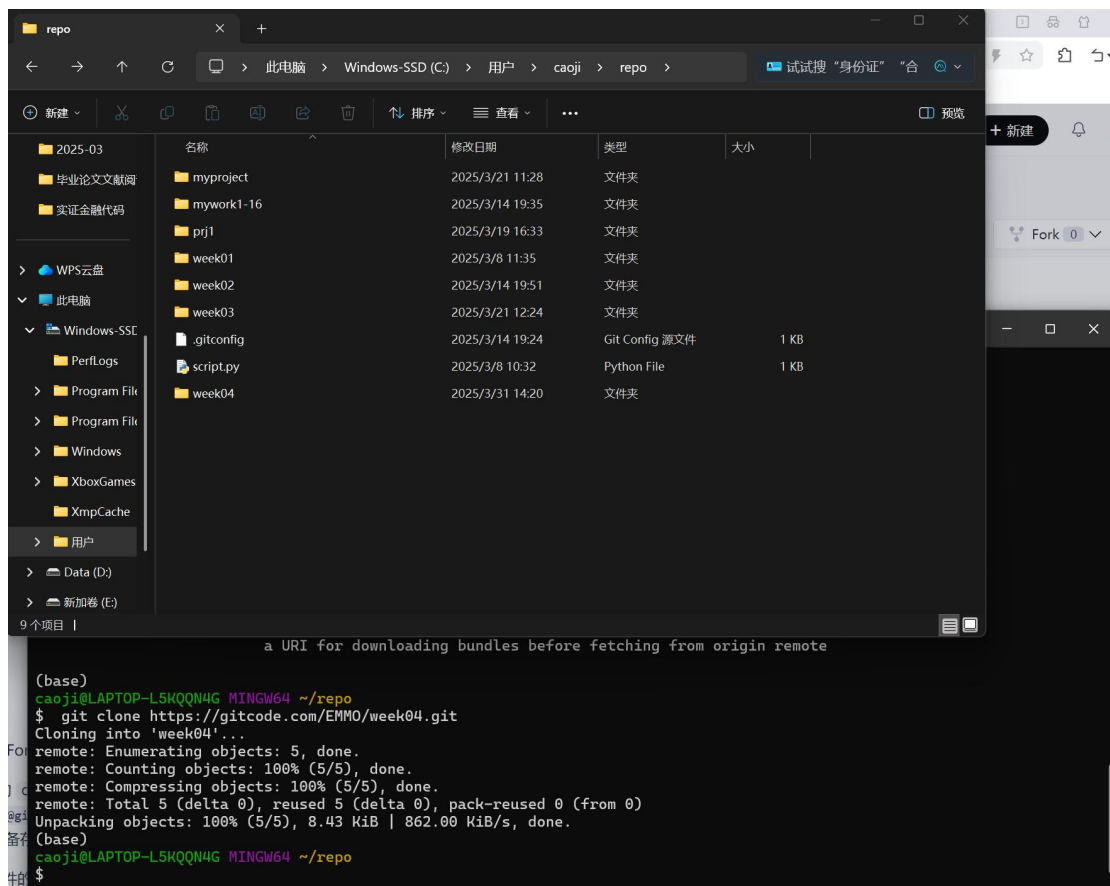


第四周学习笔记

```
MINGW64/c/Users/caoji/repo x + v
Use quit() or Ctrl-Z plus Return to exit
>>> print('week end')
week end
>>> quit()
(base)
caoji@LAPTOP-L5KQQN4G MINGW64 ~
$ cd repo
(base)
caoji@LAPTOP-L5KQQN4G MINGW64 ~/repo
$ ls -l
bash: ls-l: command not found
(base)
caoji@LAPTOP-L5KQQN4G MINGW64 ~/repo
$ ls -l
ls: cannot access '-': No such file or directory
ls: cannot access 'l': No such file or directory
(base)
caoji@LAPTOP-L5KQQN4G MINGW64 ~/repo
$ ls -l
total 17
drwxr-xr-x 1 caoji 197121 0 3月 21 11:28 myproject/
drwxr-xr-x 1 caoji 197121 0 3月 14 19:35 mywork1-16/
drwxr-xr-x 1 caoji 197121 0 3月 19 16:33 prj1/
-rw-r--r-- 1 caoji 197121 189 3月 8 10:32 script.py
drwxr-xr-x 1 caoji 197121 0 3月 8 11:35 week01/
drwxr-xr-x 1 caoji 197121 0 3月 14 19:51 week02/
drwxr-xr-x 1 caoji 197121 0 3月 21 12:24 week03/
(base)
caoji@LAPTOP-L5KQQN4G MINGW64 ~/repo
$
```

