**Python学习和开发环境的建立**

**东南大学能源与环境学院 程懋华**

“工欲善其事，必先利其器”，软件开发少不了软件工具。首先，会用，纸上谈兵和软件开发无缘，然后，擅用，高效学习和工作

Python解释器、标准库及扩展包构成Python的基本开发环境，可以满足学习和软件开发的基本要求。Python解释器及标准库可从Python官网下载；其他扩展包可用pip命令等方法安装。

较大些的Python软件开发宜采用集成开发环境，有: Eclipse、MS Visual Studio、PyCharm等，其中，Eclipse加PyDev插件的环境，为较多Python程序开发人员使用。此外，还有使用文本编辑器(如Visual Studio Code)构建开发环境的方法。

本文档给出了Windows操作系统下Python开发环境的建立过程，是同学们动手实践的入门文档。

在同学们本文档开始实践之前，和大家一起读下人民教育家陶行知先生的一句话：

**行是知之始，知是行之成**

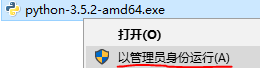
开发环境建立过程中，需要从网络上下载软件。注意：**务必从官网下载软件**，以免下载了被“污染”了的软件。

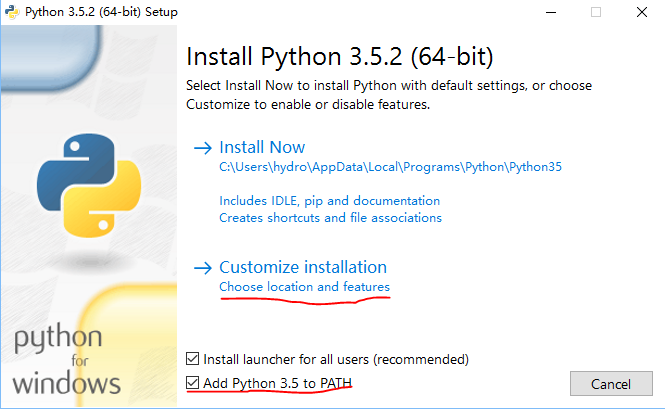
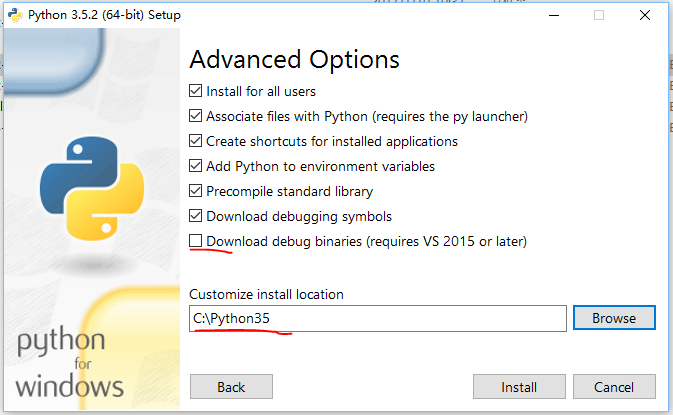
1. **Python基本开发环境**

**1.1 安装Python软件**

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。建议安装次新版（在语言新特征和软件包兼容性之间取一个较好的平衡），2017年初的次新版是Python3.5.2。Windows10 64位操作系统下，安装要点如下（**使用定制方式安装**）：

1. 初次安装时，电脑务必**联网**；使用**管理员权限运**行安装软件
2. 勾选“**Add Python3.5 to Path**” 将Python3.5的安装目录添加到Windows系统的环境变量PATH中；
3. 点“Customize installation”,进如下一步，勾选“Advanced options”项目中**除“Downloaded debug binaries”以外**的所有项目；
4. **不要使用默认安装目录，自定义目录（目录名带主版本号**）。现在安装Python3.5.2版，目录为：C:\Python35（注意：Python的安装目录可以自定义任意名称，因为可在一台机器上安装多个Python版本，然后，选择使用某个版本，因此，目录名称中含版本号为宜）

