

第六周打卡

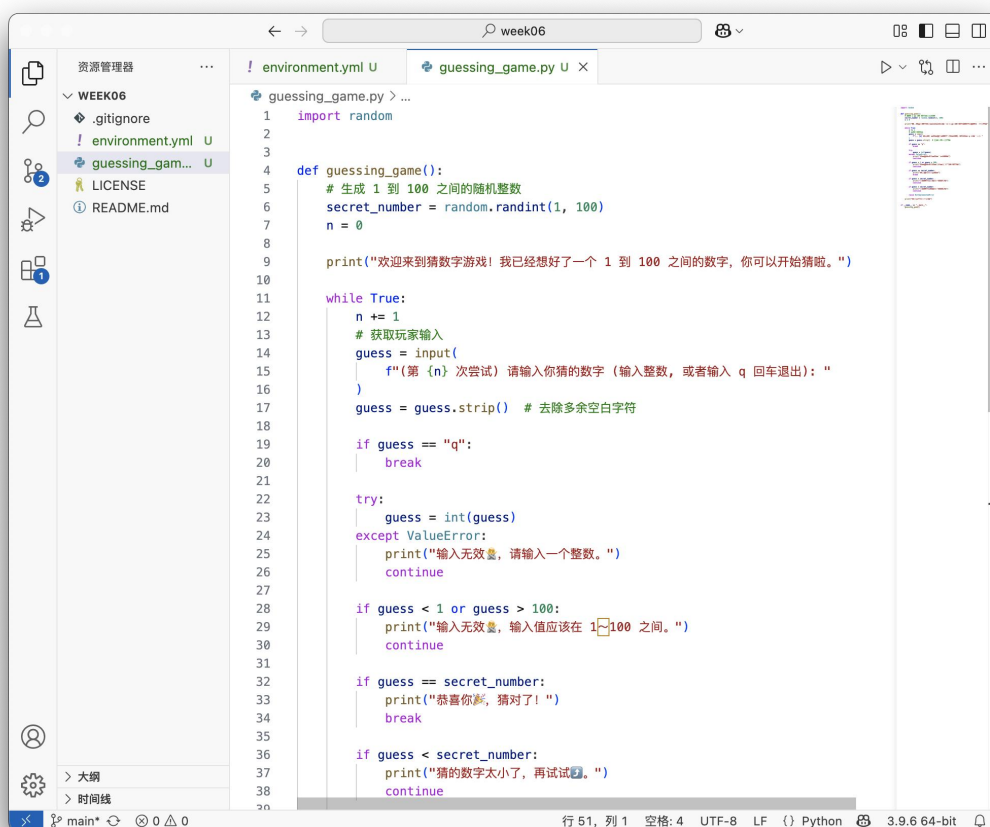
Python 代码组织

一、guessing_game.py 文件

1、将第五周 environment.yml 文件复制到第六周文件中

```
[(base) sjy@sunjiayideMacBook-Pro week06 % cp ../week05/environment.yml ./]
[(base) sjy@sunjiayideMacBook-Pro week06 % ls -l]
total 56
-rw-r--r--  1 sjy  staff  18411  4 13 20:51 LICENSE
-rw-r--r--  1 sjy  staff   2216  4 13 20:51 README.md
-rw-r--r--  1 sjy  staff    89  4 13 21:04 environment.yml
(base) sjy@sunjiayideMacBook-Pro week06 %
```

2、创建 guessing_game.py, 复制粘贴代码



3、使用 pdb 调试流程

```
(week06) sjy@sunjiayideMacBook-Pro week06 % python -m pdb guessing_game.py
> /Users/sjy/cm36/week06/guessing_game.py(1)<module>()
-> import random
(Pdb)
```

```

(Pdb) n
> /Users/sjy/cm36/week06/guessing_game.py(6)guessing_game()
-> secret_number = random.randint(1, 100)
(Pdb) p random.randint(1,100)
48
(Pdb) p random.randint(1,100)
7
(Pdb) p random.randint(1,100)
79
(Pdb) p random.randint(1,100)
96
(Pdb)

```

```

(Pdb) n
> /Users/sjy/cm36/week06/guessing_game.py(7)guessing_game()
-> n = 0
(Pdb) 1
2
3
4     def guessing_game():
5         # 生成 1 到 100 之间的随机整数
6         secret_number = random.randint(1, 100)
7     ->     n = 0
8
9         print("欢迎来到猜数字游戏！我已经想好了一个 1 到 100 之间的数字，你
可以开始猜啦。")
10
11         while True:
12             n += 1
(Pdb) p secret_number
62
(Pdb) n
> /Users/sjy/cm36/week06/guessing_game.py(9)guessing_game()
-> print("欢迎来到猜数字游戏！我已经想好了一个 1 到 100 之间的数字，你可以开始猜
啦。")
(Pdb) 1
4     def guessing_game():
5         # 生成 1 到 100 之间的随机整数
6         secret_number = random.randint(1, 100)
7         n = 0
8
9     ->     print("欢迎来到猜数字游戏！我已经想好了一个 1 到 100 之间的数字，你
可以开始猜啦。")
10
11         while True:
12             n += 1
13             # 获取玩家输入
14             guess = input(
(Pdb) p n
0

```

➤ while 条件循环

```

11 ->     while True:
12         n += 1
13         # 获取玩家输入
14         guess = input(
15             f"(第 {n} 次尝试) 请输入你猜的数字 (输入整数, 或者输入 q 回
车退出): "

