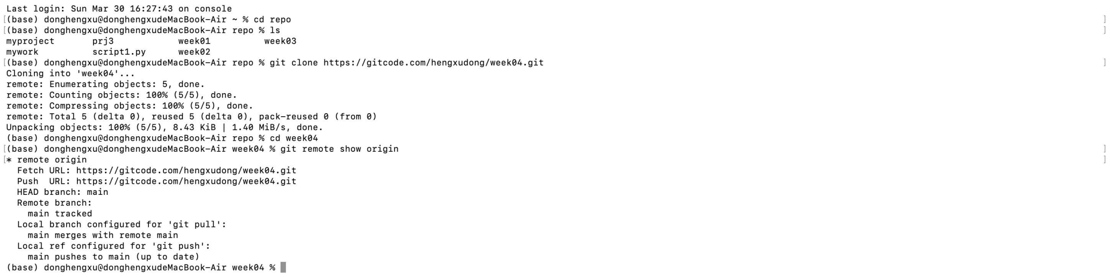
第四周学习报告

计算机组成原理，操作系统的概念：

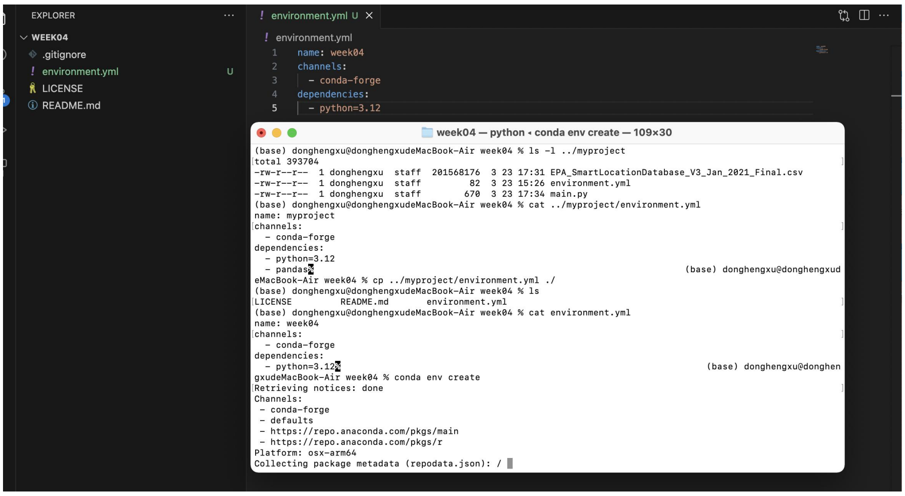
数据在不通电的情况下可以长期持久地 (persistently) 存储在 磁盘 (如固态硬盘 SSD、机械硬盘 HDD) 或磁带 (常用于数据备份、长期归档) 里。但在需要呈现 (print、render、show、display、play)、计算加工 (compute、transform、analyze、machine learning、deep learning) 或编解码 (encode、decode) 时，就需要通电的 CPU 和 内存 (硬件)，在操作系统 (软件) 里以 进程 (process) 为单元 (相互隔离) 进行处理。例如，Microsoft Word 启动后就是一个进程，我们在 Word 进程里打开某个 .docx 文档，将其从磁盘加载 (读取) 到内存，然后在图形界面 (GUI) 里查看和编辑 (计算) 内存中的文档，最后将内存数据保存 (写入) 到磁盘。同理，Python 解释器 (interpreter) 启动后也是一个进程，她按照流程 (flow) 执行我们准备好的 Python 代码，根据我们代码的要求，转告 (即 调用，call) 操作系统或其他软件 (即 依赖项，dependency)，委托她们替我们执行各种 “读取——计算——写入” 等工作。我们并不需要完全理解依赖项内部的工作细节 (黑箱)，只需要清楚每个调用的主体 (即 对象，object) 是什么 类型 (type)，每个调用的输入 (即 参数，parameter/argument)、输出 (即 返回值，return value) 是什么类型，以及调用会对内存数据、磁盘文件做什么修改，就足以支持我们自动批量地完成工作了。

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



```
Last login: Sun Mar 30 16:27:43 on console
(base) donghengxu@donghengxudeMacBook-Air ~ % cd repo
(base) donghengxu@donghengxudeMacBook-Air repo % ls
myproject       prj3            week01          week03
mywork          script1.py      week02
(base) donghengxu@donghengxudeMacBook-Air repo % git clone https://gitcode.com/hengxudong/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 | 1.40 MiB/s, done.
(base) donghengxu@donghengxudeMacBook-Air repo % cd week04
(base) donghengxu@donghengxudeMacBook-Air week04 % git remote show origin
* remote origin
  Fetch URL: https://gitcode.com/hengxudong/week04.git
  Push  URL: https://gitcode.com/hengxudong/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) donghengxu@donghengxudeMacBook-Air week04 % 
```

