*#面向对象 属性+方法 属性：可以作用于任何一个函数的参数(一个类里面) 方法(函数)  
#将函数进行分类和封装，使开发更高效  
#函数：具有某种功能可重复使用的代码块  
#1. 定义一个类 class 类名: def 函数1 def 函数2  
#2. 调用 对象 asd() 类名()赋值给对象  
#类的实例(实际的例子) 方便在类的外部调用函数  
#类的实例(实际的例子) 方便在类的内部调用函数  
#self=小艾  
#self 类的实例 在类里面不是参数  
#3. 继承 新的类名(旧的类名) 一个新的类可以继承旧的类中的所有函数  
#嵌套继承  
# 多继承 一个新类可以同时继承多个类  
# 如果继承的类里面有两个函数名一样时，继承第1个类里面的(就近原则)  
#4.多态 方法重写 把原有类里的函数功能重写  
#5.私有方法 不能被继承的方法也叫函数 只能在自己的类的内部使用  
# 函数名左边加上两个下划线 只能在自己的类里面使用 不能被继承  
#6.类的专有方法 也叫专有函数  
#运行不需要手动调用  
# class asd():  
# def 打电话(self):  
# print('ok')  
# def \_\_发短信(self):  
# print('想的美')  
# def \_\_聊天(self):  
# print('你好')***class new\_asd:  
 def \_\_init\_\_**(self,*x* **= '小红'**)**:** *#初始化属性* self.name **=** *x* **def 对话**(self)**:** print(**'hello {}'**.format(self.name))  
 **def 聊天**(self)**:** print(**'{}今年19岁'**.format(self.name))  
a **=** new\_asd(**'ace'**)  
b **=** new\_asd(**'110'**)  
a.对话()  
b.对话()