```

```

12         n += 1
13         # 获取玩家输入
14         guess = input(
15     ->         f"(第 {n} 次尝试) 请输入你猜的数字 (输入整数, 或者输入 q 回
车退出): "
16         )
17         guess = guess.strip() # 去除多余空白字符
18
19         if guess == "q":
20             break
(Pdb) n
> /Users/sjy/cm36/week06/guessing_game.py(14)guessing_game()
-> guess = input(
(Pdb) 1
9         print("欢迎来到猜数字游戏! 我已经想好了一个 1 到 100 之间的数字, 你
可以开始猜啦。")
10
11         while True:
12             n += 1
13             # 获取玩家输入
14     ->             guess = input(
15             f"(第 {n} 次尝试) 请输入你猜的数字 (输入整数, 或者输入 q 回
车退出): "
16             )
17             guess = guess.strip() # 去除多余空白字符
18
19             if guess == "q":
(Pdb) n
(第 1 次尝试) 请输入你猜的数字 (输入整数, 或者输入 q 回车退出): 35
> /Users/sjy/cm36/week06/guessing_game.py(17)guessing_game()
-> guess = guess.strip() # 去除多余空白字符
(Pdb) 1
12         n += 1
13         # 获取玩家输入
14         guess = input(
15     ->         f"(第 {n} 次尝试) 请输入你猜的数字 (输入整数, 或者输入 q 回
车退出): "
16         )
17     ->         guess = guess.strip() # 去除多余空白字符
18
19         if guess == "q":
20             break
21
22         try:
(Pdb) p guess
'35'
(Pdb)

```

▶ 应该先转化成整数, 再进行比较

```

try:
    guess = int(guess)
except ValueError:
    print("输入无效⚠️, 请输入一个整数。")
    continue

if guess < 1 or guess > 100:
    print("输入无效⚠️, 输入值应该在 1~100 之间。")
    continue

```

4、使用 python 直接调试 (运行)

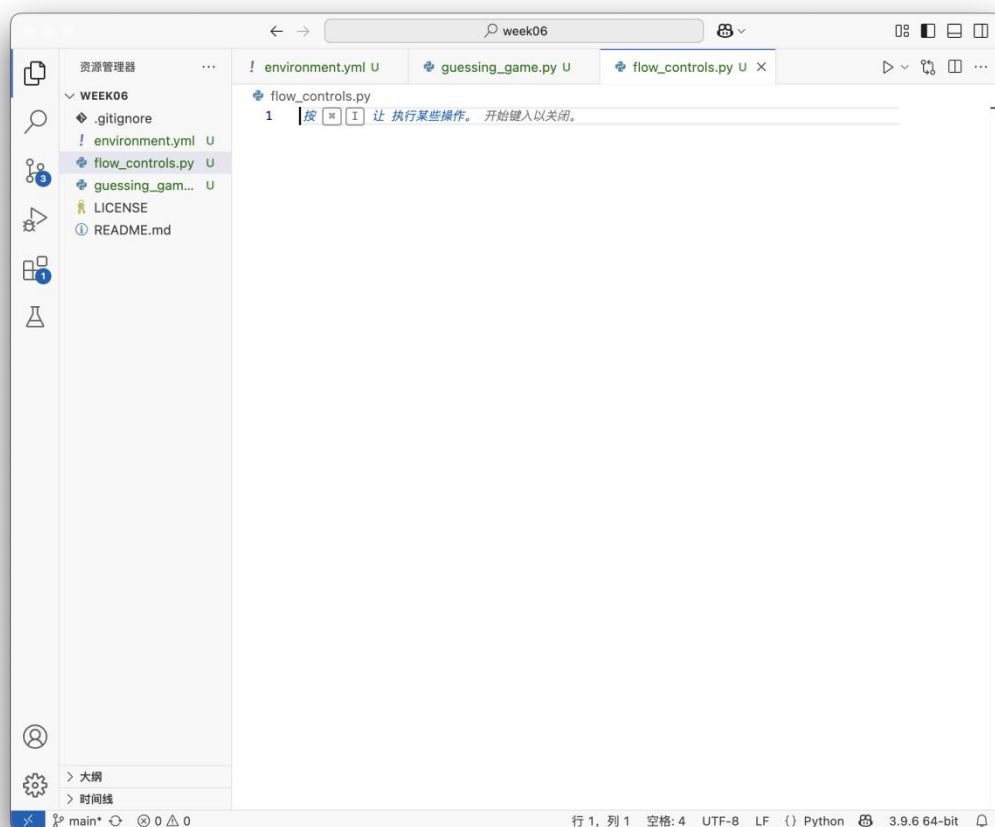
```
(week06) sjy@sunjiayideMacBook-Pro week06 % python guessing_game.py
欢迎来到猜数字游戏！我已经想好了一个 1 到 100 之间的数字，你可以开始猜啦。
(第 1 次尝试) 请输入你猜的数字 (输入整数，或者输入 q 回车退出): q
游戏结束，再见 🙋。
(week06) sjy@sunjiayideMacBook-Pro week06 %
```

```
(Pdb) n
欢迎来到猜数字游戏！我已经想好了一个 1 到 100 之间的数字，你可以开始猜啦。
(第 1 次尝试) 请输入你猜的数字 (输入整数，或者输入 q 回车退出): 88ss
输入无效 🙋，请输入一个整数。
```

```
(Pdb) q
(week06) sjy@sunjiayideMacBook-Pro week06 % python guessing_game.py
欢迎来到猜数字游戏！我已经想好了一个 1 到 100 之间的数字，你可以开始猜啦。
(第 1 次尝试) 请输入你猜的数字 (输入整数，或者输入 q 回车退出): 88
猜的数字太小了，再试试 🙋
(第 2 次尝试) 请输入你猜的数字 (输入整数，或者输入 q 回车退出): 90
猜的数字太小了，再试试 🙋
(第 3 次尝试) 请输入你猜的数字 (输入整数，或者输入 q 回车退出): 94
猜的数字太大了，再试试 🙋
(第 4 次尝试) 请输入你猜的数字 (输入整数，或者输入 q 回车退出): 92
恭喜你 🎉，猜对了！
游戏结束，再见 🙋。
(week06) sjy@sunjiayideMacBook-Pro week06 %
```

