



# 实验1 Python环境搭建和运行

ANDYWWW

LAST MODIFIED: 25 FEB 2019



# Python的安装

## 1. 最基础的Python的安装 (<https://www.python.org/downloads/release/python-372/>)

Files					
Version	Operating System	Description	MD5 Sum	File Size	GPG
<a href="#">Gzipped source tarball</a>	Source release		02a75015f7cd845e27b85192bb0ca4cb	22897802	<a href="#">SIG</a>
<a href="#">XZ compressed source tarball</a>	Source release		df6ec36011808205beda239c72f947cb	17042320	<a href="#">SIG</a>
<a href="#">macOS 64-bit/32-bit installer</a>	Mac OS X	for Mac OS X 10.6 and later	d8ff07973bc9c009de80c269fd7efcca	34405674	<a href="#">SIG</a>
<a href="#">macOS 64-bit installer</a>	Mac OS X	for OS X 10.9 and later	0fc95e9f6d6b4881f3b499da338a9a80	27766090	<a href="#">SIG</a>
<a href="#">Windows help file</a>	Windows		941b7d6279c0d4060a927a65dcab88c4	8092167	<a href="#">SIG</a>
<a href="#">Windows x86-64 embeddable zip file</a>	Windows	for AMD64/EM64T/x64	f81568590bef56e5997e63b434664d58	7025085	<a href="#">SIG</a>
<a href="#">Windows x86-64 executable installer</a>	Windows	for AMD64/EM64T/x64	ff258093f0b3953c886192dec9f52763	26140976	<a href="#">SIG</a>
<a href="#">Windows x86-64 web-based installer</a>	Windows	for AMD64/EM64T/x64	8de2335249d84fe1eeb61ec25858bd82	1362888	<a href="#">SIG</a>
<a href="#">Windows x86 embeddable zip file</a>	Windows		26881045297dc1883a1d61baffeecafo	6533256	<a href="#">SIG</a>
<a href="#">Windows x86 executable installer</a>	Windows		38156b62c0cbcb03bfddeb86e66c3a0f	25365744	<a href="#">SIG</a>
<a href="#">Windows x86 web-based installer</a>	Windows		1e6c626514b72e21008f8cd53f945f10	1324648	<a href="#">SIG</a>

随便点一个下载即可

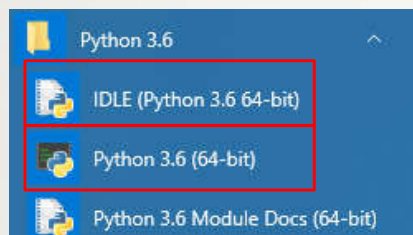


以第七个为例，  
下载后如图

# Python的安装

## 1. 最基础的Python的安装 (<https://www.python.org/downloads/release/python-364/>)

双击即可安装，注意将Python添加进入环境变量，安装后在开始菜单即可找到IDLE



IDLE

Python (In cmd)

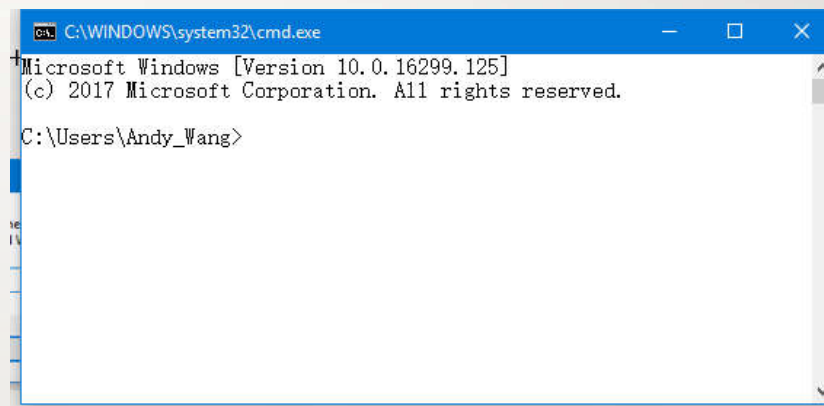
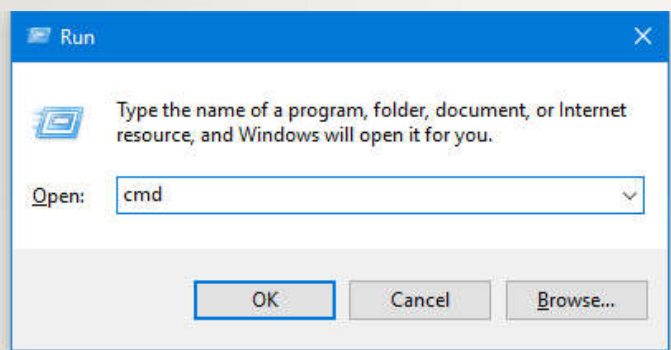
※ 不推荐使用官方自带IDLE，不便于对package进行管理，不容易Debug，虽然界面简单，但是很丑

# Python的安装

## 1. 最基础的Python的安装 (<https://www.python.org/downloads/release/python-364/>)

与之对应的Package管理工具为pip，需要在cmd（命令提示符）运行

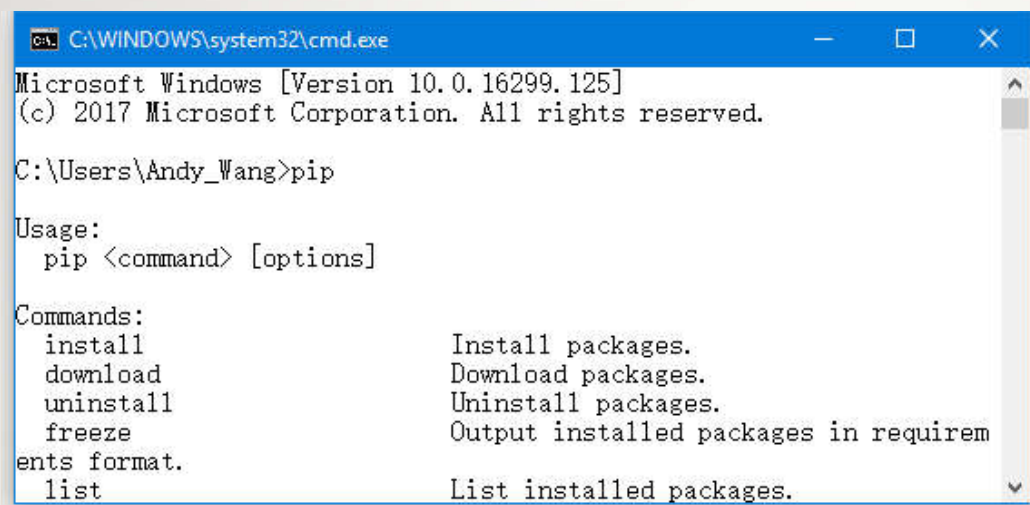
① 按下徽标+R组合键，调出运行提示框，输入cmd，点击确定，即可调出命令提示符（右图）



