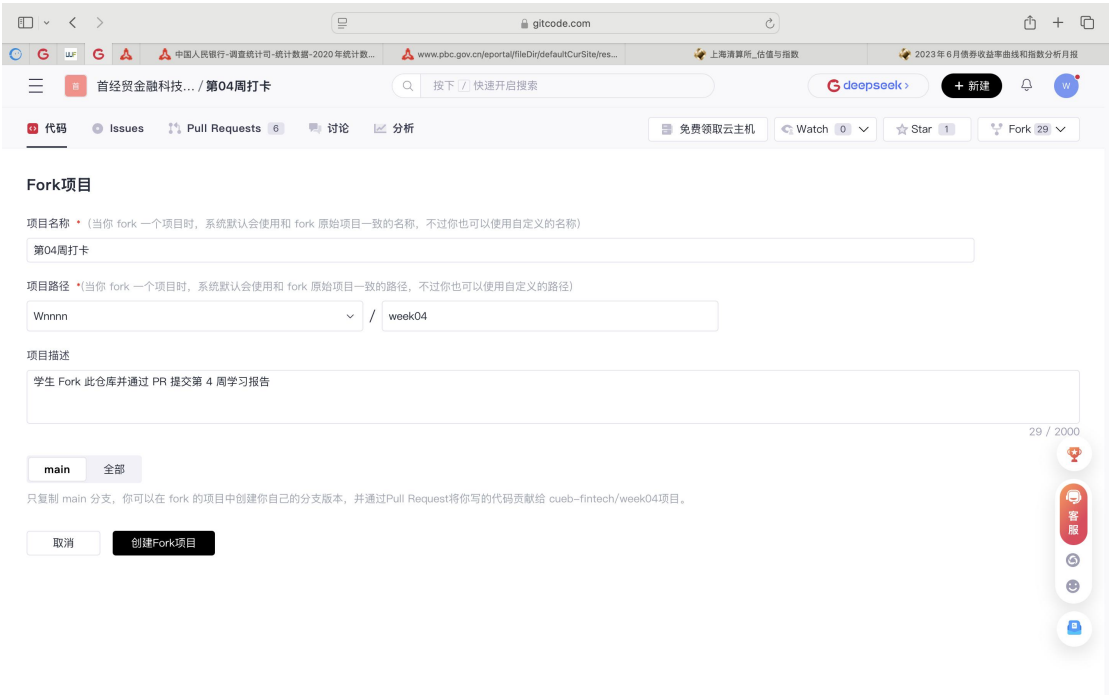
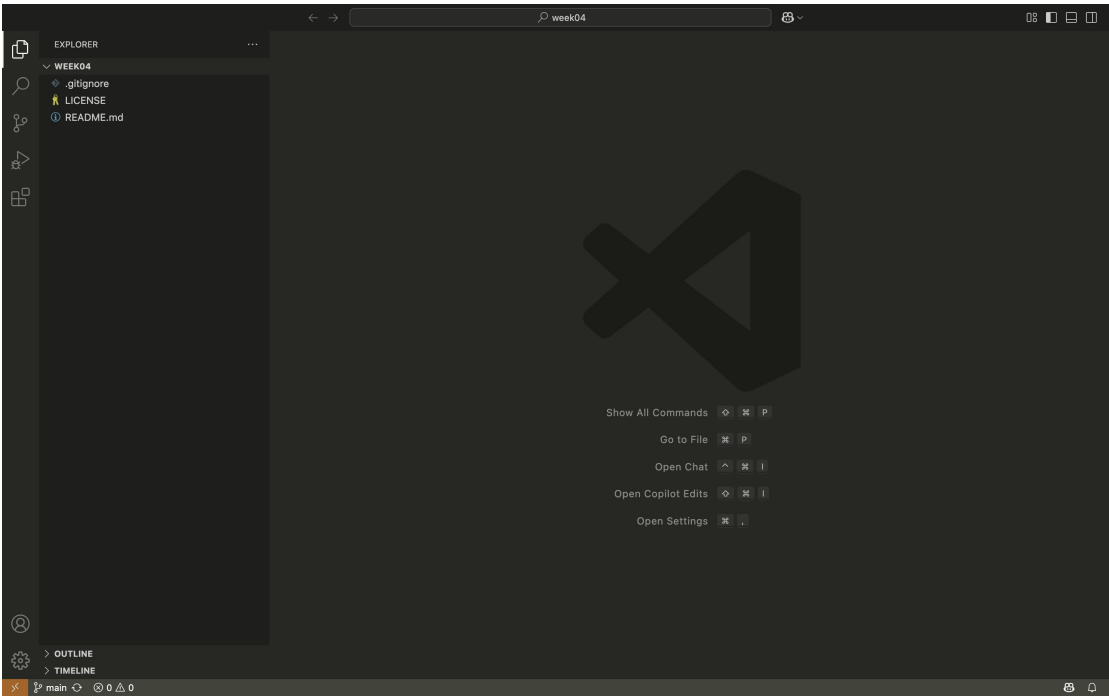


