# python 常用类库

## 1.  基本安装

<http://www.python.org/> 官方标准Python开发包和支持环境，同时也是Python的官方网站；  
<http://www.activestate.com/> 集成多个有用插件的强大非官方版本，特别是针对Windows环境有不少改进；

## 2.  Python文档

Python库参考手册  
Python Tutorial中文版  
Byte of Python 可以代替Tutorial使用，有中文译版的入门书籍。  
Dive into Python 一本比较全面易懂的入门书，中文版翻译最近进步为很及时的5.4了。英文版则正在写针对Python 3.x的版本。  
缺点是此书针对的Python版本比较陈旧，会给出一些过时的优化建议，建议同时阅读赖勇浩的＜Dive into Python＞大补贴。  
《深入 Python 3》中文版  
Python社区官方建议采用的Python编码风格（啄木鸟社区上的中文译文）。相关的代码检查工具，参考：用PEP 8检查Python Coding风格。  
Google SoC 建议的 Python 编码风格 （Python 编码风格指南中译版（Google SOC））  
Google 原版的 Python 编码风格指南 SoC 的那个指南基本上是在这个上面做的修改和增补。  
Zoom.Quiet的文档集（包括大量Python内容）  
豆瓣上的Python相关书籍推荐  
Pythonic到底是什么玩意儿？  
啄木鸟社区上的Python资源索引

### 2.1 推荐资源站点

Parnassus山的拱顶 巨大的Python代码库，包罗万象。既可以从上面下载代码参考学习，同时也是与Python有关程序的大列表。  
Python号星际旅行船 著名Python社区，代码、文档、高人这里都有。  
faqts.com的Python程序设计知识数据库 Python程序设计知识库，都是与Python有关的程序设计问题及解决方法。  
啄木鸟 Pythonic 开源社区 著名的（也可以说是最好的）国内Python开源社区。  
赖勇浩推荐的几个好玩又有难度的编程网站

### 2.2  其他参考资料

ZDNET文章：学习Python语言必备的资源  
各种实用代码片段  
Python编程技巧笔记  
如何获得Python脚本所在目录的位置  
如何方便地给Python环境注册新类库  
Python几种并发实现方案的性能比较  
Pythonic Web 应用平台对比 现在个人对web.py（大致和KARRIGELL类似，都是类似于PHP那样比较简单的Python Web应用框架）和web2py非常感兴趣。近来Zope社区又出现了repoze.bfg这个解决方案，于是做应用时可以自由选用Zope中的合适部分了（ Zope 社区的另一个发展方向是Grok，目标是敏捷易用又强大）。另外还有Paste这样的wsgi参考实现来帮助做自己的框架。  
在wxPython下进行图像处理的经验 （其实，仅使用wxPython也可以完成很多比较基础的图像处理工作，具体可以参照《wxPython in Action》一书的第12节）  
序列化存储 Python 对象  
通过win32扩展接口使用Python获得系统进程列表的方法  
py2exe使用中遇到的问题  
idle的中文支持问题

### 2.3  代码示例

<http://newedit.tigris.org/technical.htm> Limodou的NewEdit编辑器的技术手册，讨论了一些关于插件接口实现、i18实现、wxPython使用有关的问题，值得参考。  
关于插件扩展机制：  
A Simple Plugin Framework 讲解了一种非常聪明而简单的办法来为代码提供插件接口。如果参考Java社区的类似框架，则有Java Plug-in Framework (JPF)以及OSGi标准。  
另外limodou先生在Ulipad中实现插件和MixIn的机制也十分简单有效，见设计说明(一)--概述，当然在UliPad最新版本的代码中对这个机制的实现又有实用的改进。（发布插件则有不少项目采用Python Eggs）  
比较复杂的应用程序则可以使用Zope Component Architecture这样的机制来实现灵活易用的扩展接口（注意：zca虽然是Zope平台的一个组成部分，但其自身是不需要依赖Zope平台即可使用的）。

