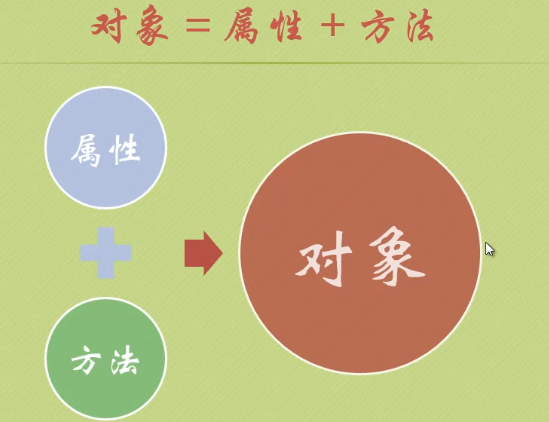
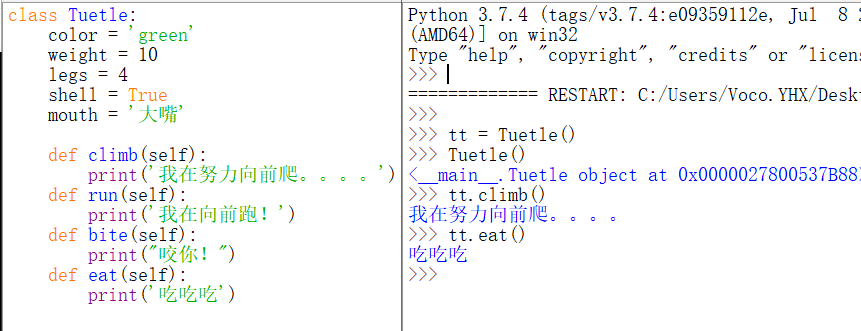
1. 对象



静态为属性 动态是方法

1. 类

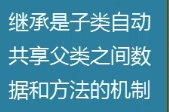
大写字母开头

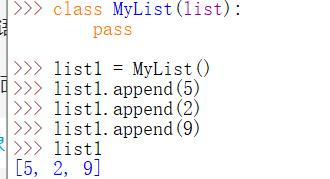


1. 面向对象



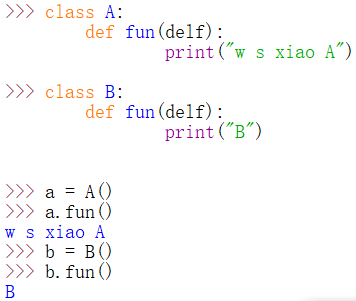
封装：只给需要的方法 调用实现 不需要哦知道该方法如何实现

继承：

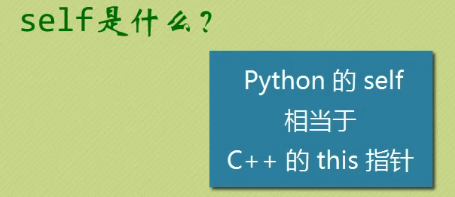


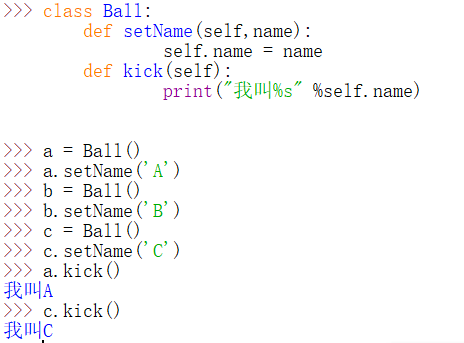
MyList 继承list list1 为MyList的对象 有list的属性和方法

多态：不同对象对同一方法不同的响应

 delf 为 self

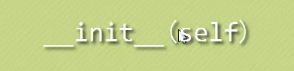
1. Self



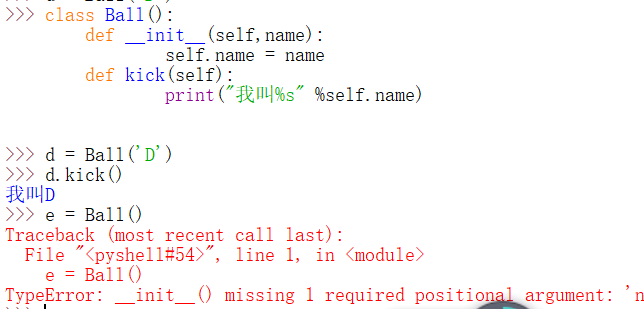


在py中 把 self 默认写为第一个参数

1. 构造方法 \_\_init\_\_(self,pram1,pram2....)