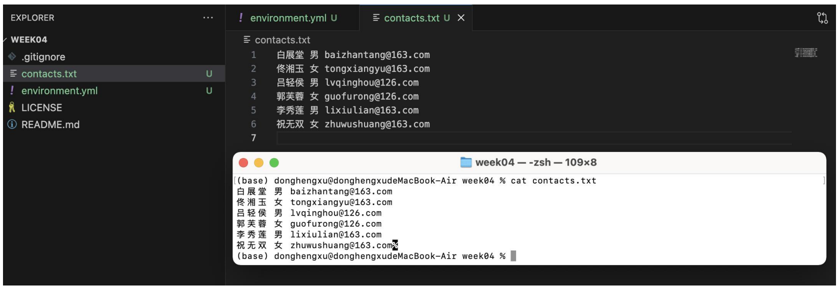
2、用 VS Code 打开项目目录，新建一个 environment.yml 文件，指定安装 Python 3.12，然后运行 conda env create 命令创建 Conda 环境



```
name: week04
channels:
  - conda-forge
dependencies:
  - python=3.12
```

```
(base) donghengxu@donghengxudeMacBook-Air week04 % ls -l ../myproject
total 393704
-rw-r--r--  1 donghengxu  staff  201568176  3 23 17:31 EPA_SmartLocationDatabase_V3_Jan_2021_Final.csv
-rw-r--r--  1 donghengxu  staff         82  3 23 15:26 environment.yml
-rw-r--r--  1 donghengxu  staff        670  3 23 17:34 main.py
(base) donghengxu@donghengxudeMacBook-Air week04 % cat ../myproject/environment.yml
name: myproject
channels:
  - conda-forge
dependencies:
  - python=3.12
  - pandas%                                                          (base) donghengxu@donghengxud
eMacBook-Air week04 % cp ../myproject/environment.yml ./
(base) donghengxu@donghengxudeMacBook-Air week04 % ls
LICENSE         README.md       environment.yml
(base) donghengxu@donghengxudeMacBook-Air week04 % cat environment.yml
name: week04
channels:
  - conda-forge
dependencies:
  - python=3.12%                                                         (base) donghengxu@donghen
gxudeMacBook-Air week04 % conda env create
Retrieving notices: done
Channels:
 - conda-forge
 - defaults
 - https://repo.anaconda.com/pkgs/main
 - https://repo.anaconda.com/pkgs/r
Platform: osx-arm64
Collecting package metadata (repodata.json): / 
```

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



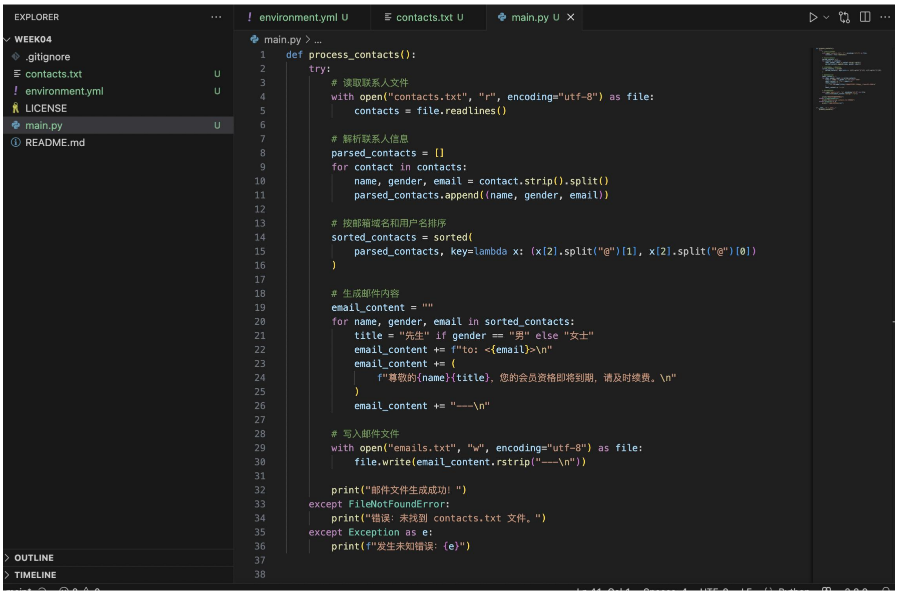
```
白展堂 男 baizhantang@163.com
佟湘玉 女 tongxiangyu@163.com
吕轻侯 男 lvqinghou@126.com
郭芙蓉 女 guofurong@126.com
李秀莲 男 lixiulian@163.com
祝无双 女 zhuwushuang@163.com
```

```
(base) donghengxu@donghengxudeMacBook-Air week04 % 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) donghengxu@donghengxudeMacBook-Air week04 %
```