cd repo: 打开 repo 文件夹

ls -l: 查看文件夹里的内容



```
(base)
caoji@LAPTOP-L5KQQN4G MINGW64 ~/repo
$ git clone https://gitcode.com/EMMO/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 | 862.00 KiB/s, done.
(base)
caoji@LAPTOP-L5KQQN4G MINGW64 ~/repo
$
```

git clone https://gitcode.com/EMMO/week04.git (将 week04 复制到 repo 里)

```
MINGW64/c/Users/caoji/repco x + v
$ git clone https://gitcode.com/EMMO/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 | 862.00 KiB/s, done.
(base)
caoji@LAPTOP-L5KQQN4G MINGW64 ~/repo
$ cd week04
(base)
caoji@LAPTOP-L5KQQN4G MINGW64 ~/repo/week04 (main)
$ pwd
/c/Users/caoji/repco/week04
(base)
caoji@LAPTOP-L5KQQN4G MINGW64 ~/repo/week04 (main)
$ git remote show origin
* remote origin
Fetch URL: https://gitcode.com/EMMO/week04.git
Push URL: https://gitcode.com/EMMO/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)
caoji@LAPTOP-L5KQQN4G MINGW64 ~/repo/week04 (main)
$
```

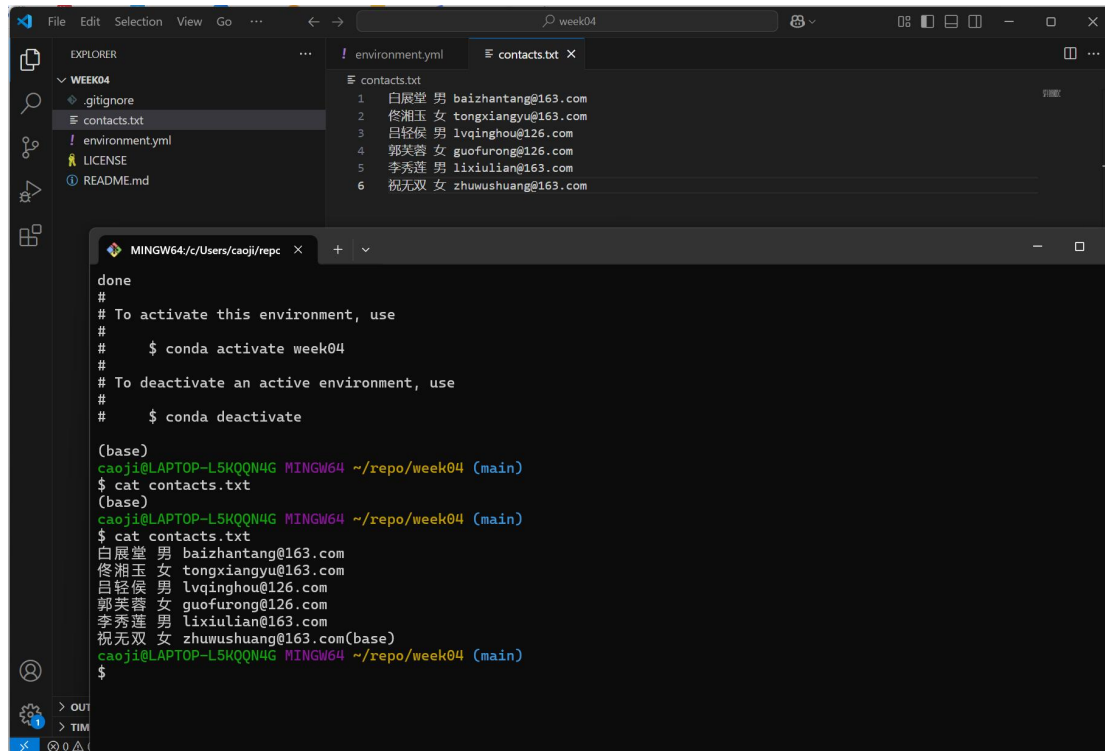
pwd:查看当前文件夹所在位置

git remote show origin: 查看当前文件夹的远程地址

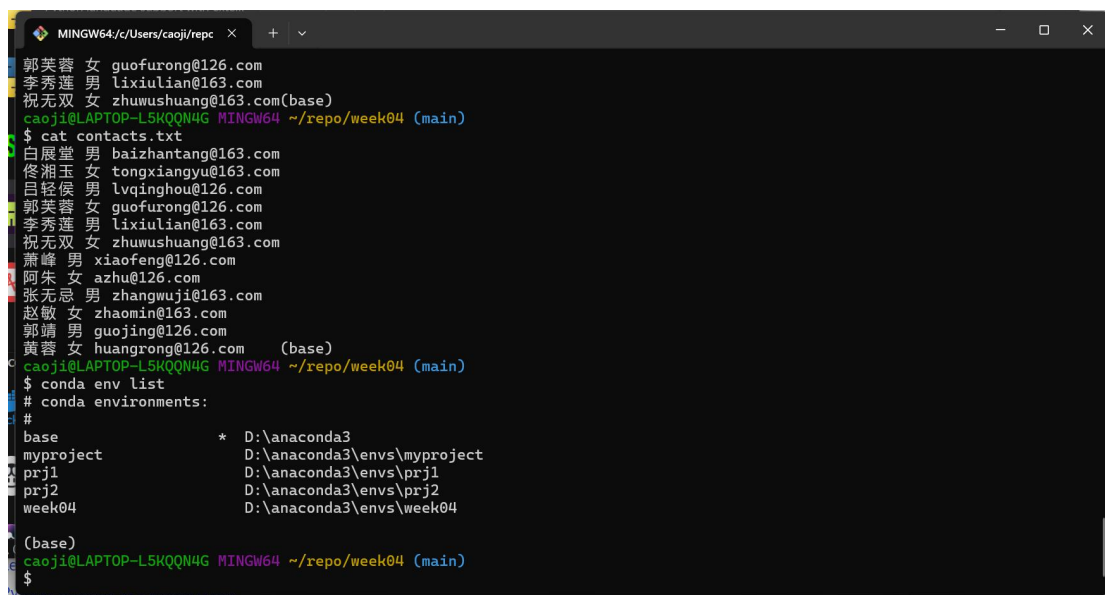
```
MINGW64/c/Users/caoji/repco x + v
