

INDEX



- Python教程
  - Python简介
  - 安装Python
  - 第一个Python程序
  - Python基础
  - 函数
  - 高级特性
  - 函数式编程
- 模块
  - 使用模块
  - 安装第三方模块
- 面向对象编程
- 面向对象高级编程
- 错误、调试和测试
- IO编程
- 进程和线程
- 正则表达式
- 常用内建模块
- 常用第三方模块
- virtualenv
- 图形界面
- 网络编程
- 电子邮件
- 访问数据库
- Web开发
- 异步IO
- 实战
- FAQ
- 期末总结

关于作者



廖雪峰北京 朝阳区

+ 加关注

安装第三方模块

Reads: 15332276

在Python中，安装第三方模块，是通过包管理工具pip完成的。

如果你正在使用Mac或Linux，安装pip本身这个步骤就可以跳过了。

如果你正在使用Windows，请参考[安装Python](#)一节的内容，确保安装时勾选了 `pip` 和 `Add python.exe to Path`。

在命令提示符窗口下尝试运行 `pip`，如果Windows提示未找到命令，可以重新运行安装程序添加 `pip`。

注意：Mac或Linux上有可能并存Python 3.x和Python 2.x，因此对应的pip命令是 `pip3`。

例如，我们要安装一个第三方库——Python Imaging Library，这是Python下非常强大的处理图像的工具库。不过，PIL目前只支持到Python 2.7，并且有年头没有更新了，因此，基于PIL的Pillow项目开发非常活跃，并且支持最新的Python 3。

一般来说，第三方库都会在Python官方的[pypi.python.org](#)网站注册，要安装一个第三方库，必须先知道该库的名称，可以在官网或者pypi上搜索，比如Pillow的名称叫Pillow，因此，安装Pillow的命令就是：

```
pip install Pillow
```

耐心等待下载并安装后，就可以使用Pillow了。

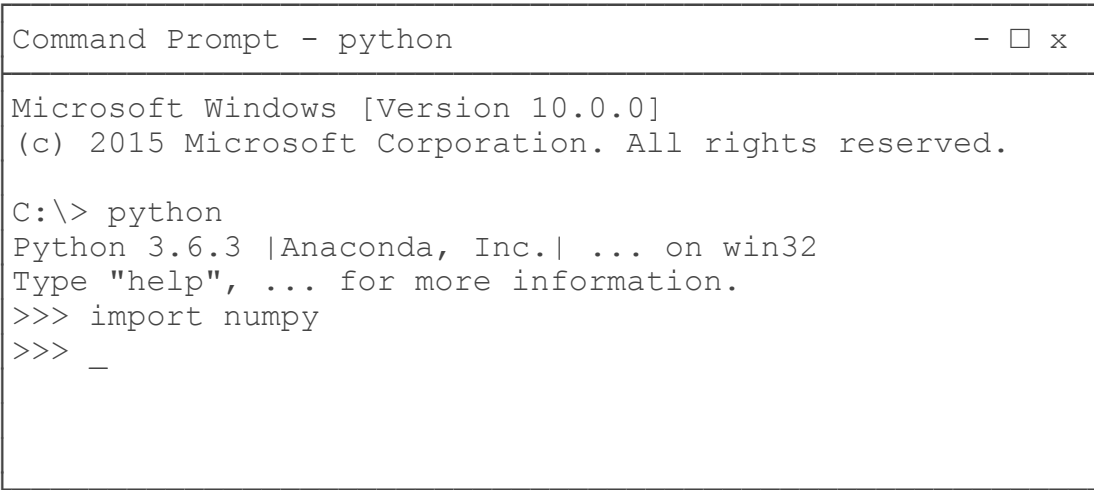


安装常用模块

在使用Python时，我们经常需要用到很多第三方库，例如，上面提到的Pillow，以及MySQL驱动程序，Web框架Flask，科学计算Numpy等。用pip一个一个安装费时费力，还需要考虑兼容性。我们推荐直接使用[Anaconda](#)，这是一个基于Python的数据处理和科学计算平台，它已经内置了许多非常有用的第三方库，我们装上Anaconda，就相当于把数十个第三方模块自动安装好了，非常简单易用。

可以从[Anaconda官网](#)下载GUI安装包，安装包有500~600M，所以需要耐心等待下载。网速慢的同学请移步[国内镜像](#)。下载后直接安装，Anaconda会把系统Path中的python指向自己自带的Python，并且，Anaconda安装的第三方模块会安装在Anaconda自己的路径下，不影响系统已安装的Python目录。

安装好Anaconda后，重新打开命令行窗口，输入python，可以看到Anaconda的信息：



可以尝试直接 `import numpy` 等已安装的第三方模块。

模块搜索路径

当我们试图加载一个模块时，Python会在指定的路径下搜索对应的.py文件，如果找不到，就会报错：

```
>>> import mymodule
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ImportError: No module named mymodule
```

默认情况下，Python解释器会搜索当前目录、所有已安装的内置模块和第三方模块，搜索路径存放在 `sys` 模块的 `path` 变量中：

```
>>> import sys
>>> sys.path
['', '/Library/Frameworks/Python.framework/Versions/3.6/lib/python36.zip',
'/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6', ...,
'/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages']
```

如果我们要添加自己的搜索目录，有两种方法：

一是直接修改 `sys.path`，添加要搜索的目录：

```
>>> import sys
>>> sys.path.append('/Users/michael/my_py_scripts')
```

这种方法是在运行时修改，运行结束后失效。

第二种方法是设置环境变量 `PYTHONPATH`，该环境变量的内容会被自动添加到模块搜索路径中。设置方式与设置Path环境变量类似。注意只需要添加你自己的搜索路径，Python自己本身的搜索路径不受影响。

读后有收获可以支付宝请作者喝咖啡，读后有疑问请加微信群讨论：



还可以分享给朋友：

分享到微博

◀ Previous Page

Next Page ▶

Comments

Make a comment

Sign in to make a comment

