Python语言基础与应用

高级特性 / 面向对象：类的定义与调用

陈斌 北京大学 gischen@pku.edu.cn

面向对象：类的定义与调用

- 什么是类
- 定义类
- 调用类

什么是类

- 类(class)是对象的模版，封装了对应现实实体的性质和行为
- 实例对象(Instance Objects)是类的具体化
- 把类比作模具，对象则是用模具制造出来的零件

什么是类

- **类的出现，为面向对象编程的三个最重要的特性提供了实现的手段**

  封装性、继承性、多态性

- **和函数相似，类是一系列代码的封装**

  Python中约定，类名用大写字母开头，函数用小写字母开头，以便区分

定义类

class语句

```
class <类名>:
    <一系列方法的调用>
```

类的初始化

```
class <类名>:
    def __init__(self, <参数表>):
    def <方法名>(self, <参数表>):
```

- __init__()是一个特殊的函数名，用于根据类的定义创建实例对象，第一个参数必须为self

调用类

- **<类名>(<参数>)**

  调用类会创建一个对象，（注意括号！）

  ```
  obj = <类名>(<参数表>)
  ```

  返回一个对象实例

  类方法中的self指这个对象实例！

- **使用点(.)操作符来调用对象里的方法**

  ```
  t = turtle.Pen()
  t.forward(100)
  t.left(90)
  …
  ```

类的定义与调用

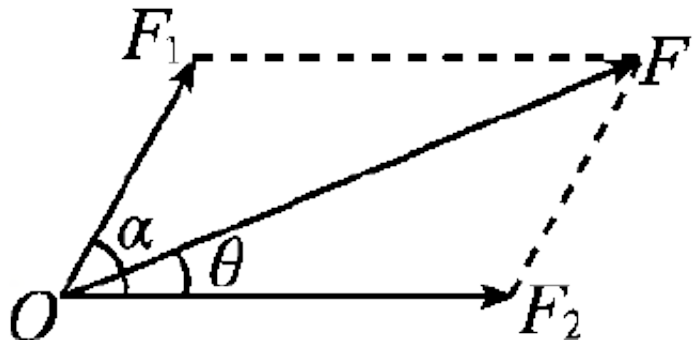
```python
class Force:  # 力
    def __init__(self, x, y):  # x,y方向分量
        self.fx, self.fy = x, y

    def show(self):  # 打印出力的值
        print("Force<%s,%s>" % (self.fx, self.fy))

    def add(self, force2):  # 与另一个力合成
        x = self.fx + force2.fx
        y = self.fy + force2.fy
        return Force(x, y)


# 生成一个力对象
f1 = Force(0, 1)
f1.show()

# 生成另一个力对象
f2 = Force(3, 4)
# 合成为新的力
f3 = f1.add(f2)
f3.show()
```




```
Force<0,1>
Force<3,5>
```