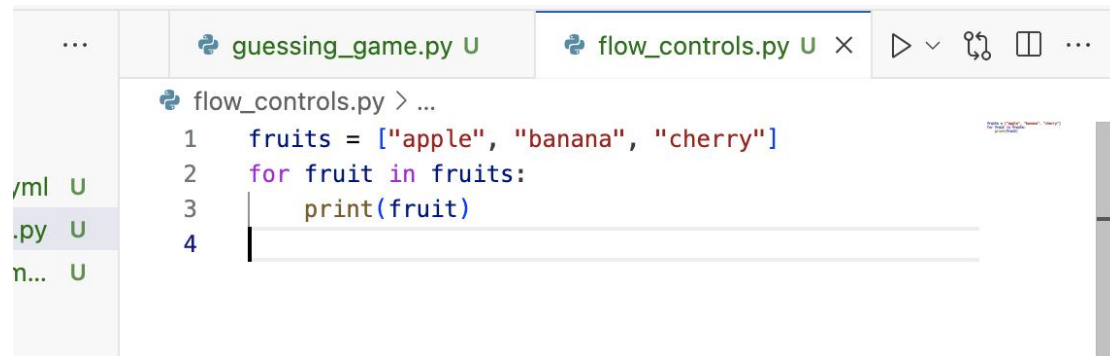
二、体会 python 流程控制语句

创建 flow_controls.py 文件

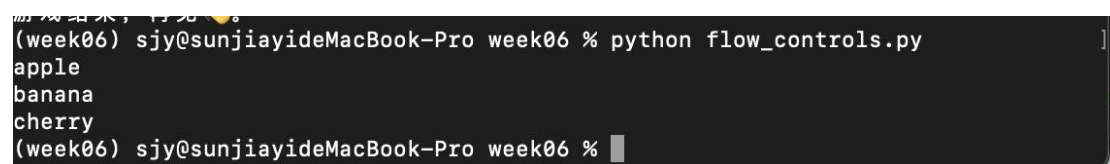


1、for 迭代循环

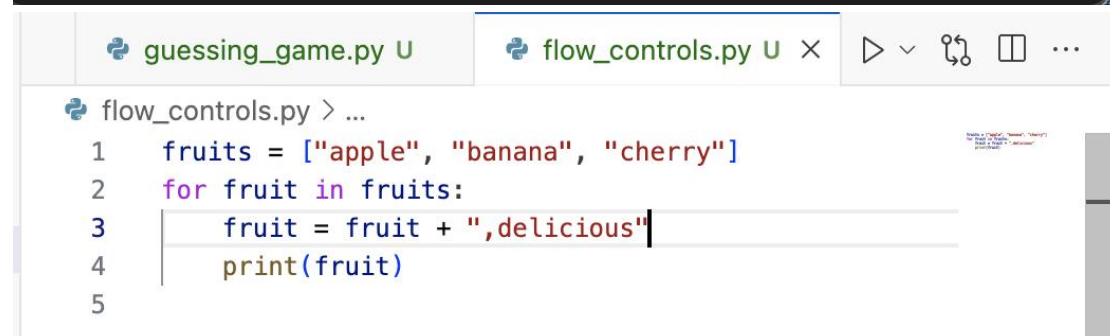
(1) 遍历列表



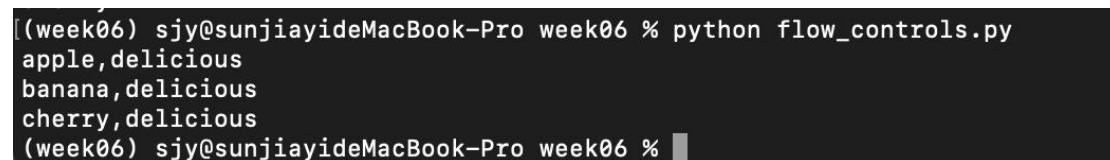
```
...  
guessing_game.py U  
flow_controls.py U ×  
flow_controls.py > ...  
1 fruits = ["apple", "banana", "cherry"]  
2 for fruit in fruits:  
3     print(fruit)  
4
```



```
(week06) sjy@sunjiayideMacBook-Pro week06 % python flow_controls.py  
apple  
banana  
cherry  
(week06) sjy@sunjiayideMacBook-Pro week06 %
```