# Python的安装

1. 最基础的Python的安装 (<https://www.python.org/downloads/release/python-364/>)

② 如果成功安装了pip，输入pip后界面如图。



```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [Version 10.0.16299.125]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Users\Andy_Wang>pip

Usage:
  pip <command> [options]

Commands:
  install          Install packages.
  download         Download packages.
  uninstall        Uninstall packages.
  freeze           Output installed packages in requirem
ents format.
  list             List installed packages.
```

返回的即为帮助，比如可以使用pip  
install astropy安装astropy包，  
pip uninstall astropy卸载astropy包

# Python的安装

---

1. 最基础的Python的安装 (<https://www.python.org/downloads/release/python-364/>)

③ 如果没有安装pip, 在<https://pip.pypa.io/en/stable/installing/> 下载get-pip.py。

在cmd下将路径更改至下载路径, 或将get-pip.py复制至当前目录。

如何更改路径?

如果需要更改盘符, 只需要输入X: (其中X为需要改至的盘符)

如果需要在本地盘符内更改路径, 输入cd xxx (其中xxx为路径)

已经安装了Python或Anaconda提示Python不可识别如何解决?

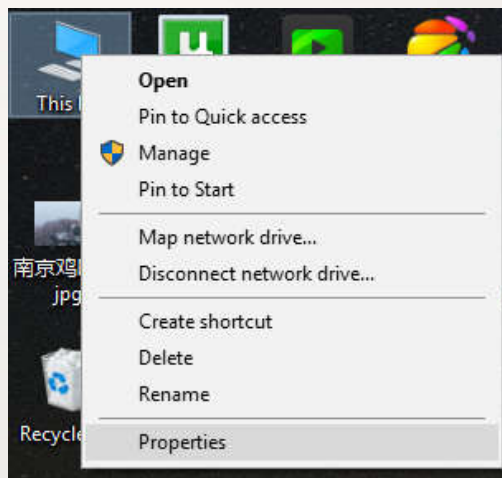
(转下页)

已经安装了Python或Anaconda提示Python不可识别如何解决？

---

对于输入的命令，系统会在环境变量\$PATH里的路径下寻找命令.exe是否存在，如果不存在就会返回这个错误，解决办法就是把命令.exe所在的路径添加到环境变量\$PATH内。（这里以Windows10为例，Linux和Mac有类似的命令行命令，比如在cshell下可以用setenv命令）

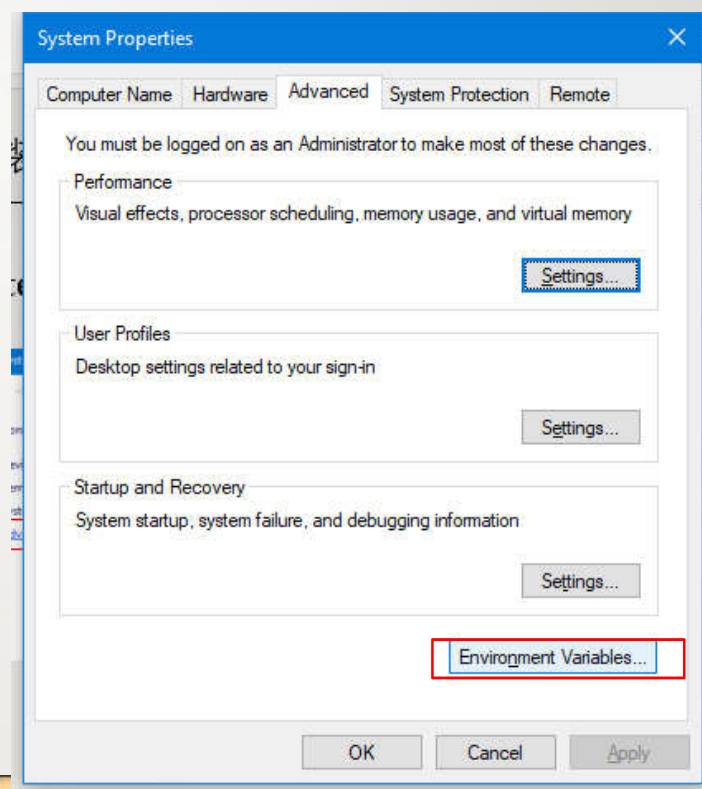
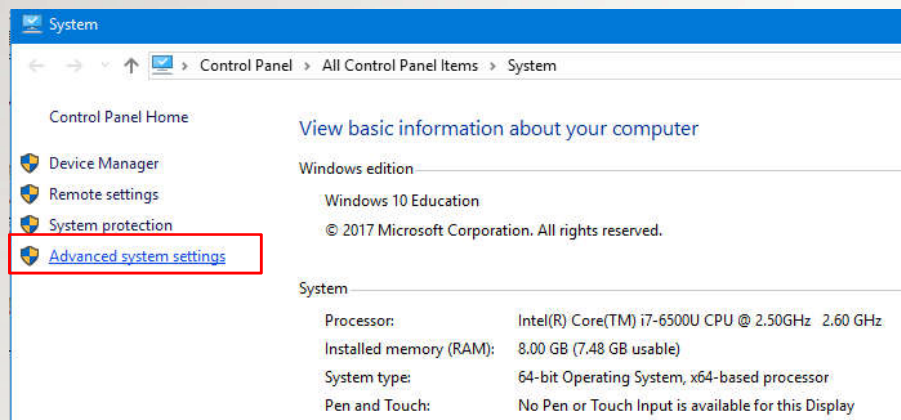
Step1 右键我的电脑—属性





