# Python学习笔记

# Python的简介

Python，是一种面向对象、直译式计算机程序设计语言，也是一种功能强大而完善的通用型语言，已经具有十多年 的发展历史，成熟且稳定。这种语言具有非常简捷而清晰的语法特点，适合完成各种高层任务，几乎可以在所有的操作系统中运行。Python 具有[脚本语言](http://baike.baidu.com/view/76320.htm)中最丰富和强大的类库，足以支持绝大多数日常应用。它的名字来源于一个喜剧,也许最初设计Python这种语言的人并没有想到今天Python会在工业和科研上获得如此广泛的使用。由于Python对于[C](http://baike.baidu.com/view/10075.htm)和其他语言的良好支持，很多人还把Python作为一种“[胶水语言](http://baike.baidu.com/view/2993364.htm)”（glue language）使用。使用Python将其他语言编写的程序进行集成和封装。在谷歌内部的很多项目使用[C++](http://baike.baidu.com/view/824.htm)编写性能要求极高的部分，然后用Python调用相应的模块。Python是一种解释性的语言，但是这种说法是不正确的，实际上，Python在执行时，首先会将.py文件中的源代码编译成Python的byte code（字节码），然后再由Python Virtual Machine来执行这些编译好的byte code。这种机制的基本思想跟Java，.NET是一致的。然而，Python Virtual Machine与Java或.NET的Virtual Machine不同的是，Python的Virtual Machine是一种更高级的Virtual Machine。这里的高级并不是通常意义上的高级，不是说Python的Virtual Machine比Java或.NET的功能更强大，而是说和Java 或.NET相比，Python的Virtual Machine距离真实机器的距离更远。或者可以这么说，Python的Virtual Machine是一种抽象层次更高的Virtual Machine。

基于C的Python编译出的字节码文件，通常是.pyc格式。

在实际开发中，python常被昵称为胶水语言，这不是说他会把你的手指粘住，而是说他能够很 轻松的把用其他语言制作的各种模块（尤其是C/C++）轻松地联结在一起。常见的一种应用情形是，使用python快速生成程序的原型（有时甚至是程序的 最终界面），然后对其中有特别要求的部分，用更合适的语言改写，比如3D游戏中的图形渲染模块，速度要求非常高，就可以用C++重写。

可扩充性可说是Python作为一种编程语言的特色。新的内置模块（module）可以用C 或 C++写成。而我们也可为现成的模块加上Python的接口。Python可以使用户避免过分的语法的羁绊而将精力主要集中到所要实现的程序任务上。

特点

简单————Python是一种代表简单主义思想的语言。阅读一个良好的Python程序就感觉像是在读英语一样，尽管这个英语的要求非常严格！Python的这种伪代码本质是它最大的优点之一。它使你能够专注于解决问题而不是去搞明白语言本身。

　　易学————就如同你即将看到的一样，Python极其容易上手。前面已经提到了，Python有极其简单的语法。

　　免费、开源————Python是FLOSS（自由/开放源码软件）之一。简单地说，你可以自 由地发布这个软件的拷贝、阅读它的源代码、对它做改动、把它的一部分用于新的自由软件中。FLOSS是基于一个团体分享知识的概念。这是为什么 Python如此优秀的原因之一——它是由一群希望看到一个更加优秀的Python的人创造并经常改进着的。

　　高层语言————当你用Python语言编写程序的时候，你无需考虑诸如如何管理你的程序使用的内存一类的底层细节。

　　可移植性————由于它的开源本质，Python已经被移植在许多平台上（经过改动使它能够工 作在不同平台上）。如果你小心地避免使用依赖于系统的特性，那么你的所有Python程序无需修改就可以在下述任何平台上面运行。这些平台包括 Linux、Windows、FreeBSD、Macintosh、Solaris、OS/2、Amiga、AROS、AS/400、BeOS、 OS/390、z/OS、Palm OS、QNX、VMS、Psion、Acom RISC OS、VxWorks、PlayStation、Sharp Zaurus、Windows CE甚至还有PocketPC、Symbian以及Google基于linux开发的android平台！

