

Python 开发环境搭建与基础

1.关于 Anaconda 的版本与下载	2
1.1 版本	2
1.2 下载	2
1.3 安装主版本 Anaconda2	2
1.3.1.选择安装类型	2
1.3.2 选择安装目录	3
1.3.3 选择添加到环境变量和默认 Python 版本:	3
1.4 在主版本 Anaconda2 文件夹下安装 Anaconda3	4
1.5 安装完验证	5
2.关于 Pycharm 的版本下载与安装	7
2.1 下载 Pycharm	7
2.2Pycharm 的安装	7
3.Pycharm 使用的 python 环境设置.....	9
4.配置和使用 conda 来管理需要的包.....	10
4.1 anaconda 的配置.....	10
4.2 anaconda 的包管理.....	11
5.python+pycharm+pyqt5+QT Designer 前端开发完整配置.....	13
5.1 环境安装	13
5.2 配置 Pycharm	14
5.2.1 配置 PyQt5.....	14
5.2.2 配置 QtDesigner 以便快速打开 QtDesigner，可视化地拖控件写 UI	14
5.2.3 配置 Pyuic 用来把 QtDesigner 生成的 UI 文件转换成.py 文件.	15
5.2.3 配置 Pyrcc 用来将资源文件如图片等转成 python 代码能识别的文件	16
5.2 测试 Pycharm + python3 + QtDesigner	17

Data	Author	Version	History
2018/10/09	CME	V1.0	This is the initial version.

1.关于 Anaconda 的版本与下载

1.1Anaconda 的两个版本

Anaconda 是一个 Python 的科学计算发行版，包含了超过 300 个流行的用于科学、数学、工程和分析的 Python Packages。由于 Python 有 2 和 3 两个版本，因此 Anaconda 也在 Python2 和 Python3 的基础上推出了两个发行版，即 Anaconda2 和 Anaconda3。有时候我们会在电脑上同时使用 Anaconda2 和 Anaconda3 两个版本，因此有必要考虑它们的共存问题。一般的使用场景是以其中一个版本为主，另外一个版本为辅，因此只需要达到能方便的切换到备用版本，且备用版本的 pip 等命令能正常使用即可。

1.2Anaconda 的下载

下载有两种方式：

A.官网（不推荐）<https://www.anaconda.com/download/>不推荐，因为尤其的慢，而且通常是安装到一般就错误。

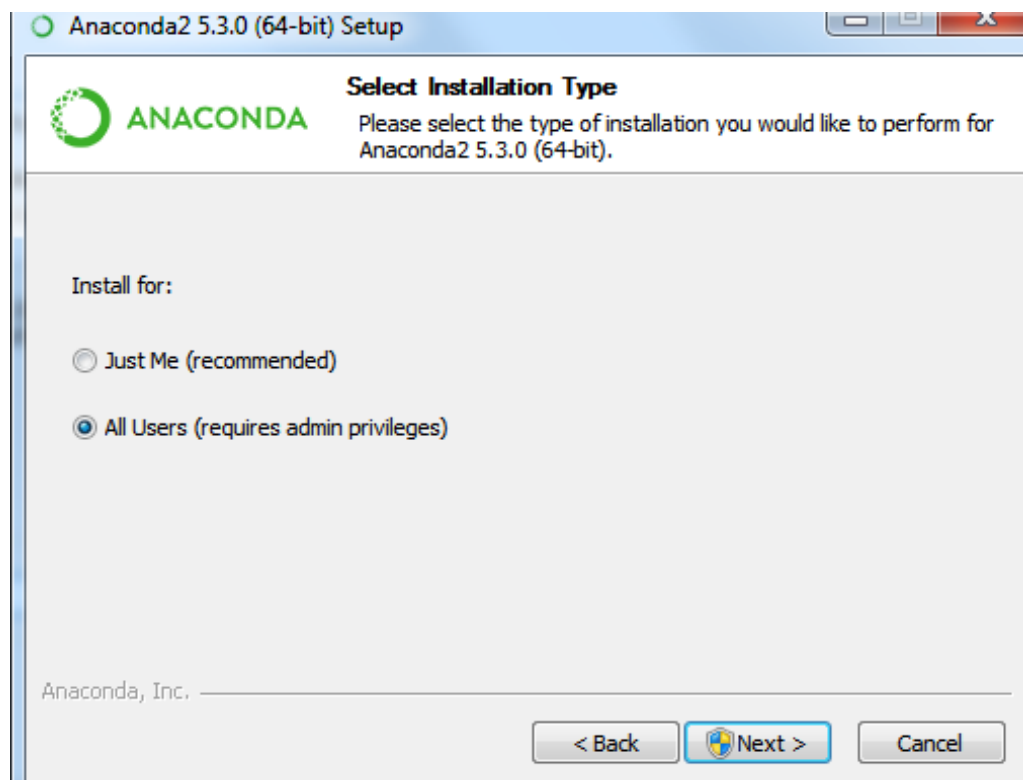
B.镜像网站安装。<https://mirrors.tuna.tsinghua.edu.cn/anaconda/archive/>这是清华大学的镜像网，直接找到你需要的下载即可，又快又省力。

