

# Vamei

编程，数学，设计

博客园 首页 订阅 管理

随笔-209 文章-1 评论-3802

## Python基础08 面向对象的基本概念

作者: Vamei 出处: <http://www.cnblogs.com/vamei> 欢迎转载，也请保留这段声明。谢谢！

谢谢逆水寒龙, topmad和Liqing纠错

Python使用类(class)和对象(object)，进行**面向对象** (object-oriented programming, 简称OOP) 的编程。

面向对象的最主要目的是提高程序的**重复使用性**。我们这么早切入面向对象编程的原因是，Python的整个概念是基于对象的。了解OOP是进一步学习Python的关键。

下面是对面向对象的一种理解，基于分类。

### 相近对象，归为类

在人类认知中，会根据**属性相近**把东西**归类**，并且给类别命名。比如说，鸟类的共同属性是有羽毛，通过产卵生育后代。任何一只特别的鸟都在鸟类的**原型**基础上的。

面向对象就是模拟了以上人类认知过程。在Python语言，为了听起来酷，我们把上面说的“东西”称为对象(object)。

先定义鸟类

```
class Bird(object):  
    have_feather = True  
    way_of_reproduction = 'egg'
```

我们定义了一个类别(class)，就是鸟(Bird)。在隶属于这个类比的语句块中，我们定义了两个变量，一个是有羽毛(have\_feather)，一个是生殖方式

(way\_of\_reproduction), 这两个变量对应我们刚才说的属性(attribute)。我们暂时先不说明括号以及其中的内容, 记为**问题1**。

假设我养了一只小鸡, 叫summer。它是个对象, 且属于鸟类。使用前面定义的类:

```
summer = Bird()
print summer.way_of_reproduction
```


通过第一句创建对象, 并说明summer是类别鸟中的一个对象, summer就有了鸟的类属性, 对属性的引用是通过 **对象.属性(object.attribute)** 的形式实现的。

可怜的summer, 你就是个有毛产的蛋货, 好不精致。

## 动作

日常认知中, 我们在通过属性识别类别的时候, 有时根据这个东西**能做什么事情**来区分类别。比如说, 鸟会移动。这样, 鸟就和房屋的类别区分开了。这些动作会带来一定的结果, 比如移动导致位置的变化。

这样的一些“行为”属性为**方法(method)**。Python中通过在类的内部定义函数, 来说明方法。



```
class Bird(object):
    have_feather = True
    way_of_reproduction = 'egg'
    def move(self, dx, dy):
        position = [0,0]
        position[0] = position[0] + dx
        position[1] = position[1] + dy
        return position

summer = Bird()
print 'after move:', summer.move(5, 8)
```



我们重新定义了鸟这个类别。鸟新增一个方法属性, 就是表示移动的方法move。(我

承认这个方法很傻，你可以在看过下一讲之后定义个有趣些的方法)

(它的参数中有一个`self`，它是为了方便我们引用对象自身。**方法的第一个参数必须是`self`，无论是否用到**。有关`self`的内容会在下一讲展开)

另外两个参数，`dx`，`dy`表示在`x`、`y`两个方向移动的距离。`move`方法会最终返回运算过的`position`。

在最后调用`move`方法的时候，我们只传递了`dx`和`dy`两个参数，不需要传递`self`参数(因为`self`只是为了内部使用)。

我的`summer`可以跑了。

## 子类

---

类别本身还可以进一步细分成子类

比如说，鸟类可以进一步分成鸡，大雁，黄鹂。

在OOP中，我们通过**继承(inheritance)**来表达上述概念。



```
class Chicken(Bird):
    way_of_move = 'walk'
    possible_in_KFC = True

class Oriole(Bird):
    way_of_move = 'fly'
    possible_in_KFC = False

summer = Chicken()
print summer.have_feather
print summer.move(5,8)
```



新定义的鸡(`Chicken`)类的，增加了两个属性：移动方式(`way_of_move`)，可

能在KFC找到 (`possible_in_KFC`)

在类定义时，括号里为了Bird。这说明，Chicken是属于鸟类 (Bird) 的一个子类，即Chicken继承自Bird。自然而然，Bird就是Chicken的父类。Chicken将享有Bird的所有属性。尽管我只声明了summer是鸡类，它通过继承享有了父类的属性（无论是变量属性`have_feather`还是方法属性`move`）