　　解释性————这一点需要一些解释。一个用编译性语言比如C或C++写的程序可以从源文件 （即C或C++语言）转换到一个你的计算机使用的语言（二进制代码，即0和1）。这个过程通过编译器和不同的标记、选项完成。当你运行你的程序的时候，连 接/转载器软件把你的程序从硬盘复制到内存中并且运行。而Python语言写的程序不需要编译成二进制代码。你可以直接从源代码 运行 程序。在计算机内部，Python[解释器](http://baike.baidu.com/view/592974.htm)把 源代码转换成称为字节码的中间形式，然后再把它翻译成计算机使用的机器语言并运行。事实上，由于你不再需要担心如何编译程序，如何确保连接转载正确的库等 等，所有这一切使得使用Python更加简单。由于你只需要把你的Python程序拷贝到另外一台计算机上，它就可以工作了，这也使得你的Python程 序更加易于移植。

　　面向对象————Python既支持面向过程的编程也支持面向对象的编程。在“面向过程” 的语言中，程序是由过程或仅仅是可重用代码的函数构建起来的。在“面向对象”的语言中，程序是由数据和功能组合而成的对象构建起来的。与其他主要的语言如 C++和Java相比，Python以一种非常强大又简单的方式实现面向对象编程。

　　可扩展性————如果你需要你的一段关键代码运行得更快或者希望某些算法不公开，你可以把你的部分程序用C或C++编写，然后在你的Python程序中使用它们。

　　可嵌入性————你可以把Python嵌入你的C/C++程序，从而向你的程序用户提供脚本功能。

　　丰富的库————Python标准库确实很庞大。它可以帮助你处理各种工作，包括正则表达式、 文档生成、单元测试、线程、数据库、网页浏览器、CGI、FTP、电子邮件、XML、XML-RPC、HTML、WAV文件、密码系统、GUI（图形用户 界面）、Tk和其他与系统有关的操作。记住，只要安装了Python，所有这些功能都是可用的。这被称作Python的“功能齐全”理念。除了标准库以 外，还有许多其他高质量的库，如wxPython、Twisted和Python图像库等等。

　　概括————Python确实是一种十分精彩又强大的语言。它合理地结合了高性能与使得编写程序简单有趣的特色。

前景

著名的搜索引擎 Google 也大量使用Python。更加令人吃惊的是，在Nokia智能手机所采用的Symbian操作系统上，Python成为继C++,Java之后的第三个编程语言！

### 应用范围

### 系统编程

　　提API，能方便进行系统维护和管理，Linux下标志性语言之一，是很多系统管理员理想的编程工具。

### 图形处理

　　有PIL、Tkinter等图形库支持，能方便进行图形处理。

### 数学处理

　　NumPy扩展提供大量与许多标准数学库的接口。

### 文本处理

　　python提供的re模块能支持正则表达式，还提供SGML，XML分析模块，许多程序员利用python进行XML程序的开发。

### 数据库编程

　　程序员可通过遵循Python DB-API（数据库应用程序编程接口）规范的模块与Microsoft SQL Server，Oracle，Sybase，DB2，Mysql、SQLite等数据库通信。python自带有一个Gadfly模块，提供了一个完整的SQL环境。

### 网络编程

　　提供丰富的模块支持sockets编程，能方便快速地开发分布式应用程序。

### Web编程

　　应用的开发语言，支持最新的XML技术。

### 多媒体应用

　　Python的PyOpenGL模块封装了“OpenGL应用程序编程接口”，能进行二维和三维图像处理。PyGame模块可用于编写游戏软件。

# Python的安装

在 Windows 上, 安装 Python 有两种选择：

一种选择是ActiveState 制作了一个 Windows 上的 Python 安装程序称为 ActivePython, 它包含了一个完整的 Python 发布; 一个适用于 Python 编程的 IDE 附加了一些 Python 的 Windows 扩展, 提供了全部的访问 Windows APIs 的服务, 以及 Windows 注册表的注册信息。虽然 ActivePython 不是开源软件, 但它可以自由下载。

ActivePython 的安装过程:

1. 从 <http://www.activestate.com/Products/ActivePython/> 下载 ActivePython 。
2. 双击安装程序 ActivePython-2.2.2-224-win32-ix86.msi。
3. 按照安装程序的提示信息一步步地执行。
4. 在安装完后之后, 关闭安装程序, 打开 Start->Programs->ActiveState ActivePython 2.2->PythonWin IDE。 您将看到类似如下的信息:

PythonWin 2.2.2 (#37, Nov 26 2002, 10:24:37) [MSC 32 bit (Intel)] on win32.

