Python的开发工具

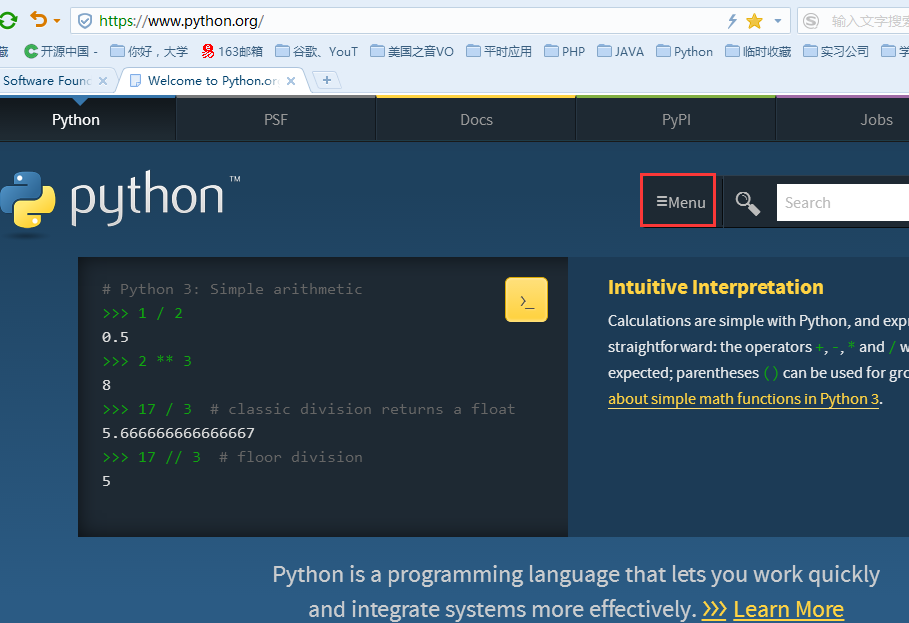
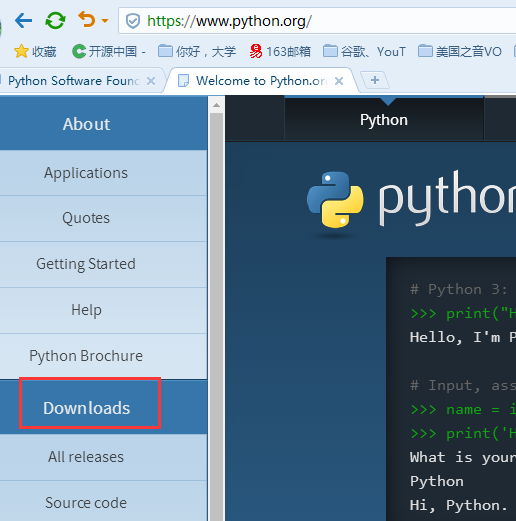
# 在windows平台下Python环境的安装