3、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

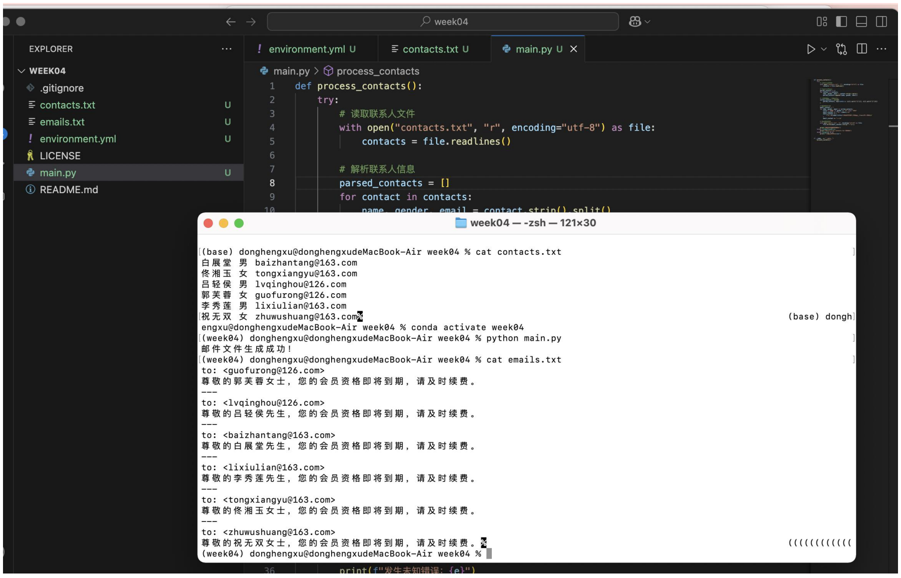
已在上周任务中完成 python ruff 扩展安装，因此没有截屏记录

（1）将大模型提供的代码复制粘贴进 main.py 文件，记得保存



```python
def process_contacts():
    try:
        # 读取联系人文件
        with open("contacts.txt", "r", encoding="utf-8") as file:
            contacts = file.readlines()

        # 解析联系人信息
        parsed_contacts = []
        for contact in contacts:
            name, gender, email = contact.strip().split()
            parsed_contacts.append((name, gender, email))

        # 按邮箱域名和用户名排序
        sorted_contacts = sorted(
            parsed_contacts, key=lambda x: (x[2].split("@")[1], x[2].split("@")[0])
        )

        # 生成邮件内容
        email_content = ""
        for name, gender, email in sorted_contacts:
            title = "先生" if gender == "男" else "女士"
            email_content += f"to: <{email}>\n"
            email_content += (
                f"尊敬的{name}{title}，您的会员资格即将到期，请及时续费。\n"
            )
            email_content += "---\n"

        # 写入邮件文件
        with open("emails.txt", "w", encoding="utf-8") as file:
            file.write(email_content.rstrip("---\n"))

        print("邮件文件生成成功！")
    except FileNotFoundError:
        print("错误：未找到 contacts.txt 文件。")
    except Exception as e:
        print(f"发生未知错误：{e}")


```

（2）运行代码

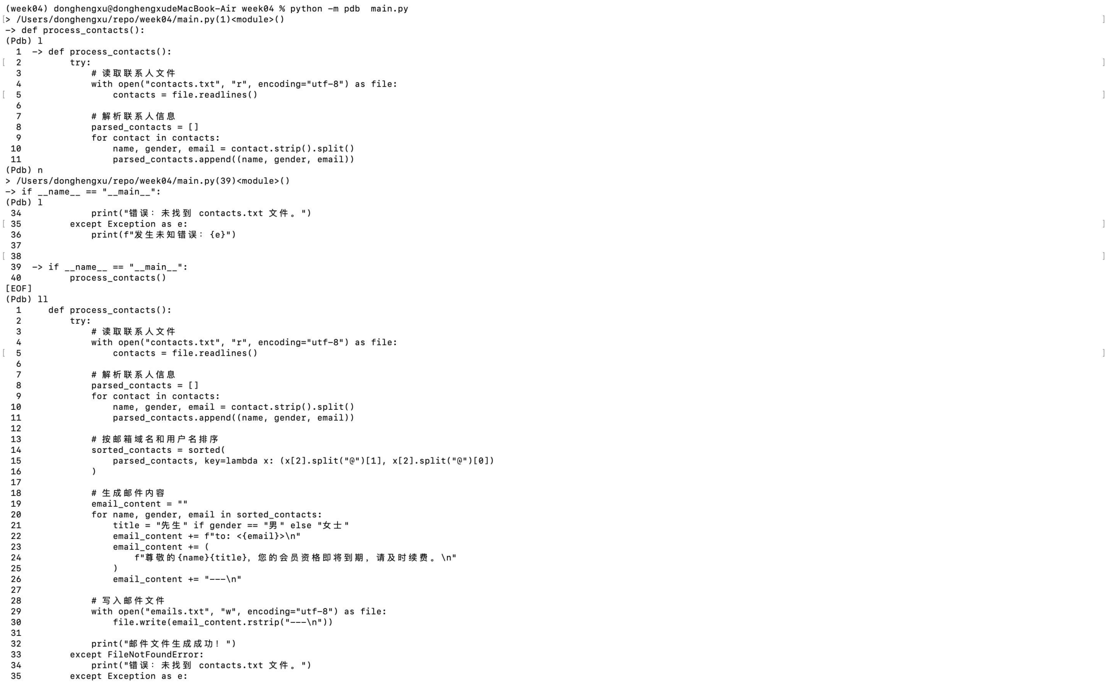


```python
def process_contacts():
    try:
        # 读取联系人文件
        with open("contacts.txt", "r", encoding="utf-8") as file:
            contacts = file.readlines()

        # 解析联系人信息
        parsed_contacts = []
        for contact in contacts:
```

