

# PYTHON

## 编程基础

# 子类的定义

# 子类的定义

定义子类时需要指定父类，其语法格式为：

```
class 子类名(父类名1, 父类名2, ..., 父类名M):
```

```
    语句1
```

```
    语句2
```

```
    ...
```

```
    语句N
```

- 当M等于1时，则为单继承；当M大于1时，则为多重继承。
- 例：继承示例。

# 子类的定义



```
1 class Person: #定义Person类
2     def SetName(self, name): #定义SetName方法
3         self.name=name #将self对应对象的name属性赋为形参name的值
4 class Student(Person): #以Person类作为父类定义子类Student
5     def SetSno(self, sno): #定义SetSno方法
6         self.sno=sno #将self对应对象的sno属性赋为形参sno的值
```

# 子类的定义



html

```
7      class Teacher(Person): #以Person类作为父类定义子类Teacher
8          def SetTno(self, tno): #定义SetTno方法
9              self.tno=tno #将self对应对象的tno属性赋为形参tno的值
10     class TA(Student,Teacher): #以学生类和Teacher类作为父类
                                     #定义子类TA
11         def SetTeacher(self, teacher): #定义SetTeacher方法
12             self.teacher=teacher #将self对象的teacher属性赋为形参
                                     #teacher的值
```

## 子类的定义

```
13     if __name__ == '__main__':
14         stu=Student() #定义Student类对象stu
15         stu.SetSno('1810100') #调用Student类中定义的SetSno方法
16         stu.SetName('李晓明') #调用Student类从Person类继承过来的SetName
                                   #方法
17         print('学号： %s， 姓名： %s'%(stu.sno,stu.name)) #输出学号和姓名
18         t=Teacher() #定义Teacher类对象t
19         t.SetTno('998012') #调用Teacher类中定义的SetTno方法
20         t.SetName('马红') #调用Teacher类从Person类继承过来的SetName方法
21         print('教工号： %s， 姓名： %s'%(t.tno,t.name)) #输出教工号和姓名
```

# 子类的定义

学号：1810100，姓名：李晓明

教工号：998012，姓名：马红