## Python 2 与Python 3的安装

分别下载**python2** 与**python3** ，安装在不同的目录下，并且配置好**Path路径**。

<https://www.python.org/downloads/>

首先打开： https://www.python.org/ 然后点击MENU，选择Downloads打开即可。



为了避免麻烦，建议安装Python3即可。**现在Python3已经逐渐占据上风。**

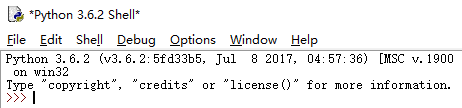
## 配置环境变量

### 配置安装的根目录路径D:\Tools\Python\Python36\添加到Path下；

### 配置路径D:\Tools\Python\Python36\Lib\idlelib到Path下，因为这样可以在命令行启动IDLE。

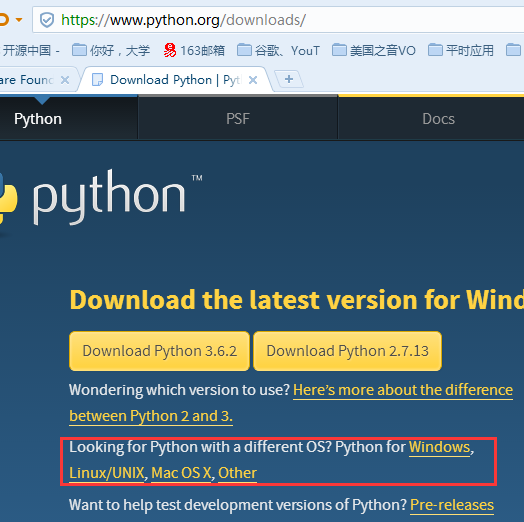
**然后Win+R，输入Idle即可启动Python Shell**。

**或者Win+R，输入cmd，再输入idle也可以启动Python Shell。**



输入**exit()** 即可退出**Python Shell**。

# 在Linux等其他平台下搭建Python开发环境



在linux的终端下，首先解压，然后编译。



**make 即可安装。**

# 命令行下的Python编辑

## 利用cmd命令进入命令行：

### 系统只安装一个版本的Python

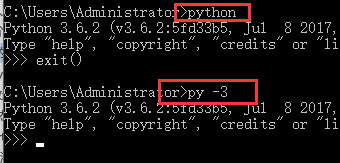
输入python 即可进入Python编辑状态。

或者 py -2 或者py -3

### 安装两个版本的Python

输入**py –2** 即可进入python2.x

输入py -3 进入python3.x（注意中间有个空格）

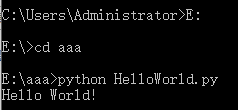


## 在命令行下如何运行py文件

**python E://.../\*\*.py**



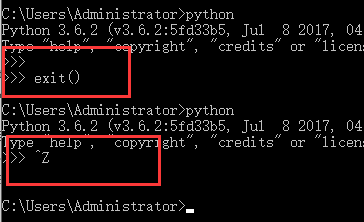
或者



## 在命令行下退出Python编辑状态

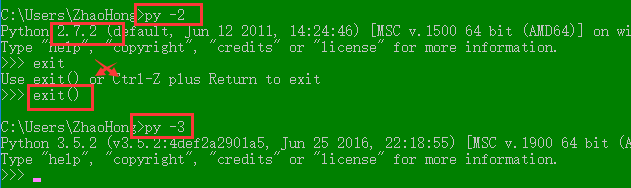
### 利用exit() 函数退出。注意必须加小括号。

### 按住Ctrl+Z，然后回车即可。



## 切换Python2与Python3

在切换到python3前，需要先从python2中退出，然后输入**py -3**即可进入python3。



# Python开发的多种方法

### 在命令行输入python即可，进入Python编辑环境；

### 在命令行输入idle，在idle下编辑；

### 在idle下，Ctrl+N 创建文件，在新文件中编辑命令，并按住Fn+F5执行该文件；

### 随便创建一个文件.py，然后再命令行，进入到该目录下，然后利用python \*.py 执行该文件；

### 安装PyCharm、Eclipse等IDE进行开发。

# Python库文件及源文件查看方法

## 在Python中一个库一般对应一个\*\*.py文件或一个文件夹

**文件夹下一般对应多个库，即包括多个\*\*.py文件**。

例如**re库**就是re.py文件，**查看源代码，只需要打开D:\Python\Python36\Lib\re.py文件即可**。类似还有**os.py、io.py**等。

另外，还有些库如urllib库，urllib是一个文件夹名字，其内部有多个\*\*.py文件，一个py文件对应一个库，如urllib.request、urllib.response、urllib.error等库。

**文件夹下**一般有个文件叫**\_\_init\_\_.py**，里面对应的就是文件夹名对应的库，如bs4库，一般使用from bs4 import BeautifulSoup，其中BeautifulSoup就是在\_init\_.py文件中定义的，其实**BeautifulSoup**就是一个类。