新定义的黄鹂 (Oriole) 类，同样继承自鸟类。在创建一个黄鹂对象时，该对象自动拥有鸟类的属性。

通过继承制度，我们可以减少程序中的重复信息和重复语句。如果我们分别定义两个类，而不继承自鸟类，就必须把鸟类的属性分别输入到鸡类和黄鹂类的定义中。整个过程会变得繁琐，因此，面向对象提高了程序的可重复使用性。

（回到问题1，括号中的object，当括号中为object时，说明这个类没有父类（到头了））

将各种各样的东西分类，从而了解世界，从人类祖先开始，我们就在练习了这个认知过程，面向对象是符合人类思维习惯的。所谓面向过程，也就是执行完一个语句再执行下一个，更多的是机器思维。通过面向对象的编程，我们可以更方便的表达思维中的复杂想法。

## 总结

---

将东西根据属性归类 （将object归为class）

方法是一种属性，表示动作

用继承来说明父类-子类关系。子类自动具有父类的所有属性。

`self`代表了根据类定义而创建的对象。

建立一个对象： 对象名 = 类名()

引用对象的属性： `object.attribute`

标签: Python

好文要顶

关注我

收藏该文



Vamei

关注 - 26

粉丝 - 4985

荣誉: 推荐博客

+加关注

40

0

(请您对文章做出评价)

« 上一篇: Python基础07 函数

» 下一篇: Python基础09 面向对象的进一步拓展

posted @ 2012-06-02 07:51 Vamei 阅读(59177) 评论(49) 编辑 收藏

评论列表

#1楼 2012-09-26 22:34 sunrisevv

博主你括号里的内容好可爱!

支持(0) 反对(0)

#2楼 2012-11-12 21:28 左耳醉梦

讲解得很生动

支持(0) 反对(0)

#3楼[楼主] 2012-11-12 21:52 Vamei

@ 左耳醉梦

谢谢

支持(0) 反对(0)

#4楼 2012-11-18 16:42 逆水寒龙

way\_of\_move = 'walk'

支持(0) 反对(0)

#5楼[楼主] 2012-11-18 17:24 Vamei

@ 逆水寒龙

好像当时用成中文输入法了, 谢谢!

支持(0) 反对(0)

## #6楼 2012-11-18 23:34 nowherekai

从开头看到这篇，适合有其他语言基础的入门，写的不错，比较简洁有重点。

支持(0) 反对(0)

## #7楼 2013-02-23 00:15 红烧狮子头

十分简洁明了推荐啊

支持(0) 反对(0)

## #8楼 2013-03-27 13:46 hawkuu

楼主你写的太好了

支持(0) 反对(0)

## #9楼[楼主] 2013-03-27 19:56 Vamei

@ hawkuu

呵呵，谢谢！

支持(0) 反对(0)

## #10楼 2013-04-11 10:46 海 子

引用

方法的第一个参数必须是self，无论是否用到

这个地方有点问题的，参数不一定必须是self，可以指定为任何名字的参数，取为sel、this、it等等都可以，python中原意是为：类中的方法第一个参数必须是调用该方法的对象本身，由于大家都习惯用self，所以很多教材都说必须用self，这种说法是不准确的。

支持(4) 反对(0)

## #11楼[楼主] 2013-04-11 11:35 Vamei

@ 海 子

嗯，你说的对，它的名字不一定是self，只是惯例上使用self. 但使用self更容易让别人理解。

我一直使用self指代该参数，也想让读到的人跟随这个惯例。

支持(0) 反对(0)

## #12楼 2013-05-23 19:25 xiujie\_jiang

透彻，写的让我们这些纯白，都能理解！看了好几篇博主的文章，感谢分享

支持(0) 反对(0)

#13楼[楼主] 2013-05-25 09:07 Vamei

@ xiujie\_jiang

谢谢

支持(0) 反对(0)

#14楼 2013-06-18 11:19 Eve.月

楼主讲解的方式很好啊～去繁取简，先从最基础的东西来说～

支持(0) 反对(0)

#15楼 2013-07-09 11:51 boke\_hh

楼主，请问创建一个类时，括号里面有object和没有object的区别是什么？

支持(0) 反对(0)

#16楼 2013-07-11 17:14 石@头

```
class Chicken(Bird):  
    way_of_move = 'walk'  
    possible_in_KFC = True
```

```
class Oriole(Bird):  
    way_of_move = 'fly'  
    possible_in_KFC = False
```

```
summer = Chicken()  
print summer.have_feather  
print summer.move(5,8)
```

刚接触，楼主请教执行后为什么报Bird未定义？

支持(0) 反对(0)

