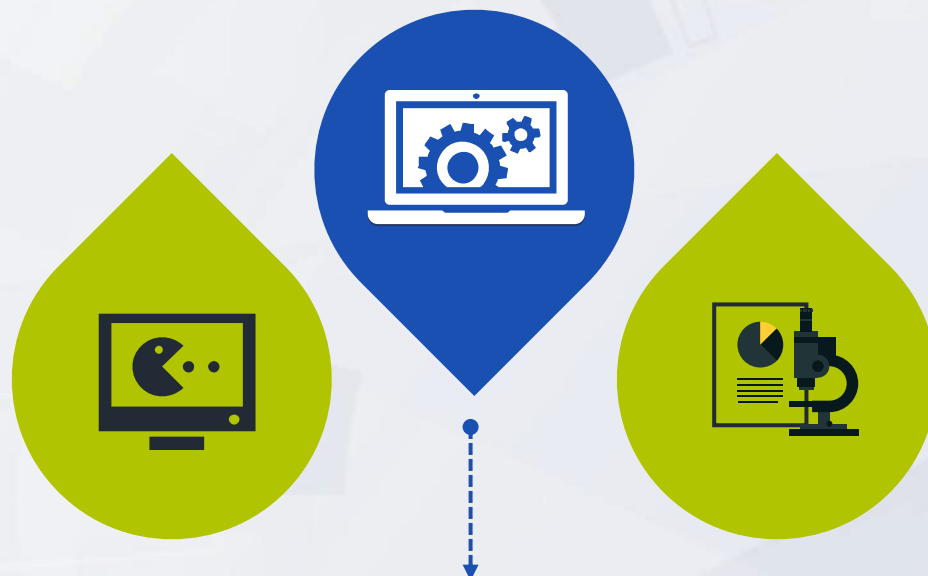


# PYTHON

## 编程基础

# 装饰器

# 概述



利用装饰器，可以在不修改已有函数的情况下向已有函数中注入代码，使其具备新的功能。

一个装饰器可以为多个函数注入代码，一个函数也可以注入多个装饰器的代码。

利用装饰器可以将日志处理、执行时间计算等较为通用的代码注入到不同的函数中，从而使代码更加简洁。



## 例：装饰器示例1

```
1     def deco1(func): #定义函数deco1
2         def inner1(*args, **kwargs): #定义函数inner1
3             print('deco1 begin')
4             func(*args, **kwargs)
5             print('deco1 end')
6         return inner1 #返回函数inner1的引用
```



## 例：装饰器示例1

```
7      def deco2(func): #定义函数deco2
8          def inner2(*args, **kwargs): #定义函数inner2
9              print('deco2 begin')
10             func(*args, **kwargs)
11             print('deco2 end')
12         return inner2 #返回函数inner2的引用
```

## 例：装饰器示例1

```
13     @deco1
14     def f1(a,b): #定义函数f1
15         print('a+b=',a+b)
16     @deco1
17     @deco2
18     def f2(a,b,c): #定义函数f2
19         print('a+b+c=',a+b+c)
20     if __name__ == '__main__':
21         f1(3,5) #调用f1函数
22         f2(1,3,5) #调用f2函数
```

```
print('deco1 begin')
print('a+b=',a+b)
print('deco1 end')

print('deco1 begin')
print('deco2 begin')
print('a+b+c=',a+b+c)
print('deco2 end')
print('deco1 end')
```

```
deco1 begin
a+b= 8
deco1 end
deco1 begin
deco2 begin
a+b+c= 9
deco2 end
deco1 end
```



## 例：装饰器示例2

```
1     def deco1(func): #定义函数deco1
2         def inner1(x,y): #定义函数inner1
3             print('deco1 begin')
4             func(x,y)
5             print('deco1 end')
6         return inner1 #返回函数inner1的引用
7     def deco2(func): #定义函数deco2
8         def inner2(): #定义函数inner2
9             print('deco2 begin')
10            func()
11            print('deco2 end')
12        return inner2 #返回函数inner2的引用
```

**提示：**

▶ deco1只能用于装饰带两个参数的函数，而deco2只能用于装饰没有参数的函数。

## 例：装饰器示例2

```
13  @deco1
14  def f1(a,b): #定义函数f1
15      print('a+b=',a+b)
16  @deco2
17  def f2(): #定义函数f2
18      print('f2 is called')
19  if __name__=='__main__':
20      f1(3,5) #调用f1函数
21      f2() #调用f2函数
```

```
print('deco1 begin')
print('a+b=',a+b)
print('deco1 end')

print('deco2 begin')
print('f2 is called')
print('deco2 end')
```

```
deco1 begin
a+b= 8
deco1 end
deco2 begin
f2 is called
deco2 end
```