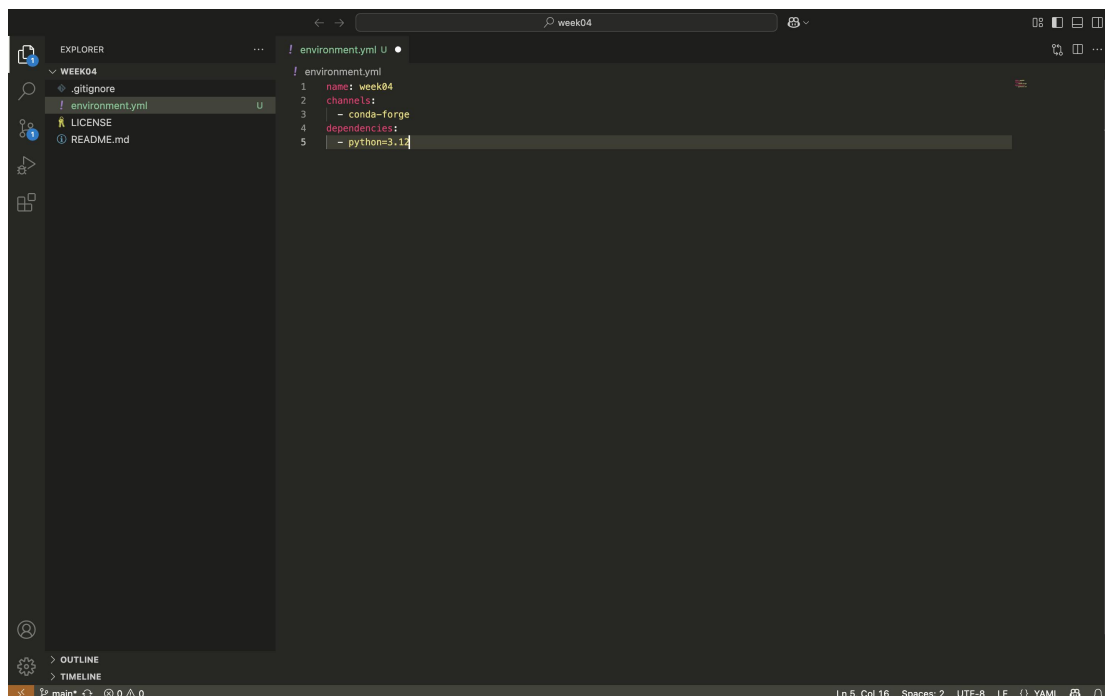
1. 创建 fork



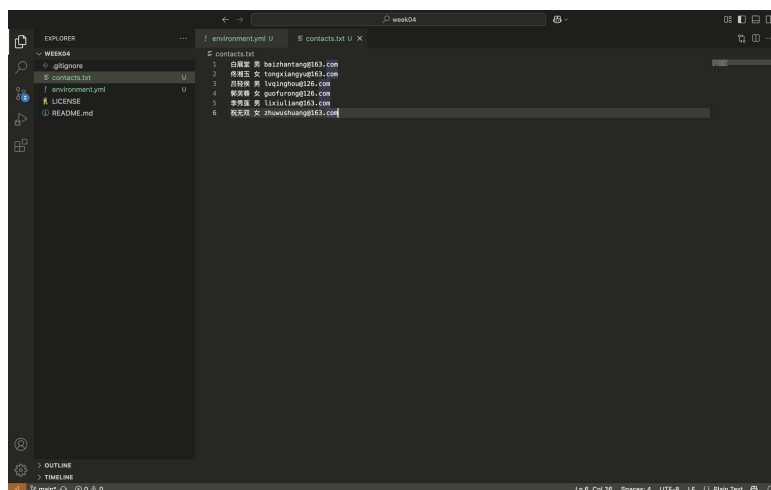
2. 用 VS Code 打开项目目录, 新建一个 `environment.yml` 文件, 指定安装 Python 3.12, 然后运行 `conda env create` 命令创建 Conda 环境