#17楼 2013-07-29 20:18 topmad

```
1  def judg(year):  
2      if year%4==0:  
3          print(year,'is a leap year')  
4      else:  
5          print(year,'is a mono year')  
6      return  
7
```

```
8
9  for year in range(1900,2000):
10     print(judg(year));
```

tom 我在python3.3下面运行上面的代码 每一次执行完print 总要出来一个none行 不知道是哪里来的 请tom帮忙释疑 谢谢

支持(0) 反对(0)

---

#18楼 2013-07-31 16:25 LittleBird99

@ 石@头

因为你肯定是把Bird类的定义漏掉了，这是继承，没有父类怎么继承呢？  
加上：

```
1  class Bird(object):
2      have_feather = True
3      way_of_reproduction = 'egg'
4      def move(self, dx, dy):
5          position = [0,0]
6          position[0] = position[0] + dx
7          position[1] = position[1] + dy
8          return position
```

就行了

支持(0) 反对(0)

---

#19楼 2013-08-20 13:26 RobertTech

讲的很容易理解~3Q

支持(0) 反对(0)

---

#20楼 2013-10-28 21:53 keyword

@ topmad

去掉 for 循环里的 print() 。

支持(1) 反对(0)

---

#21楼 2014-02-03 13:40 kingboo501



楼主写得超级棒，我想请问下楼主，您学了python大概多久？还有学习路线大概是什么样的？鄙人本身有一些C语言的基础，觉得python挺好玩的，想了解下，不知道如何去入手.....能否请楼主提些建议，感激不尽

支持(0) 反对(0)

---

#22楼 2014-03-18 16:32 Jerry\_L

楼主讲的很生动，赞一个。nice！

支持(0) 反对(0)

---

#23楼 2014-04-20 22:49 cc1e

nice

支持(0) 反对(0)

---

#24楼 2014-04-22 09:53 red.hu

博主的文章可以让w3school收录了，你的讲解有点像w3school上面的php教程，通俗易懂，容易上手。

支持(0) 反对(0)

---

#25楼 2014-05-16 09:46 死侍

膜拜大神！！！！

支持(0) 反对(0)

---

#26楼 2014-06-11 00:55 ZJA+

```
class Duck(Bird):  
    have_feather=False
```

```
summer=Duck()  
print summer.have_feather
```

Python 跑出来的是False

也就是说，子类里面可以定义和父类一模一样的属性但赋予新的值，结果跟着子类的新定义走？

支持(0) 反对(0)

---

#27楼 2014-07-28 17:33 snapshot560

看到 Chicken 类后,笑坏了 哈哈

支持(0) 反对(0)

## #28楼 2014-08-05 17:54 biadq

之前草草看了一下简明简称，对类不是很理解，经过博主的讲解理解很多，非常感谢

支持(0) 反对(0)

## #29楼 2014-08-15 21:30 伤逝水

写得很好

支持(0) 反对(0)

## #30楼 2014-08-21 10:53 zjl-1110

哈哈哈.....很有简明python教程的感觉，不过讲得更细致易懂！！给你32个赞！

支持(0) 反对(0)

## #31楼 2014-08-21 12:40 zjl-1110

之前一直搞不懂为什么不能直接调用方法，汗原来还要定义成类方法，不然都是实例方法

```
1  #!/usr/bin/python
2  #-*-coding:utf-8-*-
3  class a():
4      a_1=0
5      a_2='aaaa'
6      #classmethod
7      @classmethod
8      def d(self,a):
9          print(a+"a1a2a3")
10
11     def c(self,a):
12         print(a+"11111")
13
14     @staticmethod
15     def b(a):
16         print(a+"2222")
17
18 x=a()
19 x.c('d')
20
21 a.d('c')
22
23 a.b('bb')
```

分别是定义和调用类函数，静态函数和实例函数

#32楼 2014-08-21 12:54 zjl-1110

```
1  #!/usr/bin/python
2  # -*- coding: utf-8 -*-
3  class a():
4      a_1=0
5      a_2='aaaa'
6      #classmethod
7      @classmethod
8      def d(self,a):
9          print(a+"a1a2a3")
10
11     def c(self,a):
12         print(a+"11111")
13
14     @staticmethod
15     def b(a):
16         print(a+"2222")
17
18 x=a()
19 x.c('d')
20
21 a.d('c')
22
23 a.b('bb')
24
25 class r(a):
26     r_1=111
27     r_2='rrrrr'
28
29 rr=r()
30 rr.c('r22222')
31
32 r.d('r11111')
33
34 r.b('r33333')
```