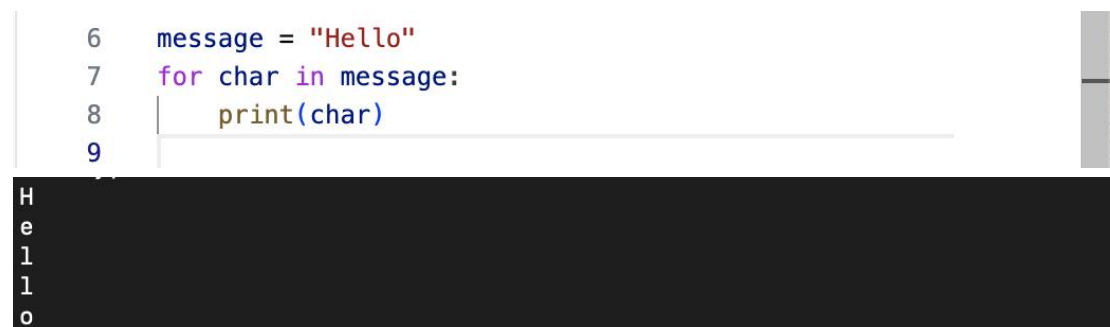


```
guessing_game.py U  
flow_controls.py U ×  
flow_controls.py > ...  
1 fruits = ["apple", "banana", "cherry"]  
2 for fruit in fruits:  
3     fruit = fruit + ",delicious"  
4     print(fruit)  
5
```




```
(week06) sjy@sunjiayideMacBook-Pro week06 % python flow_controls.py  
apple,delicious  
banana,delicious  
cherry,delicious  
(week06) sjy@sunjiayideMacBook-Pro week06 %
```

(2) 遍历字符串



```
6 message = "Hello"  
7 for char in message:  
8     print(char)  
9
```



```
H  
e  
l  
l  
o
```

(3) 遍历数字: 从 0 开始

```
9
10 for i in range(6):
11     print(i)
12
```

```
0
1
2
3
4
5
```

(4) 遍历字典

```
13 student = {"name": "John", "age": 20, "grade": "A"}
14 for key in student:
15     print(key, ":", student[key])
16
```

```
4
5
name : John
age : 20
grade : A
(week06) sjy@sunjiayideMacBook-Pro week06 %
```

▶ key 和 value 的区别

```
13 student = {"name": "John", "age": 20, "grade": "A"}
14 for key in student:
15     print(key, ":", student[key])
16
17 student = {"name": "John", "age": 20, "grade": "A"}
18 for value in student.values():
19     print(value)
```

```
name : John
age : 20
grade : A
John
20
A
```


2、while 条件循环

(1) 计数循环

```
21 count = 0
22 while count < 5:
23     print(count)
24     count = count + 1
```

```
0
1
2
3
4
```

(2) 从列表中移除元素 (**while** 后面是个表达式)

```
26 numbers = [1, 2, 3, 4, 5]
27 while numbers:
28     print(numbers.pop())
29
```

```
5
4
3
2
1
```

(3) 猜数字游戏

```
31 import random
32
33 secret_number = random.randint(1, 10)
34 guess = None
35 while guess != secret_number:
36     guess = int(input("猜一个 1 到 10 之间的数字: "))
37     if guess < secret_number:
38         print("猜的数字太小了, 再试一次!")
39     elif guess > secret_number:
40         print("猜的数字太大了, 再试一次!")
41 print("恭喜你, 猜对了!")
```

```
猜一个 1 到 10 之间的数字: 8
猜的数字太大了, 再试一次!
猜一个 1 到 10 之间的数字: 4
猜的数字太大了, 再试一次!
猜一个 1 到 10 之间的数字: 2
恭喜你, 猜对了!
```

3、for...else 循环未被打断的处理

```
44 numbers = [2, 4, 6, 8, 10]
45 target = 3
46
47 for num in numbers:
48     if num == target:
49         print(f"找到了目标数字 {target}")
50         break
51 else:
52     print(f"未找到目标数字 {target}")
```

未找到目标数字 3

4、if 条件分支

```
55 score = 85
56 if score >= 90:
57     print("成绩等级为 A。")
58 elif score >= 80:
59     print("成绩等级为 B。")
60 elif score >= 70:
61     print("成绩等级为 C。")
62 elif score >= 60:
63     print("成绩等级为 D。")
64 else:
65     print("成绩等级为 F。")
```

成绩等级为 B。

5、if...elif...elif 多重条件分支

```
55 month = 7
56 if 3 <= month <= 5:
57     season = "春季"
58 elif 6 <= month <= 8:
59     season = "夏季"
60 elif 9 <= month <= 11:
61     season = "秋季"
62 elif month == 12 or 1 <= month <= 2:
63     season = "冬季"
64 else:
65     season = "输入的月份不合法"
66 print(f"{month} 月属于 {season}。")
```



```
7 月属于 夏季。
```

6、if...else 未满足条件的处理

```
67
68     num = -5
69     if num >= 0:
70         print(f"{num} 是一个非负数。")
71     else:
72         print(f"{num} 是一个负数。")
```

```
-5 是一个负数。
```

```
(week06) sjy@sunjiayideMacBook-Pro week06 %
```

7、try...except[...except...else...finally]捕捉异常的处理

```
75     try:
76         num = int("abc")
77     except ValueError:
78         print("错误：输入不是有效的整数！")
79     except ZeroDivisionError:
80         print("错误：不能除以零！")
```

```
错误：输入不是有效的整数！
```