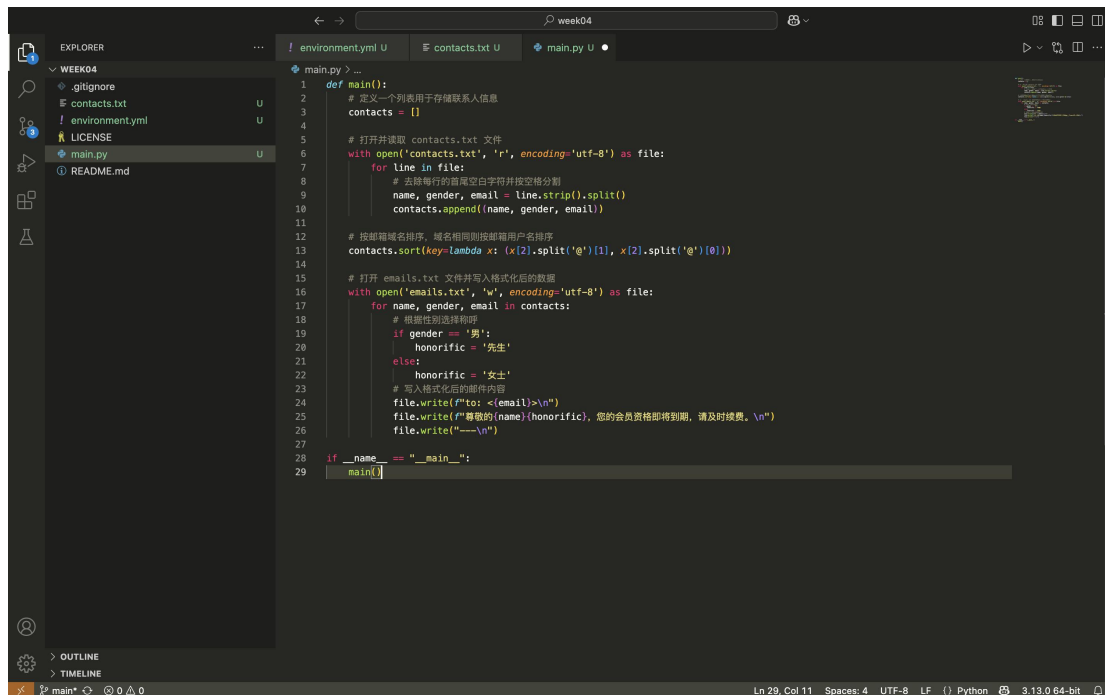
```
(base) mac@macdeMacBook-Pro-2 week04 % ls -l ../myproject
total 393704
-rw-r--r--  1 mac  staff   201568176  3 23 20:51 EPA_SmartLocationDatabase_V3_Jan_2021_Final.csv
-rw-r--r--@  1 mac  staff           82  3 23 20:01 environment.yml
-rw-r--r--@  1 mac  staff          670  3 23 20:56 main.py
(base) mac@macdeMacBook-Pro-2 week04 % cat ../myproject/environment.yml
name: myproject
channels:
  - conda-forge
dependencies:
  - python=3.12
  - pandas
(base) mac@macdeMacBook-Pro-2 week04 % cp ../myproject/environment.yml ./
(base) mac@macdeMacBook-Pro-2 week04 % ls -l
total 56
-rw-r--r--  1 mac  staff   18411  3 28 11:39 LICENSE
-rw-r--r--  1 mac  staff    2216  3 28 11:39 README.md
-rw-r--r--@  1 mac  staff     82  3 28 11:49 environment.yml
(base) mac@macdeMacBook-Pro-2 week04 %
```