已经安装了Python或Anaconda提示Python不可识别如何解决？

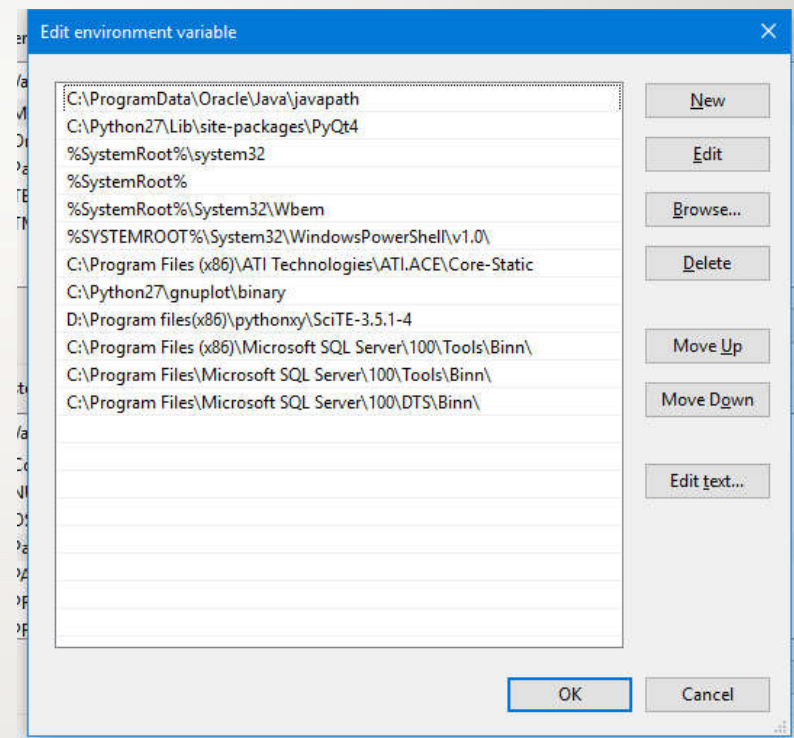
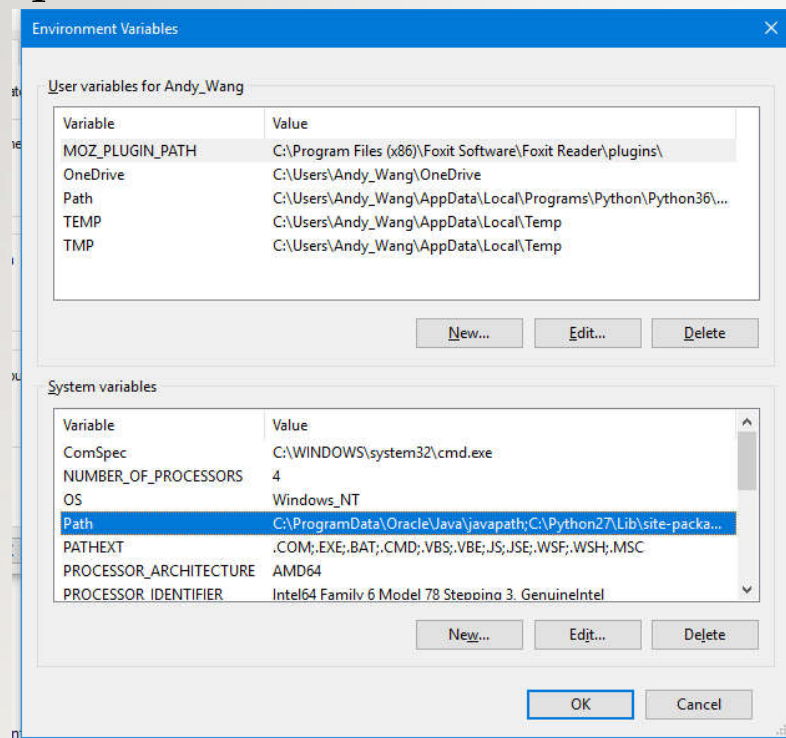
Step2 点击左侧高级系统设置，跳出系统属性选项卡，点击选环境变量。





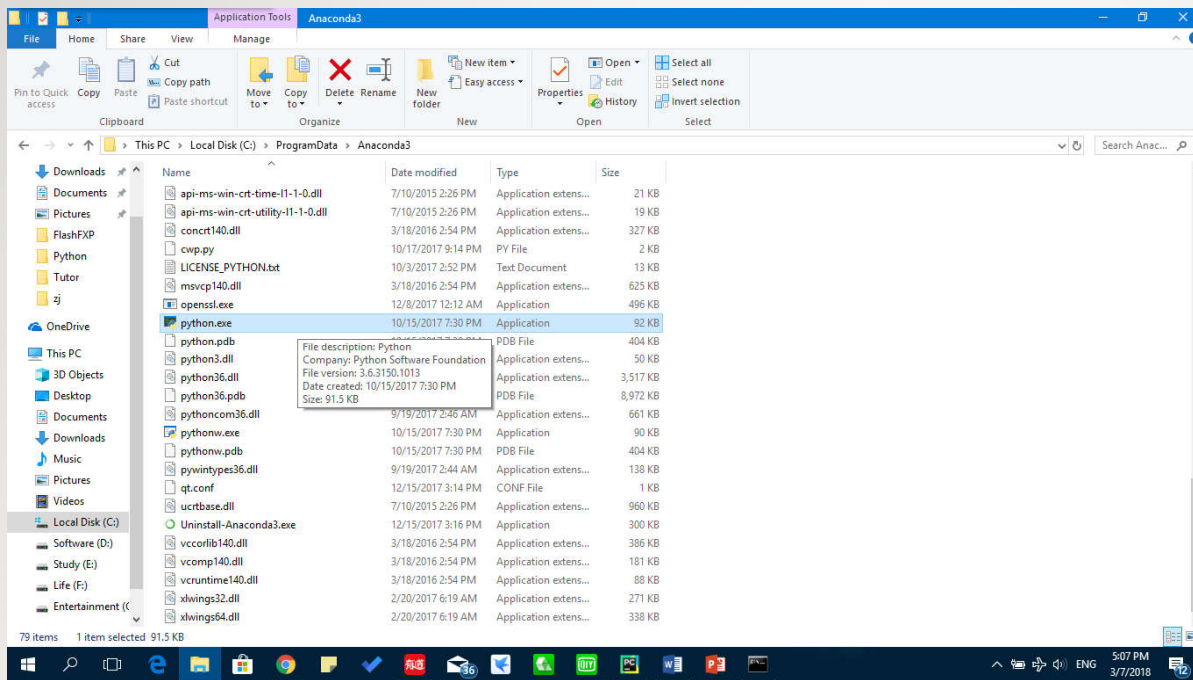
已经安装了Python或Anaconda提示Python不可识别如何解决？

Step3 选择系统变量里的Path，点击Edit...，点击New。



已经安装了Python或Anaconda提示Python不可识别如何解决？

Step4 输入python.exe所在的路径，疯狂点击确定即可。



例如，我需要调用Anaconda下安装的Python，其安装位置如图，则在环境变量里添加  
C:\ProgramData\Anaconda3  
即可

# Python的安装

---

## 2. Anaconda的安装

下载地址：

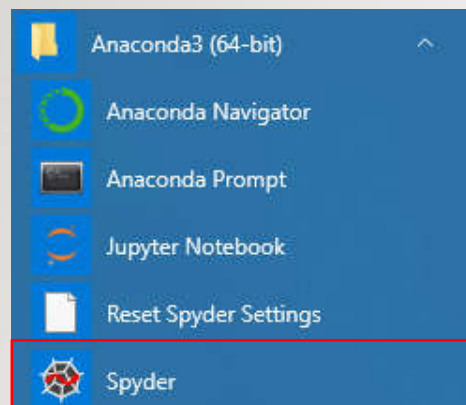
- ① 官网: <https://www.anaconda.com/download/>
- ② 学校镜像: <http://mirrors.nju.edu.cn/anaconda/archive/>