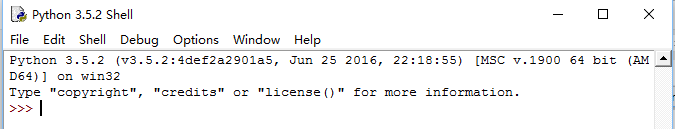


注意：python3.X安装过程中,会有进度条不动的现象。这时，先耐心等待，如果长时间不动，检查联网情况，再次尝试安装。如果还是失败，建议从Python官网zip压缩包，然后，把那个压缩包解压到电脑指定的目录，然后手动将这个目录添加到系统的环境变量PATH。

**1.2 使用IDLE编程**

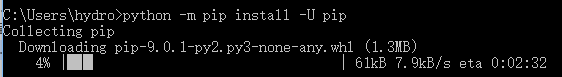
IDLE是Python软件包提供的轻量级集成编程环境：



1. **Python扩展软件包安装**

pip是在线安装Python软件包的软件。安装好Python后，DOS命令行下, 在线检查、更新pip到最新版本（电脑联网）：

>python -m pip install -U pip



在DOS命令行下,执行：

>pip install 软件包名

即可在线安装软件包，但有些软件包使用pip安装时，需要本地编译，如果本地不具备编译条件，安装会失败。这时可使用，加州大学欧文分校提供的Windows32/64位Python扩展包编译版，

<http://www.lfd.uci.edu/~gohlke/pythonlibs/>

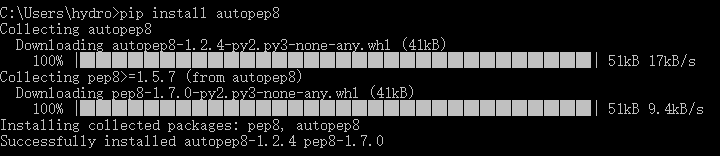
加州大学欧文分校的编译版软件包，内容丰富、更新及时、安装方便。

**2.1 编码规范**

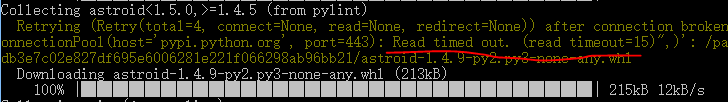
好的软件源码需遵循一定的约定，Python语言将这些约定总结为PEP8，可以使用软件包，检查修改源码，使其满足PEP8的要求，提高源码质量。

>pip install autopep8

>pip install pylint



软件包在线安装过程中，可能下载软件比较慢，提示：Read timed out（下载软件超时），这时安装过程，会自动进行再次下载尝试。。如果，多次尝试不成功，安装过程会失败。等待网络环境好转时，再次执行pip安装。



**2.2 科学计算软件包**

**2.2.1 科学计算基本包**

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

numpy

scipy

matplotlib

Scipy 软件包依赖一些C/Fortran软件库。在线安装时，会下载这些库的源码，到本地编译，但Windows环境下通常缺少这些源码的本地编译环境，因此，Windows环境下安装Scipy 软件包，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 {filename}.whl

**2.2.2 科学计算可选包**

Pandas: 基于 Numpy 构建的含有更高级数据结构和数据分析；

Scikit-learn: 机器学习软件包

Numba: 计算加速包；

安装Numba前，需要预先安装llvmlite，Windows 平台下需要安装VS2015发布包。

**2.2.3 IF97水和水蒸汽物性计算**

**2.2.3.1 SEUIF97水和水蒸汽物性计算**

基于C语言实现共享库Python的封装包:

<https://github.com/Py03013052/SEUIF97>

下载仓库的zip文件，解压，将和操作系统对应版本的：SEUIF97.dll和 seuif97.py：

1) SEUIF97.dll拷贝到 c:\windows\system

2) seuif97.py拷贝到 c:\python35\Lib

**2.2.3.2 IAPWS水和水蒸汽物性计算**

Python语言实现的计算包：

>pip install iapws

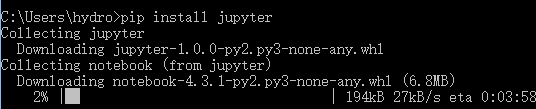
1. **交互计算 Jupyter**

**3.1 安装Jupyter环境包**

在线安装Jupyter ：

>pip install jupyter

Jupyter安装过程中，需要安装很多软件包，易出现超时现象，耐心等待。



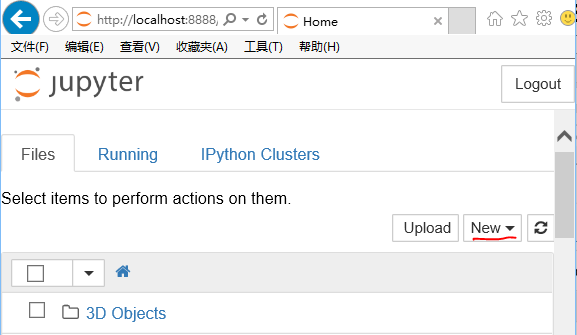
如果某些软件包，在线一直难以安装，建议从加州大学欧文分校下载相应软件包的whl,然后，用pip安装whl.