## 查看Python源代码：打开相应的py文件查看

### Python自带模块的源文件存在于D：//Python/Python36/Lib/目录下，在PyCharm中打开该Lib目录就可以找到相应库的源代码。

### 对于第三方库，一般利用pip安装后，源代码都会存在于D:\Python\Python36\Lib\site-packages下，打开查看即可。

## PyCharm中查看源文件

按住Ctrl键，鼠标点击函数名字即可。

# Python开发工具的分类

## 文本工具类

如 **IDLE、Notepad++、Vim、Sublime Text**等

## 集成工具类

如**PyCharm、PyDev、Anaconda**等

PyCharm社区版是免费的。



# Python自带的IDLE：Python Shell工具的使用

## IDLE的启动方法

IDLE是Python安装自带的编辑器，但是调用IDLE需要配置环境变量。

需要将D:\Python\Python36\Lib\idlelib 路径配置到Path下，这样才能利用命令idle启动该编辑器。

## Python IDLE常用的快捷键

## 快捷键：获取历史命令

上一条命令 Alt+P

下一条命令 Alt+N

（1） **Alt+N**：从上向下走，复制出刚才用过的命令。

（2） **Alt+P**：从下向上走，复制出刚才用过的命令。

## 常用其他命令

**Alt+/ 自动补全代码**（查找**编辑器内已经写过的代码**来补全)

**Ctrl+Shift+space** 补全提示 (默认与输入法冲突，修改之)

利用**上下键选择**就可以，自动补全。

## 文件编辑

**Ctrl + N** 启动文件编辑窗口；保存后，利用F5快捷键即可运行。（ThinkPad：Fn+F5）

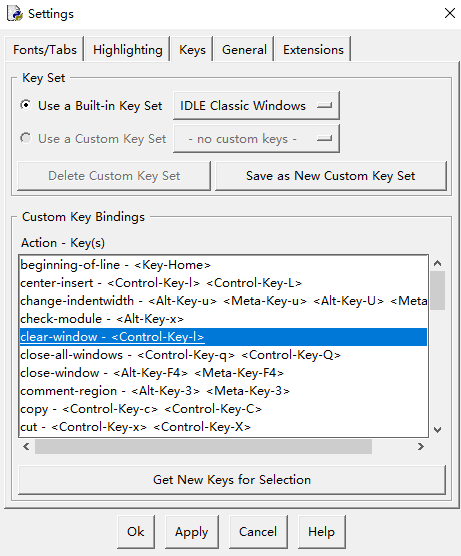
## 其他的快捷键

后退                    Ctrl+Z

重做                    Ctrl+Shift+Z  
加缩进                 Ctrl+]  
减缩进                 Ctrl+[  
加注释                 Alt+3  
去注释                 Alt+4

## 修改或添加快捷键

**Options 🡺Configure IDLE🡺Keys**



## IDLE的清屏

windows DOS下利用**命令cls**清屏；MATLAB下利用**clc** 清屏。

IDLE并没有什么命令可以清屏，也没有快捷键清屏，所以需要自己写一个命令。

### 下载文件clearwindow.py

文件原作者：Roger D. Serwy

<http://pan.baidu.com/s/1ntOzSAt>

### 拷贝到Python的安装目录\Python36\Lib\idlelib\下

拷贝到目录D:\Python\Python36\Lib\idlelib\ 下。

### 配置config-extensions.def文件

打开D:\Python\Python36\Lib\idlelib\config-extensions.def配置文件。

config-extensions.def是IDLE扩展的配置文件。

利用记事本等打开，在最后添加：

[ClearWindow]

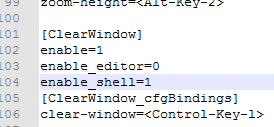
enable=1

enable\_editor=0

enable\_shell=1

[ClearWindow\_cfgBindings]