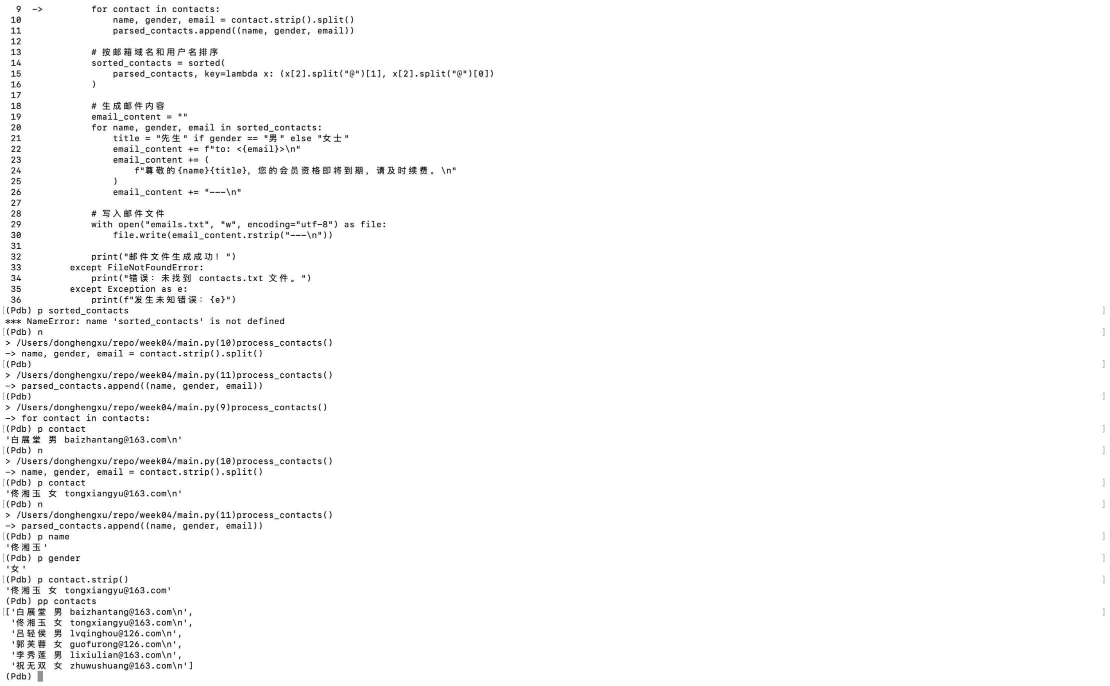
```
(base) donghengxu@donghengxudeMacBook-Air week04 % cat contacts.txt
白展堂 男 baizhantang@163.com
佟湘玉 女 tongxiangyu@163.com
吕轻侯 男 lvqinghou@126.com
郭芙蓉 女 guofurong@126.com
李秀莲 男 lixiulian@163.com
祝无双 女 zhuwushuang@163.com%                                                                    (base) dongh
engxu@donghengxudeMacBook-Air week04 % conda activate week04
(week04) donghengxu@donghengxudeMacBook-Air week04 % python main.py
邮件文件生成成功！
(week04) donghengxu@donghengxudeMacBook-Air week04 % 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。%                                     ((((((((((((
(week04) donghengxu@donghengxudeMacBook-Air week04 %
```

运行 python -m pdb main.py 命令 (作用是以调试模式 (debug mode) 启动 Python 解释器，准备执行 main.py 里的代码)

