**Python学习和开发环境的建立**

**程懋华**

Python解释器、标准库及扩展包构成Python的基本开发环境，能够满足学习和简单软件开发的要求。Python解释器及其标准库可从Python官网下载；其他扩展包可用pip命令等方法安装。

大型软件开发宜采用集成开发工具，常用的有: Eclipse，VS2015等，其中，Eclipse+PyDev 为更多Python程序开发人员使用。此外，还可以使用文本编辑器构建开发环境

**注意：务必从官网下载软件，以免下载被“污染”的软件**。

1. **Python基本开发环境建立**

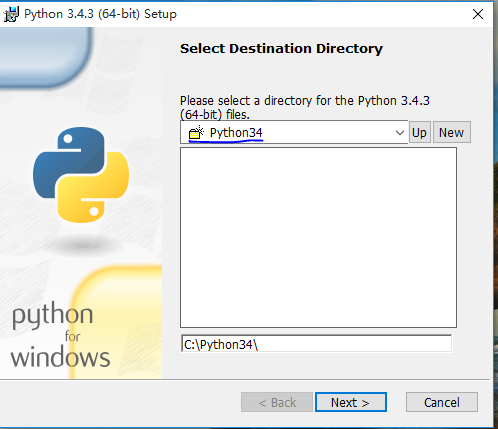
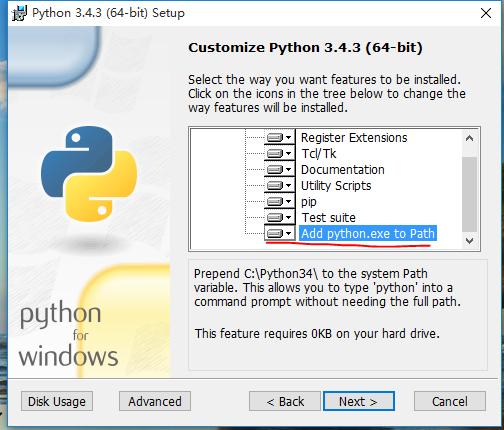
从官网：

<http://www.python.org>

下载和操作系统（32/64位）对应版本的Python3软件。下面以Python3.4.3版本安装为例做简单说明

**Windows：**

建议使用定制方式安装: 1) 加环境变量PATH; 2) 自定义安装目录： c:\python34

然后，命令行下更新pip到新版本：

>pip install --upgrade pip

**Ubuntu：**

Ubuntu内置有Python2和Python3基本包.还需要安装pip3 和IDLE3：

sudo apt-get install python3-pip

Sudo apt-get install idle3

1. **Python扩展包安装**

无论在Windows还是Linux下，pip命令都是便捷的安装方法。

加州大学欧文分校提供了很全面的Windows32/64位Python扩展包，更新及时。地址： <http://www.lfd.uci.edu/~gohlke/pythonlibs/>

**2.1 编码规范包**

**2.1.1 PEP8**

pip install pep8

pip install autopep8

pip install pytest-pep8

pip install pylint

**2.1.2 PEP257**

Docstring:

pip install pep257

**2.2 科学计算软件包**

**2.2.1 SciPy**

Scipy官网：<http://www.scipy.org/>,提供有全面的科学计算软件包信息。基本应用安装以下3个即可：

numpy

scipy

matplotlib

**Window**环境下Scipy 软件包的 安装，建议从加州大学欧文分校下载编译好的\*.whl，然后，用pip逐个本地安装：

>pip install \*.whl

**Ubuntu**环境下,，建议使用apt-get安装 :

sudo apt-get install python-numpy python-scipy python-matplotlib

**2.2.2 交互计算 Windows安装和配置**

1） 安装jupyter ：

>pip install jupyter 在线安装

2） 安装Python语言内核

>pip install ipython ，支持Python语言

3） 安装依赖包

>pip install pyreadline

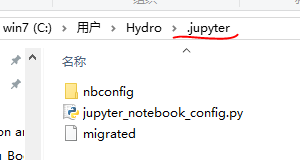
>pip install sympy

4） 配置notebook:

命名行下执行：

>jupyter notebook --generate-config

产生notebook的空配置文件 jupyter\_notebook\_config.py于 jupyter安装目录下：



用IDLE打开该空配置文件，修改如下项目：

c.NotebookApp.ip = 'localhost'

关键是修改指定notebook文件在那里（目录）：

　# The directory to use for notebooks and kernels.

c.NotebookApp.notebook\_dir = **'d:\\SEUCourse\\notebook-course'**

5) 启动notebook

命名行下执行：jupyter notebook

6）支持显示数学符号、公式，安装MathJax：

打开一个IPython的shell，然后键入如下代码：

from IPython.external.mathjax import install\_mathjax

install\_mathjax()

