

python_模块

一、模块的分类

- 1、标准模块即Python自带模块
- 2、第三方模块，比如pymysql等
- 3、自己定义模块，比如自己写一个PY文件就是一个模块

二、导入模块

导入模块的实质就是把你导入的模块运行了一遍，并且可以使用模块的方法及变量
比如 模块dd

```
def moudle():  
  
    print("模块内的方法")  
  
  
print("模块中的打印")
```

导入模块

```
import dd  
  
dd.moudle() #调用模块方法
```

C:\Users\zhaow\AppData\Local\Programs\Python\Python37\python.exe D:/study/python/test/YY.PY.py

模块中的打印 #运行模块中的代码

模块内的方法 #可调模块的方法

三、导入模块的顺序

- 1、在当前目录下找
- 2、在python的环境变量里面去找

```
import sys

print(sys.path) #python环境变量路径，非安装的是系统变量

sys.path.append(r'D:\study\python') #加环境变量路径,默认加在最后位置

sys.path.insert(0,r'D:\study') #为查找的快，插入到环境变量list的第一个路径中

print(sys.path)
```

加入的环境变量

```
C:\Users\zhaow\AppData\Local\Programs\Python\Python37\python.exe D:/study/python/test/dd.py
C:\Users\python\user\; D:\study\python\user\; C:\Users\zhaow\AppData\Local\Programs\Python\Python37\python\;
C:\Users\zhaow\AppData\Local\Programs\Python\Python37\DLLs\; C:\Users\zhaow\AppData\Local\Programs\Python\Python37\lib\
C:\Users\zhaow\AppData\Local\Programs\Python\Python37\; C:\Users\zhaow\AppData\Local\Programs\Python\Python37\Scripts\;
D:\study\; D:\study\python\user\; D:\study\python\user\; C:\Users\zhaow\AppData\Local\Programs\Python\Python37\python\;
C:\Users\zhaow\AppData\Local\Programs\Python\Python37\DLLs\; C:\Users\zhaow\AppData\Local\Programs\Python\Python37\lib\
C:\Users\zhaow\AppData\Local\Programs\Python\Python37\; C:\Users\zhaow\AppData\Local\Programs\Python\Python37\Scripts\; C:\Users\python\;
```

另外环境变量加的是临时的，程序运行结束就仍是原来的固定的Python环境变量路径

四、导入模块，会执行一遍模块中的代码，但是模块中的if __name__ == '__main__':下的代码块不执行

模块代码：

```
def moudle():

    print("模块内的方法")

print("模块中的打印")

print(__name__)

if __name__ == '__main__':

    print("我是导入模块不会执行的代码")
```

C:\Users\zhaow\AppData\Local\Programs\Python\Python37\python.exe D:/study/python/test/dd.py

模块中的打印

__main__

我是导入模块不会执行的代码

导入模块及引用

```
import dd
```

```
dd.moudle()
```

```
print(dd.__name__)
```

C:\Users\zhaow\AppData\Local\Programs\Python\Python37\python.exe D:/study/python/test/YY.PY.py

模块中的打印

```
dd
```

模块内的方法

```
dd
```