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

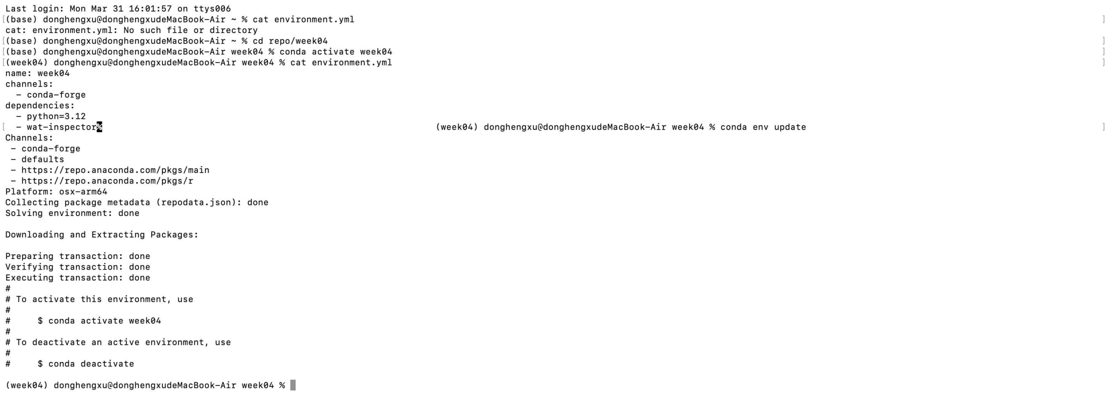


```
(week04) donghengxu@donghengxudeMacBook-Air week04 % python -m pdb  main.py
> /Users/donghengxu/repo/week04/main.py(1)<module>()
-> def process_contacts():
(Pdb) l
  1  -> def process_contacts():
  2         try:
  3             # 读取联系人文件
  4             with open("contacts.txt", "r", encoding="utf-8") as file:
  5                 contacts = file.readlines()
  6
  7             # 解析联系人信息
  8             parsed_contacts = []
  9             for contact in contacts:
 10                 name, gender, email = contact.strip().split()
 11                 parsed_contacts.append((name, gender, email))
(Pdb) n
> /Users/donghengxu/repo/week04/main.py(39)<module>()
-> if __name__ == "__main__":
(Pdb) l
 34             print("错误：未找到 contacts.txt 文件。")
 35         except Exception as e:
 36             print(f"发生未知错误：{e}")
 37
 38
 39  -> if __name__ == "__main__":
 40         process_contacts()
[EOF]
(Pdb) ll
  1     def process_contacts():
  2         try:
  3             # 读取联系人文件
  4             with open("contacts.txt", "r", encoding="utf-8") as file:
  5                 contacts = file.readlines()
  6
  7             # 解析联系人信息
  8             parsed_contacts = []
  9             for contact in contacts:
 10                 name, gender, email = contact.strip().split()
 11                 parsed_contacts.append((name, gender, email))
 12
 13             # 按邮箱域名和用户名排序
 14             sorted_contacts = sorted(
 15                 parsed_contacts, key=lambda x: (x[2].split("@")[1], x[2].split("@")[0])
 16             )
 17
 18             # 生成邮件内容
 19             email_content = ""
 20             for name, gender, email in sorted_contacts:
 21                 title = "先生" if gender == "男" else "女士"
 22                 email_content += f"to: <{email}>\n"
 23                 email_content += (
 24                     f"尊敬的{name}{title}，您的会员资格即将到期，请及时续费。\n"
 25                 )
 26                 email_content += "---\n"
 27
 28             # 写入邮件文件
 29             with open("emails.txt", "w", encoding="utf-8") as file:
 30                 file.write(email_content.rstrip("---\n"))
 31
 32             print("邮件文件生成成功！")
 33         except FileNotFoundError:
 34             print("错误：未找到 contacts.txt 文件。")
 35         except Exception as e:
```