main tracked
Local branch configured for 'git pull':
main merges with remote main
Local ref configured for 'git push':
main pushes to main (up to date)
(base)
caoji@LAPTOP-L5KQQN4G MINGW64 ~/repo/week04 (main)
$ code .
bash: code: command not found
(base)
caoji@LAPTOP-L5KQQN4G MINGW64 ~/repo/week04 (main)
$ ls -l ../myproject
total 196846
-rw-r--r-- 1 caoji 197121      87  3月 21 11:22 environment.yml
-rw-r--r-- 1 caoji 197121 201568176 3月 21 11:36 EPA_SmartLocationDatabase_V3_Jan_2021_Final.csv
-rw-r--r-- 1 caoji 197121      696  3月 21 11:52 main.py
(base)
caoji@LAPTOP-L5KQQN4G MINGW64 ~/repo/week04 (main)
$ cat ../myproject/environment.yml
name: myproject
channels:
- conda-forge
dependencies:
- python=3.12
- pandas(base)
caoji@LAPTOP-L5KQQN4G MINGW64 ~/repo/week04 (main)
$ cp ../myproject/environment.yml ./
(base)
caoji@LAPTOP-L5KQQN4G MINGW64 ~/repo/week04 (main)
$
```

cat: 查看 yml 中的内容

cp (复制) ../myproject/environment.yml (复制的源文件) ./ (到当前目录)



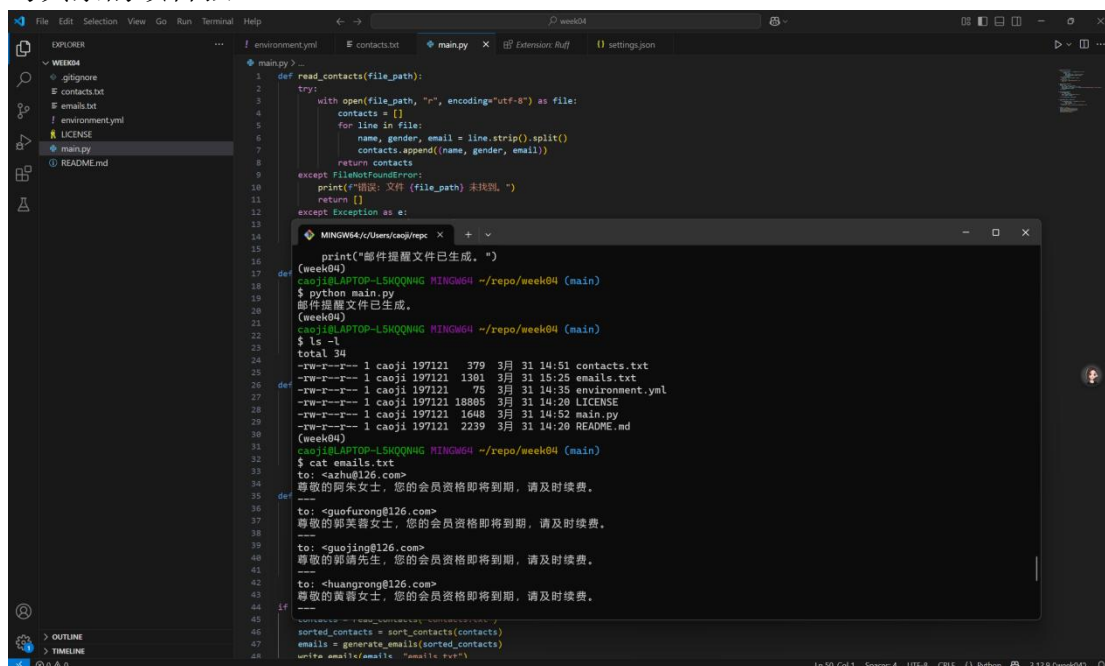
cat contacts.txt: 读取文件内容，查看文件能否顺利运行并输出
conda env create: 创建一个新的 Conda 环境



