

git clone https://gitcode.com/gtxy/week04.git:将第 04 周打卡仓库 Clone 到本地计算机。

git remote show origin: 查看远程仓库

```
(base)
ZhaoXu@LAPTOP-QD2IM9DB MINGW64 ~/repo/week04 (main)
$ git remote show origin
* remote origin
  Fetch URL: https://gitcode.com/gtxy/week04.git
  Push URL: https://gitcode.com/gtxy/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)
```

cp ../myproject/environment.yml ./ : 复制文件到当前路径

```
(base)
ZhaoXu@LAPTOP-QD2IM9DB MINGW64 ~/repo/week04 (main)
$ cat ../myproject/environment.yml
name: my-project
channels:
  - conda-forge
dependencies:
  - python=3.12
  - pandas(base)
ZhaoXu@LAPTOP-QD2IM9DB MINGW64 ~/repo/week04 (main)
$ cp ../myproject/environment.yml ./
(base)
```

cat contacts.txt: 读取 contacts.txt 内容

```
(base)
ZhaoXu@LAPTOP-QD2IM9DB 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)
```

cat contacts.txt environment.yml: 拼接 contacts.txt、environment.yml 内容

```

ZhaoXu@LAPTOP-QD2IM9DB MINGW64 ~/repo/week04 (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.com
name: week04
channels:
  - conda-forge
dependencies:
  - python=3.12(base)

```

检查 AI 生成代码:

```

(week04)
ZhaoXu@LAPTOP-QD2IM9DB MINGW64 ~/repo/week04 (main)
$ cat emails.txt
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: <lvqinghou@126.com>
尊敬的吕轻侯先生，您的会员资格即将到期，请及时续费。
---
to: <baizhantang@163.com>
尊敬的白展堂先生，您的会员资格即将到期，请及时续费。
---
to: <lixiulian@163.com>
尊敬的李秀莲先生，您的会员资格即将到期，请及时续费。
---
to: <tongxiangyu@163.com>
尊敬的佟湘玉女士，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。
---

```

python -m pdb main.py: 先由调试器加载代码。

```

(week04)
ZhaoXu@LAPTOP-QD2IM9DB MINGW64 ~/repo/week04 (main)
$ python -m pdb main.py
> c:\users\zhaoxu\repo\week04\main.py(1)<module>()
-> with open("contacts.txt", "r", encoding="utf-8") as f:
(Pdb)

```

l: 显示即将运行的代码。

```

with open('contacts.txt', 'r', encoding='utf-8') as f:
(Pdb) l
1  -> with open("contacts.txt", "r", encoding="utf-8") as f:
2      contacts = f.readlines()
3
4
5      parsed_contacts = []
6      for line in contacts:
7          name, gender, email = line.strip().split()
8          username, domain = email.split("@")
9          parsed_contacts.append((domain, username, name, gender, em
ail))

```

n : 执行当前行

```

11
(Pdb) n
> c:\users\zhaoxu\repo\week04\main.py(2)<module>()
-> contacts = f.readlines()
(Pdb)

```

p : 打印表达式

s : 步入调用

```

(Pdb) s
--Call--
> <frozen codecs>(319).decode()
(Pdb)

```

l : 显示上下 5 行

```

(Pdb) l .
314         def _buffer_decode(self, input, errors, final):
315             # Overwrite this method in subclasses: It must decode input
316             # and return an (output, length consumed) tuple
317             raise NotImplementedError
318
319 ->         def decode(self, input, final=False):
320             # decode input (taking the buffer into account)
321             data = self.buffer + input
322             (result, consumed) = self._buffer_decode(data, self.errors,
final)
323             # keep undecoded input until the next call
324             self.buffer = data[consumed:]
(Pdb)

```

Python 语法保留字：红色。有特定用途单词，不能作为变量名

```

1  with open("contacts.txt", "r", encoding="utf-8") as f:
2      contacts = f.readlines()
3
4
5      parsed_contacts = []
6      for line in contacts:
7          name, gender, email = line.strip().split()
8          username, domain = email.split("@")
9          parsed_contacts.append((domain, username, name, gender, email))
10
11
12      parsed_contacts.sort()
13
14      with open("emails.txt", "w", encoding="utf-8") as f:
15          for domain, username, name, gender, email in parsed_contacts:
16              title = "女士" if gender == "女" else "先生"