Portions Copyright 1994-2001 Mark Hammond (mhammond@skippinet.com.au) -

see 'Help/About PythonWin' for further copyright information.

>>>

第二种选择是使用由 Python 发布的 “官方” Python 安装程序。是可自由下载的开源软件, 并且您总是可以获得当前 Python 的最新版本。

安装来自 [Python.org](http://www.python.org/) 的 Python ：

1. 从 <http://www.python.org/ftp/python/>选择最新的 Python Windows 安装程序, 下载 .exe 安装文件。
2. 双击安装程序 Python-2.xxx.yyy.exe。 文件名依赖于您所下载的 Python 安装程序文件。
3. 按照安装程序的提示信息一步步地执行。
4. 如果磁盘空间不足, 可以取消 HTMLHelp 文件, 实用脚本 (Tools/), 和/或 测试套件 (Lib/test/)。
5. 如果您没有机器的管理员权限, 您可以选择, Advanced Options, 然后选择 Non-Admin Install。 这只对登记注册表和开始菜单中创建的快捷方式有影响。
6. 在安装完成之后, 关闭安装程序, 打开 Start->Programs->Python 2.3->IDLE (Python GUI)。 您将看到类似如下的信息:

Python 2.3.2 (#49, Oct 2 2003, 20:02:00) [MSC v.1200 32 bit (Intel)] on win32

Type "copyright", "credits" or "license()" for more information.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Personal firewall software may warn about the connection IDLE

makes to its subprocess using this computer's internal loopback

interface. This connection is not visible on any external

interface and no data is sent to or received from the Internet.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

IDLE 1.0

>>>

# Python程序运行

## 第一个Python程序 Hello World程序

这个程序在命令行窗口上打印Hello World

代码示例：

def buildConnectionString(params):

"""Build a connection string from a dictionary of parameters.

Returns string."""

return ";".join(["%s=%s" % (k, v) for k, v in params.items()])

if \_\_name\_\_ == "\_\_main\_\_":

mystr = 'hello world!'

print mystr

对每一行的解释

1. **运行程序**

程序运行方法 ：python odbchelper.py

# Python的语法

## 代码缩进

Python 函数没有明显的 begin 和 end, 没有花括号, 用于标识函数的开始和结束。 唯一的分隔符是一个冒号 (:)。

代码块通过它们的缩进来定义的，开始缩进表示块的开始, 取消缩进表示块的结束。

这个函数代码 (包括 doc string) 缩进了 4 个空格。不一定非要是 4, 只要一致就可以了。没有缩进的第一行则在函数体之外。

## 代码注释

井（#）字符表示从这个符号出现一直到该语句行结尾的文字将是一个程序注释。

## 对象

一切都可以赋值给变量或作为参数传递给函数。

在 Python 中, 函数同其它东西一样也是对象。Python 函数有属性, 并且这些属性在运行时是可用的。在 Python 中一切都是对象, 并且几乎一切都有属性和方法。 所有的函数都有一个内置的 \_\_doc\_\_ 属性, 它会返回在函数源代码中定义的 doc string 。 sys 模块是一个对象, 它有一个叫作 path 的属性。

## 变量

按数据类型定义划分语言

静态类型定义语言

一种在编译期间数据类型固定的语言。大多数静态类型定义语言是通过要求在使用所有变量之前声明它们的数据类型来保证这一点的。 Java 和 C 是静态类型定义语言。

动态类型定义语言

一种在运行期间才去确定数据类型的语言, 与静态类型定义相反。 VBScript 和 Python 是动态类型定义的, 因为它们确定一个变量的类型是在您第一次给它赋值的时候。

强类型定义语言

一种总是强制类型定义的语言。 Java 和 Python 是强制类型定义的。您有一个整数, 如果不明确地进行转换 , 不能将把它当成一个字符串。

弱类型定义语言

一种类型可以被忽略的语言, 与强类型定义相反。 VBScript 是弱类型定义的。在 VBScript 中, 您可以将字符串 '12' 和整数 3 进行连接得到字符串'123', 然后可以把它看成整数 123 , 所有这些都不需要任何的显示转换。

