

## Python : \_\_init\_\_.py的作用 原



cloud-coder 发布于 2014/02/19 17:41 字数 558 阅读 6655 收藏 4 点赞 2 评论 0

python

\_\_init\_\_.py

package

module

作用

### 一、\_\_init\_\_.py的作用：

在使用PyDev插件，在Eclipse创建package时，会自动在package所在的目录自动创建一个\_\_init\_\_.py文件，且文件内容为空。这个文件有什么有呢？

\_\_init\_\_.py的作用有如下几点：

1. 相当于class中的def \_\_init\_\_(self):函数，用来初始化模块。
2. 把所在目录当作一个package处理
3. from-import 语句导入子包时需要用到它。 如果没有用到, 他们可以是空文件。

如引入package.module下的所有模块

```
from package.module import *
```

这样的语句会导入哪些文件取决于操作系统的文件系统. 所以我们在\_\_init\_\_.py 中加入 \_\_all\_\_变量.

该变量包含执行这样的语句时应该导入的模块的名字. 它由一个模块名字符串列表组成.

### 二、python在执行import语句时的步骤

- 1: 创建一个新的，空的module对象（它可能包含多个module）；
- 2: 把这个module对象插入sys.module中
- 3: 装载module的代码（如果需要，首先必须编译）
- 4: 执行新的module中对应的代码。

在执行第3步时，首先要找到module程序所在的位置，其原理为：如果需要导入的module的名字是module1，则解释器必须找到module1.py。

它首先在当前目录查找，然后是在环境变量PYTHONPATH中查找。PYTHONPATH可以视为系统的PATH变量一类的东西，其中包含若干个目录。如果PYTHONPATH没有设定，或者找不到module1.py，则继续搜索与python的安装设置相关的默认路径，在Unix下，通常是/usr/lib64/python2.6/。

事实上，搜索的顺序是：当前路径（以及从当前目录指定的sys.path），然后是PYTHONPATH，然后是python的安装设置相关的默认路径。正因为存在这样的顺序，如果当前 路径或PYTHONPATH中存在与标准module同样的module，则会覆盖标准module。也就是说，如果当前目录下存在xml.py，那么执行import xml时，导入的是当前目录下的module，而不是系统标准的xml。