**clear-window=<Control-Key-l>**



重新启动IDLE，就可以利用***Ctrl + L*** 清屏了。

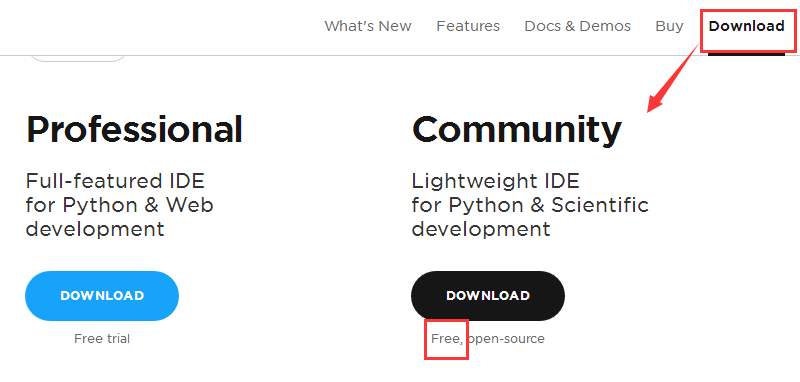
# PyCharm的介绍

## PyCharm IDE的简单介绍



官方网址：<https://www.jetbrains.com/pycharm/>

<http://www.jetbrains.com/pycharm/download/>



**不同系统的PyCharm工具**：



**分为专业版和社区版，社区版是免费的，功能足以个人开发所用**。

## PyCharm中查看源代码快捷键

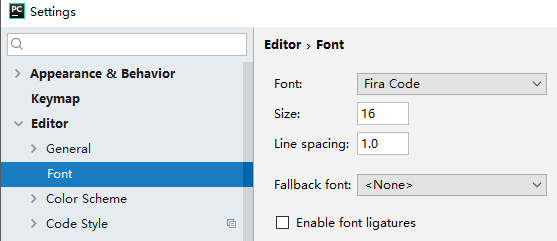
查看源代码方法：**按住Ctrl，鼠标点击需要查看的函数名字即可**打开对应的py文件。

## PyCharm的设置Settings

**File –>Settings** 就可以打开设置面板。 **快捷键：Ctrl + Alt + S**

### 字体设置

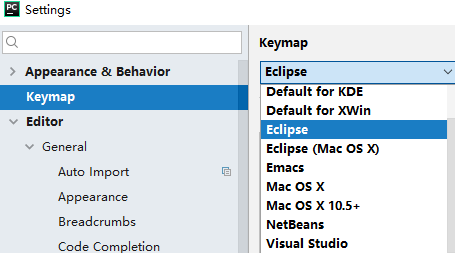
**Editor🡪General🡪Font**



### 快捷键设置：更改为Eclipse的快捷键：

File—>settings—>**keymap**🡪更改为Eclipse或Visual Studio即可。

**Keymap**：



### 自动保存

pycharm默认是自动保存的，习惯自己按**ctrl + s** 的可以进行如下设置：

**file -> Setting -> General -> Synchronization -> Save files on frame deactivation** 和 Save files automatically if application is idle for .. sec 的勾去掉