提供的压缩包 AnacondaAndPycharm.7z 中有两个版本。

1.3 安装主版本 Anaconda2

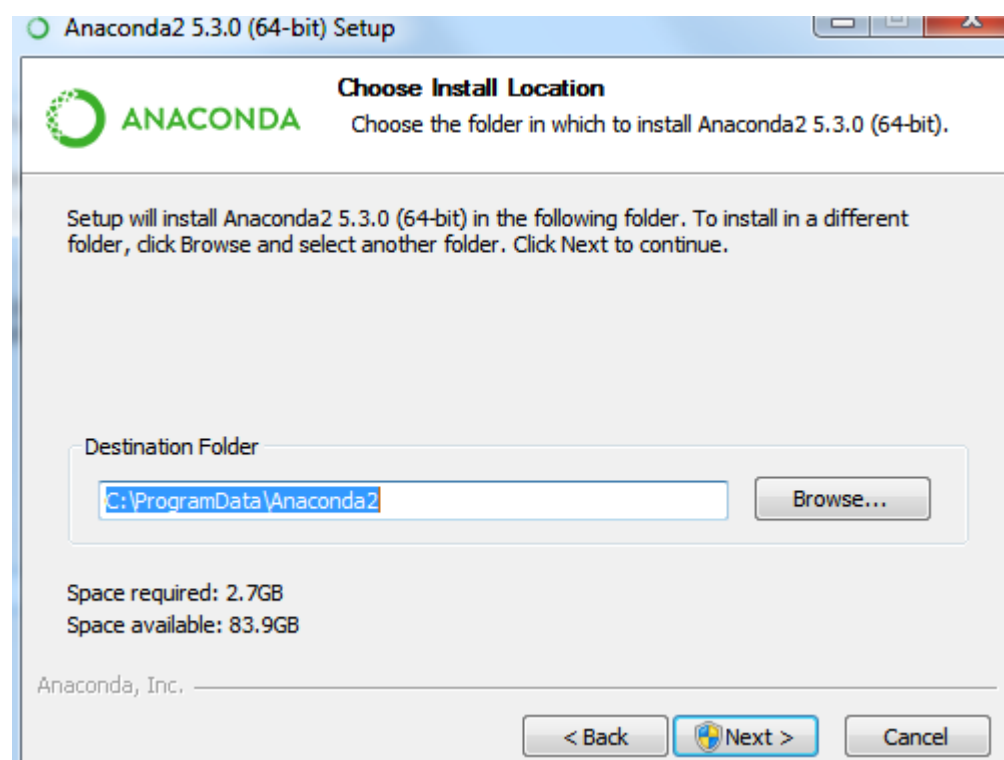
1.3.1.选择安装类型

这里选择 All Users,还要与 Pycharm 的选择一致。（测试中发现如果使用 Just Me，在后期会有莫名其妙的错误，很可能是环境变量的问题，所以推荐使用 All Users）