带有MacOSX的适配苹果电脑，带有Linux的适配Linux及其衍生系统，带有Windows的适配Windows系统。x86\_64为64位软件，仅带有x86的为32位软件。

双击即可安装，一定要等安装条到最后再关闭，否则会安装失败。

# Python的安装

## 2. Anaconda的安装



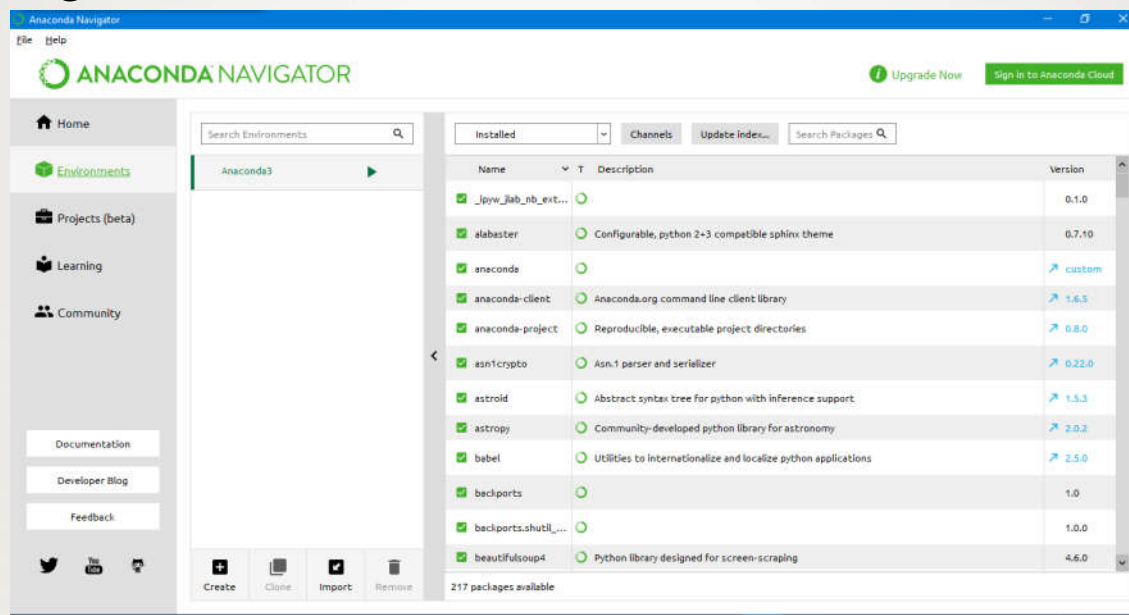
以后常用的Spyder

与pip相对应的，Anaconda的命令为conda用来进行包管理，如果没有将conda.exe所在路径设置至Path里，需要用Anaconda Prompt使用这个命令，其使用方法与pip相似。这里介绍可视化界面的方法。

# Python的安装

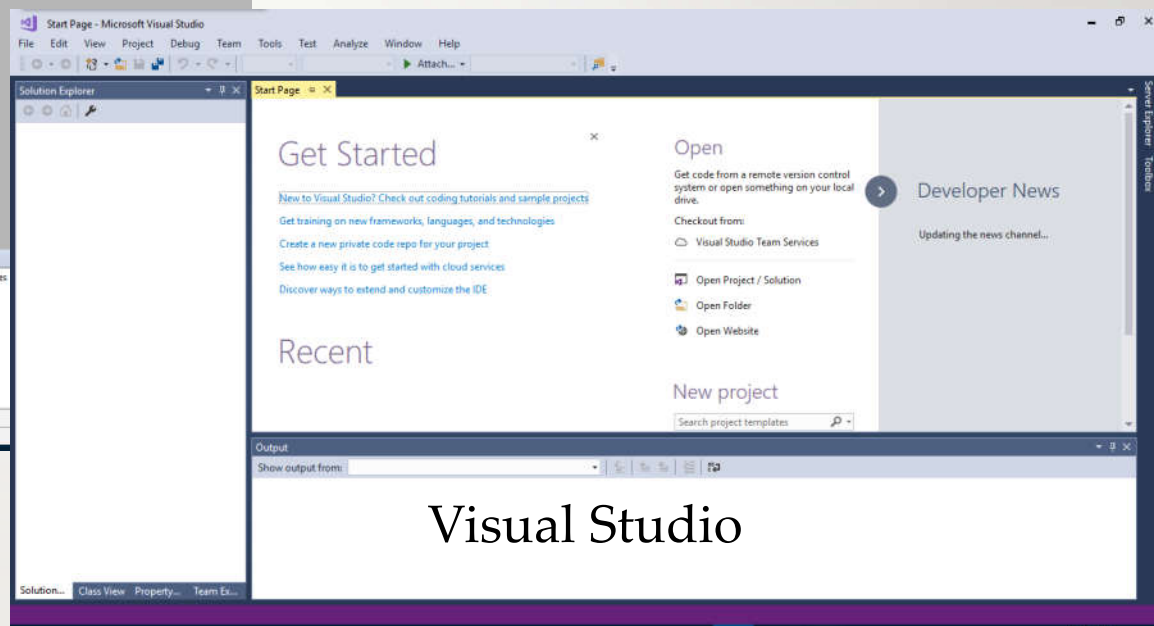
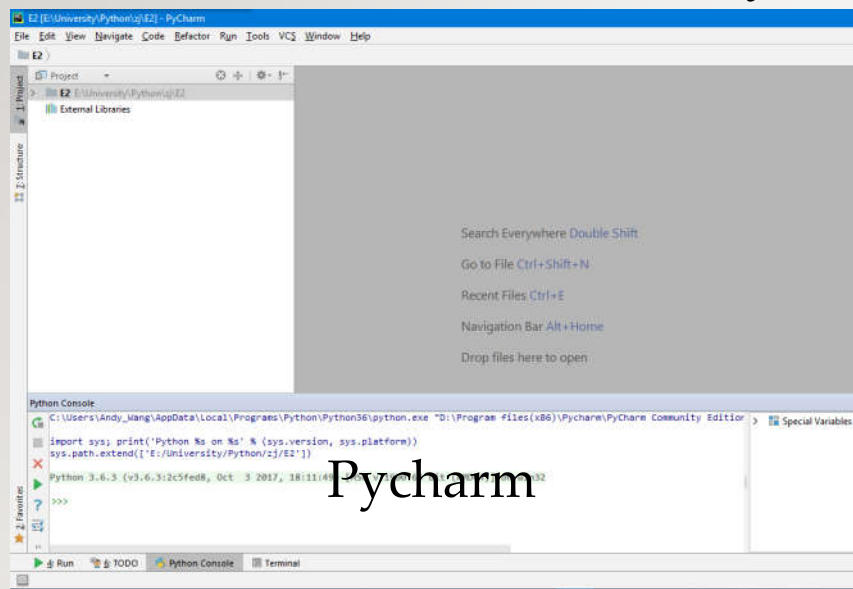
## 2. Anaconda的安装