conda env list: 查看已经创建的环境

```
MINGW64/c/Users/caoji/rep/ x + v
(base)
caoji@LAPTOP-L5KQQN4G MINGW64 ~/repo/week04 (main)
$ conda activate week04
(week04)
caoji@LAPTOP-L5KQQN4G MINGW64 ~/repo/week04 (main)
$ ls -l
total 30
-rw-r--r-- 1 caoji 197121 379 3月 31 14:51 contacts.txt
-rw-r--r-- 1 caoji 197121 75 3月 31 14:35 environment.yml
-rw-r--r-- 1 caoji 197121 18805 3月 31 14:20 LICENSE
-rw-r--r-- 1 caoji 197121 1648 3月 31 14:52 main.py
-rw-r--r-- 1 caoji 197121 2239 3月 31 14:20 README.md
(week04)
caoji@LAPTOP-L5KQQN4G 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
萧峰 男 xiaofeng@126.com
阿朱 女 azhu@126.com
张无忌 男 zhangwuji@163.com
赵敏 女 zhaomin@163.com
郭靖 男 guojing@126.com
黄蓉 女 huangrong@126.com
(week04)
caoji@LAPTOP-L5KQQN4G MINGW64 ~/repo/week04 (main)
$
```

conda activate week04: 将当前终端的环境更改为 week04 中创建的 conda 环境, 与其余的项目独立



The screenshot shows the Visual Studio Code editor with the `main.py` file open. The file contains a function `read_contacts` that reads a `contacts.txt` file and returns a list of contacts. Below the editor, a terminal window is open, showing the output of the `python main.py` command. The terminal output indicates that the email reminder file has been generated successfully. The terminal also shows the `ls -l` command output, which lists the files in the current directory, including `contacts.txt`, `environment.yml`, `main.py`, and `README.md`.

```
def read_contacts(file_path):
    try:
        with open(file_path, "r", encoding="utf-8") as file:
            contacts = []
            for line in file:
                name, gender, email = line.strip().split()
                contacts.append((name, gender, email))
            return contacts
    except FileNotFoundError:
        print(f"错误: 文件 {file_path} 未找到.")
        return []
    except Exception as e:
        print(f"异常: {e}")

print("邮件提醒文件已生成.")

(week04)
caoji@LAPTOP-L5KQQN4G MINGW64 ~/repo/week04 (main)
$ python main.py
邮件提醒文件已生成.
(week04)
caoji@LAPTOP-L5KQQN4G MINGW64 ~/repo/week04 (main)
$ ls -l
total 34
-rw-r--r-- 1 caoji 197121 379 3月 31 14:51 contacts.txt
-rw-r--r-- 1 caoji 197121 1301 3月 31 15:25 emails.txt
-rw-r--r-- 1 caoji 197121 75 3月 31 14:35 environment.yml
-rw-r--r-- 1 caoji 197121 18805 3月 31 14:20 LICENSE
-rw-r--r-- 1 caoji 197121 1648 3月 31 14:52 main.py
-rw-r--r-- 1 caoji 197121 2239 3月 31 14:20 README.md
(week04)
caoji@LAPTOP-L5KQQN4G MINGW64 ~/repo/week04 (main)
$ cat emails.txt
to: <azhu@126.com>
尊敬的阿朱女士, 您的会员资格即将到期, 请及时续费.
to: <guofurong@126.com>
尊敬的郭芙蓉女士, 您的会员资格即将到期, 请及时续费.
to: <guojing@126.com>
尊敬的郭靖先生, 您的会员资格即将到期, 请及时续费.
to: <huangrong@126.com>
尊敬的黄蓉女士, 您的会员资格即将到期, 请及时续费.
if
sorted_contacts = sort_contacts(contacts)
emails = generate_emails(sorted_contacts)
write_emails(emails, "emails.txt")
```

