conda env create
回到 base 环境 conda deactivate
删除环境 conda env remove -n week06
conda env list
清除 conda env remove -n prj1
生成 1-100 随机数

```
p random.randint(1,100)
```

遍历列表

python

```
fruits = ['apple', 'banana', 'cherry']
for fruit in fruits:
    print(fruit)
```

For I in (5): 0-4

```
student = {"name": "John", "age": 20, "grade": "A"}
for key in student:
    print(key)

student = {"name": "John", "age": 20, "grade": "A"}
for key, value in student.items():
    print(f"{key}: {value}")
```

可以循环列表

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

移除最后一个元素=a.pop()

`break` 打断跳出循环

`continue` 跳至下一轮循环

`for...else` 循环未被打断的处理

`if` 条件分支

`if...elif[...elif]` 多重条件分支

`if...else` 未满足条件的处理

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

`raise` 主动抛出异常

无形参和返回值 = return None print 该函数=None

```
y = mylib.func3(45) #位置传参数
print(y)

y = mylib.func3(x=47) #命名传参数
print(y)
```

有默认值的函数

```
def func4(x=50):
    y = x**0.5 - 7
    return y
```

有默认值的形参应在无默认值的形参后面
斜杠前面的形参只能位置传入，*后面只能命名传入实参

```
def func7(a, /, b, *, operation="add"):
    if operation == "add":
        return a + b
    elif operation == "subtract":
        return a - b
    else:
        return None
```

*接受任意数量的实参

```python
def func8(*args):
    result = 0
    for num in args:
        result += num
    return result
```

**接受任意数量命名实参

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

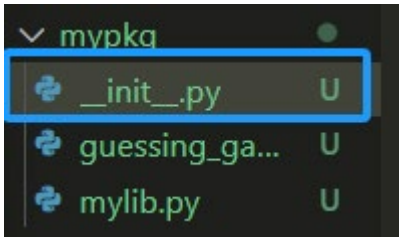
解包传入

```python
iterable = (10, 20)
mylib.func10(*iterable, c="new_value")
```

类型注解

```python
12(a: str, b: int, c: str = "default_value") -> None:
"参数调用的例子"
t(f"a: {a}, b: {b}, c: {c}")
```

文件夹里面有__init__.py 表示这个文件夹是软件包



描述软件包，在软件包文件夹外面

安装 pip install -e .

调用 import mypkg