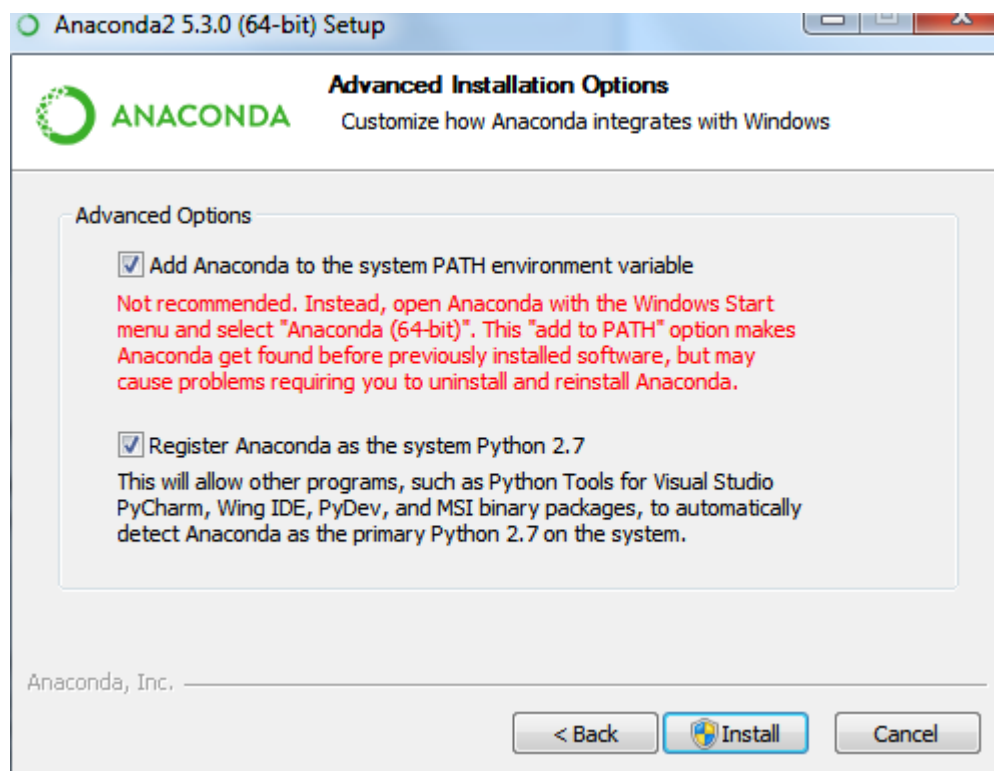
1.3.2 选择安装目录

这里安装目录为默认 C:\ProgramData\Anaconda2。推荐使用默认路径，如果使用自定义的目录，一定不要有中文路径和空格等符号，全英文是最好的。



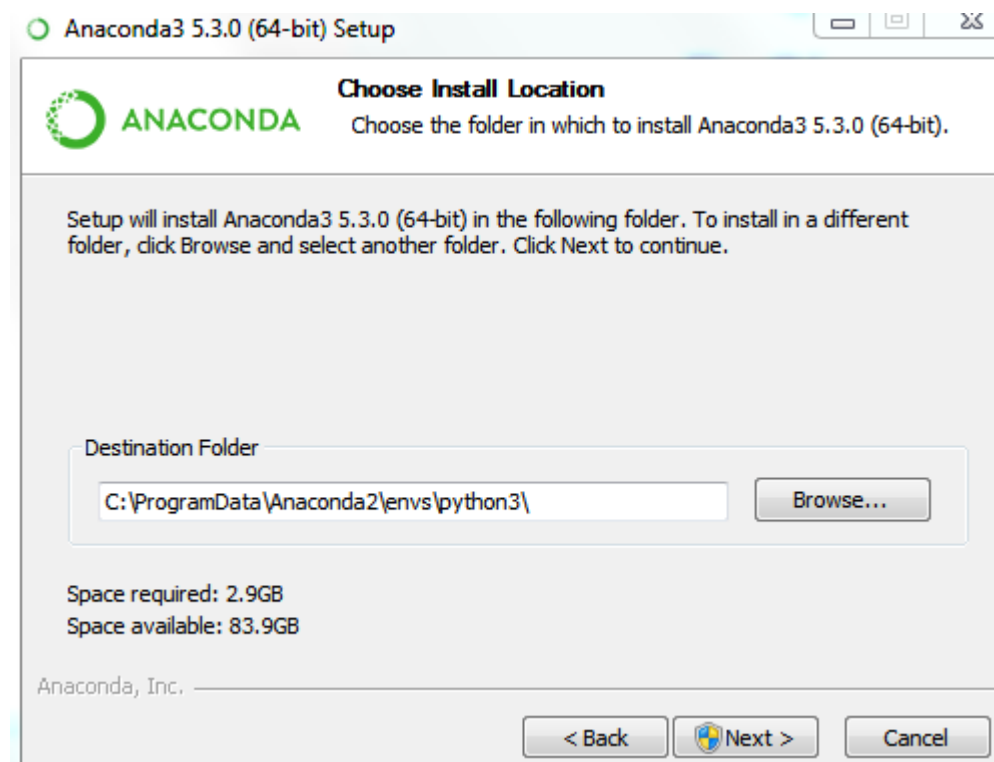
1.3.3 选择添加到环境变量和默认 Python 版本：

第一个选项是将安装目录加入到系统的 PATH 环境变量中，以后在 CMD 中便可以直接用 python 命令启动 python。第二个选项是让其他 IDE 能够检测到 Anaconda2 并将 Anaconda2 作为默认的 Python 2.7。