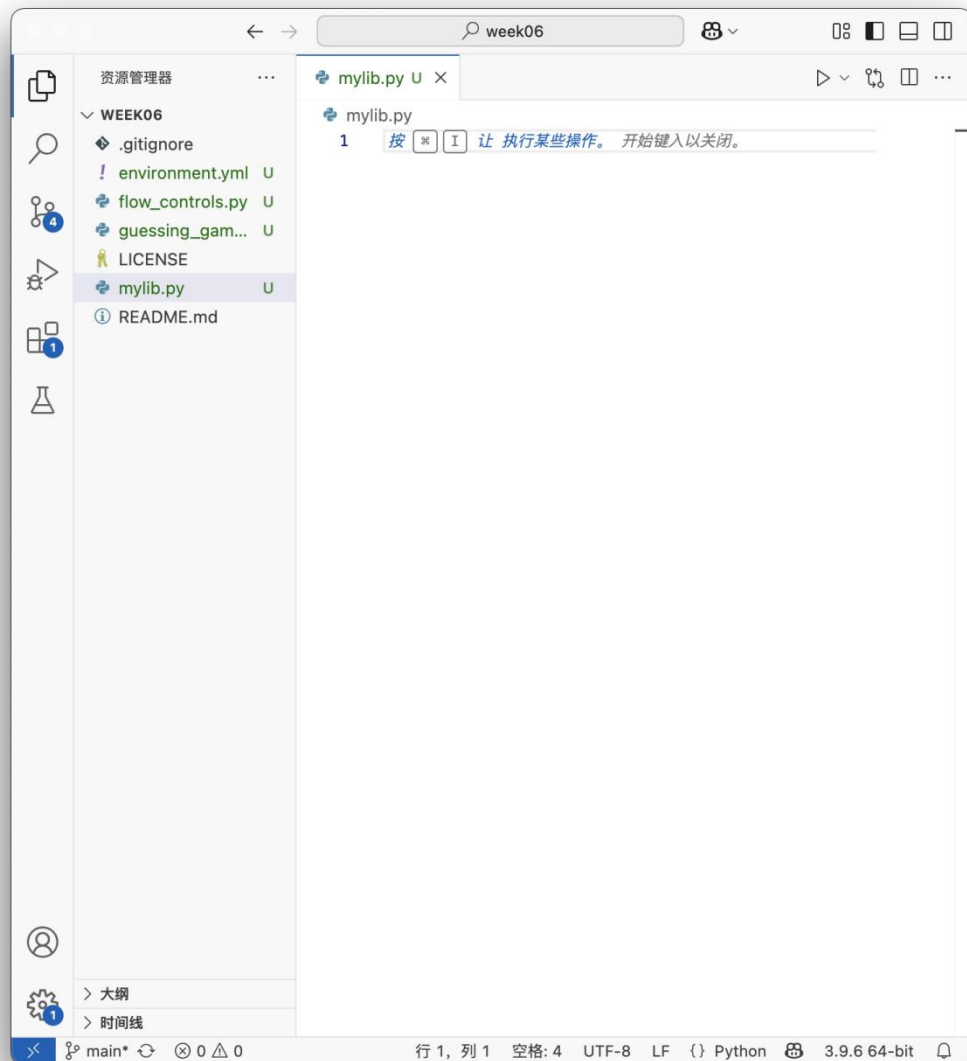
8、raise 主动抛出异常

```
82
83     numbers = [1, 2, -3, 4, 5]
84     for num in numbers:
85         if num < 0:
86             raise ValueError(f"发现负数：{num}")
87         print(num)
88
```

```
1
2
Traceback (most recent call last):
  File "/Users/sjy/cm36/week06/flow_controls.py", line 86, in <module>
    raise ValueError(f"发现负数：{num}")
ValueError: 发现负数：-3
```

三、导入函数并尝试调用

1、创建一个 mylib.py 模块



2、定义函数

(1) 定义函数 func1，没有形参，没有返回值——会返回 none

```
1 def func1():
2     x = 50
3     y = x**0.8 - 82
4     print(y)
```

```
((week06) sjy@sunjiayideMacBook-Pro week06 % python myjob.py
-59.13474740363368
(week06) sjy@sunjiayideMacBook-Pro week06 %
```

```

1 import mylib # noqa: F401
2
3 y = mylib.func1()
4 print(y)
5

```

```

[week06] sjy@sunjiayideMacBook-Pro week06 % python myjob.py
-59.13474740363368
None
[week06] sjy@sunjiayideMacBook-Pro week06 %

```

```

6 try:
7     y = mylib.func1(0)
8 except TypeError as e:
9     print(e)
10

```

```

[week06] sjy@sunjiayideMacBook-Pro week06 % python myjob.py
-59.13474740363368
None
func1() takes 0 positional arguments but 1 was given
[week06] sjy@sunjiayideMacBook-Pro week06 %

```

(2) 定义函数 func2，没有形参，有返回值

```

7 def func2():
8     x = 100
9     y = x**0.8 - 82
10    print(y)
11    return y

```

```

-42.189282944650266
-42.189282944650266
[week06] sjy@sunjiayideMacBook-Pro week06 %

```

(3) 定义函数 func3，只有一个位置形参

✖ 不传实参会报错

```

14 def func3(x):
15     y = x**0.8 - 82
16     return y
--
14 y = mylib.func3()
15 print(y)
16

```

```

Traceback (most recent call last):
  File "/Users/sjy/cm36/week06/myjob.py", line 14, in <module>
    y = mylib.func3()
    ^^^^^^^^^^^^^^^^^
TypeError: func3() missing 1 required positional argument: 'x'
(week06) sjy@sunjiayideMacBook-Pro week06 %