### 自动导包（自动添加库）：Alt+Enter

file ->Setting -> Editor -> Editor Tabs -> Mark modified tabs with asterisk 打上勾

**利用Alt + Enter快捷键可以实现自动导包功能。**

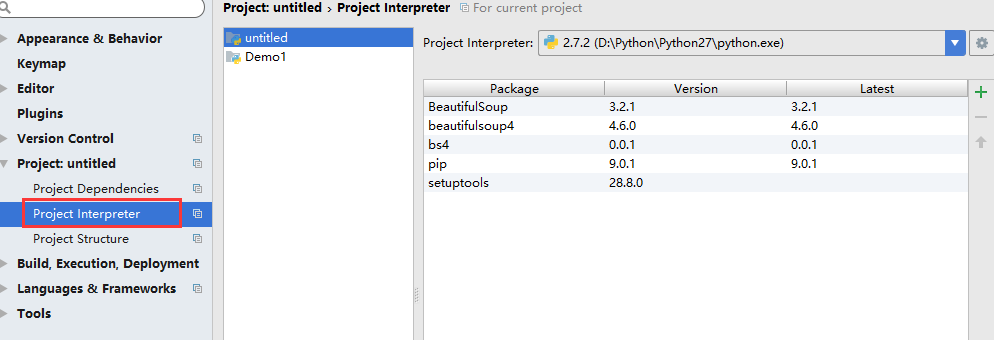
### 主题

在**setting中搜索theme可以改变主题**，所有配色统一改变

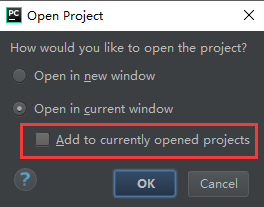
### 添加插件：导入BeautifulSoup

**File—>settings—>Project Interpreter🡪点击加号** 就可以。

删除，选中点击减号即可。



## PyCharm 在一个窗口下，只能是一个工程。



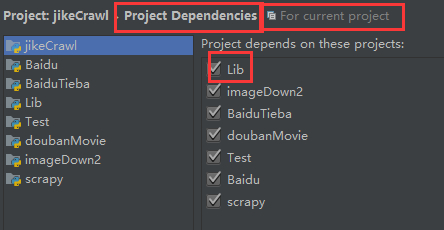
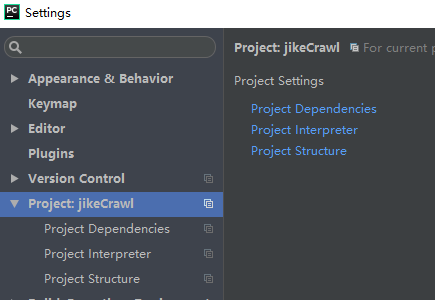
**在选中Add to currently opened projects的情况下，只是把新工程添加到当前工程，作为当前工程的一个依赖工程，对于依赖工程的编写，此时并没有很强的代码提示功能且一般的包都不能正常导入。只有在当前窗口对应的工程下，才可以正常使用。**

### 对于依赖工程编写问题

依赖工程py文件编写的时候，很容易出现错误，比如：**no module named os**

**解决办法**：利用File—>open🡪Python36/Lib文件夹，把Lib文件夹当成是一个工程，然后在File🡪settings🡪Project：\*\*🡪Project Dependencies添加进来。

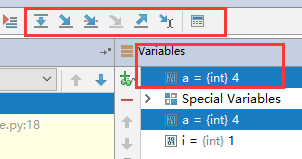
注意：在没有依赖工程情况下，不存在Project Dependencies选项。



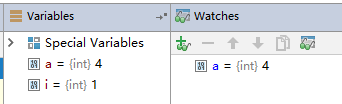
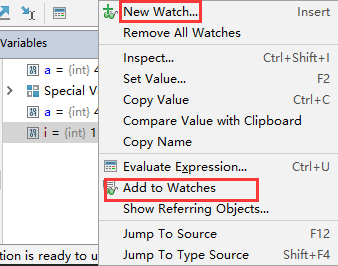
## PyCharm不能自动补全第三方库的函数和属性问题

**Python是动态语言**，所以很多类型必须到运行时才能确定，所以工具无法判断类型并给出方法的自动补齐。

## PyCharm IDE 的调试功能Debug

变量比较多的时候，可以把某些需要关注的变量添加到**Watches窗口**：



# 在Eclipse中开发Python

PyDev：

首先需要指明编码方式：

**# coding=gbk 或者 #!coding=gbk**

**# coding=utf-8或者#!coding=utf-8用等号或者冒号都可**以。

# 工具的选取

## 利用IDLE比较简单方便。

## 集成开发环境Pycharm 社区版免费。

## Anaconda主要用于科学计算和数学分析。

**Canopy 收费，价格较高，Anaconda免费**。



# 附录： PyCharm中的快捷键

PyCharm Default Keymap

* PyCharm3.0默认快捷键(翻译的)  
  1、编辑（Editing）  
  Ctrl + Space 基本的代码完成（类、方法、属性）  
  Ctrl + Alt + Space 快速导入任意类  
  Ctrl + Shift + Enter 语句完成  
  Ctrl + P 参数信息（在方法中调用参数）  
  Ctrl + Q 快速查看文档  
  Shift + F1 外部文档  
  Ctrl + 鼠标 简介  
  Ctrl + F1 显示错误描述或警告信息  
  Alt + Insert 自动生成代码  
  Ctrl + O 重新方法  
  Ctrl + Alt + T 选中  
  Ctrl + / 行注释  
  Ctrl + Shift + / 块注释  
  Ctrl + W 选中增加的代码块  
  Ctrl + Shift + W 回到之前状态  
  Ctrl + Shift + ]/[ 选定代码块结束、开始  
  Alt + Enter 快速修正  
  Ctrl + Alt + L 代码格式化  
  Ctrl + Alt + O 优化导入  
  Ctrl + Alt + I 自动缩进  
  Tab / Shift + Tab 缩进、不缩进当前行  
  Ctrl+X/Shift+Delete 剪切当前行或选定的代码块到剪贴板  
  Ctrl+C/Ctrl+Insert 复制当前行或选定的代码块到剪贴板  
  Ctrl+V/Shift+Insert 从剪贴板粘贴  
  Ctrl + Shift + V 从最近的缓冲区粘贴  
  Ctrl + D 复制选定的区域或行  
  Ctrl + Y 删除选定的行  
  Ctrl + Shift + J 添加智能线  
  Ctrl + Enter 智能线切割  
  Shift + Enter 另起一行  
  Ctrl + Shift + U 在选定的区域或代码块间切换  
  Ctrl + Delete 删除到字符结束  
  Ctrl + Backspace 删除到字符开始  
  Ctrl + Numpad+/- 展开折叠代码块  
  Ctrl + Numpad+ 全部展开  
  Ctrl + Numpad- 全部折叠  
  Ctrl + F4 关闭运行的选项卡