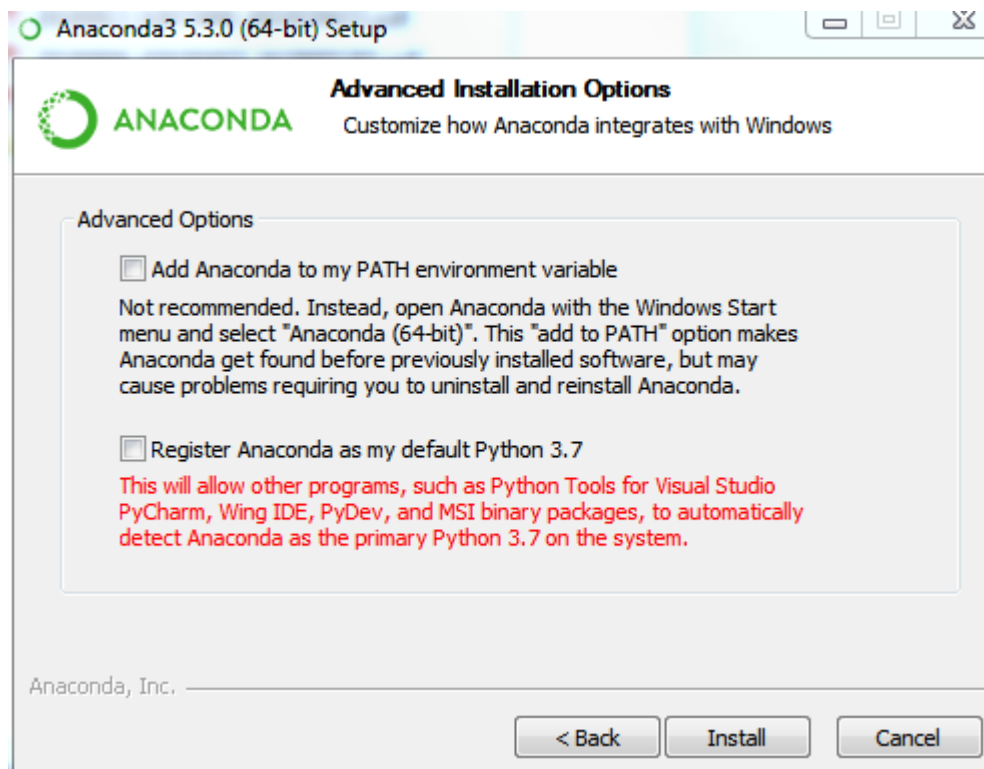


1.4 在主版本 Anaconda2 文件夹下安装 Anaconda3

这里 Anaconda3 的安装目录必须选在 Anaconda2 安装目录的\envs 子目录下，如图所示，最后的"pythob3"可以自己另取其他的名字，该名字用于启用 python3.