python main.py: 使用 Python 运行 main.py 这个程序

```
MINGW64/c/Users/caoji/repco x + v
尊敬的赵敏女士，您的会员资格即将到期，请及时续费。
---
to: <zhuwushuang@163.com>
尊敬的祝无双女士，您的会员资格即将到期，请及时续费。
---
(caoji@LAPTOP-L5KQON4G MINGW64 ~/repo/week04 (main))
$ ls -l
total 34
-rw-r--r-- 1 caoji 197121 379 3月 31 14:51 contacts.txt
-rw-r--r-- 1 caoji 197121 1301 3月 31 15:25 emails.txt
-rw-r--r-- 1 caoji 197121 75 3月 31 14:35 environment.yml
-rw-r--r-- 1 caoji 197121 18805 3月 31 14:20 LICENSE
-rw-r--r-- 1 caoji 197121 1648 3月 31 14:52 main.py
-rw-r--r-- 1 caoji 197121 2239 3月 31 14:20 README.md
(caoji@LAPTOP-L5KQON4G MINGW64 ~/repo/week04 (main))
$ rm emails.txt
(caoji@LAPTOP-L5KQON4G MINGW64 ~/repo/week04 (main))
$ ls -l
total 30
-rw-r--r-- 1 caoji 197121 379 3月 31 14:51 contacts.txt
-rw-r--r-- 1 caoji 197121 75 3月 31 14:35 environment.yml
-rw-r--r-- 1 caoji 197121 18805 3月 31 14:20 LICENSE
-rw-r--r-- 1 caoji 197121 1648 3月 31 14:52 main.py
-rw-r--r-- 1 caoji 197121 2239 3月 31 14:20 README.md
(caoji@LAPTOP-L5KQON4G MINGW64 ~/repo/week04 (main))
$
```

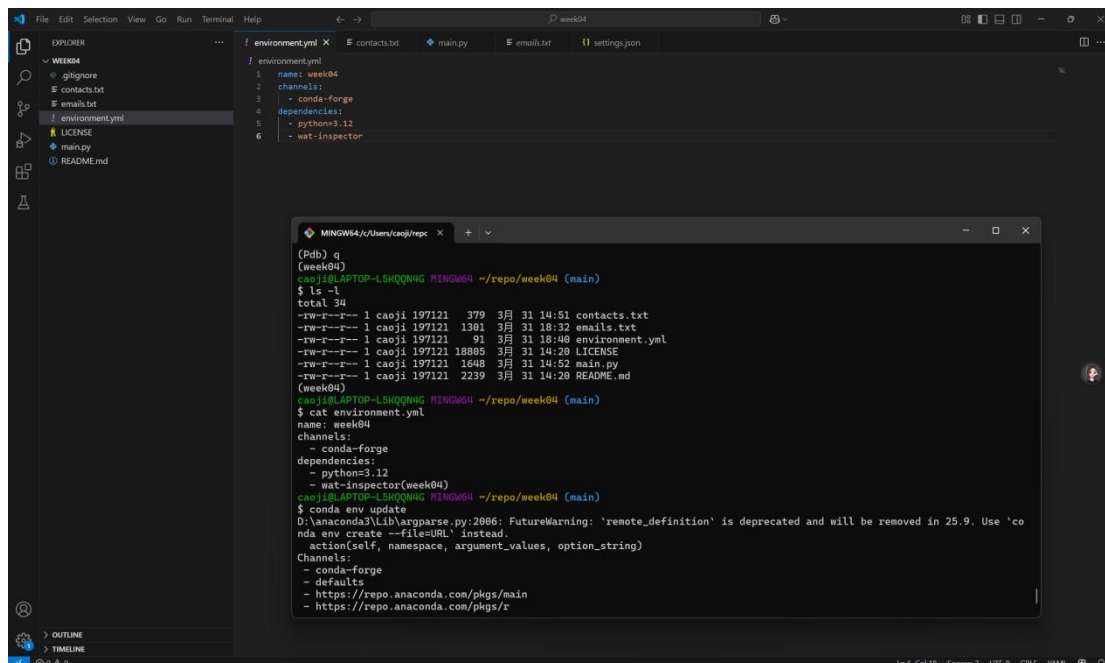
rm emails.txt: 删除 emails.txt 这一文件

```
MINGW64/c/Users/caoji/repco x + v
(caoji@LAPTOP-L5KQON4G MINGW64 ~/repo/week04 (main))
$ rm emails.txt
(caoji@LAPTOP-L5KQON4G MINGW64 ~/repo/week04 (main))
$ ls -l
total 30
-rw-r--r-- 1 caoji 197121 379 3月 31 14:51 contacts.txt
-rw-r--r-- 1 caoji 197121 75 3月 31 14:35 environment.yml
-rw-r--r-- 1 caoji 197121 18805 3月 31 14:20 LICENSE
-rw-r--r-- 1 caoji 197121 1648 3月 31 14:52 main.py
-rw-r--r-- 1 caoji 197121 2239 3月 31 14:20 README.md
(caoji@LAPTOP-L5KQON4G MINGW64 ~/repo/week04 (main))
$ python -m pdb main.py
> c:\users\caoji\repo\week04\main.py(1)<module>()
-> def read_contacts(file_path):
(Pdb) l
1 --> def read_contacts(file_path):
2     try:
3         with open(file_path, "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             return contacts
9     except FileNotFoundError:
10        print(f"错误: 文件 {file_path} 未找到。")
11        return []
(Pdb)
```

python -m pdb main.py: 先运行调试器，然后运行 main.py 这一程序，就可以做到逐行运行

- l (查看代码) -> 后的是即将运行的代码
- n (执行当前代码)
- ll (查看全部代码)
- p (打印表达式)
- s (步入调用) 类似于进入到了门里面运行
- pp (美观打印)
- c (全部执行直到结束)
- l . (箭头所指程序的上下五行)
- l 1,5 (查看第 1 行到第 5 行)

