



# [问与答]Python 中 `__all__` 的作用？

2015年11月15日 12:59:42    [orangeleiu](#)    阅读数：21772

你要是看Python的源码或者相关框架的源码，总是在 `__init__.py` 或者是源文件的开头看到一个 `__all__` 变量的定义，今天就说说它的作用. orangeleiu

## 问题出处

Can someone explain `all` in Python?

## 问题

我越来越多的使用Python了，经常看到 `__all__` 变量再各种 `__init__.py` 文件中，谁能解释为什么那么做呢？

## 解答

它是一个string元素组成的list变量，定义了当你使用 `from <module> import *` 导入某个模块的时候能导出的符号（这里代表变量，函数，类等）。

举个例子，下面的代码在 `foo.py` 中，明确的导出了符号 `bar`，`baz`

```
1  __all__ = ['bar', 'baz']
2
3  waz = 5
4  bar = 10
5  def baz(): return 'baz'
```

导入实现如下：

```
1  from foo import *
2
3  print bar
4  print baz
5
6  # The following will trigger an exception, as "waz" is not exported by the module
7  # 下面的代码就会抛出异常，因为 "waz"并没有从模块中导出，因为 __all__ 没有定义
8  print waz
```

如果把 `foo.py` 中 `__all__` 给注释掉，那么上面的代码执行起来就不会有问题，`import *` 默认的行为是从给定的命名空间导出所有的符号（当然下划线开头的私有变量除外）。

## 注意

需要注意的是 `__all__` 只影响到了 `from <module> import *` 这种导入方式，对于 `from <module> import <member>` 导入方式并没有影响，仍然可以从外部导入。