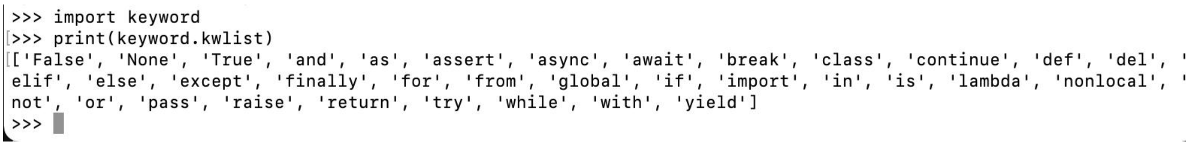
```
  9  ->         for contact in contacts:
 10                 name, gender, email = contact.strip().split()
 11                 parsed_contacts.append((name, gender, email))
 12
 13             # 按邮箱域名和用户名排序
 14             sorted_contacts = sorted(
 15                 parsed_contacts, key=lambda x: (x[2].split("@")[1], x[2].split("@")[0])
 16             )
 17
 18             # 生成邮件内容
 19             email_content = ""
 20             for name, gender, email in sorted_contacts:
 21                 title = "先生" if gender == "男" else "女士"
 22                 email_content += f"to: <{email}>\n"
 23                 email_content += (
 24                     f"尊敬的{name}{title}，您的会员资格即将到期，请及时续费。\n"
 25                 )
 26                 email_content += "---\n"
 27
 28             # 写入邮件文件
 29             with open("emails.txt", "w", encoding="utf-8") as file:
 30                 file.write(email_content.rstrip("---\n"))
 31
 32             print("邮件文件生成成功！")
 33         except FileNotFoundError:
 34             print("错误：未找到 contacts.txt 文件。")
 35         except Exception as e:
 36             print(f"发生未知错误：{e}")
(Pdb) p sorted_contacts
*** NameError: name 'sorted_contacts' is not defined
(Pdb) n
> /Users/donghengxu/repo/week04/main.py(10)process_contacts()
-> name, gender, email = contact.strip().split()
(Pdb)
> /Users/donghengxu/repo/week04/main.py(11)process_contacts()
-> parsed_contacts.append((name, gender, email))
(Pdb)
> /Users/donghengxu/repo/week04/main.py(9)process_contacts()
-> for contact in contacts:
(Pdb) p contact
'白展堂 男 baizhantang@163.com\n'
(Pdb) n
> /Users/donghengxu/repo/week04/main.py(10)process_contacts()
-> name, gender, email = contact.strip().split()
(Pdb) p contact
'佟湘玉 女 tongxiangyu@163.com\n'
(Pdb) n
> /Users/donghengxu/repo/week04/main.py(11)process_contacts()
-> parsed_contacts.append((name, gender, email))
(Pdb) p name
'佟湘玉'
(Pdb) p gender
'女'
(Pdb) p contact.strip()
'佟湘玉 女 tongxiangyu@163.com'
(Pdb) pp contacts
['白展堂 男 baizhantang@163.com\n',
 '佟湘玉 女 tongxiangyu@163.com\n',
 '吕轻侯 男 lvqinghou@126.com\n',
 '郭芙蓉 女 guofurong@126.com\n',
 '李秀莲 男 lixiulian@163.com\n',
 '祝无双 女 zhuwushuang@163.com\n']
(Pdb) █
```

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



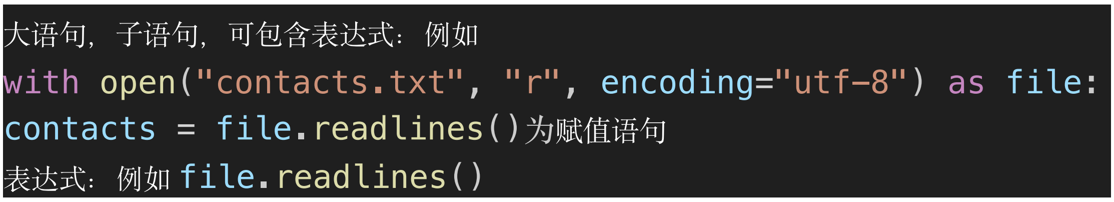
```
Last login: Mon Mar 31 16:01:57 on ttys006
(base) donghengxu@donghengxudeMacBook-Air ~ % cat environment.yml
cat: environment.yml: No such file or directory
(base) donghengxu@donghengxudeMacBook-Air ~ % cd repo/week04
(base) donghengxu@donghengxudeMacBook-Air week04 % conda activate week04
(week04) donghengxu@donghengxudeMacBook-Air week04 % cat environment.yml
name: week04
channels:
  - conda-forge
dependencies:
  - python=3.12
  - wat-inspector%                                          (week04) donghengxu@donghengxudeMacBook-Air week04 % conda env update
Channels:
 - conda-forge
 - defaults
 - https://repo.anaconda.com/pkgs/main
 - https://repo.anaconda.com/pkgs/r
Platform: osx-arm64
Collecting package metadata (repodata.json): done
Solving environment: done

Downloading and Extracting Packages:

Preparing transaction: done
Verifying transaction: done
Executing transaction: done
#
# To activate this environment, use
#
#     $ conda activate week04
#
# To deactivate an active environment, use
#
#     $ conda deactivate

(week04) donghengxu@donghengxudeMacBook-Air week04 %
```

Python 语法保留字 (reserved key words)（紫色字符）



```
>>> import keyword
>>> print(keyword.kwlist)
['False', 'None', 'True', 'and', 'as', 'assert', 'async', 'await', 'break', 'class', 'continue', 'def', 'del', '
elif', 'else', 'except', 'finally', 'for', 'from', 'global', 'if', 'import', 'in', 'is', 'lambda', 'nonlocal', '
not', 'or', 'pass', 'raise', 'return', 'try', 'while', 'with', 'yield']
>>>
```

