

图解

Python

可能是迄今为止

最易懂的Python视频教程



```
"""静态方法"""
```

```
.....
```

类对象的静态方法只是一个普通函数。把某个普通函数归属于类对象，可能只是为了易于代码管理。

在类对象中定义静态方法时，必须使用装饰器`@staticmethod`进行装饰。静态方法只是一个普通函数，因此，第一个形参没有特殊含义和要求。

静态方法可以被类对象所调用，语法格式为：类对象.方法名([实参]) 或：cls.方法名([实参])。

静态方法也可以被实例对象所调用，语法格式为：实例对象.方法名([实参]) 或：self.方法名([实参])。

类对象的所有实例对象都有一个指向类对象的指针，所以，类对象的所有实例对象都可以调用类对象中定义的类方法。

调用静态方法时的参数传递与调用普通函数是一样的。

```
.....
```

```
class MyClass(object):
    # 在类对象中定义静态方法
    @staticmethod
    def sm(p1, p2):
        print(p1, p2)

    @classmethod
    def cm(cls):
        # 通过类对象调用静态方法
        MyClass.sm(1, 2)
        cls.sm(1, 2)

    def im(self):
        # 通过实例对象调用静态方法
        self.sm(1, 2)
```

通过类对象调用静态方法

MyClass.sm(1, 2) # 1 2

mc = MyClass()

通过实例对象调用静态方法

mc.sm(1, 2) # 1 2

MyClass.cm()

mc.im()