# 第1章 搭建编程环境

所有现代操作系统都支持Python，但在不同的操作系统中，搭建Python编程环境的过程存在细微的差别：

## 1.1 在Linux系统中搭建编程环境

### 1.1.1 检查当前安装的Python版本

你的系统可能已经安装了Python，要确定默认安装的版本，可使用命令python --version：

$ python --version

Python 2.7.6

如果输出类似于上面这样，说明默认安装的是Python 2.7。你还应核实是否安装了Python 3：

$ python3 --version

Python 3.4.0

如果安装了Python 3.4或更高的版本，可直接使用它；如果安装的是Python 3.3或更低的版本，可能需要安装Python 3.5。

### 1.1.2 安装Python 3.5

下面的安装说明适用于Ubuntu系统，还有大多数基于Debian的使用包管理器apt的系统。

添加deadsnakes包，再安装Python 3.5：

$ sudo add-apt-repository ppa:fkrull/deadsnakes

$ sudo apt-get update

$ sudo apt-get install python3.5

要核实是否成功地安装了Python 3.5，可像下面这样做：

$ python3.5 --version

Python 3.5.0

现在可启动Python终端会话了，为此可使用命令python3.5：

$ python3.5

Python 3.5.0 (default, Sep 17 2015, 00:00:00)

[GCC 4.8.4] on linux

Type "help", "copyright", "credits" or "license" for more information.

>>>

这个命令还可用于配置文本编辑器以及从终端运行程序。

### 1.1.3 安装Geany

在Ubuntu和其他使用包管理器apt的系统中，可在线安装Geany：

$ sudo apt-get install geany

如果这样做行不通，可参阅http://geany.org/Download/ThirdPartyPackages/的安装说明。

#### 配置Geany

启动终端会话时，如果你使用的是简单命令python，就根本不需要配置Geany；但如果你使用的是诸如python3、python3.5等命令，就必须对Geany稍作修改，使其使用正确的Python版本来运行程序。

为此，在Geany中新建一个空文件，将其保存为hello\_world.py。这个文件应只包含如下一行代码：

print("Hello Python world!")

选择菜单Build>Set Build Commands，在出现的对话框中包含字样Compile，该字样旁边有一个命令。将这个命令改成下面这样：

python3 -m py\_compile "%f"

如果你启动终端会话时使用的是命令python3.5，务必将上述命令中的python3替换为python3.5。

在字样Execute旁边，输入如下命令：

python3 "%f"

同样，如果你启动终端会话时使用的是命令python3.5，务必将上述命令中的python3替换为python3.5。

现在可以运行程序了，为此可选择菜单Build>Execute、单击Execute图标（两个齿轮）或按F5键。

## 1.2 在OS X系统中搭建编程环境

### 1.2.1 检查当前安装的Python版本

在你的系统中，可能已经安装了Python。要核实这一点，在文件夹Applications/Utilities中，单击Terminal（也可按Command + 空格键，再输入terminal并按回车）。

在打开的终端窗口中，执行命令python --version，以确定安装的是哪个Python版本：

$ python --version

Python 2.7.5

如果输出类似于上面这样，说明默认安装的是Python 2.7。你还应检查是否安装了Python 3：

$ python3 --version

Python 3.4.0

如果安装了Python 3.4或更高的版本，可直接使用它；如果安装的是Python 3.3或更低的版本，可能需要安装Python 3.5。

### 1.2.2 安装Python 3.5

先得安装Homebrew，这样可轻松地安装最新的Python版本，但在此之前，必须安装一些Apple Xcode工具：

$ xcode-select --install

安装过程可能持续一段时间，这取决于你的网络连接的速度。接下来安装Homebrew：

$ ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)"

要核实是否成功地安装了Homebrew，可执行命令brew doctor：

$ brew doctor

Your system is ready to brew.

现在可以使用Homebrew来安装Python包了。要安装Python 3，可执行如下命令：

$ brew install python3

你可核实是否成功地安装了Python 3，为此可像下面这样做：

$ python3 --version

Python 3.5.0

你将使用命令python3来配置文本编辑器、启动Python终端会话以及从终端运行程序。

### 1.2.3 安装Sublime Text

你可下载Sublime Text安装程序，为此可访问http://www.sublimetext.com/3并单击链接OS X。Sublime Text采取的是自由许可策略，你想免费使用多长时间都可以，但如果你喜欢这个程序，并打算长时间使用它，开发者请求你购买许可。

下载安装程序后打开它，再将图标Sublime Text拖放到文件夹Applications中。

#### 配置Sublime Text使其使用Python 3

启动终端会话时，如果你使用的是简单命令python，就根本不需要配置Sublime Text；但如果你使用的是诸如python3、python3.5等命令，就必须对Sublime Text稍作修改，使其使用正确的Python版本来运行程序。

为此，首先确定Python解释器的路径：

$ type -a python3

python3 is /usr/local/bin/python3

在Sublime Text中新建一个空文件，将其保存为hello\_world.py。这个文件应只包含如下一行代码：

print("Hello Python world!")

