

# 图解

# Python

可能是迄今为止

最易懂的Python视频教程



---

"""

如果在一个函数的内部嵌套定义了另外一个函数（姑且将外部的函数和内部的函数分别称之为外函数和内函数），内函数中引用了外函数中的变量，并且外函数的返回值是内函数，这样，就构成了一个闭包。

"""

```
def outer():  
    a = 10  
    def inner():  
        print(a)  
    return inner
```

"""

通常情况下，在函数调用结束后，函数内定义的变量将不再可用。

但是，对于闭包而言，在外函数调用结束后，外函数中被内函数引用的变量仍然是可用的，因为外函数中被内函数引用的变量会被绑定到内函数的特殊属性\_\_closure\_\_中。

"""

```
def do_sth():  
    temp = 8  
    print(temp)
```

```
do_sth()  
# print(temp)
```

```
outer_result = outer()  
outer_result() # 10  
outer()()      # 10
```

```
print(outer_result.__closure__) # (<cell at 0x10eebe228: int object at 0x10eb4d8e0>,)  
print(outer_result.__closure__[0].cell_contents) # 10
```