

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10 回过头来看看

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

从最初的“Hello World”，走到面向对象。该回过头来看看，教程中是否遗漏了什么。

我们之前提到一句话，“Everything is Object”。那么我们就深入体验一下这句话。

需要先要介绍两个内置函数，`dir()`和`help()`

`dir()`用来查询一个类或者对象所有属性。你可以尝试一下

```
>>>print dir(list)
```

`help()`用来查询的说明文档。你可以尝试一下

```
>>>print help(list)
```

(`list`是Python内置的一个类，对应于我们之前讲解过的列表)

list是一个类

在上面以及看到，表是Python已经定义好的一个类。当我们新建一个表时，比如：

```
>>>n1 = [1,2,5,3,5]
```

实际上，`n1`是类`list`的一个对象。

实验一些`list`的方法：

```
>>>print nl.count(5)          # 计数，看总共有多少个5

>>>print nl.index(3)          # 查询 nl 的第一个3的下标

>>>nl.append(6)                # 在 nl 的最后增添一个新元素6

>>>nl.sort()                   # 对nl的元素排序

>>>print nl.pop()              # 从nl中去除最后一个元素，并将该元素返回。

>>>nl.remove(2)                # 从nl中去除第一个2

>>>nl.insert(0,9)              # 在下标为0的位置插入9
```

总之，list是一个类。每个列表都属于该类。

Python补充中有list常用方法的附录。

运算符是特殊方法

使用dir(list)的时候，能看到一个属性，是__add__()。从形式上看是特殊方法（下划线，下划线）。它特殊在哪呢？

这个方法定义了"+"运算符对于list对象的意义，两个list的对象相加时，会进行的操作。

```
>>>print [1,2,3] + [5,6,9]
```

运算符，比如+，-，>，<，以及下标引用[start:end]等等，从根本上都是定义在类内部的方法。

尝试一下

```
>>>print [1,2,3] - [3,4]
```

会有错误信息，说明该运算符“-”没有定义。现在我们继承list类，添加对“-”的定义



```
class superList(list):
    def __sub__(self, b):
        a = self[:]      # 这里，self是superList的对象。由于
                           # superList继承于list，它可以利用和list[:]相同的引用方法来表示整个对象。
        b = b[:]
        while len(b) > 0:
            element_b = b.pop()
            if element_b in a:
                a.remove(element_b)
        return a

print superList([1,2,3]) - superList([3,4])
```



内置函数`len()`用来返回`list`所包含的元素总数。内置函数`__sub__()`定义了“-”的操作：从第一个表中去掉第二个表中出现的元素。如果`__sub__()`已经在父类中定义，你又在子类中定义了，那么子类的对象会参考子类的定义，而不会载入父类的定义。任何其他的属性也是这样。

（教程最后也会给出一个特殊方法的清单）

定义运算符对于复杂的对象非常有用。举例来说，人类有多个属性，比如姓名，年龄和身高。我们可以把人类的比较（>，<，=）定义成只看年龄。这样就可以根据自己的目的，将原本不存在的运算增加在对象上了。

下一步

希望你已经对Python有了一个基本了解。你可能跃跃欲试，要写一些程序练习一下。这会对你很有好处。

但是，Python的强大很大一部分原因在于，它提供有很多已经写好的，可以现成用的对象。我们已经看到了内置的比如说`list`，还有`tuple`等等。它们用起来很方便。在Python的标准库里，还有大量可以用于操作系统互动，Internet开发，多线程，文本处理的对象。而在所有的这些的基础上，又有很多外部的库包，定义了更丰富的对象，比如`numpy`，`tkinter`，`django`等用于科学计算，GUI开发，

web开发的库，定义了各种各样的对象。对于一般用户来说，使用这些库，要比自己去从头开始容易得多。我们要开始攀登巨人的肩膀了。

谢谢你的关注，

欢迎来到Python的世界。

总结

`len()` `dir()` `help()`

数据结构`list` (列表) 是一个类。

运算符是方法

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](#)

26

0

(请您对文章做出评价)

« 上一篇: [Python基础09 面向对象的进一步拓展](#)

» 下一篇: [Python进阶01 词典](#)

posted @ 2012-06-02 22:02 Vamei 阅读(42427) 评论(66) 编辑 收藏

< Prev

1

2

评论列表

#51楼 2014-06-06 00:24 人生苦短，gopython

新人，看不懂这个

`a = self[:]`

```
b = b[:]
```

支持(0) 反对(0)

#52楼 2014-07-08 10:52 990973848

调用superList([1,2,3]) - superList([3,4]) 这个的时候, 怎么知道使用这个__sub__这个方法呢。

支持(2) 反对(0)

#53楼 2014-09-02 09:30 刘文壮

@ 990973848

__sub__这个方法是个内置方法, 对应与减法运算, 是基类定义的, 在这里只不过是实现了这个接口

支持(2) 反对(1)

#54楼 2014-10-24 16:40 pocket

@ Vamei

我觉得执行print superList([1,2,3]) - superList([3,4])这一句的时候
先执行print superList([1,2,3]) 得到self=list[1,2,3]
遇到 "-"号时, 执行了superList([3,4])里的__sub__, 对吗?