Python 既是 动态类型定义语言 (因为它不使用显示数据类型声明) , 又是 强类型定义语言 (因为一旦一个变量具有一个数据类型, 它实际上就一直是这个类型了)

## 语句

代码块 ：是指函数、if 语句、for 循环、while 循环等等。

## 函数

函数声明方法 ：

以关键字 def 开始, 接着为函数名, 再往后为参数, 参数放在小括号里； 多个参数之间用逗号分隔。

例如：def buildConnectionString(params):

函数参数定义 ：

Python 中参数params 不需要指定数据类型。 Python 会判定一个变量是什么类型, 并在内部将其记录下来。

函数返回值 ：

Python 不需要指定返回值的数据类型；不需要指定是否有返回值。实际上, 每个 Python 函数都返回一个值；如果函数执行过 return 语句,它将返回指定的值, 否则将返回 None ( Python 的空值) 。

## 类

Python类是静态数据成员或函数声明的包容器，包含在类中德函数叫方法

定义一个类

class class\_name[(base\_classes\_if\_any)] :

“optional documentation string”

Static\_member\_declarstions

Method\_declarations

类是使用关键字class定义的，如果定义了一个子类，就要在括号里给出它是从哪个母类或者基类中被推导出来的。示例：

## 模块

是指可以交互使用的一个代码段, 或者来自一个大规模的 Python 程序。一旦导入了一个模块,就可以引用它的任何公共的函数、类或属性。模块可以通过这种方法来使用其它模块的功能。

# IDE的使用

操作 Python IDE （Python 的辅助开发）：

访问函数的 doc string - 在命令行中启动 Python.exe,然后在 Python 提示信息下输入命令

>>> import odbchelper

>>> params = {"server":"mpilgrim", "database":"master", "uid":"sa", "pwd":"secret"}

>>> print odbchelper.buildConnectionString(params)

server=mpilgrim;uid=sa;database=master;pwd=secret

>>> print odbchelper.buildConnectionString.\_\_doc\_\_

Build a connection string from a dictionary

Returns string.

>>> import sys

>>> sys.path

['', '/usr/local/lib/python2.2', '/usr/local/lib/python2.2/plat-linux2',

'/usr/local/lib/python2.2/lib-dynload', '/usr/local/lib/python2.2/site-packages',

'/usr/local/lib/python2.2/site-packages/PIL', '/usr/local/lib/python2.2/site-packages/piddle']

>>> sys

<module 'sys' (built-in)>

>>> sys.path.append('/my/new/path')

>>> import odbchelper

>>> odbchelper.\_\_name\_\_

'odbchelper'

模块 ： 是指可以交互使用的一个代码段, 或者来自一个大规模的 Python 程序。一旦导入了一个模块,

就可以引用它的任何公共的函数、类或属性。模块可以通过这种方法来使用其它模块的功能。

Python IDE操作的解释 ：

[import odbchelper] 将 odbchelper 程序作为模块导入。import 一个 Python 模块, 可以使用 module.function 来访问它的函数；

[print odbchelper.buildConnectionString(params)] 使用在被导入模块中定义的函数时, 必须包含模块的名字。

[print odbchelper.buildConnectionString.\_\_doc\_\_] 访问函数的 \_\_doc\_\_ 属性。

[import sys] 导入 sys 模块, 使得它的所有函数和属性都有效。模块导入的搜索路径，当导入一个模块时, Python 在几个地方进行搜索。

它会按所有定义在 sys.path 中的目录进行搜索。它只是一个list (列表), 可以查看它或通过标准的list方法来修改它。

[sys.path] 是一个组成当前搜索路径的目录列表。

# Python应用

## 访问文件

示例代码

filename = raw\_input('Enter file name:')

file = open(filename, 'r')

allline = file.readlines()

file.close()

for eachline in allline:

print eachline

## 多线程

## 网络编程

# 参考资料

## Python的库

　　由于设计者和开源社区的共同努力，在python中有大量优秀的库可以被直接调用以高效地完成不同需求的工作。这里列举一些常见常用的库：

　　Tkinter———— Python默认的图形界面接口。

　　Tkinter是一个和Tk接口的Python模块，Tkinter库提供了对Tk API的接口，它属于Tcl/Tk的GUI工具组。Tcl/Tk是由John Ousterhout发展的书写和图形设备。Tcl(工具命令语言)是个宏语言，用于简化shell下复杂程序的开发，Tk工具包是和Tcl一起开发的， 目的是为了简化用户接口的设计过程。Tk工具包由许多不同的小部件，如一个按钮、一个滚动条等。通过Tk提供的这些小部件，我们就可快速地进行GUI开 发。Perl、Scheme等语言也利用Tk库进行GUI开发。Tkinter是跨平台，在各种平台下都能使用。