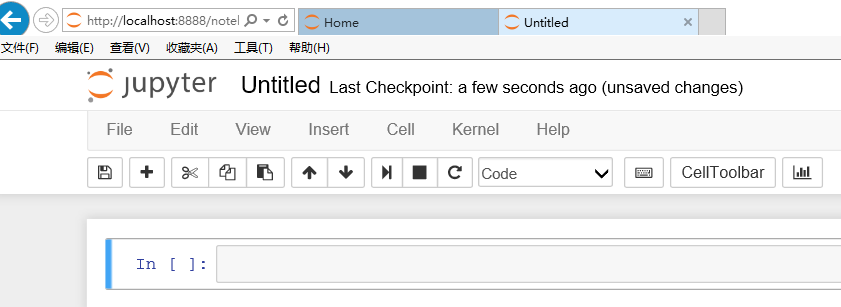
**3.2 创建Jupyter Notebook文件**

DOS命令行执行：

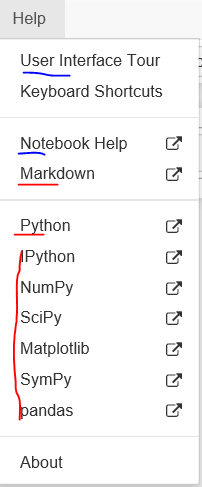
>jupyter notebook

启动jupyter notebook服务，服务会打开浏览器，供编写jupyter notebook, 点“New”, 新建一个jupyter notebook 文件。





点“help”可见Jupyter Notebook、、Makedown、Python、科学计算软件包等在线帮助链接，点击可在线查看相关帮助。



1. **基于Eclipse的Python开发环境**

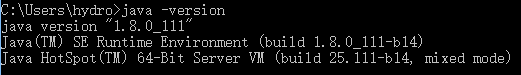
**4.1 安装和配置Eclipse**

**4.1.1安装Java SDK**

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 ,会显示有关版本信息：



如果没有安装Java包 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-i**64**.exe.

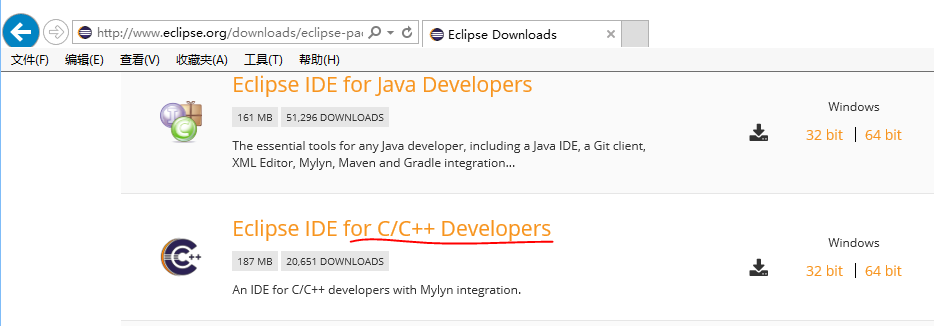
运行下载Java安装包（安装时注意**配置环境变量**），即可安装好Java环境。

**4.1.2 安装Eclipse**

Eclipse IDE是Eclipse基金会开发的插件型集成开发环境，Eclipse基金会在：<http://www.eclipse.org/users/> ，提供了大量资源。安装和使用中的问题，都可以在其中找到相应的帮助文档。

Eclipse IDE有很多版本。本课程使用Python语言，可能会涉及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/eclipse-packages/



