

PCS111 类

概述

在 Python 中可以通过关键字 `class` 定义一个类，后面加上类名称，在类中可以定义类成员和类方法。自定义的类也可以继承现有的类，Python 中类的继承也有单继承和多继承。

应用

类定义

自定义的类是基于以下形式的：

```
In [1]: class A(object):
...:     """This is Class A"""
...:     def __init__(self, a):
...:         self.a = a
...:
```

```
In [2]: class B(object):
...:     """This is Class B"""
...:     def __init__(self, b):
...:         self.b = b
```

```
In [3]: print B.__dict__
```

```
{'__dict__': <attribute '__dict__' of 'B' objects>, '__module__': '__main__',
 '__weakref__': <attribute '__weakref__' of 'B' objects>, '__doc__': 'This
 is Class B', '__init__': <function __init__ at 0x842ed14>}
```

在这个小例子中，关键字 `class` 定义了两个类，`A` 和 `B` 分别是类名称，他们都继承了 `object` 类（这个是基类，Python 中所有的 `class` 都是从 `object` 继承而来的，也可以不写）。和函数一样，`class` 也可以定义文档字符串，同样是通过 `__doc__` 访问。和大多数编程语言不同的是，Python 中构造函数叫做 `__init__`，第一个参数传递的就是将要初始化的实例对象本身，类似许多语言中的 `this`。同样，Python 也有类似的析构函数叫做 `__del__`。

构造函数中的 `self.a = a` 便给 `self` 这个对象增加了一个名为 `a` 的属性，其绑定的对象就是传入的参数 `a` 所绑定的对象。对象的 `__dict__` 属性可以查看对象的所有属性。在任何时候都可以给对象增加属性，只要给不存在的属性绑定对象即可，Python 会自动创建不存在的属性。

类也可以有自己的属性。

```
In [4]: class A(object):
...:     """This is Class A"""
...:     aa = 'sth'
...:     def __init__(self, a):
...:         self.a = a
...:

In [5]: A.aa
Out[5]: 'sth'
```

同样也可以通过类名称、类属性名的形式访问类属性。

类中的方法和普通的函数定义是类似的。

```
In [7]: class A(object):
...:     """This is Class A"""
...:     aa = 'sth'
...:     def __init__(self, a):
...:         self.a = a
...:     def do_sth(self):
...:         pass
```

类继承

Python 中允许单继承和有限的多继承。

```
In [13]: class C(A, B):
...:     def __init__(self, a, b, c):
...:         A.__init__(self, a)
```

```
.....:      B.__init__(self, b)
.....:      self.c = c
.....:

In [18]: obj c = C(1, 2, 3)

In [19]: obj c.__dict__
Out[19]: {'a': 1, 'b': 2, 'c': 3}
```

如果所需要的一个属性在类 C 中没有找到，则从左到右依次按 A, B 的顺序在基类中寻找。通过__dict__可以看到对象的所有属性。

小结

类的定义和使用在面向对象编程中是十分重要的内容，应该熟练掌握。有关类和面向对象的进一步资料如下所示。

- 类与实例：<http://wiki.woodpecker.org.cn/moin/ObpLovelyPython/LpyQLearn-5-object>
精巧地址：<http://bit.ly/36elwi>
- 对象和面向对象：
http://www.woodpecker.org.cn/diveintopython/object_oriented_framework/index.html#fileinfo.divein
精巧地址：<http://bit.ly/1W0EJk>
- A First Look at Classes：<http://www.python.org/doc/2.6/tutorial/classes.html>
精巧地址：<http://bit.ly/3VhOJ3>