语句 (statement) 和表达式 (expression)



大语句，子语句，可包含表达式：例如

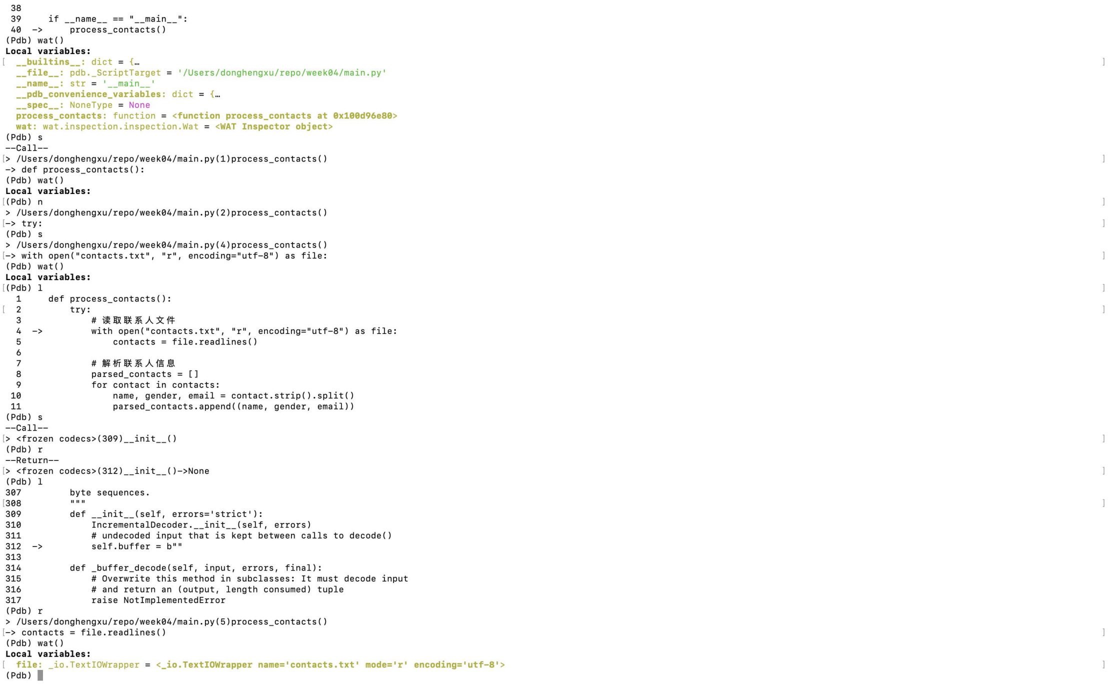
```
with open("contacts.txt", "r", encoding="utf-8") as file:
contacts = file.readlines()
```

contacts = file.readlines()为赋值语句

表达式：例如 file.readlines()