　　Python Imaging Library(PIL)————python提供强大的图形处理的能力，并提供广泛的图形文件格式支持，该库能进行图形格式的转换、打印和显示。还能进行一些图形效果的处理，如图形的放大、缩小和旋转等。是Python用户进行图象处理的强有力工具。

　　Pmw(Python megawidgets)Python超级GUI组件集————一个在python中利用Tkinter模块构建的高级GUI组件，每个Pmw都合并了一个或多个Tkinter组件，以实现更有用和更复杂的功能。

　　PyXML———— 用Python解析和处理XML文档的工具包，包中的4DOM是完全相容于W3C DOM规范的。它包含以下内容：

　　xmlproc: 一个符合规范的XML解析器。

　　Expat: 一个快速的，非验证的XML解析器。 还有其他

　　和他同级别的还有 PyHtml PySGML

　　PyGame———— 用于多媒体开发和游戏软件开发的模块。

　　PyOpenGL———— 模块封装了“OpenGL应用程序编程接口”，通过该模块python程序员可在程序中集成2D和3D的图形。

　　NumPy、NumArray和SAGE———— NumArray是Python的一个扩展库，主要用于处理任意维数的固定类型数组，简单说就是一个矩阵库。它的低层代码使用C来编写，所以速度的优势很 明显。NumPy是Numarray的后继者，用来代替NumArray。SAGE是基于NumPy和其他几个工具所整合成的数学软件包，目标是取代 Magma, Maple, Mathematica和Matlab 这类工具。

　　MySQLdb模块———— 用于连接[MySQL数据库](http://baike.baidu.com/view/673475.htm)。还有用于zope的ZMySQLDA模块，通过它就可在zope中连接mysql数据库。

　　PyGTK ———— 用于python GUI程序开发的GTK+库。GTK就是用来实现GIMP和Gnome的那个库。有了它，你完全可以自信的尝试自己制造Photoshop

　　PyQt ———— 用于python的Qt开发库。QT就是实现了KDE环境的那个库，由一系列的模块组成，有qt, qtcanvas, qtgl, qtnetwork, qtsql, qttable, qtui and qtxml，包含有300个类和超过5750个的函数和方法。PyQt还支持一个叫qtext的模块，它包含一个QScintilla库。该库是Scintillar编辑器类的Qt接口。

　　PyMedia ———— 用于多媒体操作的python模块。它提供了丰富而简单的接口用于多媒体处理(wav, mp3, ogg, avi, divx, dvd, cdda etc)。可在Windows和Linux平台下使用。

　　Psyco ———— 一个Python代码加速度器，可使Python代码的执行速度提高到与编译语言一样的水平。

　　Python-ldap ———— 提供一组面向对象的API，可方便地在python中访问ldap目录服务，它基于OpenLDAP2.x。

　　smtplib模块 ———— 发送电子邮件。

　　ftplib模块 ———— 定义了FTP类和一些方法，用以进行客户端的ftp编程。我们可用python编写一个自己的ftp客户端程序，用于下载文件或镜像站点。如果想了解ftp协议的详细内容，请参考RFC959。

　　xmpppy模块 ———— Jabber服务器采用开发的XMPP协议，Google Talk也是采用XMPP协议的IM系统。在Python中有一个xmpppy模块支持该协议。也就是说，我们可以通过该模块与Jabber服务器通信，是不是很Cool。

