

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进阶03 模块

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

我们之前看到了函数和对象。从本质上来说，它们都是为了更好的组织已有的程序，以方便重复利用。

模块 (module) 也是为了同样的目的。在Python中，一个 **.py** 文件就构成一个模块。通过模块，你可以**调用其它文件中的程序**。

引入模块

我们先写一个first.py文件，内容如下：

```
def laugh():  
    print 'HaHaHaHa'
```

再写一个second.py，并引入first中的程序：

```
import first  
  
for i in range(10):  
    first.laugh()
```

在second.py中，我们使用了first.py中定义的laugh()函数。

引入模块后，可以通过**模块.对象**的方式来调用引入模块中的某个对象。上面例子中，first为引入的模块，laugh()是我们所引入的对象。

Python中还有其它的引入方式,

```
import a as b                # 引入模块a, 并将模块a重命名为b

from a import function1      # 从模块a中引入function1对象。调用a中对象时, 我们不用再说明模块, 即直接使用function1, 而不是a.function1。

from a import *              # 从模块a中引入所有对象。调用a中对象时, 我们不用再说明模块, 即直接使用对象, 而不是a.对象。
```

这些引用方式, 可以方便后面的程序书写。

搜索路径

Python会在以下路径中搜索它想要寻找的模块:

1. 程序所在的文件夹
2. 标准库的安装路径
3. 操作系统环境变量PYTHONPATH所包含的路径

如果你有自定义的模块, 或者下载的模块, 可以根据情况放在相应的路径, 以便Python可以找到。

模块包

可以将功能相似的模块放在同一个文件夹(比如说this_dir)中, 构成一个模块包。通过

```
import this_dir.module
```

引入this_dir文件夹中的module模块。

该文件夹中必须包含一个__init__.py的文件, 提醒Python, 该文件夹为一个模块包。__init__.py可以是一个空文件。

总结

import module

module.object

__init__.py

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](#)

27

0

(请您对文章做出评价)

« 上一篇: [Python进阶02 文本文件的输入输出](#)

» 下一篇: [Python进阶04 函数的参数对应](#)

posted @ 2012-07-03 11:14 Vamei 阅读(35825) 评论(31) 编辑 收藏

评论列表

#1楼 2013-11-29 16:27 woody.song

原来 __init__.py 是这个作用

支持(0) 反对(0)

#2楼 2013-12-07 17:33 quarryman

模块的搜索路径应为:

1. 程序所在的文件夹
2. 环境变量PYTHONPATH所包含的路径
3. 标准库的安装路径

参考:

<http://docs.python.org/2/tutorial/modules.html>

建议更正!

支持(4) 反对(0)

#3楼 2013-12-28 11:12 itfanr

```
import dir.module
```

这句话有点误导

改为mod之类的吧 dir是关键词 不能定义为目录

支持(1) 反对(0)

#4楼[楼主] 2013-12-28 22:43 Vamei

@ itfanr

嗯，确实不用关键字好。谢谢提醒。

支持(0) 反对(0)

#5楼 2013-12-31 16:44 huang303513

谢谢撸主。

支持(0) 反对(0)

#6楼[楼主] 2013-12-31 17:03 Vamei

@ huang303513

不用谢哈，文章写出来就是给人读的。有人读，是我比较开心。

支持(0) 反对(0)

#7楼 2014-02-18 10:21 只是_曾经

特意注册账号，以表示感谢。

前面的基础和java类比学习挺简单。到如模块这里和java区别比较大。

继续学习。

支持(1) 反对(0)

#8楼 2014-02-18 10:57 只是_曾经

Traceback (most recent call last):

File "C:\hello.py", line 5, in <module>

```
import this_dir.module
```

```
ImportError: No module named module
```

网上也没什么解决方案。win xp ,python 2.7

支持(0) 反对(0)