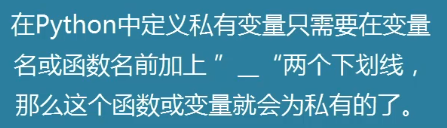


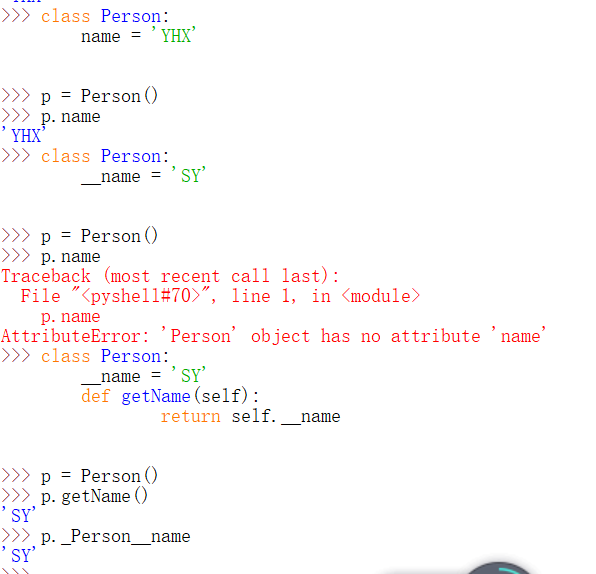
在初始化对象时可进行重写



例子中 对 \_\_init\_\_(self)进行了重写 加了一个name参数 这是创建对象时必须在括号里给定一个参数 否则会报错 若只有self一个参数则可以写也可以为空

1. 公有和私有



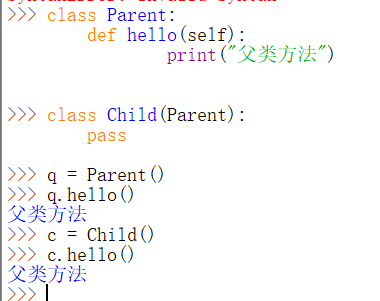


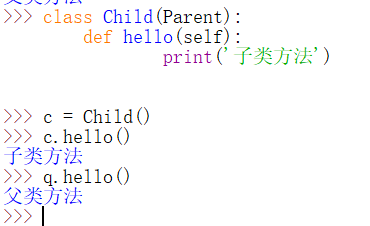
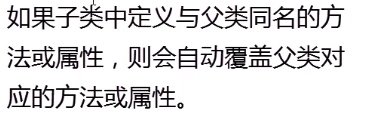
变量为私有时 可通过使用内部方法getName()进行读取

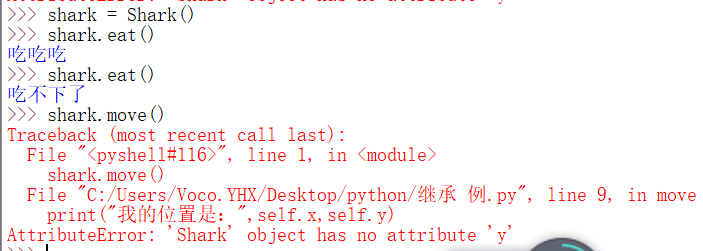
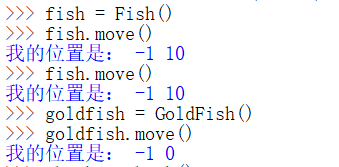
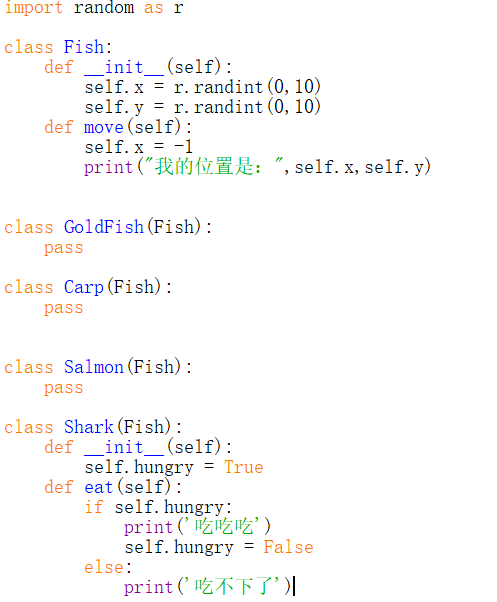
或使用 对象名.\_类名\_\_私有变量名 例：p.\_Person\_\_name