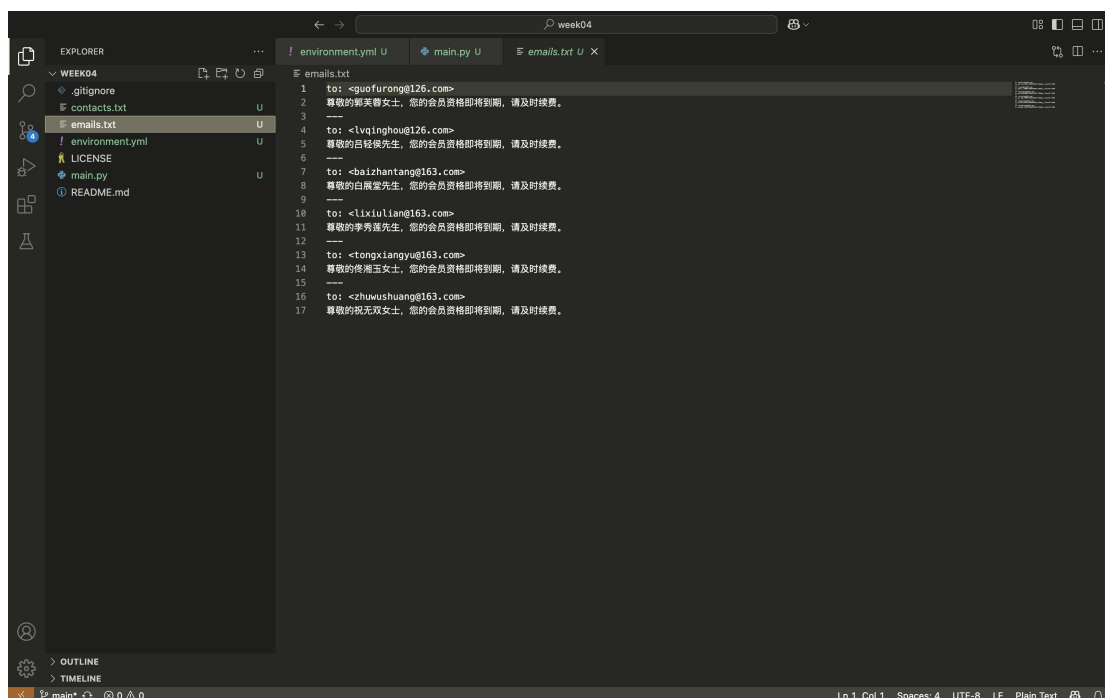
3. 新建一个 contacts.txt 文件



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



```
1 def main():
2     # 定义一个列表用于存储联系人信息
3     contacts = []
4
5     # 打开并读取 contacts.txt 文件
6     with open('contacts.txt', 'r', encoding='utf-8') as file:
7         for line in file:
8             # 去除每行的首尾空白字符并按空格分割
9             name, gender, email = line.strip().split()
10            contacts.append([name, gender, email])
11
12    # 按邮箱域名排序, 域名相同则按邮箱用户名排序
13    contacts.sort(key=lambda x: (x[2].split('@')[1], x[2].split('@')[0]))
14
15    # 打开 emails.txt 文件并写入格式化后的数据
16    with open('emails.txt', 'w', encoding='utf-8') as file:
17        for name, gender, email in contacts:
18            # 根据性别选择称谓
19            if gender == '男':
20                honorific = '先生'
21            else:
22                honorific = '女士'
23            # 写入格式化后的邮件内容
24            file.write(f"to: <{email}>\n")
25            file.write(f"尊敬的{name}{honorific}, 您的会员资格即将到期, 请及时续费.\n")
26            file.write("-----\n")
27
28 if __name__ == "__main__":
29     main()
```



```
1 to: <guofurong@126.com>
2 尊敬的郭芙蓉女士, 您的会员资格即将到期, 请及时续费。
3 -----
4 to: <luqinhuang@126.com>
5 尊敬的吕钦侯先生, 您的会员资格即将到期, 请及时续费。
6 -----
7 to: <baizhantang@163.com>
8 尊敬的白展堂先生, 您的会员资格即将到期, 请及时续费。
9 -----
10 to: <lixiliang@163.com>
11 尊敬的李秀莲先生, 您的会员资格即将到期, 请及时续费。
12 -----
13 to: <tongxiangyu@163.com>
14 尊敬的佟湘玉女士, 您的会员资格即将到期, 请及时续费。
15 -----
16 to: <zhuwushuang@163.com>
17 尊敬的祝无双女士, 您的会员资格即将到期, 请及时续费。
```

5. python 调试, 在 (pdb) 提示符下练习使用 l(显示代码)、n (执行当前行)、p (打印表达式)、s (步入调用)、pp (美观打印)、c (继续执行) 等命令

```
[(Pdb) n
> /Users/mac/repo/week04/main.py(4)<module>()
-> print(f"当前工作目录: {os.getcwd()}")
```

```

(Pdb) l
1  -> import os
2
3      # 打印当前工作目录
4      print(f"当前工作目录: {os.getcwd()}")
5
6      try:
7          with open("contacts.txt", "r", encoding="utf-8") as file:
8              contacts = file.readlines()
9
10         # 处理联系人数据
11         processed_contacts = []
(Pdb) █