```

✓ 位置实参

```

14 y = mylib.func3(56)
15 print(y)
16

```

```

-56.96483710157652
(week06) sjy@sunjiayideMacBook-Pro week06 %

```

✓ 命名实参

```

14 y = mylib.func3(56)
15 print(y)
16 y = mylib.func3(x=73)
17 print(y)
--

```

```

-56.96483710157652
-51.05016855979905
(week06) sjy@sunjiayideMacBook-Pro week06 %

```

(4) 定义函数 func4，只有一个命名形参

```

19 def func4(x=732):
20     y = x**0.8 - 82
21     return y
22

```

✓ 位置实参

```

23 y = mylib.func4(22)
24 print(y)
25

```

```

-70.14400125516002
(week06) sjy@sunjiayideMacBook-Pro week06 %

```

✓ 命名实参

```

25 y = mylib.func4(x=88)
26 print(y)
27

```

```

-70.14400125516002
-46.059332579839094
(week06) sjy@sunjiayideMacBook-Pro week06 %

```

✓ 不传实参(取默认值)

```
27 y = mylib.func4()  
28 print(y)
```

```
113.70812891412885
```

```
(week06) sjy@sunjiayideMacBook-Pro week06 %
```

(5) 定义函数 func5, 接受多个位置形参和命名形参——豆包举例

```
--  
23  
24 # 定义一个函数, 包含位置形参和命名形参  
25 def calculate_area(length, width=10):  
26     return length * width  
27  
28  
29 # 调用函数, 使用位置实参和命名实参  
30 area1 = calculate_area(5) # 使用位置实参传递 length, width  
31 area2 = calculate_area(length=7, width=3) # 使用命名实参  
32
```

```
113.70812891412885
```

```
50
```

```
21
```

(6) 定义函数 func6, 在形参列表中使用/来限定只接受位置实参的形参

```
--  
34 def func6(a, /, b, operation="add"):  
35     if operation == "add":  
36         return a + b  
37     elif operation == "subtract":  
38         return a - b  
39     else:  
40         return None  
41
```

```
--  
34 print(mylib.func6(a=18, b=179))
```

```
21
```

```
Traceback (most recent call last):
```

```
File "/Users/sjy/cm36/week06/myjob.py", line 34, in <module>
```

```
    print(mylib.func6(a=18, b=179))
```

```
^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
```

```
TypeError: func6() got some positional-only arguments passed as keyword arguments: 'a'
```

```
(week06) sjy@sunjiayideMacBook-Pro week06 %
```

(7) 定义函数 func7, 在形参列表中使用*来限定只接受命名实参的形参

```
43 def func7(a, /, b, *, operation="add"):
44     if operation == "add":
45         return a + b
46     elif operation == "subtract":
47         return a - b
48     else:
49         return None
50
51
39
40 print(mylib.func7(18, 179))
41
```

```
197
(week06) sjy@sunjiayideMacBook-Pro week06 %
```

(8) 定义函数 func8, 在位置形参的最后, 在形参名称前使用*允许传入任意数量的位置实参

```
51
52 def func8(*args):
53     total = 0
54     for num in args:
55         total = total + num
56     return total
57
41
42 print(mylib.func8(18, 179, 82837))
43 print(mylib.func8(18, 179, 82837, 73628, 2891))
44 print(mylib.func8(18, 179))
```

```
83034
159553
197
(week06) sjy@sunjiayideMacBook-Pro week06 %
```

(9) 定义函数 func9, 在命名形参的最后, 在形参名称前使用**允许传入任意数量的命名实参(被打包为字典)

```
59 def func9(**kwargs):
60     for key, value in kwargs.items():
61         print(f"{key}: {value}")
62
```



```

46 mylib.func9(name="Alice", age=25, city="New York")
47 |

```

```

name: Alice
age: 25
city: New York
(week06) sjy@sunjiayideMacBook-Pro week06 %

```

(10) 定义函数 func10，接受两个位置形参，一个命名形参，尝试在调用时使用*将可迭代对象(如元组或列表)自动解包，按位置实参传入

```

6 # 定义一个包含两个元素的元组
7 tuple_args = (2, 3)
8 result = func10(*tuple_args)
9 print(f"三个参数的总和: {result}")
48 tuple_args = (2, 3)
49 mylib.func10(*tuple_args)

```

```

第一个位置形参的值: 2
第二个位置形参的值: 3
命名形参的值: 10

```

```

48 tuple_args = (2, 3)
49 mylib.func10(*tuple_args)
50 list_args = [5, 7]
51 mylib.func10(*list_args, named_arg=20)

```

```

第一个位置形参的值: 2
第二个位置形参的值: 3
命名形参的值: 10
第一个位置形参的值: 5
第二个位置形参的值: 7
命名形参的值: 20