## 3.  常用工具

<http://www.scons.org/> Java有Ant这个巨火的构建工具，Python的特性允许我们构建更新类型的构建工具，就是scons了。  
Python Sidebar for Mozilla FireFox的一个插件，提供一个用来查看Python文档、函数库的侧边栏。  
IPython 很好用的Python Shell。  
Easy Install 快速安装Python模块的易用性解决方案（使用这种方式发行的包通常以Python Eggs的格式被Easy Install自动处理）。有人推荐先用 virtualenv 建个虚拟python环境再使用 easy\_install (之后就不用加 --prefix参数了）。  
Mac下说setuptools版本过低的问题

### 3.1  Python IDE

我的IDE选择经验 其实我现在只用Vim和Pydev了  
Pydev 基于Eclipse的，非常棒的Python环境，改进速度非常快，现在是我最喜欢的IDE。  
ulipad Python+wxPython实现的轻量级全功能跨平台IDE，非常好用。做一些简单小东西的时候我会用这个。  
Eric Python IDE 基于QT实现的非常不错的跨平台PYTHON IDE。支持调试，支持自动补全，甚至也支持重构。如果没有ulipad，并且在我的gnome下基于QT的Eric界面别走形的话，我会选这个。  
Spyder 几乎是个轻量化的 Eric ，也是基于 QT 和 QScintilla2 实现。它是科学计算包 python(x,y) （集成了科学计算常用的python第三方库，有400多M）的一个组成部分。  
<http://www.xored.com> Trustudio 一个基于Eclipse的、同时支持Python和PHP的插件，曾经是我最喜欢的Python IDE环境，功能相当全了，不过试用时感觉有些细节不完善以致不大好用。  
<http://www-900.ibm.com/developerWorks/cn/opensource/os-ecant/index.shtml> 用 Eclipse 和 Ant 进行 Python 开发  
<http://www.scintilla.org/> 同时支持Win和Linux的源代码编辑器，似乎支持Python文件的编辑。  
<http://boa-constructor.sourceforge.net/> 著名的基于WxPython的GUI快速生成用的Python IDE，但是开发进度实在太差了……  
<http://pype.sourceforge.net/> 成熟的Python代码编辑器，号称功能介于EMACS和IDLE之间的编辑器。  
<http://www.stani.be/python/spe> SPE：号称是一个Full Featured编辑器，集成WxGlade支持GUI设计。  
eggy 用Python和QT实现的Python IDE，据说轻量又好用～  
(Edit Section ↓)

### 3.2  内置类库使用参考

Python正则表达式操作指南  
为方便调试正则表达式，可以用KODOS（基于 PyQT ）、kiki（基于 wxPython ）、Pyreb（基于 wxPython ）中的任何一个来辅助。  
我目前用的是 kiki，但最近它的官方网站登陆不上去，我从 Debian 包把它的源码扒出来，应用了 stani 做的wxPython 2.8 兼容性补丁，并且用py2app把它封装成 Mac 应用程序供 QuickSilver 调用。修改以后的源代码包下载。