```

语句：可以折叠，逻辑完整的一句话。

缩进 (Indent): Python 使用缩进（空格或 Tab）表示代码块

局部变量 (Local Variable): 函数内部定义，仅在函数内有效。

全局变量 (Global Variable): 函数外部定义，全局有效。

LEGB 规则（变量查找顺序）: Local - 局部作用域、Enclosing - 闭包作用域、Global - 全局作用域、Built-in - 内置作用域。

函数 (Function): 定义: 使用 `def` 关键字。调用: 函数名加括号。

字符串 (str): "文本" 或 '文本'。

整数 (int): 123, -456

列表 (list): [1, 2, 3]

字典 (dict): {"键": "值"}

元组 (tuple): (1, 2, 3)

运算符 (Operator): 算术: +, -, \*, /, //, %, \*\*, 比较: ==, !=, >, <, >=, <=、逻辑: and, or, not、成员: in, not in

形参 (Parameter): 函数定义时声明的变量

实参 (Argument): 调用函数时传入的实际值

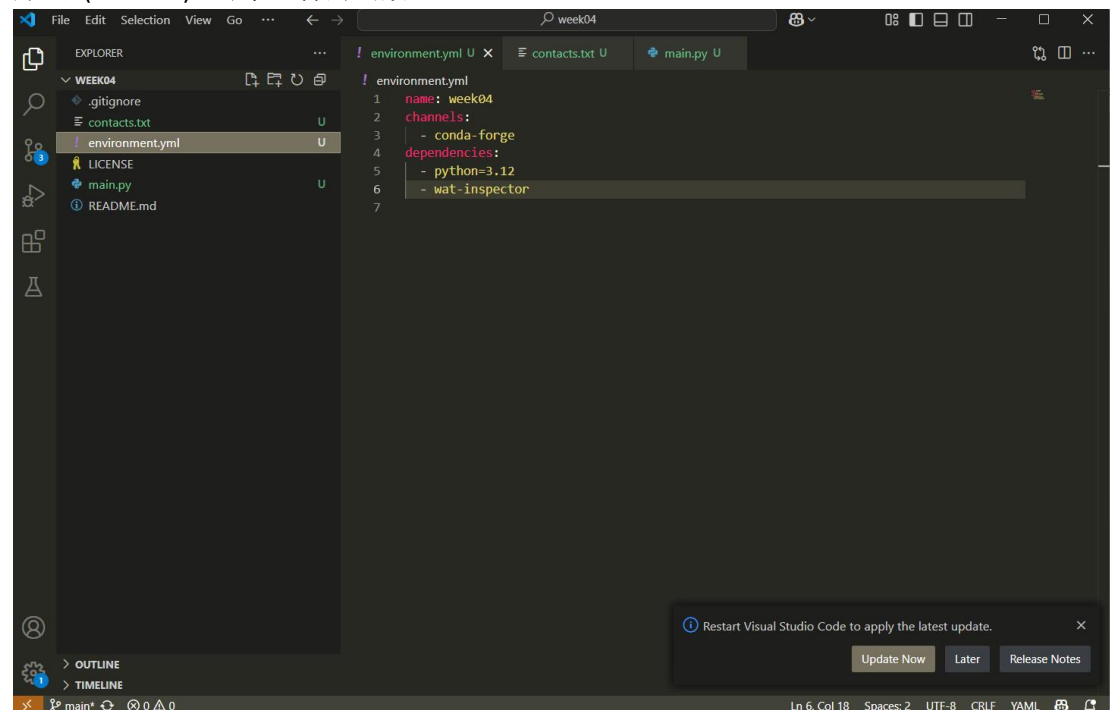
返回值: 使用 `return` 语句返回结果

对象 (Object): Python 中一切皆对象

类型 (Type): 对象的类别（如 `int`, `str`）

属性 (Attribute): 对象包含的变量

方法 (Method): 对象包含的函数



```
13
14 ∨ with open("emails.txt", "w", encoding="utf-8") as f:
15 ∨     for domain, username, name, gender, email in parsed_contacts:
16         title = "女士" if gender == "女" else "先生"
17         message = f"to: <{email}>\n尊敬的{name}{title}, 您的会员资格即将到期, 请及时续费。"
18         f.write(message)
19
20 print("emails.txt 已生成! ")
21
```

contacts.txt

```
1 白展堂 男 baizhantang@163.com
2 佟湘玉 女 tongxiangyu@163.com
3 吕轻侯 男 lvqinghou@126.com
4 郭芙蓉 女 guofurong@126.com
5 李秀莲 男 lixiulian@163.com
6 祝无双 女 zhuwushuang@163.com
```