**2.2.3 交互计算Ubuntu安装**

　　sudo apt-get install ipython3

**2.2.4 IF97 for Python**

Windows 32/64位版: 从 <https://github.com/SEU03013050C/SEUIF97> 下载：SEUIF97.dll和 seuif97.py，然后：

1) SEUIF97.dll拷贝到 c:\windows\system

2) seuif97.py拷贝到 c:\python34\Lib

**2.3 其他扩展包**

**2.2.1 微软excel文件处理**

xlrd是专门用来在python中读取微软excel文件的模块，可以直接下载安装，也可以通过包管理器安装。

官方资料：

　　下载地址：http://pypi.python.org/pypi/xlrd

　　官网地址：http://www.python-excel.org/

　　文档地址:https://secure.simplistix.co.uk/svn/xlrd/trunk/xlrd/doc/xlrd.html

　　文档pdf下载：<http://www.simplistix.co.uk/presentations/python-excel.pdf>

方便快捷的安装方式是：

**Windows:** pip install xlrd

**Ununtu :**  $ sudo pip3 install xlrd

**2.2.2 Tornado Web 服务器框架**

Tornado 和现在的主流 Web 服务器框架（包括大多数 Python 的框架）有着明显的区别：它是非阻塞式服务器，而且速度相当快。得利于其非阻塞的方式和对epoll的运用，Tornado 每秒可以处理数以千计的连接，这意味着对于实时 Web 服务来说，Tornado 是一个理想的 Web 框架。

Tornado Web框架是轻量级的框架，实现方式简洁、强壮。Tornado是快速开发网络的适用框架。

　官网：http://www.tornadoweb.org/en/stable/

　安装：

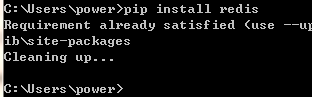
**Windows:** pip install tornado

**Ununtu :**  $ sudo pip3 install tornado

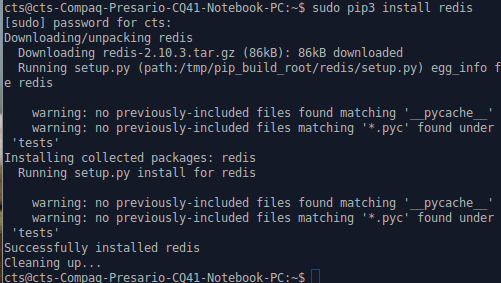
**2.2.3 Redis数据库API**

**1）Redis-py安装：**

**Windows安装：**>pip install redis



**Ubuntu安装：** $ sudo pip3 install redis

****

**测试：**

import redis

r = redis.StrictRedis(host='localhost', port=6379, db=0)

r.set('foo', 'bar')

print(r.get('foo'))

**2） Hiredis-py解析器安装**

Parser可以控制如何解析redis响应的内容。redis-py包含两个Parser类，PythonParser和HiredisParser。如果已经安装了hiredis模块，redis-py会默认使用HiredisParser，否则会使用PythonParser。

HiredisParser是C编写的，由redis核心团队维护，性能要比PythonParser提高10倍以上。高性能系统推荐安装。

安装hiredis-py 需要 Python 2.6 及更高版本。.目前HiredisParser只支持[MacOS](https://pypi.python.org/pypi?:action=browse&c=145)、[POSIX](https://pypi.python.org/pypi?:action=browse&c=160)，在Windows 下无法编译安装使用 。

**Ubuntu安装：**

需要确保机器上有安装hiredis-py需要的Python开发头文件，在Ubuntu/Debian系统，可以通过：

$sudo apt-get install python-dev

安装这些头文件，然后，执行：

$ sudo pip3 install hiredis

1. **基于Eclipse的Python开发环境**

**3.1 安装Eclipse IDE**

**3.1.1安装JavaSDK**

Eclipse IDE是使用Java开发的，电脑中需要预先安装好Java JRE/JDK软件包，因此，安装前，在命名行下，输入：

>java –version

检查是否已经安装了Java软件包。 如果电脑中已经安装了Java ,会显示有关版本**。**



如果没有安装java 或者版本不能满足Eclipse IDE最低要求，需要从Oracle下载Java包。

Java JDK的Oracle官方下载地址如下：

[**http://www.oracle.com/technetwork/java/javase/downloads/index.html**](http://www.oracle.com/technetwork/java/javase/downloads/index.html)



下载时需要注意操作系统32/64位，选择相应版本下载（\*是Java版本号）：

Windows 32位：\*-windows-i586.exe

Windows 64位：\*-windows-i64.exe.

