

图解

Python

可能是迄今为止

最易懂的Python视频教程



```
"""深拷贝"""
```

```
.....
```

```
    可以调用标准库模块copy中的函数deepcopy()实现深拷贝。
```

```
.....
```

```
import copy
```

```
.....
```

```
    对于没有嵌套子对象的不可变对象，例如：整数对象、字符串对象和元组对象等，不会进行拷贝，也就是说，不会创建另一个对象。
```

```
.....
```

```
i = 18
```

```
ic = copy.deepcopy(i)
print(ic)
print('id(i): %s' % id(i))
print('id(ic): %s' % id(ic))
```

```
t = (1, 2, 3)
```

```
tc = copy.deepcopy(t)
print(tc)
print('id(t): %s' % id(t))
print('id(tc): %s' % id(tc))
```

.....

所谓深拷贝，指的是：对于某个对象，创建与该对象具有相同值的另一个对象，同时，这两个对象内部嵌套的对应可变子对象全都不是同一个对象。简单地说，外部和内部都进行了拷贝。

.....

```
L1 = [[3, 6], 8]
L2 = copy.deepcopy(L1) # 调用标准库模块copy中的函数deepcopy()
print(L2)    # [[3, 6], 8]

print('id(L1):%s' % id(L1))
print('id(L2):%s' % id(L2))

print('id(L1[0]):%s' % id(L1[0]))
print('id(L2[0]):%s' % id(L2[0]))

print('id(L1[1]):%s' % id(L1[1]))
print('id(L2[1]):%s' % id(L2[1]))

L1[0][1] = 7
L1[1] = 9
print(L1)    # [[3, 7], 9]
print(L2)    # [[3, 6], 8]
```

.....

如果不可变对象内部又嵌套的可变子对象，深拷贝之后，会创建一个与该不可变对象具有相同值的另一个对象。

.....

```
t1 = ([3, 6], 8)
t2 = copy.deepcopy(t1)
print(t2)    # ([3, 6], 8)

print('id(t1):%s' % id(t1))
print('id(t2):%s' % id(t2))

print('id(t1[0]):%s' % id(t1[0]))
print('id(t2[0]):%s' % id(t2[0]))

print('id(t1[1]):%s' % id(t1[1]))
print('id(t2[1]):%s' % id(t2[1]))
```