1. 继承





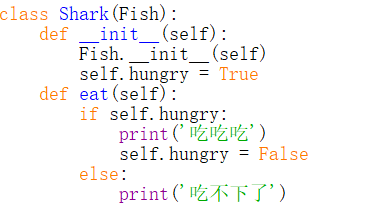




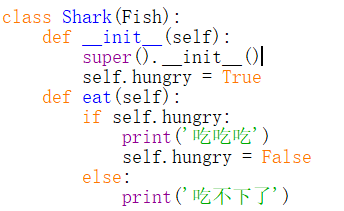
因为在Shark类中 重写了 \_\_init\_\_(self) 函数 其中不包含坐标位置 所以 shark类中不能使用move函数

此时解决方法有两种

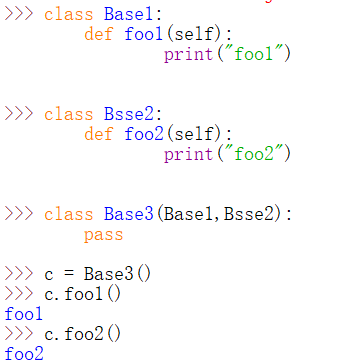
1. 调用父类中未绑定的方法 未绑定： 给调用的父类函数中传入的是子类的参数 不是父类参数