运行下载Oracle的Java安装包（安装时注意配置环境变量），即可安装好Java环境。

**3.1.2 Eclipse IDE**

　　Eclipse IDE是插件型开发环境，有很多版本可以下载。本课程主要使用Python语言，会涉及C/C++开发，所以，建议下载Eclipse CDT(Eclipse IDE　for　C/C++)版.

Eclipse CDT官方下载地址：

http://www.eclipse.org/downloads/



根据操作系统32/63位，下载相应的版本，然后将下载的Eclipse CDT解压到指定目录下，运行解压目录下的：eclipse.exe即可。

如果使用Windows7 以上版本操作系统，建议将运行eclipse.exe，固定到任务栏。（在eclipse.exe文件名上，点鼠标右键即可）





以后，从任务栏启动Eclipse方便很多（同样也可以将IDLE固定到任务栏）。

**3.2 安装Eclipse开发Python插件**

使用Eclipse IDE作为Python开发环境，需要：1） 安装PyDev插件；2）配置解释器为Python3.4；

**3.2.1安装PyDev插件**

通过Windows->Eclipse Marketplaces进入市场，输入Pydev，找到Pydev安装/更新项目，在线安装即可：



在线安装过程简单，但是，安装速度受网络环境影响。

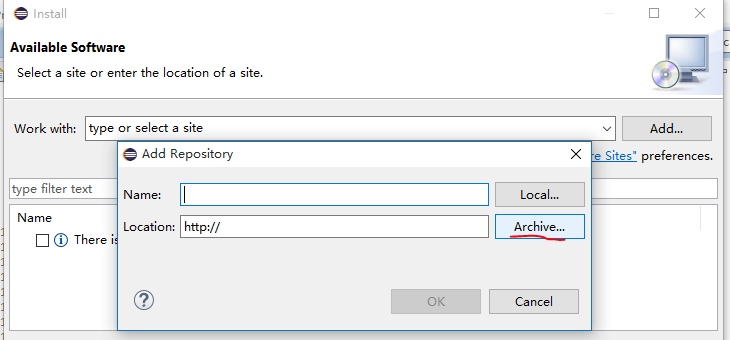
如果很慢，可从pydev官网（<http://www.pydev.org/>）提供的下载地址：

http://sourceforge.net/projects/pydev/files/

下载插件包，然后，通过：

Help->Install->New Software->Add->Archive

离线安装.



**3.2.2 配置开发使用的 Python解释器版本**

安装好后，重新启动。通过：

Windows->Preference->Pydev->Interperters->Python Interperter点其中的：Advanced Auto-config配置开发使用的 Python解释器版本:



配置好后切换到Python场景，就可以进行Python开发了。

**3.2.3　Python开发简单示例**

1. **基于PyDev的静态检查**

PyDev中集成了PEP8,AutoPEP8 和Pylint代码检查功能，这些功能默认状态都是关闭的。

程序开发过程中，需要有规范意识，但是。不可能有高的规范性，那样只能影响程序开发的进程，这是如果一直开启代码规范检查，经常提示不规范。会影响开发，所以，默认关闭时合适的。在程序开发一个阶段结果实现时，进行规范性检查更合适。

**4.1 PEP8检查和修改**

1）启动 pep8 检查：

Window > Preferences

PyDev > Editor > Code Analysis > pep8.py

选择Errors/Warnings其中之一..



右键Python工程，选择 PyDev，点 "code analysis"，即可对工程中所有Python源码进行PEP8检查： .



1）启动autopep8自动修改：

点Windows -> Preferences ->输入 'autopep8' 作为搜索串.



|  |
| --- |
| 选择（Check）: Use autopep8.py for code formatting? |

在Python源码窗口，按 CTRL-SHIFT-F 就可以自动修改代码

**4.2 使用Pylint**

PyDev 默认不开启Pylint。通过

     Window -> preferences -> Pydev -> Pylint,选中"Use pylint?",

找到安装好的lint.py的地址,例如"C:\Python34\Lib\site-packages\pylint\lint.py"



配置参数，限制Pylint的输出



选中Project->Build Automatically,这样程序修改，保存时pylint就会自动检查项目中的代码,也可用Ctrl+B手动build触发pylint。



1. **其他开发环境**

Python开发环境很多也很全面，除了主流的Eclipse IDE加插件Pydev方式外，还有以下一些值得关注：

1）移动平台： Python For Android：Python For IOS：

2） 微软Visual Studio: VS 2015以上版本集成支持Python,可使用社区版

3) Canopy: 科学计算开发环境和扩展包集成

**小结**

Python 学习和开发环境多种多样，建议同学们首先需要掌握IDLE 使用，其他，根据自己的进展，逐步掌握，进一步掌握Eclipse+Pydev，其他，了解即可。