根据操作系统32/64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,点“固定到任务栏”即可）





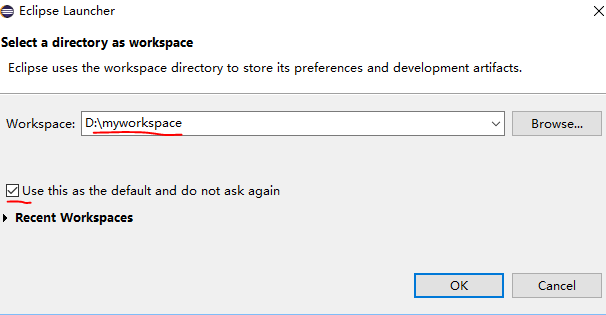
从任务栏启动Eclipse，工作起来方便很多（同样也可将IDLE固定到任务栏）。

**4.1.3 配置Eclipse工作空间**

首次打开Eclipse软件时,软件会提示设置默认的工作空间（就是存放所有开发项目信息的目录）。

**注意：不要使用Eclipse软件默认的工作空间目录**。

建议配置自己的工作空间：首先，在非系统盘中，如D盘中建立一个目录，如:D:\myworkspace，然后，打开Eclipse，在其提示设置默认的工作空间时，“Browser”到D:\myworkspace，勾选“Use this as the default and do not ask again",就将该目录配置为Eclipse默认当前工作空间目录了，你开发的程序代码等都在这个目录下。



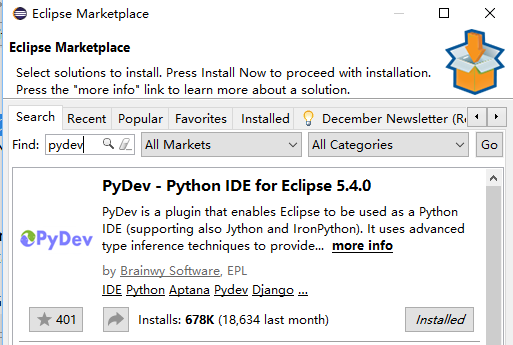
可以根据工作需要建立多个工作空间目录，通过“File->Switch Workspace->other”改变你的当前工作空间目录。

**4.2 Python开发插件PyDev**

使用Eclipse IDE作为Python开发环境，需要：1）安装PyDev插件；2）配置使用的Python解释器版本。

**4.2.1 安装PyDev插件**

通过Help->Eclipse Marketplaces进入市场，输入pydev，找到Pydev项目，点“install”在线安装。



在线安装过程简单，但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.

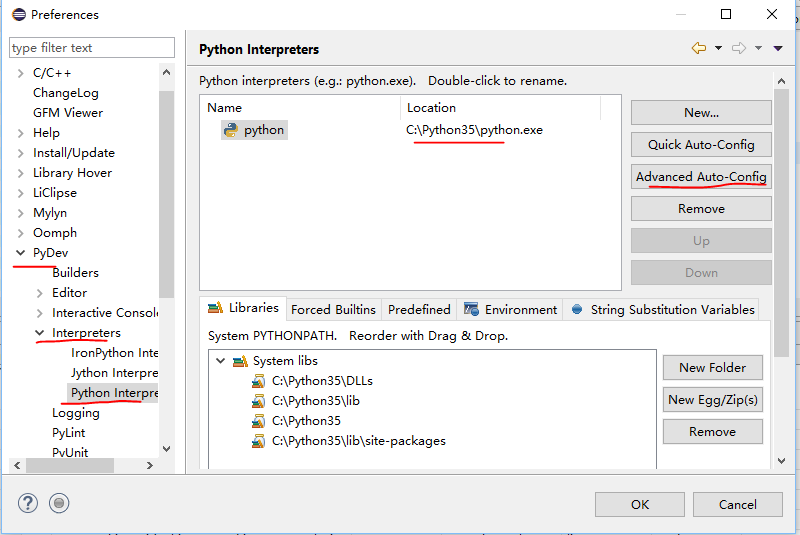


正常情况，Eclipse会自动解压，安装好插件。如解压中异常，可手工解压插件包，将解压后的所有文件，拷贝到Eclipse的插件目录plugins下。

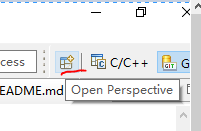
**4.2.2 配置 Python解释器**

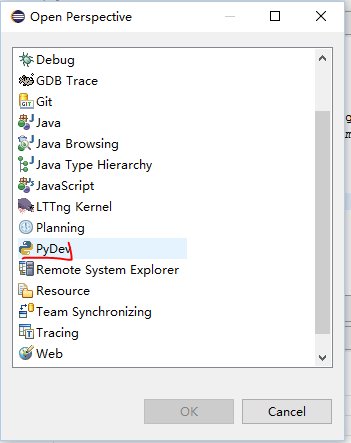
安装好后，重新启动。通过：

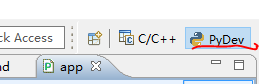
Windows->Preference->Pydev->Interperters->Python Interperter点其中的：Advanced Auto-config配置开发使用的 Python解释器版本:



配置好后，添加Python场景到工作页面：







就可以进行Python开发了。

1. **配置 PyDev提高效率**

**5.1 修改编辑器配色提高可读行**

从Window → Preferences→PyDev→Editor,进入配置界面：





**5.2 配置任务标签**

程序开发中，可在代码中标识当前任务状态，计划开发工作。使用“任务标签”在代码中标识任务，然后，让开发环境将其识别出来，加入工作空间的任务列表中。

1. **配置任务标签：**

通过PyDev->Task Tags中配置任务标签:



可以根据不同的任务类型，自己增加自定义标签

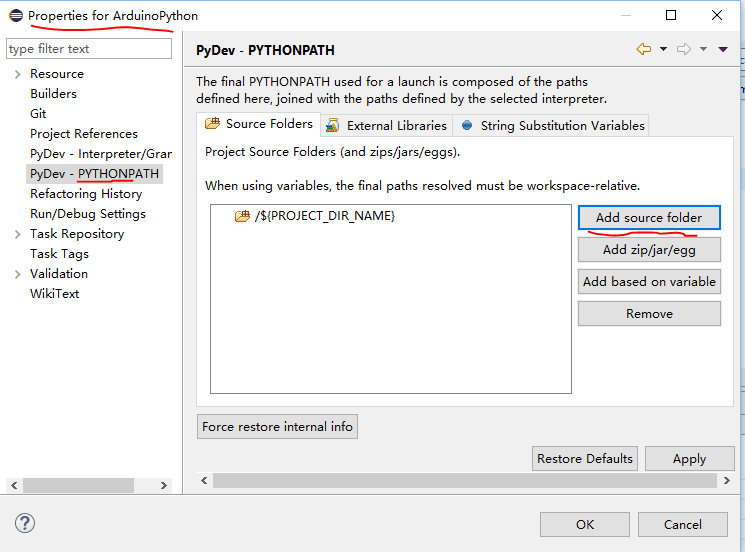
1. **添加任务标签注释**

在程序源码中加入使用任务标签的注释（是注释，但是使用了“任务标签”关键词开头，PyDev用任务标签作为关键字，检索生成任务列表的注释）：



1. **自动识别任务**

需要配置当前工程的PyDev-PYTHONPATH的source folder。在源码工程的Properties->PyDev-PYTHONPATH配置项中，点“Add source folder”将源码目录加入。



保存新修改或运行程序一次（Ctrl+b）或选择 Project → Clean：

就可将使用任务标签注释加入任务列表,相应的任务注释行左边会有标识#加任务标签关键字所在行为任务标识行。



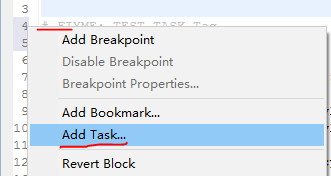
如果当前任务窗口包含“Tasks”，刚加的任务，就会立刻显示在“Tasks”窗口中。



如“Tasks”窗口没有打开，可Window->Show View->Tasks显示任务窗口。

1. **手动添加任务**

在PyDev 环境下，也可手动添加任务。将鼠标移动到需要添加为任务行的最左侧，点右键，选择“Add Task”,即将该行加入任务列表。



1. **基于PyDev的语言规范静态检查**

PyDev中集成了PEP8、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，要有代码规范意识，但过分注意规范会影响开发进程。如果一直开启代码规范检查，经常提示不规范，会对开发形成负面影响，所以，开发进程中默认关闭，在程序开发一个阶段结束时，开启规范性检查检查更好。