子类对父类函数的调用

#33楼 2014-08-21 13:20 zjl-1110

```
1  #!/usr/bin/python
2  #-*- coding:utf-8 -*-
3
4  #单继承
5  class A(object):
6      .....
7  class C(A):
8      .....
9
10 #多继承
11 class A(object):
12     .....
13 class B(object):
14     .....
15 class C(A,B):
16     .....
```

支持(0) 反对(0)

---

#34楼 2015-03-18 11:49 yayun

好，谢谢博主诙谐幽默的讲解。顶一个。

支持(0) 反对(0)

---

#35楼 2015-03-27 21:54 疯狂乱抓

回头再看一遍，简直不能再恍然大悟。好评。。赞赞赞

支持(0) 反对(0)

---

#36楼 2015-04-01 16:54 conleytan

```
class Bird(object):
    have_feather = True
    way_of_reproduction = 'egg'
    def move(self, dx, dy):
        position = [0,0]
        position[0] = position[0] + dx
        position[1] = position[1] + dy
        return position
```

```
summer = Bird()
print 'after move:',summer.move(5,8)
```

请问，为什么我在Windows平台上，把上面这段代码直接粘到我的IDLE后，显示

summer标红，说invalid syntax?

支持(0) 反对(0)

#37楼 2015-04-29 14:35 哈哈大笑

生动啊！楼主有教学一套！大哈哈

支持(0) 反对(0)

#38楼 2015-05-08 16:35 elivans

浅显易懂，写得真好

支持(0) 反对(0)

#39楼 2015-09-02 14:14 蚂蚁快跑

楼主讲解的生动易懂，很nice

支持(0) 反对(0)

#40楼 2015-10-30 11:27 Hear\_Yang

nice

支持(0) 反对(0)

#41楼 2015-11-26 23:55 irunner

@ Vamei

方法中的第一个参数 是不是可以不显式的写，默认就会是self

支持(0) 反对(0)

#42楼 2016-01-17 15:32 我的风儿

在动作上的一段

可怜的summer，你就是个有毛产的蛋货，好不精致。

应该

可怜的summer，你就是个有毛产蛋的货，好不精致。

原谅我的较真

支持(0) 反对(0)

#43楼 2016-01-20 21:54 Lz阿飞

possible\_in\_kfc

就为这句我得点个赞！

支持(0) 反对(0)

#44楼 2016-01-22 02:18 yeayee

哈哈，很好的例子

支持(0) 反对(0)

---

#45楼 2016-03-27 09:54 hitwh\_Gypsy

楼主再接再厉，  
继续写出好文章！

支持(0) 反对(0)

---

#46楼 2016-04-01 00:05 wqh2016

在加班的空堂看完此篇

支持(0) 反对(0)

---

#47楼 2016-04-16 00:28 冯大圆

```
1  class Bird(object):
2      have_feather=True
3      way_of_reproduction='egg'
4      position =[0,0]
5
6      def move(self,dx,dy):
7          position[0] = position[0] +dx
8          position[1] = position[1] +dy
9          return position
10
11 summer = Bird()
12
13 print 'after move:',summer.move(5,8)
14
15 print 'after move twice:',summer.move(1,1)
```

我试着将positon改成Attribute，但是报错

> NameError: global name 'position' is not defined

难道在class内，方法不能共享Attribute？

支持(0) 反对(0)

---

#48楼 2016-05-30 23:41 Tonyzhzh

生动形象 让人恍然大悟啊

支持(0) 反对(0)

---

#49楼 2016-05-31 16:51 coco413

很少评论，写的好真好！值得评论

支持(0) 反对(0)

刷新评论 刷新页面 返回顶部

注册用户登录后才能发表评论，请 [登录](#) 或 [注册](#)，[访问](#)网站首页。

【推荐】50万行VC++源码：大型组态工控、电力仿真CAD与GIS源码库

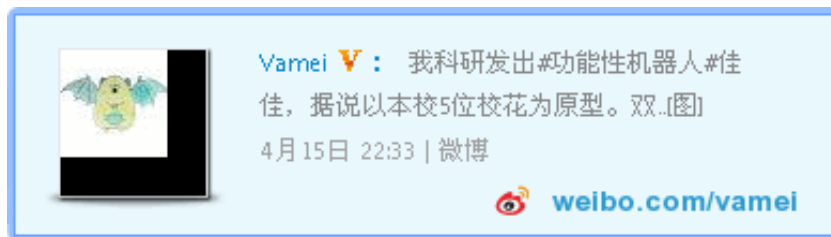
【推荐】融云即时通讯云—豆果美食、Faceu等亿级APP都在用



公告

你好，这里是Vamei，一名编程爱好者。我在博客里写了**Python/Linux/网络协议/算法/Java/数据科学**系列文章，从这里开始阅读。非常期待和你的交流。