### 3.3  常用第三方类库

**NumPy** Python的数学运算库，有时候一些别的库也会调用里面的一些功能，比如数组什么的；  
**Pil** Python下著名的图像处理库Pil；  
SimPy 利用Python进行仿真、模拟的解决方案；  
Matplotlib 据说是一个用来绘制二维图形的Python模块，它克隆了许多Matlab中的函数， 用以帮助Python用户轻松获得高质量(达到出版水平)的二维图形；  
Graphviz 与 Matplotlib 不同，这个东西能画：有向图、网络结构、有限状态机等等。本身不是 Python 实现的，但是能找到 Python 封装。  
Crypto python的加解密扩展模块；  
Python for CJK 提供与python有关的CJK语言支持功能：转码、显示之类。  
Psyco、Pyrex：两个用于提高Python代码运行效率的解决方案；如果对性能还不满意，可以试试这样：用 Python 写自己的 C Module。  
Pyflakes、PyChecker、PyLint：都是用来做Python代码语法检查的工具。  
图形界面开发库：  
wxPython 基于wxWindows的易用且强大的图形界面开发包wxPython；wxPython发行版还自带了PyCrust、PyShell、PyAlaCarte和PyAlaMode等几个工具，分别是图形界面Shell和代码编辑器等，分别具有不同特点可以根据自己的需要选用。  
PyQt 一般认为综合各平台上的表现，PyQt比wxPython能提供更为可靠的表现，较少遇到稀奇古怪的Bug。只是Qt虽然LGPL了，可是PyQt对商业软件仍然是收费的，Nokia刚开始做LGPL的Qt绑定——PySide，但是还谈不上成熟。  
在Mac上安装PyQt的详细步骤 PyQt在Win和Linux上都有可直接安装的二进制包，只有Mac没有。不过按照这篇文档来做，也是不难（我已试过，一切靠谱）。  
PyGtk Qt永远的竞争者，在Win和Linux下表现都还不错，Mac下新的本地化移植绕过了系统自带的X11服务，性能大为提升，只是对输入法和字体支持等还有欠缺。  
先按照Building GTK-OSX安装GTK-OSX，然后按照Gtk+ Python页的说明安装PyGtk即可，非常容易。  
Tornado 非阻塞、高性能、可扩展的 Web Server 实现，源自FriendFeed。  
<http://avc.inrim.it/html/> 比MVC模式更方便易用的开发框架，主要是帮助透明地完成界面控件与后台数据之间的内容绑定。该框架支持GTK、QT、TK、wxWidgets等主要具有Python封装的GUI开发框架。  
**pygame** 用Python帮助开发游戏的库，也可以用这个来播放视频或者音频什么的，大概依靠的是SDL；  
py2exe win下将Python程序编译为可执行程序的工具，是一个让程序脱离Python运行环境的办法，也可以生成Windows服务或者COM组件。其他能完成Python脚本到可执行文件这个工作的还有Gordon McMillan's Installer、Linux/Unix专用的freeze、Mac专用的py2app，另外setuptools可能也能帮上点。不过此类工具难免与一些模块有一些兼容性的问题，需要现用现测一下。总之这个主题可以参考How to Distribute Commercial Python Applications。  
嵌入式数据库：BerkeleyDB的Python版，当然还有其他的好多。  
PEAK提供一些用于实现自动化集群测试的重要基础类库，比如超轻量线程框架、分布式代码执行等。  
greenlet 是PEAK给出的一种进程内的线程模型实现，自称是Stackless项目的副产品。其实这是一种伪线程，也就是说greenlet间切换需要明确指定下一步切换到哪块指令继续执行，而没有被执行的代码就被暂时挂起了。这里面greenlet间的执行顺序是完全确定的，并且是非并发的，就不能解决长时间大量资源占用的操作会导致程序无响应的问题。但greenlet可以用来减少资源死锁、互斥造成的资源消耗（代码被阻塞时，可以切换先去执行其他未被阻塞的部分），因此用来响应按钮点击等是挺好用的。  
greenlet 现在不再是 pylib 的组成部分了，而是一个独立的名为 greenlet 的 pypi 包，可以通过 easy\_install 安装。  
Eventlet 是基于 greenlet 完成的一个高并发网络库，提供“线程”池、消息队列等许多非常方便的机制，比直接用 greenlet 要方便得多。并且 Eventlet 源自著名的虚拟现实项目——第二人生，是经过现实考验的可靠解决方案。  
Pyro 一个Python的分布式对象系统，提供面向对象风格的RPC功能。  
PycURL 传说这是实现Python下多线程网页抓取的效率最高的解决方案，本质是对libcurl C语言库的封装。其实Twisted也可以定制成为一个网页抓取工具的。  
Protocol Buffers Google提供的高效数据序列化解决方案，比基于XML的数据序列化方案在数据大小和编解码效率上都大约高一个数量级。当然，Google自己也说这种办法也不是用哪都合适，一来Protocol Buffers不大适合用来处理格式化的文本，另一方面其本质是一个传输协议的生成语言，可能在传输的内容相对固定时，才值得专门生成一套这样的解析器。  
pyinotify 利用操作系统自身提供的Notify机制以最高的效率监控文件变化。  
Scrapy 是一个网络爬虫框架，可以非常容易地通过定制规则得到所需的爬网器。基于 Twisted 实现。  
chardet 一个猜测网页编码（比如utf-8还是gb18030）的库，会根据HTTP参数、HTML标签、XML标签中的相关声明来进行猜测。另有一个非Python实现的Enca也可以提供类似的功能。  
JPype Python里调用Java类库、Java代码的桥接解决方案（因Limodou在ChinaUnix上提及而得知）。JPype使用要点  
µTidylib 著名html整理、排错、自动修正工具HTML Tidy Library的Python封装。通常在正式开始分析html之前，先用这个东西预处理一下，后面的工作会轻松、准确得多～  
Fudge 被HD的项目组认为是最好的Python Mock模块（也即冒烟测试）。  
nose 测试脚本自动发现、自动执行辅助工具。文档见nose documentation。  
Coverage.py Python 测试代码覆盖率统计工具，已经内置于 nose 。  
Heapy 对 Python 程序进行内存占用剖析的模块，Guppy-PE 项目的组成部分之一，被 Scrapy 等项目用作内存泄露的调试工具。类似的还有：  
PySizer，感觉已经太旧了。  
Dowser，和 Heapy 特点不太一样，并且使用CheeryPy作为调试结果的输出界面。  
Gluttony Python 模块之间依赖关系图的自动生成工具。