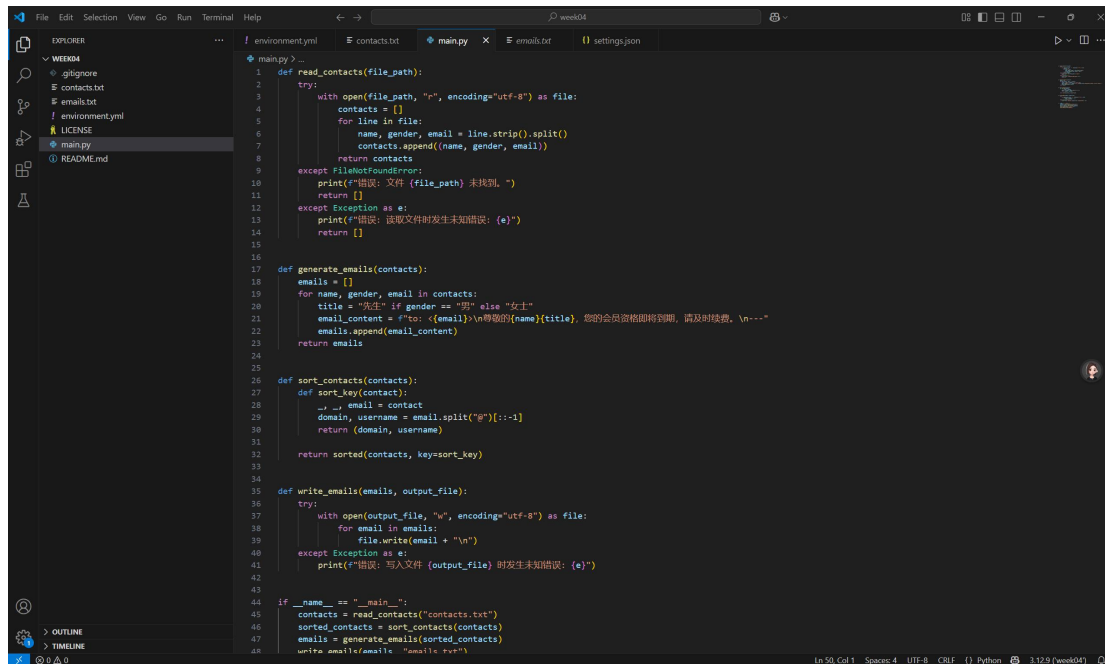
```
*** NameError: name 'contacts_file' is not defined
(Pdb) p read_contacts
<function read_contacts at 0x0000235408BF9C0>
(Pdb) p contacts
*** NameError: name 'contacts' is not defined
(Pdb) n
> c:\users\caoji\repo\week04\main.py(46)<module>()
-> sorted_contacts = sort_contacts(contacts)
(Pdb) n
> c:\users\caoji\repo\week04\main.py(47)<module>()
-> emails = generate_emails(sorted_contacts)
(Pdb) p contacts
[('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', 'lixliulan@163.com'), ('祝无双', '女', 'zhuwushuang@163.com'), ('萧峰', '男', 'xiaofeng@126.com'), ('阿朱', '女', 'azhu@126.com'), ('张无忌', '男', 'zhangwuji@163.com'), ('赵敏', '女', 'zhaomin@163.com'), ('郭靖', '男', 'guojing@126.com'), ('黄蓉', '女', 'huangrong@126.com')]
(Pdb) pp contacts
[('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', 'lixliulan@163.com'), ('祝无双', '女', 'zhuwushuang@163.com'), ('萧峰', '男', 'xiaofeng@126.com'), ('阿朱', '女', 'azhu@126.com'), ('张无忌', '男', 'zhangwuji@163.com'), ('赵敏', '女', 'zhaomin@163.com'), ('郭靖', '男', 'guojing@126.com'), ('黄蓉', '女', 'huangrong@126.com')]
(Pdb)
```



conda env update: 对环境进行更新

Python 语法保留字 (reserved key words):