2、查找/替换(Search/Replace)  
F3 下一个  
Shift + F3 前一个  
Ctrl + R 替换  
Ctrl + Shift + F 全局查找  
Ctrl + Shift + R 全局替换

3、运行(Running)  
Alt + Shift + F10 运行模式配置  
Alt + Shift + F9 调试模式配置  
Shift + F10 运行  
Shift + F9 调试  
Ctrl + Shift + F10 运行编辑器配置  
Ctrl + Alt + R 运行manage.py任务

4、调试(Debugging)  
F8 跳过  
F7 进入  
Shift + F8 退出  
Alt + F9 运行游标  
Alt + F8 验证表达式  
Ctrl + Alt + F8 快速验证表达式  
F9 恢复程序  
Ctrl + F8 断点开关  
Ctrl + Shift + F8 查看断点

5、导航(Navigation)  
Ctrl + N 跳转到类  
Ctrl + Shift + N 跳转到符号  
Alt + Right/Left 跳转到下一个、前一个编辑的选项卡  
F12 回到先前的工具窗口  
Esc 从工具窗口回到编辑窗口  
Shift + Esc 隐藏运行的、最近运行的窗口  
Ctrl + Shift + F4 关闭主动运行的选项卡  
Ctrl + G 查看当前行号、字符号  
Ctrl + E 当前文件弹出  
Ctrl+Alt+Left/Right 后退、前进  
Ctrl+Shift+Backspace 导航到最近编辑区域  
Alt + F1 查找当前文件或标识  
Ctrl+B / Ctrl+Click 跳转到声明  
Ctrl + Alt + B 跳转到实现  
Ctrl + Shift + I查看快速定义  
Ctrl + Shift + B跳转到类型声明  
Ctrl + U跳转到父方法、父类  
Alt + Up/Down跳转到上一个、下一个方法  
Ctrl + ]/[跳转到代码块结束、开始  
Ctrl + F12弹出文件结构  
Ctrl + H类型层次结构  
Ctrl + Shift + H方法层次结构  
Ctrl + Alt + H调用层次结构  
F2 / Shift + F2下一条、前一条高亮的错误  
F4 / Ctrl + Enter编辑资源、查看资源  
Alt + Home显示导航条F11书签开关  
Ctrl + Shift + F11书签助记开关  
Ctrl + #[0-9]跳转到标识的书签  
Shift + F11显示书签

6、搜索相关(Usage Search)  
Alt + F7/Ctrl + F7文件中查询用法  
Ctrl + Shift + F7文件中用法高亮显示  
Ctrl + Alt + F7显示用法

7、重构(Refactoring)  
F5复制F6剪切  
Alt + Delete安全删除  
Shift + F6重命名  
Ctrl + F6更改签名  
Ctrl + Alt + N内联  
Ctrl + Alt + M提取方法  
Ctrl + Alt + V提取属性  
Ctrl + Alt + F提取字段  
Ctrl + Alt + C提取常量  
Ctrl + Alt + P提取参数

8、控制VCS/Local History  
Ctrl + K提交项目  
Ctrl + T更新项目  
Alt + Shift + C查看最近的变化  
Alt + BackQuote(’)VCS快速弹出

9、模版(Live Templates)  
Ctrl + Alt + J当前行使用模版  
Ctrl +Ｊ插入模版

10、基本(General)  
Alt + #[0-9]打开相应的工具窗口  
Ctrl + Alt + Y同步  
Ctrl + Shift + F12最大化编辑开关  
Alt + Shift + F添加到最喜欢  
Alt + Shift + I根据配置检查当前文件  
Ctrl + BackQuote(’)快速切换当前计划  
Ctrl + Alt + S　打开设置页  
Ctrl + Shift + A查找编辑器里所有的动作  
Ctrl + Tab在窗口间进行切换