```

(11) 定义函数 func12，给函数添加内嵌文档，给形参和返回值添加类型注解

✓ 没有什么约束力

```

91 def func12(
92     positional_arg1: str, positional_arg2: int, named_arg: str = "default"
93 ) -> None:
94     """多个参数的例子"""
95     print(f"第一个位置形参的值: {positional_arg1}")
96     print(f"第二个位置形参的值: {positional_arg2}")
97     print(f"命名形参的值: {named_arg}")
98     return positional_arg1 + positional_arg2 + named_arg
99

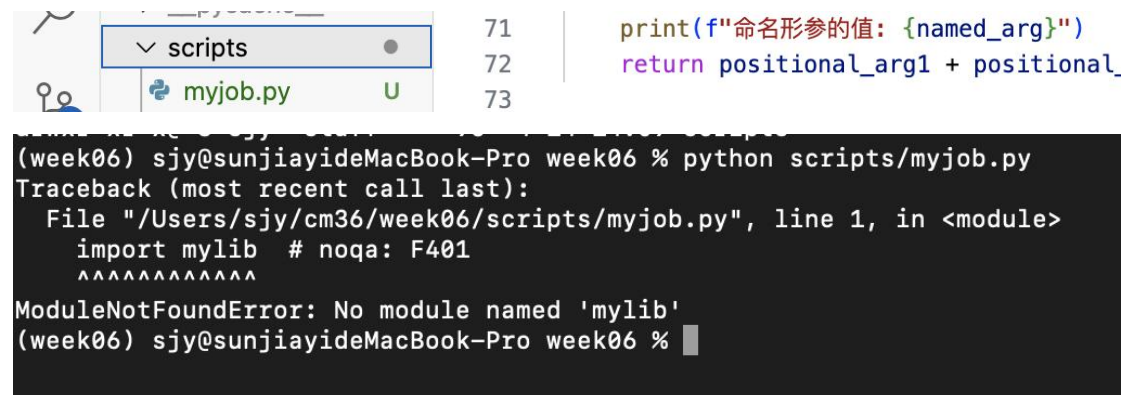
```

```
53 mylib.func12(72378, 32798, 372)
```

```
命名形参的值: 372
第一个位置形参的值: 72378
第二个位置形参的值: 32798
命名形参的值: 372
```

四、脚本打包软件包

1、把 myjob.py 移动到 scripts/myjob.py，发现运行失败




```
71 | print(f"命名形参的值: {named_arg}")
72 | return positional_arg1 + positional_
73 |
74 |
75 |
76 |
77 |
78 |
79 |
80 |
81 |
82 |
83 |
84 |
85 |
86 |
87 |
88 |
89 |
90 |
91 |
92 |
93 |
94 |
95 |
96 |
97 |
98 |
99 |
100 |
```

```
(week06) sjy@sunjiayideMacBook-Pro week06 % python scripts/myjob.py
Traceback (most recent call last):
  File "/Users/sjy/cm36/week06/scripts/myjob.py", line 1, in <module>
    import mylib # noqa: F401
    ^^^^^^^^^^^^^
ModuleNotFoundError: No module named 'mylib'
(week06) sjy@sunjiayideMacBook-Pro week06 %
```

🔧 得安装后再使用，而不是直接 import，没有打包成软件包

2、将 mylib.py 模块移动至 src/mypkg/mylib.py



```
4 | print(y)
5 |
6 |
7 | def func2():
8 |     x = 100
9 |     y = x**0.8 - 82
10 |     print(y)
```

3、创建 src/mypkg/__init__.py 文件

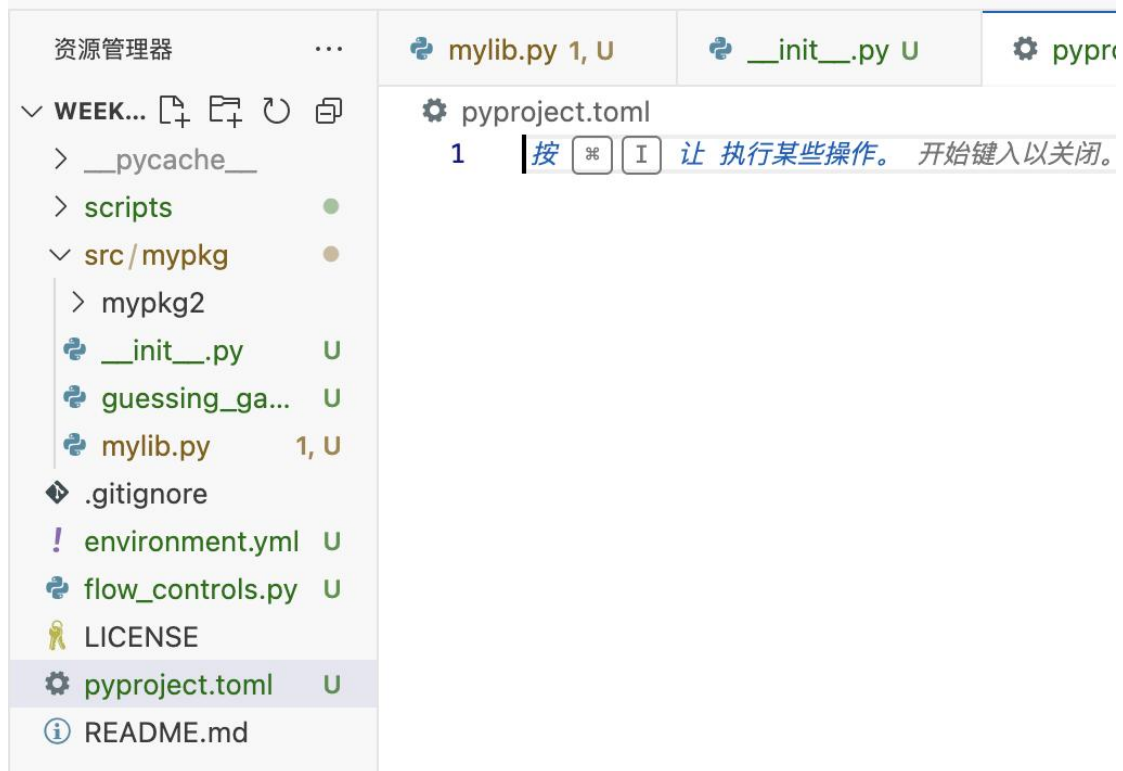
✅ 可以是空的，但一定要有这个文件，这样就知道当前文件夹是一个软件包



```
src/mypkg
├── __init__.py
├── guessing_ga...
└── mylib.py
```