### 3.4  其他东西

[http://www.forum.nokia.com/main/0,,034-821,00.html](http://www.forum.nokia.com/main/0%2C%2C034-821%2C00.html) Nokia居然发布了在Series 60系统上运行Python程序（图形界面用wxPython）的库，还有一个Wiki页是关于这个的：[http://www.postneo.com/postwiki/moin.cgi/PythonForSeries60](http://www.postneo.com/postwiki/moin.cgi/PythonForSeries60" \t "_blank) 。Python4Symbian这个页面是记录的我的使用经验。  
pyre：使用Python完成高性能计算需求的包，真的可以做到么？还没研究。  
Parallel Python：纯Python的并行计算解决方案。相关中文参考页面  
Pexpect：用Python作为外壳控制其他命令行程序的工具（比如Linux下标准的ftp、telnet程序什么的），还没有测试可用程度如何。  
pyjamas：Google GWT的Python克隆，还处在早期版本阶段。  
Durus：Python的对象数据库，当然也可以作为一种对象实例持久化的机制来使用；这是一个开源的纯Python实现，并提供一个可选的C语言插件来大幅提高运行效率。  
Missile DB：一种 Python 的、简洁高效的 DBMS，自称是Durus的一种衍生品，更能够适应Stackless Python环境。同时也是并发性能极高的Eurasia3项目的一个子项目。  
ey-lessql：收集的各种分布式键值数据库的Python版驱动。如果不考虑Python驱动的问题，A Yes for a NoSQL Taxonomy是一个非常完整的非SQL数据存储引擎列表。  
NoSQL数据库探讨之一 － 为什么要用非关系数据库？  
NOSQL Patterns  
目前我最感兴趣的键值数据库是MongoDB，有人认为它基本上已经可以替代MySQL所能提供的功能，并且性能更高！  
Ming可能是类似 ORM 对关系数据库所起作用的 MongoDB 的 Python 封装？  
pymodels MongoDB 的轻量级 ORM 封装，它本来是用于 Tokyo Tyrant 和 Tokyo Cabinet 的。  
pyzmq：消息队列 zerozmq 的 Python 封装，基本上这是一个追求性能为主的消息队列实现，全部数据在内存中保存。如果担心数据持久化的问题，可以考虑RabbitMQ 等类似方案。另据传闻，Twitter已经由RabbitMQ转换到Kestrel。  
Mulib 和 restish 都是基于 Python 实现的 REST 风格网络服务框架。  
Scapy：似乎是一个能够控制底层网络封包的交互式Python工具，对网络协议分析应该很有用。