

PYTHON

编程基础

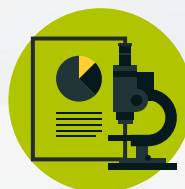
super方法

super方法

super方法用于获取父类的代理对象，以执行已在子类中被重写的父类方法，其语法格式为：

```
super([类名[, 对象名或类名]])
```

super方法



提示：

super方法有两个参数：

第一个参数是要获取父类代理对象的类名。

第二个参数如果传入对象名，则该对象所属的类必须是第一个参数指定的类或该类的子类，找到的父类对象的self会绑定到这个对象上；如果传入类名，则该类必须是第一个参数指定的类的子类。

在一个类A的定义中调用super方法时，可以将两个参数都省略，此时，`super()`等价于`super(A, self)`，即获取A的父类代理对象，且获取到的父类代理对象中的self绑定到当前A类对象的self上。

super方法



例：super方法使用示例。

```
1 class Person: #定义Person类
2     def __init__(self, name): #定义构造方法
3         print('Person类构造方法被调用！')
4         self.name=name #将self对象的name属性赋为形参name的值
5 class Student(Person): #以Person类作为父类定义子类Student
6     def __init__(self, sno, name): #定义构造方法
7         print('Student类构造方法被调用！')
8         super().__init__(name) #调用父类的构造方法
9         self.sno=sno #将self对象的sno属性赋为形参sno的值
```

super方法



例：super方法使用示例。

```
10 class Postgraduate(Student): #以Student类作为父类定义子类Postgraduate
11     def __init__(self, sno, name, tutor): #定义构造方法
12         print('Postgraduate类构造方法被调用！')
13         super().__init__(sno, name) #调用父类的构造方法
14         self.tutor=tutor #将self对象的tutor属性赋为形参tutor的值
15 if __name__=='__main__':
16     pg=Postgraduate('1810100','李晓明','马红') #创建Postgraduate类对象pg
17     print('学号： %s , 姓名： %s , 导师： %s'%(pg.sno,pg.name,pg.tutor))
```

super方法

Postgraduate类构造方法被调用！

Student类构造方法被调用！

Person类构造方法被调用！

学号：1810100，姓名：李晓明，导师：马红

super方法

提示

将 “`super().__init__(sno, name)`” 改为
“`super(Postgraduate,self).__init__(sno, name)`” ,
程序运行结果完全相同。