4、在项目中创建 pyproject.toml——用来描述软件包的基本信息

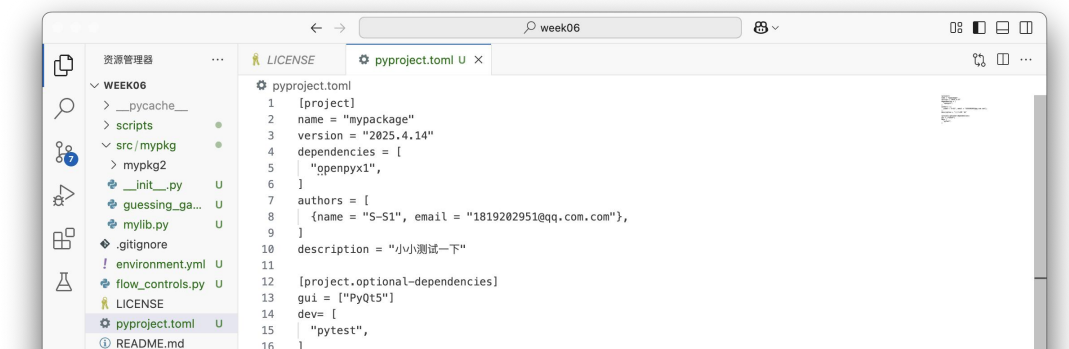
(1) 创建



3 version = "0.1.1"

- ✓ 大版本号：如果修改基本上推翻重做
- ✓ 中版本号：保证了大多数的兼容性，不会发生巨大的改变
- ✓ 小版本号：最后一个就是缺陷的修补，一定要及时更新升级

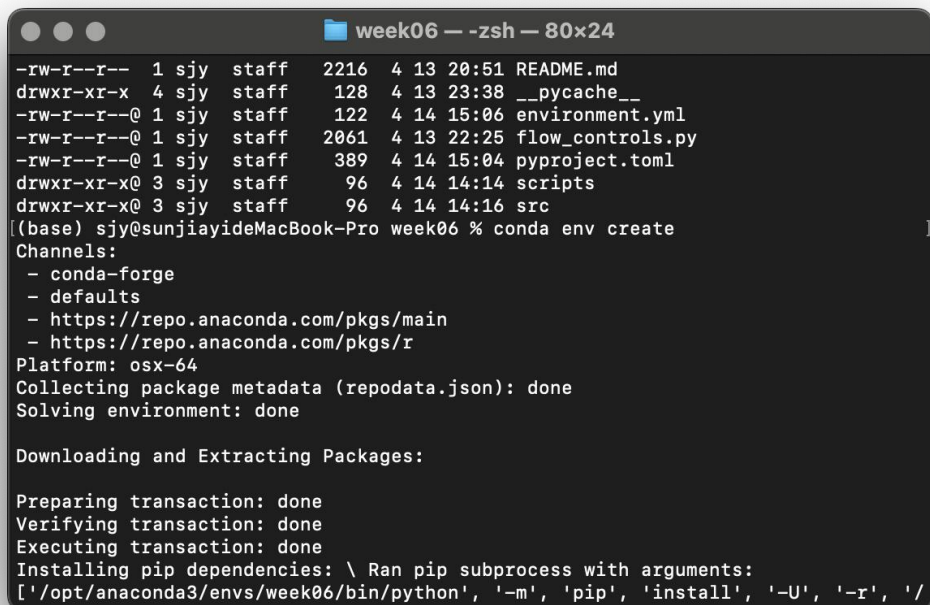
(2) 填写基本软件包信息



(3) 填写软件包的构建配置

```
18 [build-system]
19 requires = ["hatchling"]
20 build-backend = "hatchling.build"
21
22 [tool.hatch.build.targets.wheel]
23 packages = ["src/mypkg"]
24
```

5、修改 environment.yml 文件，使得 condaenvcreate 自动安装本地可编辑软件包



```
week06 — zsh — 80x24
-rw-r--r--  1 sjy  staff   2216  4 13 20:51 README.md
drwxr-xr-x  4 sjy  staff    128  4 13 23:38 __pycache__
-rw-r--r--@ 1 sjy  staff    122  4 14 15:06 environment.yml
-rw-r--r--@ 1 sjy  staff  2061  4 13 22:25 flow_controls.py
-rw-r--r--@ 1 sjy  staff   389  4 14 15:04 pyproject.toml
drwxr-xr-x@ 3 sjy  staff    96  4 14 14:14 scripts
drwxr-xr-x@ 3 sjy  staff    96  4 14 14:16 src
[(base) sjy@sunjiayideMacBook-Pro week06 % conda env create ]
Channels:
- conda-forge
- defaults
- https://repo.anaconda.com/pkgs/main
- https://repo.anaconda.com/pkgs/r
Platform: osx-64
Collecting package metadata (repodata.json): done
Solving environment: done

Downloading and Extracting Packages:

Preparing transaction: done
Verifying transaction: done
Executing transaction: done
Installing pip dependencies: \ Ran pip subprocess with arguments:
['/opt/anaconda3/envs/week06/bin/python', '-m', 'pip', 'install', '-U', '-r', '/
```