打开Anaconda Navigator, 选择Environments, 在此界面下即可管理一些常用的Package。



# Python的安装

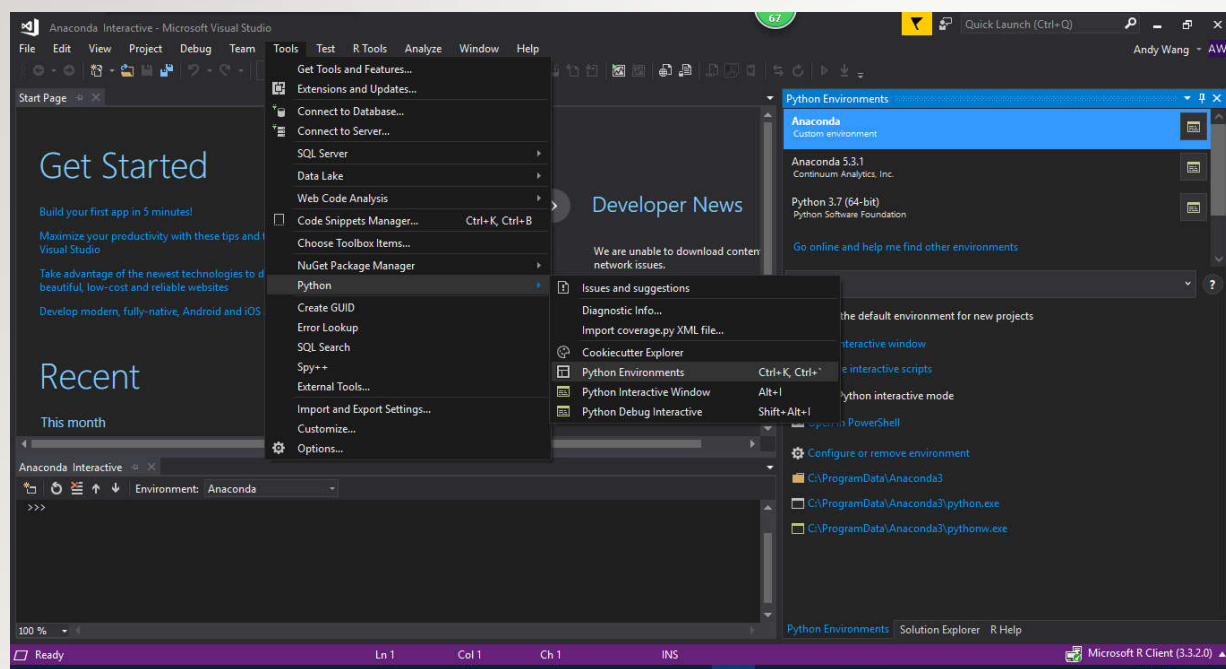
当然还有Visual Studio、Pycharm等，这里不再赘述。





# \*在Visual Studio 2017中配置Python环境

如果想要使用已经安装的Python编译器（安装visual studio 2017时也可以选择安装其自带的Anaconda）需要在VS2017中配置编译环境。

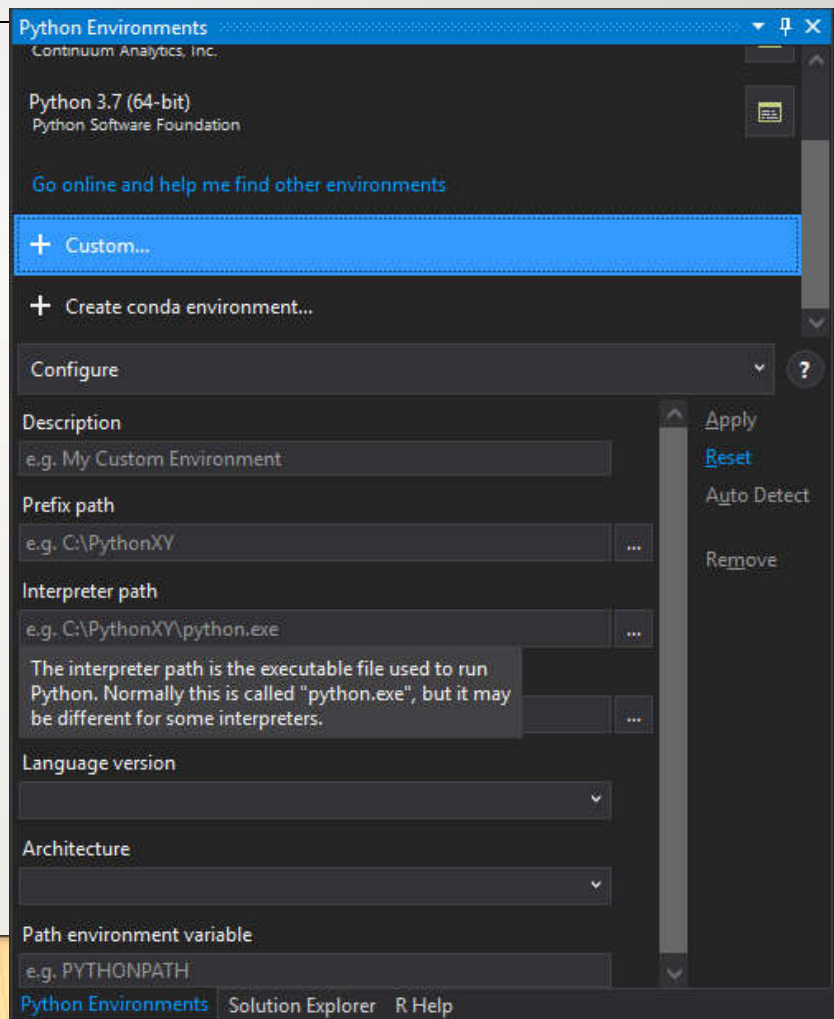
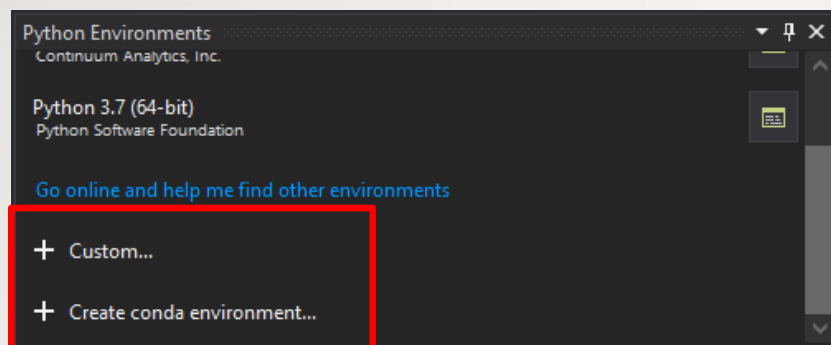