即

1. 使用super（）方法

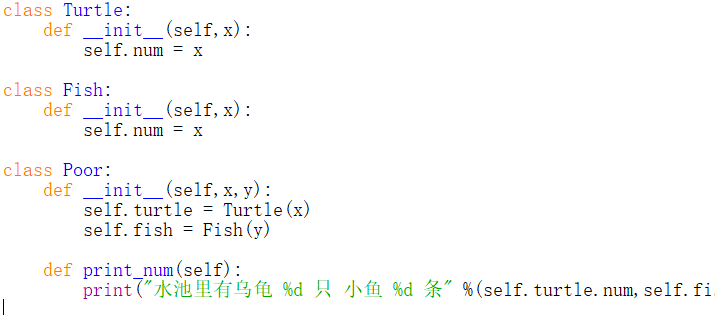


1. 多重继承

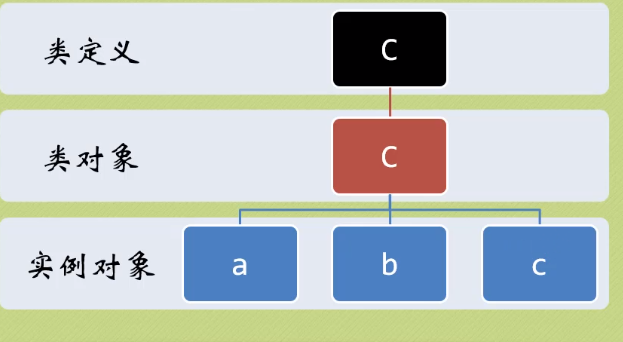


1. 组合

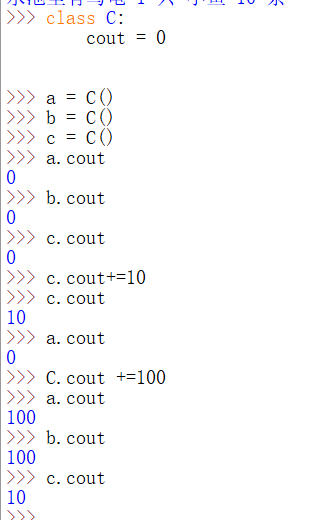
组合 是将类的实例化放到一个新类里面 将几个没有继承关系的类放在一起



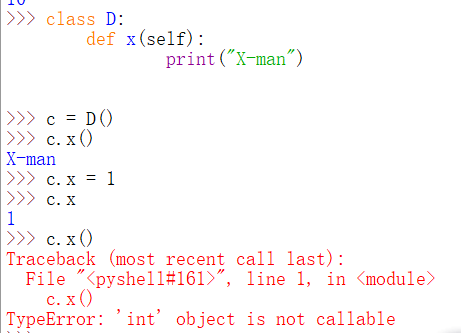
1. 类，类对象，实例对象

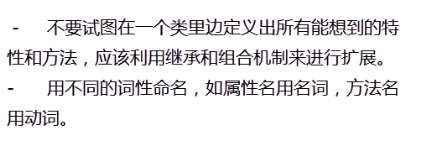


实例对象对同意属性进行修改时 不会影响类对象的值

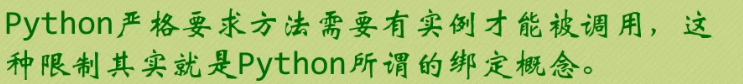


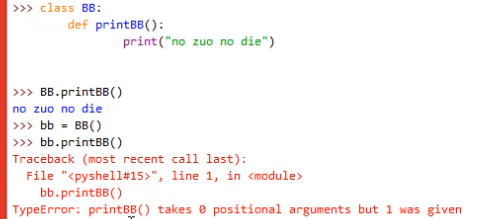
1. 属性名和方法名如果相同名字 属性会覆盖方法

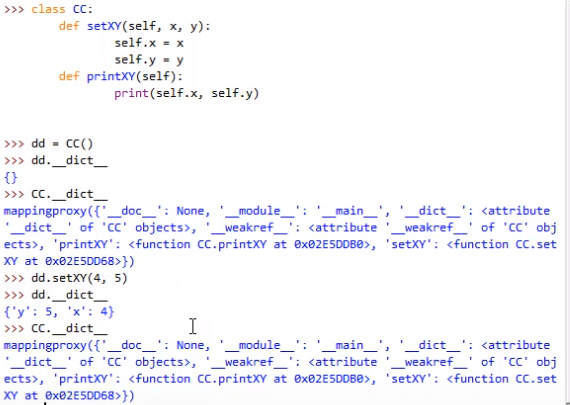




1. 绑定



实例化后需要用self传值 实际会用bb.printBB(bb)来调用

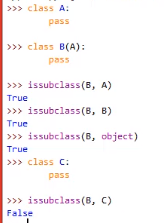


SetXY后 4，5 是独属于dd这个实例对象的 类对象不会显示 删除类对象后 不能新实例化新的实例对象 但是之前dd还是可以继续printXY的

1. 一些相关的内置函数



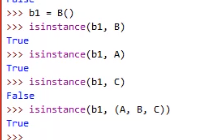
第一个参数是第二个参数的子类则返回True 两个参数实际上都是类名 一个类可以被认为是自身的子类 classinfo可以是类的对象组成的元组，只要class是其中任何一个候选类的子类，则返回True

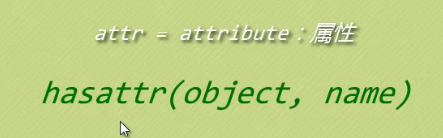




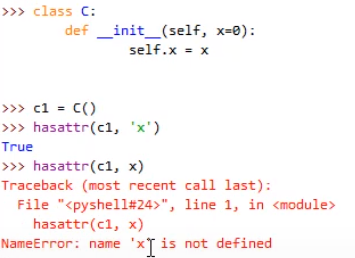
检查一个实例对象是否属于一个类 object是实例对象 class info 是类也可以是元组 一个个匹配检查

如果第一个参数不是对象 则永远返回False 如果d第二个参数不是有类或类的原则则抛出TypeError异常



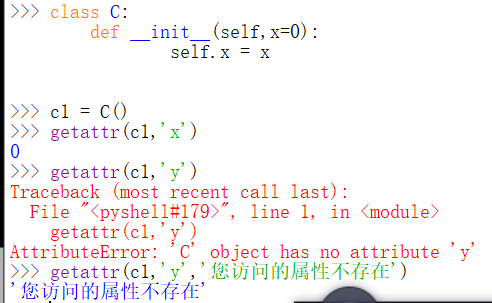


测试一个对象里面是否有指定属性 object ：对象 name ：属性 且属性必须被‘’包围起来



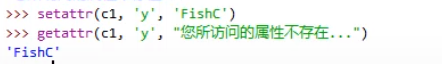


返回对象指定的属性值 若指定属性不存在 若设置了default 则会将default打印出来 否则抛出异常

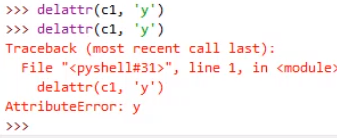


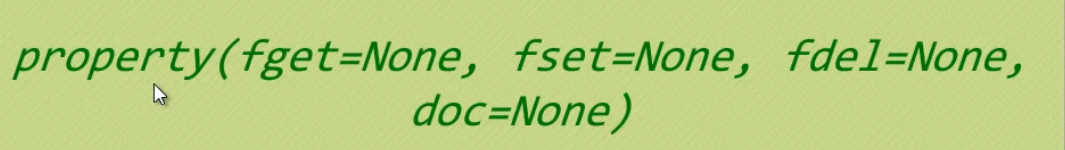


为属性设置属性值





删除指定属性 



用一个属性去设置属性 第一个参数是获取属性的方法 第二个参数 是设置属性的方法 第三个是 删除属性的方法

例 x = property（get,set,del） 用x去设置 后面方法里的属性