**6.1 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?

|  |
| --- |
|  |

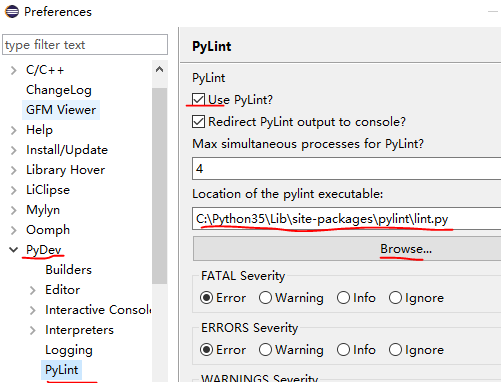
在Python源码窗口，按 CTRL-SHIFT-F 就可以自动修改代码

**6.2 使用Pylint**

PyDev 默认不开启Pylint。通过

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

“Browse…”找到安装好的lint.py的地址,例如"C:\Python36\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. **Eclipse配置和使用**

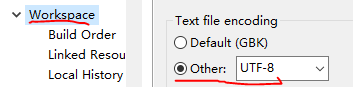
**7.1 文件字符编码方式**

中文操作系统下，Eclipse默认工作空间编码方式为GBK, 这样的编码方式下，含有中文支付的文件，在其他文本编辑器打开会乱码，因此需要修改为UTF-8。编码方式可以设定到：工作空间、工程和文件类型的不同等级的文件范围上。

建议配置编码方式UTF-8到工作空间，工程继承工作空间配置的编码方式。

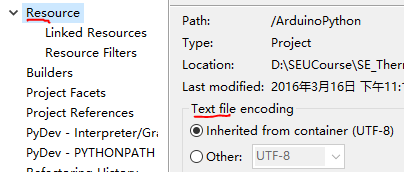
**7.1.1 工作空间编码方式**

点“Window”->“Preference”，在弹出窗口中，点击“general”-“workspace”，修改“Text file encoding”为UTF-8：



**7.1.2 工程编码方式**

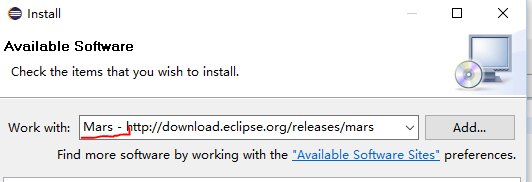
将鼠标移动到项目名上，点击右键，选择“properties”， 在弹出的对话框中 ，选中“resources”， 修改“Text file encoding”为UTF-8(如果工作空间配置为UTF-8会继承过来)：



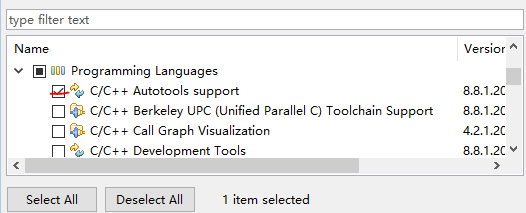
**7.2 插件补充和更新**

以Eclipse当前版本插件为例说明插件的补充和更新：

电脑联网，然后**，Help->Install New Software，**进入安装软件页面，选择插件的更新网址（图中是Eclipse的Mars版，如是其他版本，mars 对应的其版本名）：



稍后可看到插件列表，勾选需要安装的插件，按照要求继续下面步骤即可安装好指定的插件：



**7.3 联机指南**

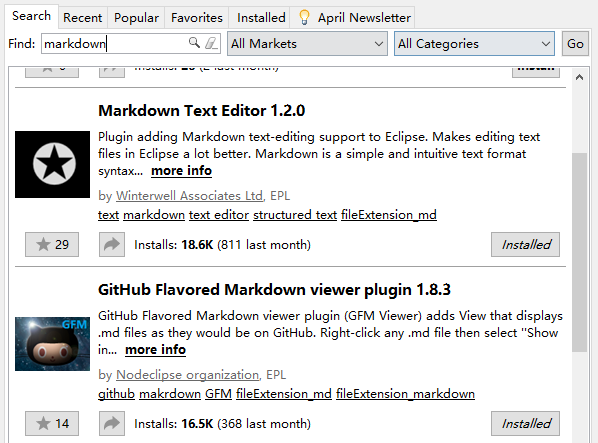
Eclipse的功能很多，软件使用中有疑问可以查询Eclipse内置的help：