工具—Python—Python环境

这个时候右侧会出现Python环境的一个对话框

## \*在Visual Studio 2017中配置Python环境

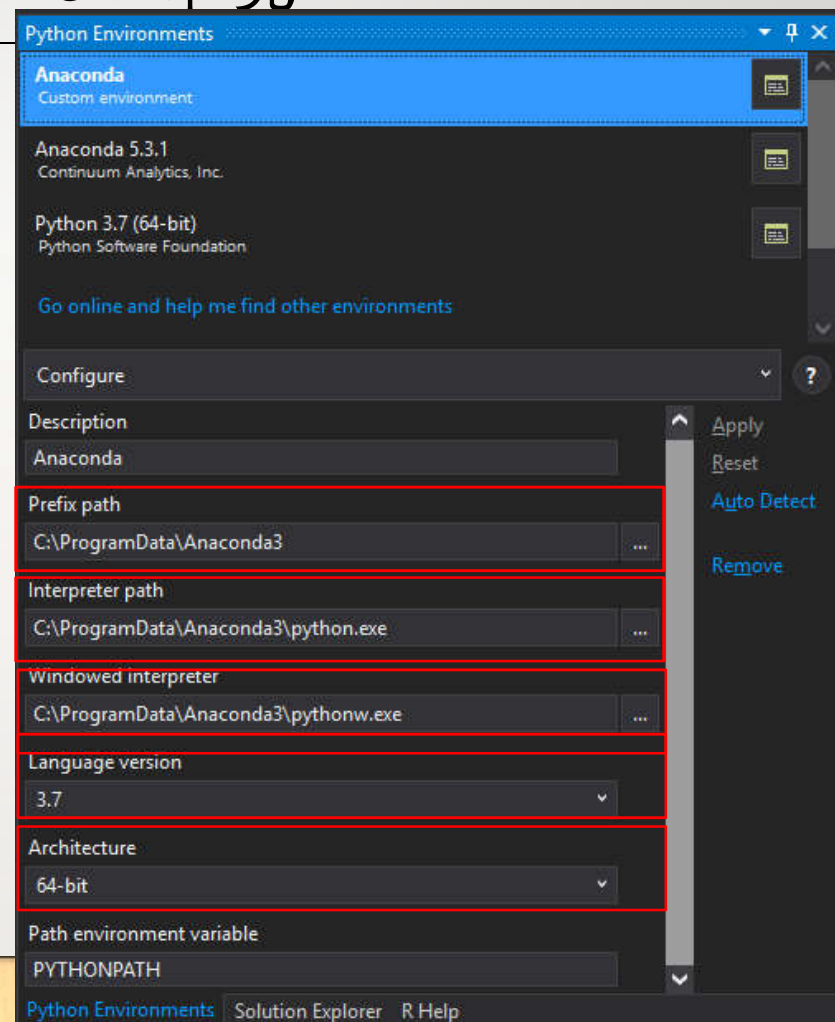
向下拉直到看到如下红框所示内容，  
点击“+Custom”，如右图。



## \*在Visual Studio 2017中配置Python环境

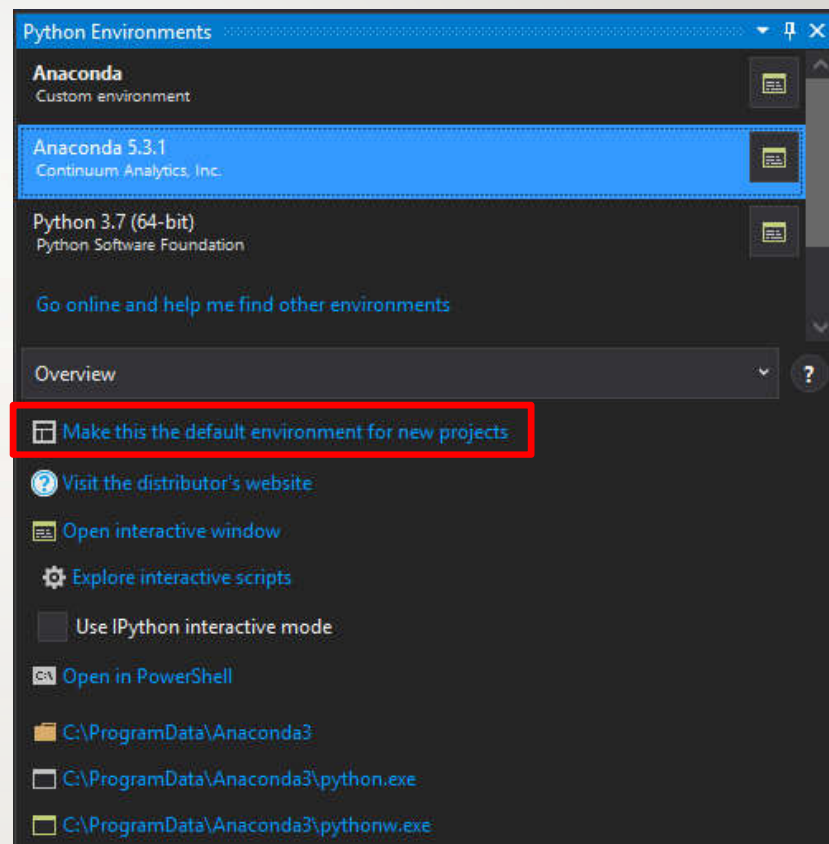
按如图所示进行配置（路径可能有所不同），注意一定要把Python中Package所在的路径加入到PYTHONPATH环境变量中（一般安装Anaconda时系统已经帮你做好）