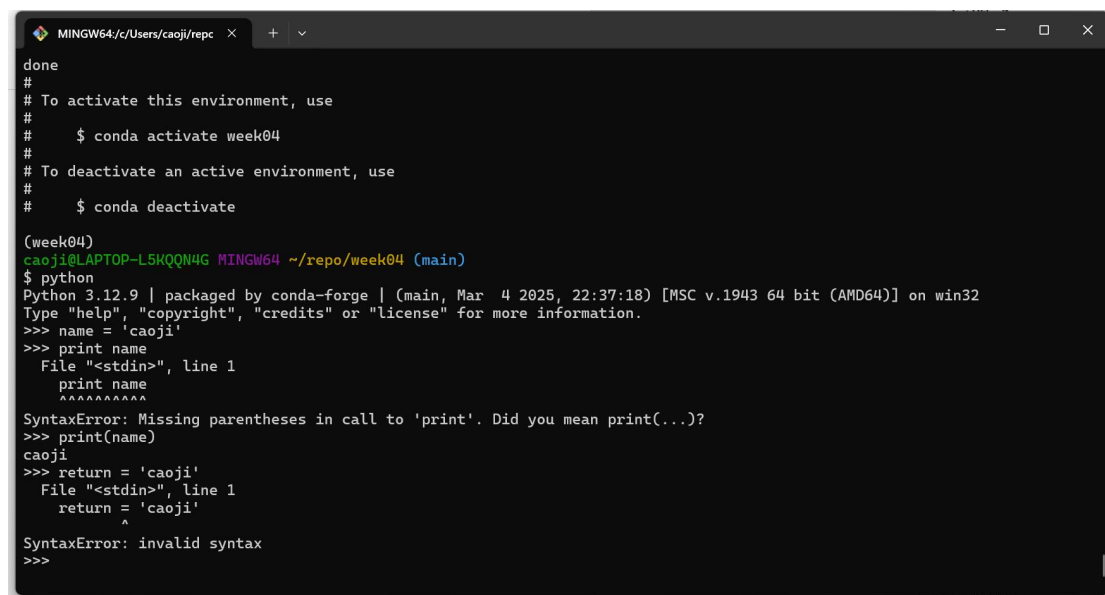
在 Python 的语法中有特殊的含义的，截图中紫色的字体，例如 try、as、return 等



```
1 def read_contacts(file_path):
2     try:
3         with open(file_path, "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             return contacts
9     except FileNotFoundError:
10        print(f"错误: 文件 {file_path} 未找到。")
11        return []
12    except Exception as e:
13        print(f"错误: 读取文件时发生未知错误: {e}")
14        return []
15
16
17 def generate_emails(contacts):
18     emails = []
19     for name, gender, email in contacts:
20         title = "先生" if gender == "男" else "女士"
21         email_content = f"to: <{email}>\n尊敬的{name}{title}, 您的会员资格即将到期, 请及时续费。 \n---"
22         emails.append(email_content)
23     return emails
24
25
26 def sort_contacts(contacts):
27     def sort_key(contact):
28         _, email = contact
29         domain, username = email.split("@")[:-1]
30         return (domain, username)
31
32     return sorted(contacts, key=sort_key)
33
34
35 def write_emails(emails, output_file):
36     try:
37         with open(output_file, "w", encoding="utf-8") as file:
38             for email in emails:
39                 file.write(email + "\n")
40     except Exception as e:
41         print(f"错误: 写入文件 {output_file} 时发生未知错误: {e}")
42
43
44 if __name__ == "__main__":
45     contacts = read_contacts("contacts.txt")
46     sorted_contacts = sort_contacts(contacts)
47     emails = generate_emails(sorted_contacts)
48     write_emails(emails, "emails.txt")
```