**Help** -> **Help Contents**



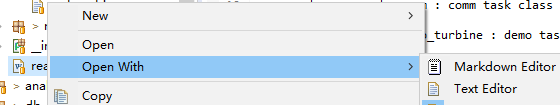
1. **Markdown插件**

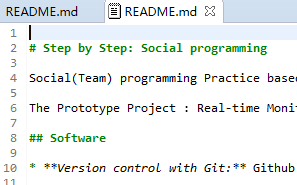
Eclipse默认安装对Markdown支持较弱，需要安装Markdown插件。Help->Eclipse Marketplace使用Markdown关键字可以找到2个插件：编辑和显示：



**8.1 Markdown Editor插件**

Eclipse Marketplace 市场安装，安装后选择其作为md文件的编辑器，即可：





Markdown Editor插件可Windows->Preferences->General->Editors->Text Editors->Markdown配置有关参数。

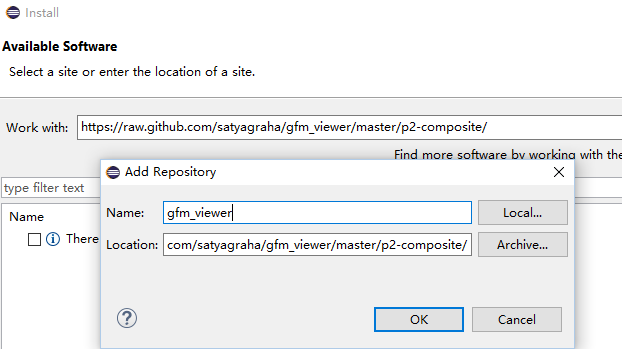
**8.2 GitHub Flavored Markdown Viewer插件**

GitHub Flavored Markdown Viewer插件源码仓库：

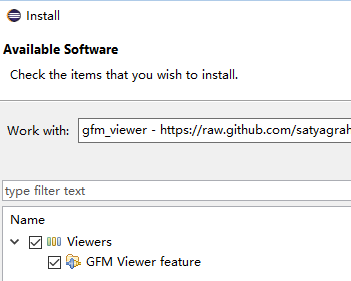
<https://github.com/satyagraha/gfm_viewer>。

其中,有插件安装地址，也可：Help -> Install New Software，拷贝插件地址到Work with:

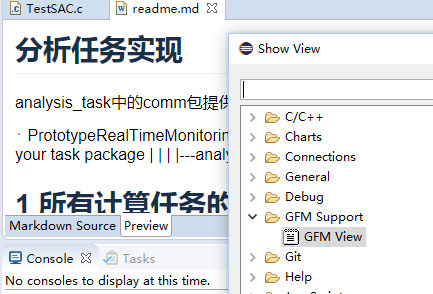
https://raw.github.com/satyagraha/gfm\_viewer/master/p2-composite/



中选择，安装Viewers插件：



安装后，Window -> Show View -> Other... and select entry GFM Support -> GFM Viewer



开启GFM Viewer窗口，点击Markdown文件就可以看到和GitHub上一样的显示效果了



**小结**

Python 学习和开发环境实现方式多种多样，没有标准答案。建议同学们，首先掌握最基本的IDLE，进一步用好Eclipse加PyDev集成开发环境。以后，根据自己的开发工作需要，选择和建立自己得心应手的开发环境。

最后再次重温人民教育家陶行知先生那句话：

**行是知之始，知是行之成**

希望同学们多实践！

**参考文献**

1. 陶行知. 中国教育的觉醒:陶行知文集(第1版).群言出版社,2013.06
2. The Eclipse Foundation. Eclipse: [Getting started](http://www.eclipse.org/users). http://www.eclipse.org/users/
3. The Eclipse Foundation. Eclipse IDE User Guide. <http://help.eclipse.org/neon/nav/0>
4. Brainwy Software Ltd. PyDev Manual. <http://www.pydev.org/manual_101_root.html>
5. 郑伟芳.PyDev for Eclipse简介. <http://www.ibm.com/developerworks/cn/opensource/os-cn-ecl-pydev/.2008.11>
6. 张颖. Python 代码调试技巧. <http://www.ibm.com/developerworks/cn/linux/l-cn-pythondebugger/,2012.05>
7. Microsoft. Python on Visual Studio Code. <https://code.visualstudio.com/docs/languages/python>