点击Apply即可



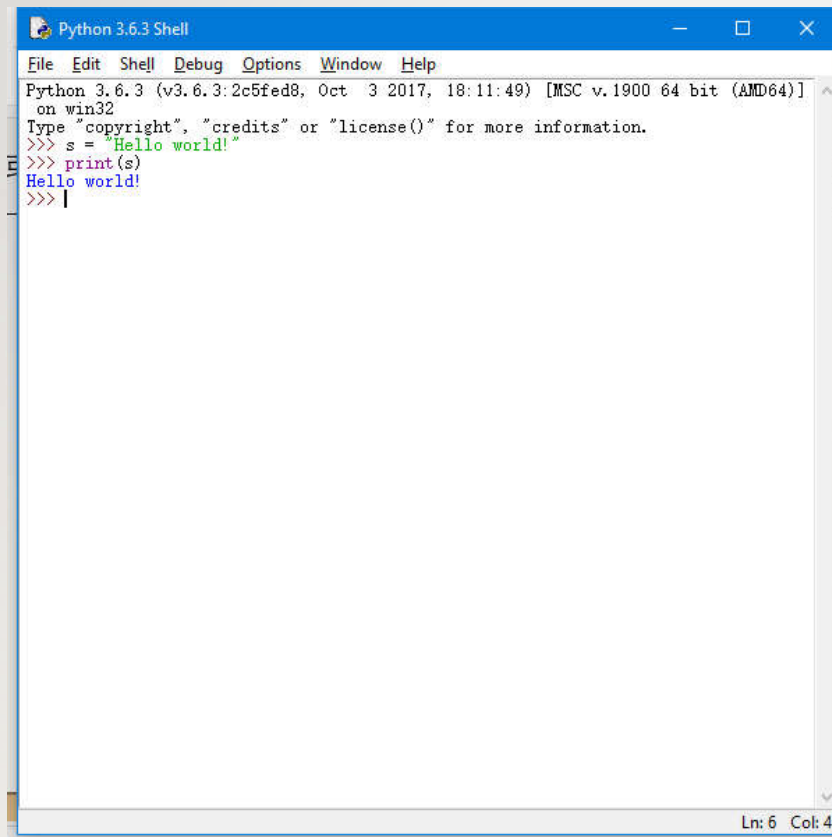
## \*在Visual Studio 2017中配置Python环境

随后，点击这个环境设置，点击右图中红框的按钮即可使用这个环境进行编译



1. 安装或直接使用Python IDLE并在Python Shell中输入如下两行代码并观察输出结果

---

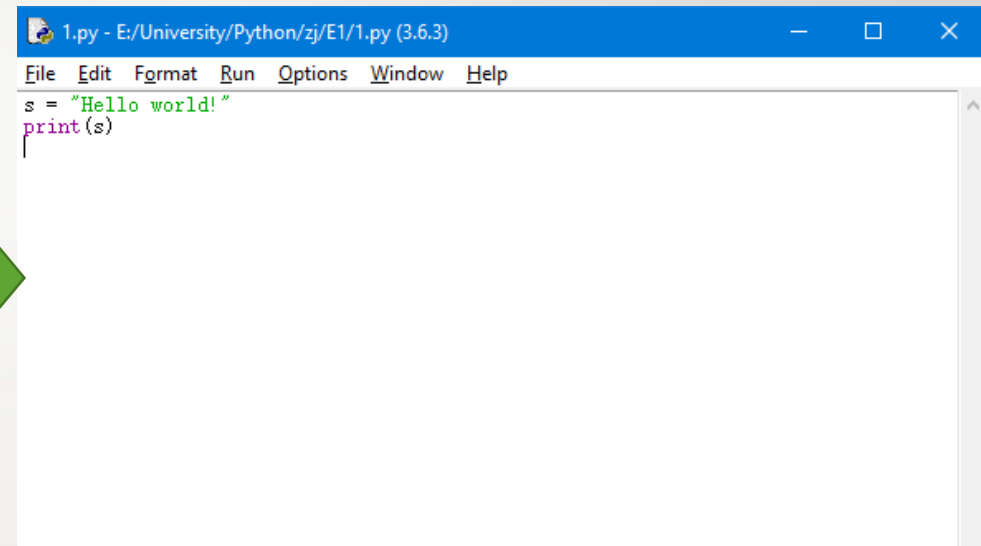
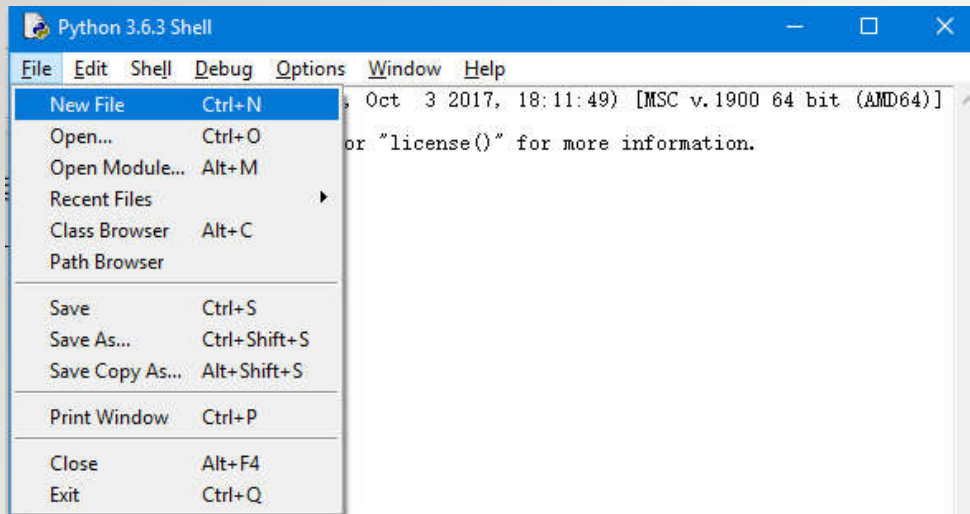


```
Python 3.6.3 Shell
File Edit Shell Debug Options Window Help
Python 3.6.3 (v3.6.3:2c5fed8, Oct 3 2017, 18:11:49) [MSC v.1900 64 bit (AMD64)]
on win32
Type "copyright", "credits" or "license()" for more information.
>>> s = "Hello world!"
>>> print(s)
Hello world!
>>> |
```