保留字不太多，但保留字不可以作为变量名

故而 `name = 'caoji'` 可以运行，但 `return = 'caoji'` 是错误语句



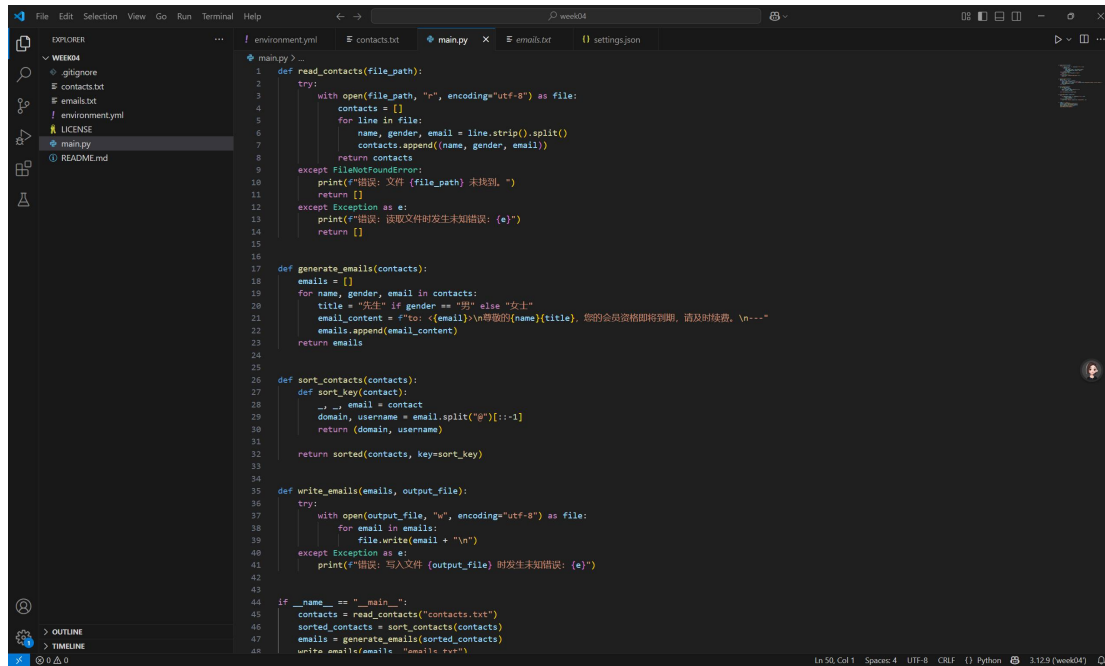
```
done
#
# To activate this environment, use
#
#     $ conda activate week04
#
# To deactivate an active environment, use
#
#     $ conda deactivate
#
(week04)
caoji@LAPTOP-L5KQQN4G MINGW64 ~/repo/week04 (main)
$ python
Python 3.12.9 | packaged by conda-forge | (main, Mar  4 2025, 22:37:18) [MSC v.1943 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> name = 'caoji'
>>> print name
File "<stdin>", line 1
    print name
    ^^^^^^^^^
SyntaxError: Missing parentheses in call to 'print'. Did you mean print(...)?
>>> print(name)
caoji
>>> return = 'caoji'
File "<stdin>", line 1
    return = 'caoji'
    ^
SyntaxError: invalid syntax
>>>
```

语句 (statement) 和表达式 (expression):

1-14 行是一个语句, 里面有很多子语句; 表达式是构成语句的元素, 类似于语言里的词汇, 语句包含表达式

缩进 (indent):

缩进的语句是原始语句的子语句



```
1 def read_contacts(file_path):
2     try:
3         with open(file_path, "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             return contacts
9     except FileNotFoundError:
10        print(f"错误: 文件 {file_path} 未找到。")
11        return []
12    except Exception as e:
13        print(f"错误: 读取文件时发生未知错误: {e}")
14        return []
15
16
17 def generate_emails(contacts):
18     emails = []
19     for name, gender, email in contacts:
20         title = "小姐" if gender == "男" else "女士"
21         email_content = f"to: <{email}>\n尊敬的{name}{title}, 您的会员资格即将到期, 请及时续费。 \n---"
22         emails.append(email_content)
23     return emails
24
25
26 def sort_contacts(contacts):
27     def sort_key(contact):
28         _, _, email = contact
29         domain, username = email.split("@")[:1]
30         return (domain, username)
31
32     return sorted(contacts, key=sort_key)
33
34
35 def write_emails(emails, output_file):
36     try:
37         with open(output_file, "w", encoding="utf-8") as file:
38             for email in emails:
39                 file.write(email + "\n")
40     except Exception as e:
41         print(f"错误: 写入文件 {output_file} 时发生未知错误: {e}")
42
43
44 if __name__ == "__main__":
45     contacts = read_contacts("contacts.txt")
46     sorted_contacts = sort_contacts(contacts)
47     emails = generate_emails(sorted_contacts)
48     write_emails(emails, "emails.txt")
```


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

```
MINGW64/c/Users/caoji/rep... x + v
.gray to disable colorful output in the console
.color to enforce colorful outputs in the console
Call wat.locals or wat() to inspect local variables.
Call wat.globals to inspect global variables.

(Pdb) wat()
Local variables:
__builtins__: dict = {...
__file__: pdb._ScriptTarget = 'C:\Users\caoji\repo\week04\main.py'
__name__: str = '__main__'
__pdb_convenience_variables__: dict = {...
__spec__: NoneType = None
wat: wat.inspection.inspection.Wat = <WAT Inspector object>

(Pdb) p wat
Try wat / object or wat.modifiers / object to inspect an object. Modifiers are:
.short or .s to hide attributes (variables and methods)
.dunder to print dunder attributes
.code to print source code of a function, method or class
.long to print non-abbreviated values and documentation
.nodocs to hide documentation for functions and classes
.caller to show how and where the inspection was called
.all to include all information
.ret to return the inspected object
.str to return the output string instead of printing
.gray to disable colorful output in the console
.color to enforce colorful outputs in the console
Call wat.locals or wat() to inspect local variables.
Call wat.globals to inspect global variables.

(Pdb)
```