　　下面这些就不详细介绍，只列出名字和功能

　　adodb ———— ADO数据库连接组件

　　bsddb3 ———— BerkeleyDB的连接组件

　　chardet ———— 编码检测

　　scons ———— 项目构建工具，写好了模板用起来还是很方便的

　　sendpkt ———— Python发包

　　setuptools ———— 一套python包管理机制

　　Cheetah ———— 构建和扩充任何种类的基于文本的内容

　　pycurl ———— URL处理工具

　　pydot ———— 画图的，graphiz

　　pyevent ———— Python的事件支持

　　pylint ———— 培养良好的编码习惯

　　Pylons ———— 又一个web framework

　　pypcap ———— 抓包的

　　pysqlite2 ———— SQLite的连接组件

　　python-dnet ———— 控制网络安全的其他设备

　　pythonwin ———— Python的Windows扩展

　　pywmi ———— 省了好多折腾功夫

　　reportlab ———— Python操作PDF的Libary。

　　scapy ———— 网络包构建分析框架,可编程的wireshark,有兴趣的google “Silver Needle in the Skype”

　　simplejson ———— JSON的支持

　　sqlalchemy ———— SQL数据库连接池

　　SQLObject ———— 数据库连接池

　　cherrypy ———— 一个WEB framework

　　ctypes ———— 用来调用动态链接库

　　Cx-oracle ———— 连接oracle的工具

　　DBUtils ———— 数据库连接池

　　django ———— 一个WEB framework

　　DPKT ———— raw-scoket网络编程

　　docutils ———— 用来写文档的

　　dpkt ———— 数据包的解包和组包

　　feedparser ———— rss解析

　　Kodos ———— 正则表达式调试工具

　　Mechanize ———— 爬虫连接网站常用

　　pefile ———— windows pe文件解析器

　　py2exe ———— 用来生成windows可执行文件

　　twisted ———— 巨无霸的网络编程框架

　　winpdb ———— 自己的程序或者用别的库不太明白的时候就靠它了

　　wxPython ———— GUI编程框架,熟悉MFC的人会非常喜欢，简直是同一架构（对于初学者或者对设计要求不高的用户来说，使用 Boa Constructor可以方便迅速的进行wxPython的开发）

　　PIL———— Python 的图像处理库，可以打开数十种图像格式，还提供许多图像处理函数，如图像增强、滤波算法等。

　　Pyro———— Python实现与JAVA RMI类似的技术

　　PLY———— 基于Python的LEX、YACC的语言工具

　　Corepy———— 使用Python开发编写x86汇编程序

　　LightCloud———— Python实现的分布式的键-值数据库

　　Parallel Python（PP）———— 轻松开发SMP、集群并行计算的库

开源社区的工作已经完成到这样的程度，基本上各种常见的编程需求都已经有非常成熟稳定高速的模块来实现，你所需要做的，只是去把它找出来然后阅读文档。

## 集成开发环境

　　适用于Python的[集成开发环境](http://baike.baidu.com/view/14867.htm)（IDE）软件，除了标准二进制发布包所附的IDLE之外，还有许多其他选择。其中有些软件设计有语法着色、语法检查、运行[调试](http://baike.baidu.com/view/322913.htm)、自动补全、智能感知等便利功能。由于Python的[跨平台](http://baike.baidu.com/view/469855.htm)出身，这些软件往往也具备各种操作系统的版本或一定的移植性。

**专门为Python设计的IDE软件：**

　　● IDLE：Python“标准”IDE

　　● Komodo和Komodo Edit：后者是前者的免费精简版

　　● PythonWin：ActivePython软件包的IDE，仅适用于Windows

　　● SPE（Stani's Python Editor）：功能较多的自由软件，依赖wxPython

　　● Ulipad：功能较全的自由软件，依赖wxPython；作者是中国Python高手limodou

　　● WingIDE：可能是功能最全的IDE，但不是自由软件

　　● Eric：基于PyQt的自由软件，功能强大

　　● 有相应[插件](http://baike.baidu.com/view/18979.htm)的通用IDE软件（使用难度往往较大）：

　　● eclipse + pydev插件：稳定性欠佳

　　● emacs + 插件

　　● Vim + 插件

　　● Visual Studio .NET 2003 + VisualPython：仅适用Windows，已停止维护，功能较差

　　● SlickEdit

　　● TextMate 另外，诸如[EditPlus](http://baike.baidu.com/view/206636.htm)、[UltraEdit](http://baike.baidu.com/view/3184.htm)、[PSPad](http://baike.baidu.com/view/284899.htm)等通用的程序员[文本编辑器](http://baike.baidu.com/view/487023.htm)软件也能对Python代码文本作出一定解释，但是否够得上集成开发环境的水平，尚有待评估。