Ln: 6 Col: 4

直接在Shell中运行

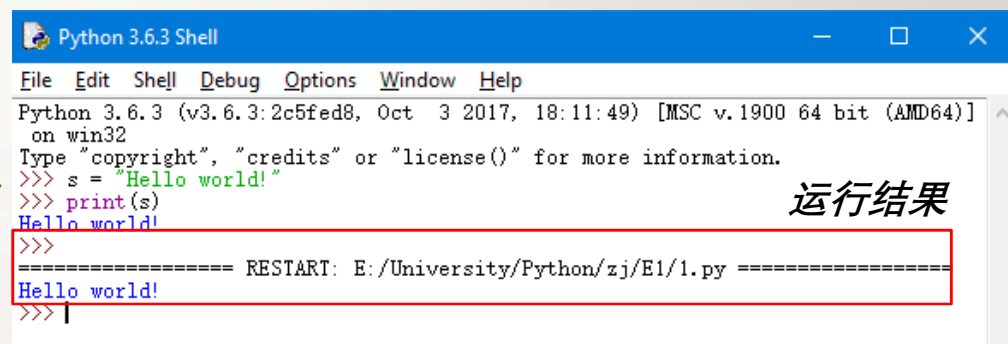
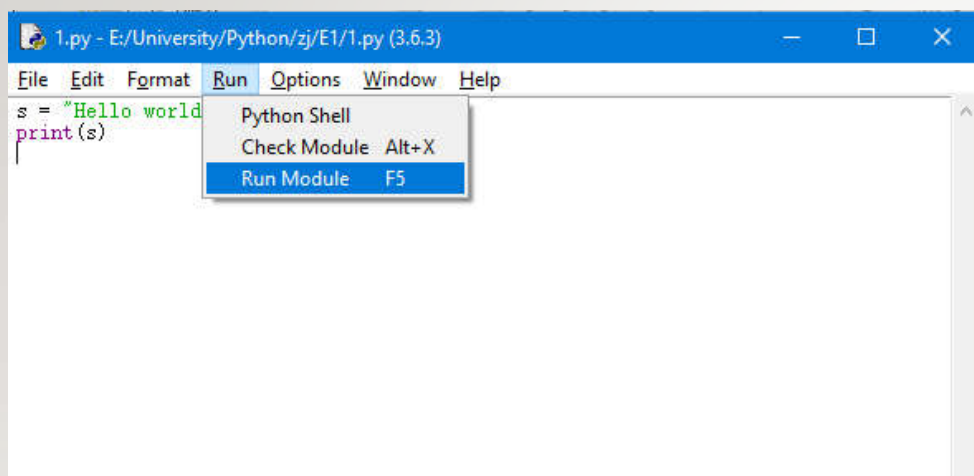
## 1. 安装或直接使用Python IDLE并在Python Shell中输入如下两行代码并观察输出结果



文件运行



## 1. 安装或直接使用Python IDLE并在Python Shell中输入如下两行代码并观察输出结果



文件运行

3. 安装Anaconda, 成功安装后打开Python IDE Spyder, 在其代码编辑窗口中输入如下代码进行测试

---

<code>import numpy as np</code>	导入numpy并简写为np,
<code>import matplotlib.pyplot as plt</code>	导入matplotlib.pyplot并简写为plt

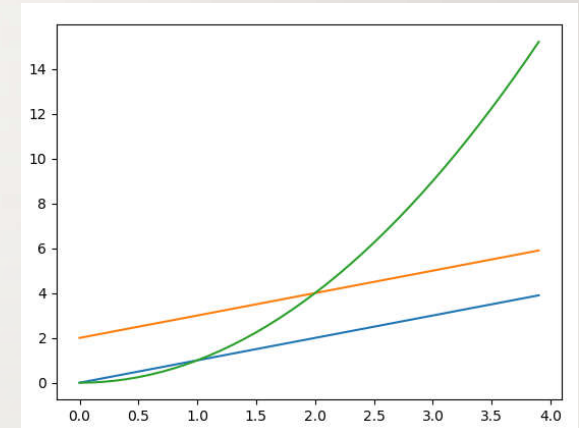
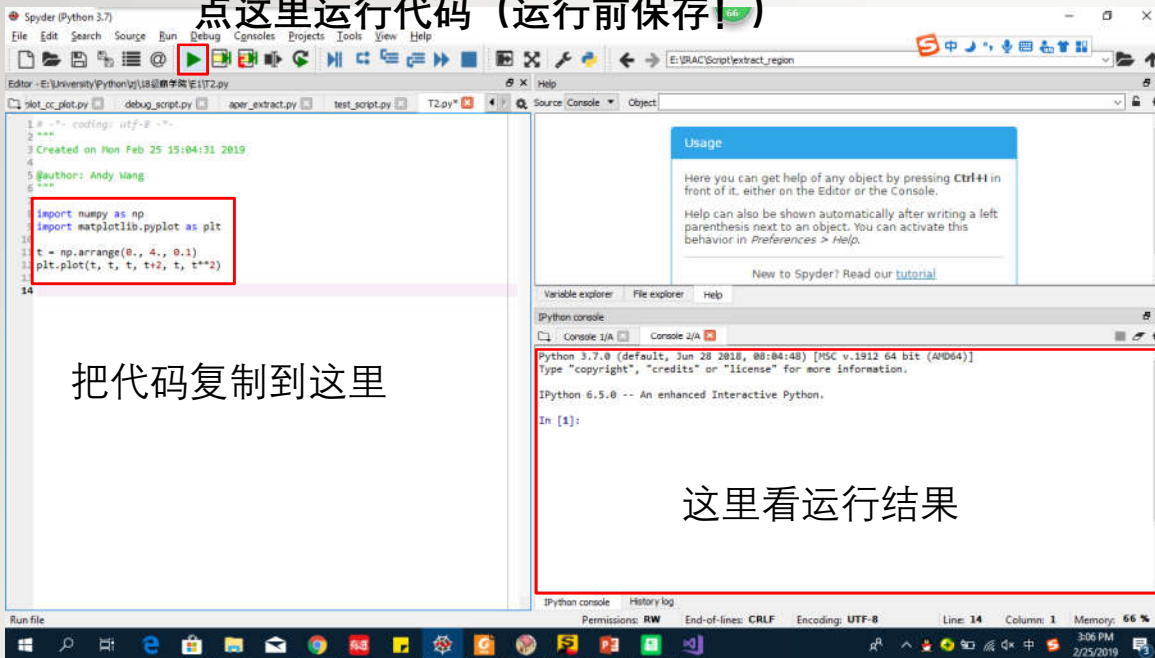
<code>t = np.arange(0., 4., 0.1)</code>	生成一个数组t, 始于0, 终于4 (不包括), 步长为0.1
<code>plt.plot(t, t, t, t+2, t, t**2)</code>	在x-y平面上绘制 $y=x$ , $y=x+2$ , $y=x^2$ 的图像, 注: 这里只是画了数组t的所在的点并将这些点连接起来

### 3. 安装Anaconda，成功安装后打开Python IDE Spyder，在其代码编辑窗口中输入如下代码进行测试



点击此按钮生成新的Script文档

点这里运行代码（运行前保存！）



运行结果