#9楼 2014-02-24 10:34 心翼飞扬

我在def_dir目录下创建了first.py文件：

```
def laugh():  
    print('hahahhah')
```

然后在与def_dir同目录下创建了second.py文件：

```
import def_dir.first
```

```
for i in range(10):  
    def_dir.first.laugh()
```

运行second文件，可以成功。

问题：

- 1.在调用对象laugh()时需要加上目录名。
- 2.我在def_dir目录下创建和不创建__init__.py(空文件)都可以成功运行啊。所以对楼主所说的“该文件夹中必须包含一个__init__.py的文件，提醒Python，该文件夹为一个模块包。__init__.py可以是一个空文件。”没有理解
- 3.使用“from first import def_dir.first.laugh”调用时发生错误：invalid syntax

支持(1) 反对(0)

#10楼 2014-03-03 11:20 星尘的天空

@ 心翼飞扬

我对这问题也不理解，我也遇到这个问题

支持(0) 反对(0)

#11楼 2014-03-17 13:01 laocan

@ 心翼飞扬

LZ是用import，你是不是就直接f5运行了

支持(0) 反对(0)

#12楼 2014-04-16 19:41 applesjump

```
>>> import first
```

Traceback (most recent call last):

```
File "<pyshell#0>", line 1, in <module>
```

```
import first
```

```
File "C:\Python25\first.py", line 1
```

```
Python 2.5.4 (r254:67916, Dec 23 2008, 15:10:54) [MSC v.1310 32 bit  
(Intel)] on win32
```

```
^
```

```
SyntaxError: invalid syntax
```

```
>>>
```

请问出现这样的错误提示是为什么呢？

我把first.py保持在shell的保存目录里

支持(0) 反对(0)

#13楼[楼主] 2014-04-17 09:42 Vamei

@ applesjump

你是first.py里import first么？

支持(0) 反对(0)

#14楼 2014-04-23 13:15 Martin11

你好楼主，前两天才开始学Python，我用的是Python 2.7.3。

我在桌面上创建了一个this_dir的文件夹，里面放了__init__.py空文件和first.py与second.py代码如同你上面写的。。

F5运行过second之后却出现了如下内容：

```
>>> ===== RESTART
```

```
=====
```

```
>>>
```

```
Traceback (most recent call last):
```

```
File "C:\Users\omaps\Desktop\this_dir\second.py", line 1, in <module>
```

```
import first
```

```
ImportError: No module named first
```

```
>>>
```

请问这是个什么情况？我是不是忽略了什么？

支持(0) 反对(0)

#15楼 2014-05-06 10:15 zhouzzy

你好.lz。我是完全零基础的初学者。

这一篇中说得（引入模块后，可以通过模块.对象的方式来调用引入模块中的某个对象。）

你的基础08 面向对象的基本概念中，def后面说得move是个方法。

这片中def后面的laugh又是个对象。这里不是很理解。

我对方法，对象，属性等概念有些不明白！

支持(0) 反对(0)

#16楼[楼主] 2014-05-06 10:42 Vamei

@ zhouzzy

眼光很犀利。在Python中，一切皆对象，所以所谓的方法、函数，都是对象。但对象可以作为另一个对象的属性，这个时候，如果是一个函数对象作为另一个对象的属性时，被称为方法。

支持(2) 反对(0)

#17楼 2014-05-06 10:51 zhouzzy

@ Vamei

感谢回复！

支持(0) 反对(0)

#18楼 2014-05-22 13:55 赵赵0_0

执行后会自动出现一个后缀是 .pyc 的文件，这个是做什么用的呢

支持(0) 反对(0)

#19楼[楼主] 2014-05-22 14:25 Vamei

@ 赵赵0_0

从源码翻译成字节码。Python会解释字节码运行。

如果删除该字节码文件，Python会根据.py文件，重新生成

支持(0) 反对(0)