```

```

[(Pdb) p
*** SyntaxError: invalid syntax
(Pdb) █

```

加范围

```

[(Pdb) l .
1      import os
2
3      # 打印当前工作目录
4      print(f"当前工作目录: {os.getcwd()}")
5
6  -> try:
7      with open("contacts.txt", "r", encoding="utf-8") as file:
8          contacts = file.readlines()
9
10         # 处理联系人数据
11         processed_contacts = []
(Pdb) █

(Pdb) l 30,39
30
31         # 去除最后一个多余的分隔符
32         if email_content:
33             email_content.pop()
34
35         # 写入邮件文件
36         with open("emails.txt", "w", encoding="utf-8") as file:
37             file.write("\n".join(email_content))
38
39     except FileNotFoundError:
(Pdb) █

```

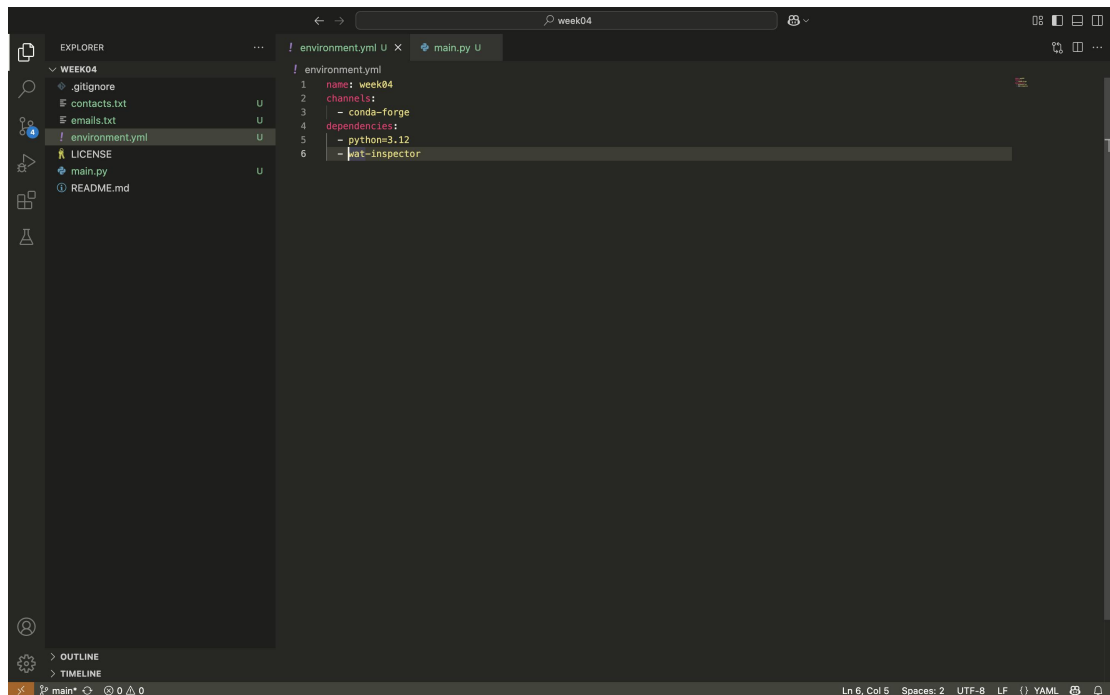
可以用 p 来翻译表达式

```

[(Pdb) p __name__
'__main__'
(Pdb) █

```

6. 在调试过程中，利用 `wat-inspector` (第三方软件包，需要安装) 检查 (inspect) 各种对象



在调试过程中，观察代码逐步运行的效果，学习理解以下 Python 基本概念 (建议观看下面的录播讲解)

Python 语法保留字 (reserved key words)

语句 (statement) 和表达式 (expression)

缩进 (indent)

局部变量 (local variable)、全局变量 (global variable)、LEGB 规则

函数 (function) 的定义 (define) 和调用 (call)

字面值 (literal) (字符串 (str)、整数 (int)、列表 (list)、字典 (dict)、元组 (tuple))

运算符 (operator)

形参 (parameter)、实参 (argument)、返回值 (return value)

对象 (object)、类型 (type)、属性 (attribute)、方法 (method)