**Python中的包**

包是一个分层次的文件目录结构，它定义了一个由模块及子包，和子包下的子包等组成的Python的应用环境。

考虑一个在Phone目录下的pots.py文件。这个文件有如下源代码：

#!/usr/bin/python

def Pots():

print "I'm Pots Phone"

同样地，我们有另外两个保存了不同函数的文件：

* Phone/Isdn.py 含有函数Isdn()
* Phone/G3.py 含有函数G3()

现在，在Phone目录下创建file \_\_init\_\_.py：

* Phone/\_\_init\_\_.py

当你导入Phone时，为了能够使用所有函数，你需要在\_\_init\_\_.py里使用显式的导入语句，如下：

from Pots import Pots

from Isdn import Isdn

from G3 import G3

当你把这些代码添加到\_\_init\_\_.py之后，导入Phone包的时候这些类就全都是可用的了。

#!/usr/bin/python

# Now import your Phone Package.

import Phone

Phone.Pots()

Phone.Isdn()

Phone.G3()