我的微博



下列教程已经做成电子出版物，内容经过修订，也方便离线阅读：  
协议森林

欢迎阅读我写的其他书籍：

现代小城的考古学家

天气与历史的相爱相杀

## 随手拍光影

昵称: **Vamei**

园龄: **4年1个月**

荣誉: 推荐博客

粉丝: **4985**

关注: **26**

**+加关注**

常用链接

我的随笔

我的评论

我的参与

最新评论

我的标签

我的标签

**Python(61)**

**Java(42)**

**大数据(22)**

**Linux(17)**

**网络(16)**

**算法(15)**

**文青(14)**

**技普(9)**

**系列索引(6)**

**开发工具(4)**

**更多**

系列文章

**Java快速教程**

**Linux的概念与体系**

**Python快速教程**

**数据科学**

**协议森林**

**纸上谈兵：算法与数据结构**



积分与排名

积分 - 659668

排名 - 122

最新评论

### 1. Re:Java基础11 对象引用

受教!

--MissLost

### 2. Re:Python快速教程

看评论区一片喝彩！看来我得在此扎营了！

--测试小蚂蚁

### 3. Re:Python进阶06 循环对象

好好地列表解析变成了表推导

--ashic

### 4. Re:“不给力啊，老湿！”：RSA加密与破解

感谢楼主精彩分享

--worldball

### 5. Re:概率论04 随机变量

你写的这一系列太棒了，刚加入博客园就在你这里学到了，我要转载过去学习一下

--yixius

### 6. Re:Python基础03 序列

挺好的教程、、、

--王小拽的号

### 7. Re:Python进阶07 函数对象

```
def func(x,y): print x**ydef test(f,a,b): print 'test' print f(a,b)test (func,3,2)
```

输出的内容:tes.....

--M-edea

### 8. Re:Python进阶02 文本文件的输入输出

@coderXT换行符: \n...

--行者之印

### 9. Re:数据科学

博主啊，这里是一枚即将大二的计算机新人，大一学了python，java，还有一些算法，数据结构，图论了，感觉我对数学又一些反感，但是听说离散数学对计算机专业的很重要，不知道怎么去学比较好呢，我想像您写.....

--Acokil

## 10. Re:为什么要写技术博

楼主是用自己自定义的模板吗？在博客园里找不到这种风格的blog模板？

--行者之印

## 11. Re:来玩Play框架01 简介

挖煤哥,我补充了一下Windows下的搭建play框架,希望有点帮助,谢谢!

--Sungeek

## 12. Re:来玩Play框架07 静态文件

```
@helper.form(action = routes.Application.upload, 'enctype ->
    "multipart/form-data") {--action = rout.....
```

--quxiaozha

## 13. Re:来玩Play框架07 静态文件

该记录将/assets/下的URL，对应到项目的/public文件夹内的文件。比如在项目的/public/images/test.jpg，就可以通过/assests/images/test.jpg这一.....

--quxiaozha

## 14. Re:来玩Play框架06 用户验证

支持挖煤哥~~~

--quxiaozha

## 15. Re:“不给力啊，老湿！”：RSA加密与破解

@maanshancss请你仔细阅读了这个文章再来评价。...

--Vamei

推荐排行榜

1. “不给力啊，老湿！”：RSA加密与破解(218)
2. Python快速教程(140)
3. 野蛮生长又五年(91)
4. Java快速教程(88)
5. 协议森林01 邮差与邮局 (网络协议概观)(79)
6. 为什么要写技术博(71)
7. 编程异闻录(54)
8. 博客一年：心理之旅(49)
9. 协议森林08 不放弃 (TCP协议与流通信)(45)
10. Python快速教程 尾声(43)
11. 协议森林(42)
12. Java基础01 从HelloWorld到面向对象(42)
13. Python基础08 面向对象的基本概念(40)

14. 一天能学会的计算机技术(34)

15. 博客第二年，杂谈(33)

Copyright ©2016 Vamei

**05370248**