选择菜单Tools>Build System>New Build System，这将打开一个新的配置文件。删除其中的所有内容，再输入如下内容：

{

"cmd": ["/usr/local/bin/python3", "-u", "$file"],

}

这些代码让Sublime Text使用命令python3来运行程序。请确保其中的路径为你在前面执行命令type -a python3时获悉的路径（可能与这里的路径不同）。将这个配置文件命名为Python3.sublime-build，并将其保存到默认目录——你选择菜单Save时Sublime Text打开的目录。

现在可以运行程序了，为此可选择菜单Build>Execute、单击Execute图标（两个齿轮）或按F5键。

## 1.3 在Windows系统中搭建编程环境

### 1.3.1 检查当前安装的Python版本

在你的系统中，可能已经安装了Python。为核实这一点，可打开一个命令窗口（方法是按住Shift键并右击桌面，再选择“在此打开命令窗口”；也可在“开始”菜单中输入“命令”并按回车），再执行命令python –version来确定安装的是哪个版本：

> python --version

Python 2.7.6

如果输出类似于上面这样，说明安装的是Python 2.7。你还应核实是否安装了Python 3：

> python3 --version

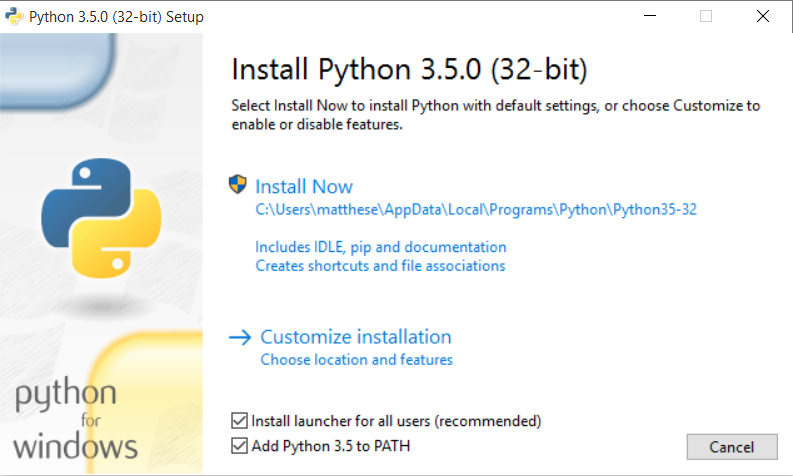
Python 3.4.0

如果安装了Python 3.4或更高的版本，可直接使用它；如果安装的是Python 3.3或更低的版本，可能需要安装Python 3.5。

如果执行上述两个命令时都出现错误消息，说明系统中没有安装Python，因此需要安装Python 3.5。

### 1.3.2 安装Python 3.5

访问https://www.python.org/downloads/并单击按钮Download Python 3.5。下载安装程序后运行它，并确保选中了复选框Add Python to PATH：



选中这个复选框让你能够使用简单命令python。如果你安装时没有选中这个复选框，请参阅后面的“在变量Path中添加Python”一节。

要核实是否安装成功，可像下面这样做：

> python --version

Python 3.5.0

现在可使用命令python来启动Python终端会话了：

> python

Python 3.5.0 (v3.5.0:374f501f4567, Sep 13 2015, 02:16:59) [MSC v.1900 32 bit (Intel)] on win32

Type "help", "copyright", "credits" or "license" for more information.

>>>

你还将使用这个命令来配置文本编辑器以及从终端运行程序。

### 1.3.3 在变量Path中添加Python

如果你在安装Python时选中了复选框Add Python to PATH且命令python管用，就可跳过这一步。

为确定Python的路径，打开Windows资源管理器并切换到C盘。查找名称以Python打头的文件夹；为找到正确的文件夹，可能需要在Windows资源管理器的搜索框中输入python。打开这个文件夹，并在其中找到名称为python（全小写）的文件。右击这个文件并选择“属性”，你将在字样“位置：”右边看到这个文件的路径。

在终端窗口中，执行命令python --version并它前面指定刚才看到的路径，以确定你安装的是哪个版本：

> C:\\Python35\python --version

Python 3.5.0

打开控制面板，依次选择“系统和安全”和“系统”，再单击“高级系统设置”。在打开的窗口中，单击“环境变量”。

在“系统变量”部分，找到并选择变量Path，再单击“编辑”按钮。在打开的对话框中，在文本框“变量值”中单击，并使用右箭头键滚动到最右边。请务必小心，不要覆盖既有的变量值；如果不小心删除了原来的值，可单击“取消”按钮并重做。在该变量的既有值末尾添加一个分号，再加上文件python.exe的路径：

%SystemRoot%\system32\...\System32\WindowsPowerShell\v1.0\;C:\

关闭所有的终端窗口，再打开一个新的终端窗口。如果你此时执行命令python --version时，将看到刚才在变量Path中指定的Python版本。现在，只需在命令提示符下输入python就可启动Python终端会话。

### 1.3.4 安装Geany

你可从http://www.geany.org/Download/Releases下载Windows版Geany安装程序。下载安装程序geany-1.25\_setup.exe后，运行它并接受所有的默认设置。

#### 配置Geany

启动终端会话时，如果你使用的是简单命令python，就根本不需要配置Geany；但如果你使用的是命令python3或C:\Python35\python，就必须对Geany稍作修改，使其使用正确的Python版本来运行程序。

为此，在Geany中新建一个空文件，将其保存为hello\_world.py。这个文件应只包含如下一行代码：

print("Hello Python world!")

选择菜单Build>Set Build Commands，在出现的对话框中包含字样Compile，该字样旁边有一个命令。将这个命令改成下面这样：

python3 -m py\_compile "%f"

在修改后的命令中，也可指定完整路径，如C:\Python35\python -m py\_compile "%f"。

在字样Execute旁边，输入如下命令：

python3 "%f"

同样，也可在其中指定完整路径，如C:\Python35\python "%f"。

现在可以运行程序了，为此可选择菜单Build>Execute、单击Execute图标（两个齿轮）或按F5键。