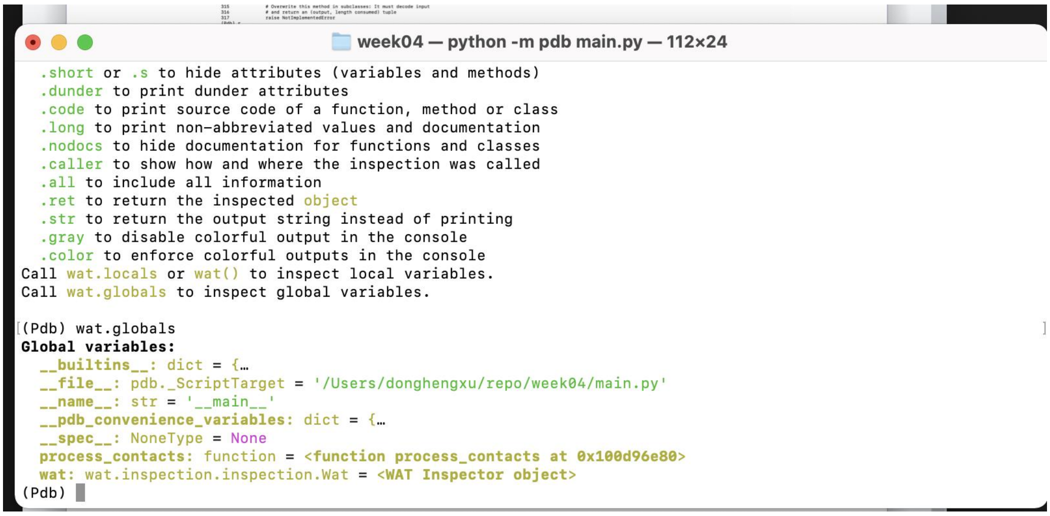
缩进 (indent)：语句的层级，界定子语句的边界

局部变量 (local variable)



```
 38
 39     if __name__ == "__main__":
 40  ->     process_contacts()
(Pdb) wat()
Local variables:
  __builtins__: dict = {…
  __file__: pdb._ScriptTarget = '/Users/donghengxu/repo/week04/main.py'
  __name__: str = '__main__'
  __pdb_convenience_variables: dict = {…
  __spec__: NoneType = None
  process_contacts: function = <function process_contacts at 0x100d96e80>
  wat: wat.inspection.inspection.Wat = <WAT Inspector object>
(Pdb) s
--Call--
> /Users/donghengxu/repo/week04/main.py(1)process_contacts()
-> def process_contacts():
(Pdb) wat()
Local variables:
(Pdb) n
> /Users/donghengxu/repo/week04/main.py(2)process_contacts()
-> try:
(Pdb) s
> /Users/donghengxu/repo/week04/main.py(4)process_contacts()
-> with open("contacts.txt", "r", encoding="utf-8") as file:
(Pdb) wat()
Local variables:
(Pdb) l
  1     def process_contacts():
  2         try:
  3             # 读取联系人文件
  4  ->         with open("contacts.txt", "r", encoding="utf-8") as file:
  5                 contacts = file.readlines()
  6
  7             # 解析联系人信息
  8             parsed_contacts = []
  9             for contact in contacts:
 10                 name, gender, email = contact.strip().split()
 11                 parsed_contacts.append((name, gender, email))
(Pdb) s
--Call--
> <frozen codecs>(309)__init__()
(Pdb) r
--Return--
> <frozen codecs>(312)__init__()->None
(Pdb) l
307         byte sequences.
308         """
309         def __init__(self, errors='strict'):
310             IncrementalDecoder.__init__(self, errors)
311             # undecoded input that is kept between calls to decode()
312  ->         self.buffer = b""
313
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
(Pdb) r
> /Users/donghengxu/repo/week04/main.py(5)process_contacts()
-> contacts = file.readlines()
(Pdb) wat()
Local variables:
  file: _io.TextIOWrapper = <_io.TextIOWrapper name='contacts.txt' mode='r' encoding='utf-8'>
(Pdb)
```

全局变量 (global variable)



```
315             # Overwrite this method in subclasses: It must decode input
316             # and return an (output, length consumed) tuple
317             raise NotImplementedError
```

```
  .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) wat.globals
Global variables:
  __builtins__: dict = {…
  __file__: pdb._ScriptTarget = '/Users/donghengxu/repo/week04/main.py'
  __name__: str = '__main__'
  __pdb_convenience_variables: dict = {…
  __spec__: NoneType = None
  process_contacts: function = <function process_contacts at 0x100d96e80>
  wat: wat.inspection.inspection.Wat = <WAT Inspector object>
(Pdb) 
```

函数 (function) 的定义 (define) 和调用 (call)

字面值 (literal) “” 中的内容

包含(字符串 (str)、整数 (int)、列表 (list)、字典 (dict)、元组 (tuple))

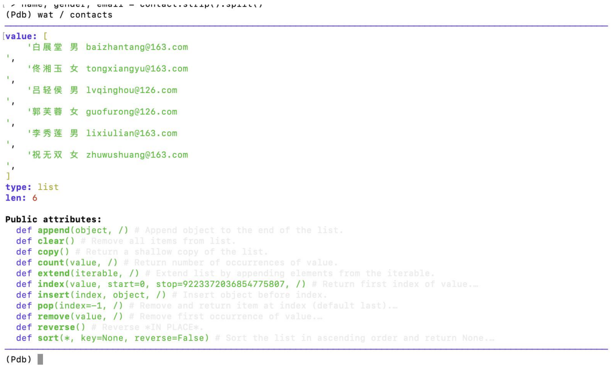
运算符 (operator)：if else == . 等

形参 (parameter)抽象

实参 (argument)调用具体的内容

返回值 (return value) ： return

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



```
> name, gender, email = contact.strip().split()
(Pdb) wat / contacts
────────────────────────────────────────────────────────────
value: [
    '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
',
]
type: list
len: 6

Public attributes:
  def append(object, /) # Append object to the end of the list.
  def clear() # Remove all items from list.
  def copy() # Return a shallow copy of the list.
  def count(value, /) # Return number of occurrences of value.
  def extend(iterable, /) # Extend list by appending elements from the iterable.
  def index(value, start=0, stop=9223372036854775807, /) # Return first index of value.…
  def insert(index, object, /) # Insert object before index.
  def pop(index=-1, /) # Remove and return item at index (default last).…
  def remove(value, /) # Remove first occurrence of value.…
  def reverse() # Reverse *IN PLACE*.
  def sort(*, key=None, reverse=False) # Sort the list in ascending order and return None.…
────────────────────────────────────────────────────────────
(Pdb) 
```