如果执行的是superList([1,2,3])里的__sub__, 那是怎么将list[3,4]赋值给b的呢?
系统是怎么识别将L2传递给b, 为什么不将L1传递给B

支持(2) 反对(0)

#55楼 2015-01-17 11:57 Supper_litt

看完朋友的python, 很适合对这门语言作为了解。

支持(0) 反对(0)

#56楼 2015-02-28 11:30 年轻的水兵湾

```
while len(b)>0:
```

```
...
```

```
return a
```

可是b里面一直都有个4, 这个while执行到什么时候结束?
我是小白

支持(0) 反对(0)

#57楼 2015-02-28 11:46 年轻的水兵湾

哦 `pop()`是取出最后一个元素，如果`b[]`的4和3调一下，这个循环什么时候结束呢？

支持(0) 反对(0)

#58楼 2015-04-22 15:59 潮爷

@ 年轻的水兵湾

不管怎么调，`pop`取出最后一个元素以后就把那个元素从`b`里删了，所以`b`里有几个元素，这个循环就执行几次

支持(0) 反对(0)

#59楼 2015-05-29 15:57 小小西红柿

@ 郭若拙

引用

如果只是`return a`，那么 `a - b - c`会报错

同问，`a`应该是一个`superList()`的对象，`return a`的`type`也应是一个`superList()`的对象啊

```
1 print(type(a-b))
2 <class 'list'>
```

why?

支持(0) 反对(0)

#60楼 2015-05-30 20:19 迷途ing

```
1 class superList(list):
2     def __sub__(self,b):
3         #a = self[:]
4         #b = b[:]
5         #请看这里
6         print type(self)
7         #print type(a)
8         print type(b)
```

```

9         while len(b)>0:
10             element_b = b.pop()
11             if element_b in self:
12                 self.remove(element_b)
13
14             #print type(self)
15             return self
16 x = superList([1,2,3,4,5])
17 y = superList([2,3])
18 z = superList([1,4])
19 #print type(x)
20 print x - y -z
21 #print type(x-y-z)

```

a b type都是list 以上是楼主示例的问题

支持(0) 反对(0)

#61楼 2015-06-02 23:40 朗月繁星

@ 年轻的水兵湾

b.pop()是取出最大脚标处的值，并将b中最大脚标处的该值删除

支持(0) 反对(0)

#62楼 2015-09-30 23:00 蜗牛搬家x

superList([1,2,3]).__sub__([3,4]) 可以得到相同结果。

好像文中没解释 __sub__ 方法跟 减号"-" 是如何自动挂钩的

支持(2) 反对(0)

#63楼 2015-10-04 16:43 挖煤Boy

@ 人生苦短，gopython

等价于 [0:]

支持(0) 反对(0)

#64楼 2015-12-03 21:51 shuo烁

@ 迷途ing

这里可能就涉及到深拷贝和浅拷贝的问题了。

支持(0) 反对(0)

#65楼 2016-04-02 15:45 wqh2016

努力把博主的所有文章 都看完

[支持\(0\)](#) [反对\(0\)](#)

#66楼 2016-05-11 04:27 newpy

写的真好，楼主回复的也很耐心，一开始看不懂，后来看评论看懂了。这样互相交流补充的形式特别好。评论内容有利于帮我们理解。看了这篇终于把类和对象搞懂了。

[支持\(0\)](#) [反对\(0\)](#)[< Prev](#) [1](#) [2](#)[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



SpreadJS 表格控件

Excel界面数据处理
可视化、可定制

[了解详情](#)



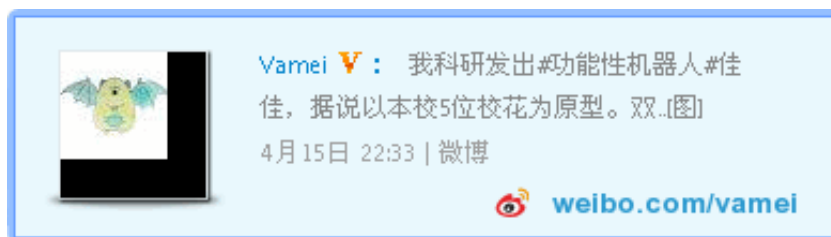
JPush 极光推送 消息推送领导品牌全面升级 JIGUANG | 极光


[详情点击](#)

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



Vamei ：我科研发出#功能性机器人#佳佳，据说以本校5位校花为原型。双...[图]

4月15日 22:33 | 微博

weibo.com/vamei

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

05370250