#20楼 2014-05-22 14:26 赵赵0_0

上边 import 文件夹.模块名 的方式貌似只可以从父目录调用子目录中的模块

我把两个文件换了位置，出错了。。。。

查了下资料：

```
1 import sys
2 sys.path.append('父目录路径')
3 import c
```

这种方式无论从父目录模块调用子目录模块，还是从子目录模块调用父目录模块都可以，但是不太理解，楼主，这里为什么要调用个sys呢，sys中的append是做什么的呢？

支持(0) 反对(0)

#21楼 2014-05-22 14:28 赵赵0_0

@ Vamei

噢~~谢谢楼主，教程写得特别好~

支持(0) 反对(0)

#22楼 2014-10-10 09:47 NUST小文

@ 心翼飞扬

我的版本2.7.3，按照lz的说法，需要一个__init__.py才能运行成功。

支持(0) 反对(0)

#23楼 2015-04-03 14:26 Moriaty0o0

感谢楼主，有学到了一些，涨姿势了

支持(0) 反对(0)

#24楼 2015-04-24 11:52 潮爷

@ 只是_曾经

不错是说没这个名字的模块，import this_dir.module这只是举个例子，其实格式就是import 模块包名.模块名，前提是你有名为this_dir的模块包，且里面包含名为module.py的文件，才可以这样用，不是固定格式。

支持(0) 反对(0)

#25楼 2015-05-15 16:10 elivans

测试了下，3.3版本不需要__init__.py这个文件，但是文件夹中自动新建了__pycache__文件夹

支持(0) 反对(0)

#26楼 2015-06-09 09:35 Kenneth_2015

@ 只是_曾经

import dir.mod

使用dir.mod里面的一个方法时，写成dir.mod.method()即可

支持(0) 反对(0)

#27楼 2015-07-01 15:05 少南，无恒

嗯，模块包的概念感觉跟java包，c#命名空间差不多

```
1  #python v2.7
2  #coding=utf-8
3  #!/usr/bin/python
4
5  from Module.TrIO import *
6  import json
7
8  helper = FileHelper("test.txt")  ## from Module.TrIO
9  helper.open()
10 helper.writeLine("22+ddsds\r\n")
11 value = ("this ia test content", 123, 456)
12 helper.writeLine(str(value))
13 helper.close()
14
15 print json.dumps([1,"ss","ddd"])
```

支持(0) 反对(0)

#28楼 2015-09-06 11:29 yana789

搜索想要的模块有没有语句能指定路径的？Python脚本语言与matlab有相似之处，博主写的也很清晰，赞一个

支持(0) 反对(0)

#29楼 2015-11-04 22:57 zjyx

请问楼主first 里的参数有办法能传递回second 里吗？

支持(0) 反对(0)

#30楼 2016-04-06 18:55 wqh2016

非常好

支持(0) 反对(0)

#31楼 2016-04-22 15:44 浪p迹

每一部分都是一小段，短小精悍，学习效率高

[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



GrapeCity
ActiveReports
企业级报表服务平台
单独部署、集成应用、报表制作、数据整合
权限管理、移动办公、二次集成开发
立即了解

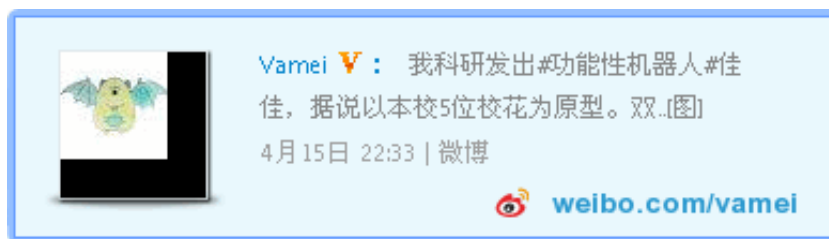


JPush 极光推送 消息推送领导品牌全面升级 JIGUANG 极光 详情点击

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 V：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/assets/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

05370297