* [面向对象编程](http://www.liaoxuefeng.com/wiki/001374738125095c955c1e6d8bb493182103fac9270762a000/0013868200385680e8cf932dba9433ea367de9aba2b4784000) 学习内容

1. 在Python当中。类当中的变量是可以在创建之后添加的。

1.1 创建Student类

classs Student(object):

pass

1.2 创建studnet类的对象

s=Student()

1.3 给Student()对象添加成员变量

s.name='sunpeng'

2. 在Python当中，也可以通过在特殊方法\_\_init\_\_当中，初始化必须绑定的属性。

class Student(object):

def \_\_init\_\_(self,name,score):

self.name = name

self.score=score

3. 在Python当中，如果想让实例的内部属性不被外部访问，可以在实例变量名称前边加上两个\_\_

class Student(object):

def \_\_init(self,name,score):

self.\_\_name=name

self.\_\_score=score

之后，定义s = Student('sunpeng',100)，之后无法通过s.\_\_name='abc'来修改实例变量s的内容。

4. 在Python当中，变量名类似\_\_XXX\_\_的，是特殊变量，特殊变量是可以直接访问的，但不是私有变量。

5. 在Python当中，\_开头的变量不是私有变量，但是尽量不要去访问这样的私有变量。

6. 在Python当中，可以用isinstance来判断某个变量是不是某个特定的类型。

7. 在Python当中，可以通过type来访问某个变量的类型。type(a).或者导入包types,使用import types

8. 使用dir，就可以得到访问某个实例变量的所有的方法的内容。

9. 在Python当中，在类当中定义\_\_len\_\_之后，就可以调用len()来求对象的长度了。调用hasattr(),setattr(),getattr()等函数，可以动态访问某个类实例变量的值。

10. 在ubantu下安装python easy\_install命令为sudo apt-get install python-setuptools

面向对象高级编程学习笔记

1. 在Python当中可以动态给类添加方法。

存在类Student，其定义方式为：

class Student(object):

pass

定义一个方法set\_age，其定义方式为

def set\_age(self,age):

self.age = age

从类型库types当中导入MethodType

from types import MethodType

s=Student()

s.set\_age=MethodType(set\_age,s,Student)

s.set\_age(25)

其中，增加的方法set\_age只会对set\_age有效，对其他Studnet实例是无效的。如果想对所有的Student实例有效，则需要直接在类当中添加。

2. 如果想限制类当中的属性。也就是不允许在运行时添加，则在类当中\_\_slots\_\_槽。如果只允许存在name/age属性，则定义方式为\_\_slots\_\_=['name','age']

3. \_\_shots\_\_功能属性，对子类是无效的。也就是说在定义\_\_slots\_\_类的子类当中还是可以定义属性的。如果想子类的属性也是特定的，则需要定义新的\_\_slots\_\_

4. @property属性，用来添加setter和getter。如在类Student当中定义age属性

class Student(object):

@property

def age(self):

self.\_age=age

@age.setter

def age(self,age):

self.\_age=age

如果只定义@property，如果没有@age.setter，则表明age是只读属性，只允许获取，不允许设置。

5. Python类当中的特殊变量

|  |  |
| --- | --- |
| 特殊属性 | 功能 |
| \_\_slots\_\_ | 类实例只允许的属性列表 |
| \_\_str\_\_ | 类对象的可视化表示。在print函数被调用的时候调用 |
| \_\_repr\_\_ | 在直接输入变量名的时候被调用。 |
| \_\_iter\_\_与next() | 在类当中添加这两个方法之后，就可以按照for s in XXX的方式来使用了。 |
| \_\_getitem\_\_ | 使得类可以像数组一样 |
| \_\_init\_\_ | 类构造函数 |
| \_\_getattr\_\_ | 当访问类的属性的时候，在没有找到属性的情况下被调用。例如s.name在s当中没有定义name，的时候调用 |