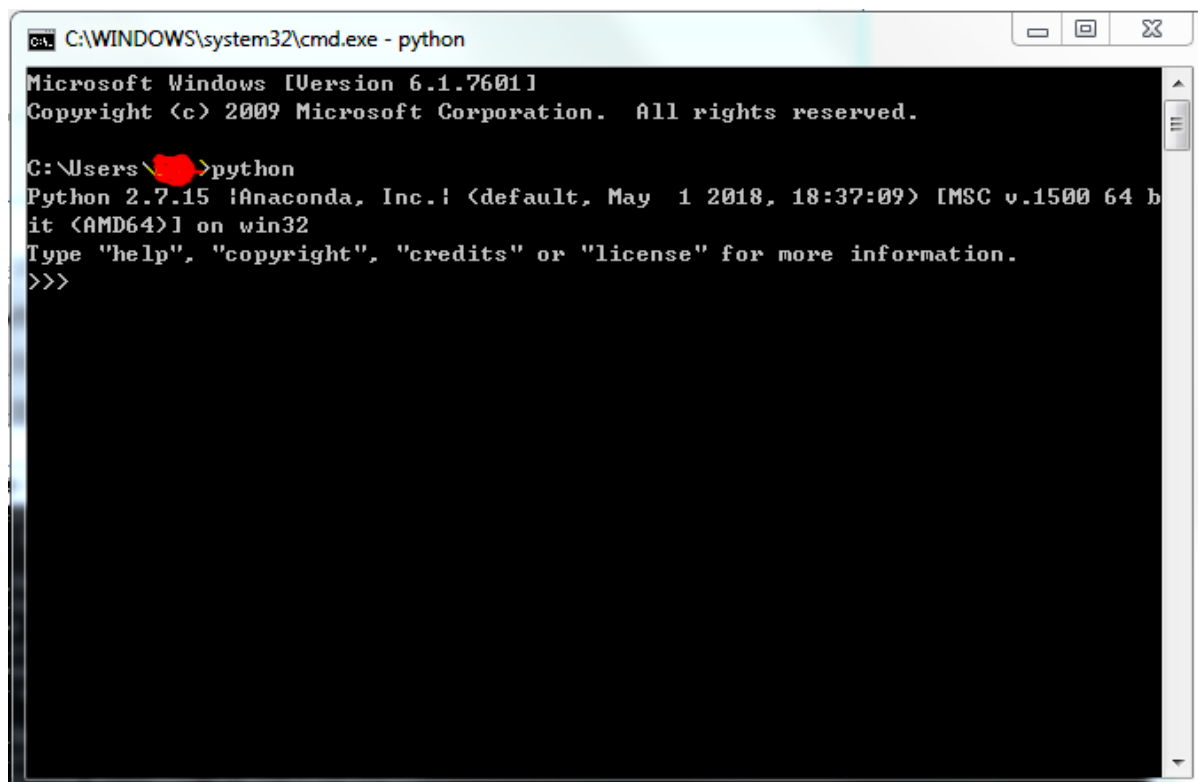


这里取消掉那两项，由于之前已经把 Anaconda2 添加到系统 PATH 路径下了，因此再把 Anaconda3 添加进去，由于顺序在 Anaconda2 的后面，几乎是没有什么作用的

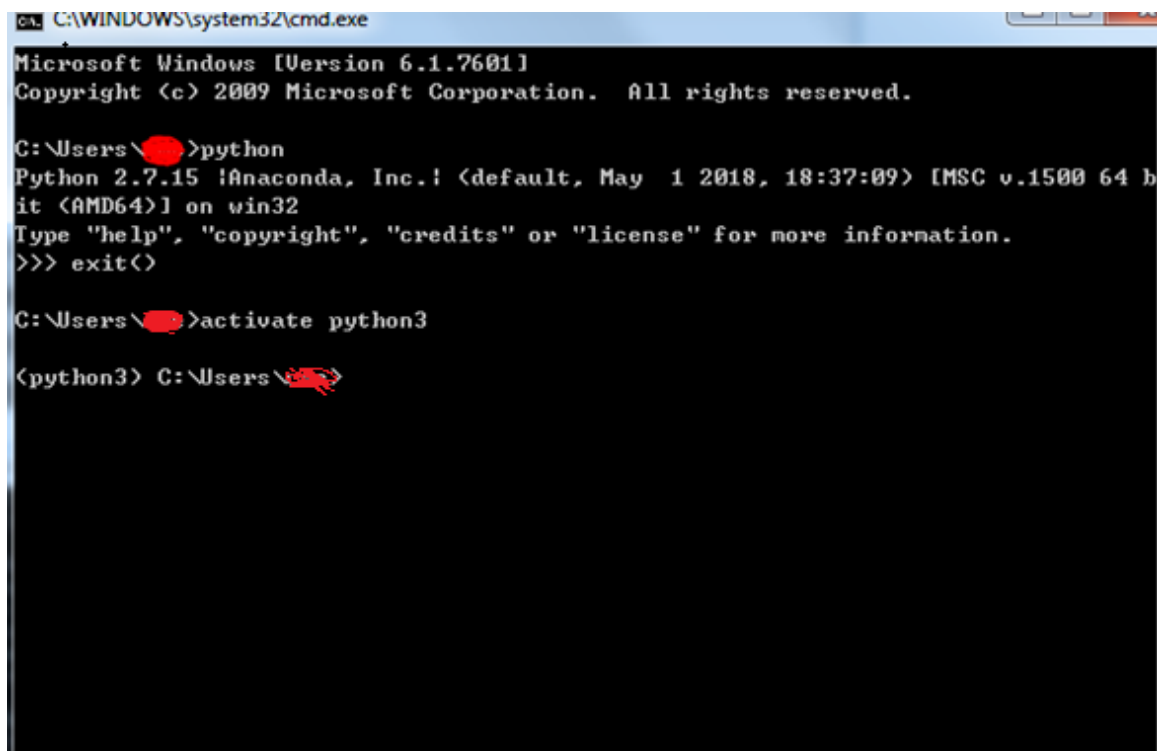


1.5 安装完验证

打开 CMD: 开始按钮+r, 或者在开始搜索栏中输入 run。在 CMD 里面直接输入 python 会启动 Python2.7:



输入 `exit()` 退出 Python2，然后使用 `activate python3`（python3 即之前 Python3 安装目录文件夹的名字）命令之后，即可切换至 Python3，在命令行前面会出现一个 `[python3]` 标记，此时使用任何的 `python` 命令都是在 Python3 下进行的。



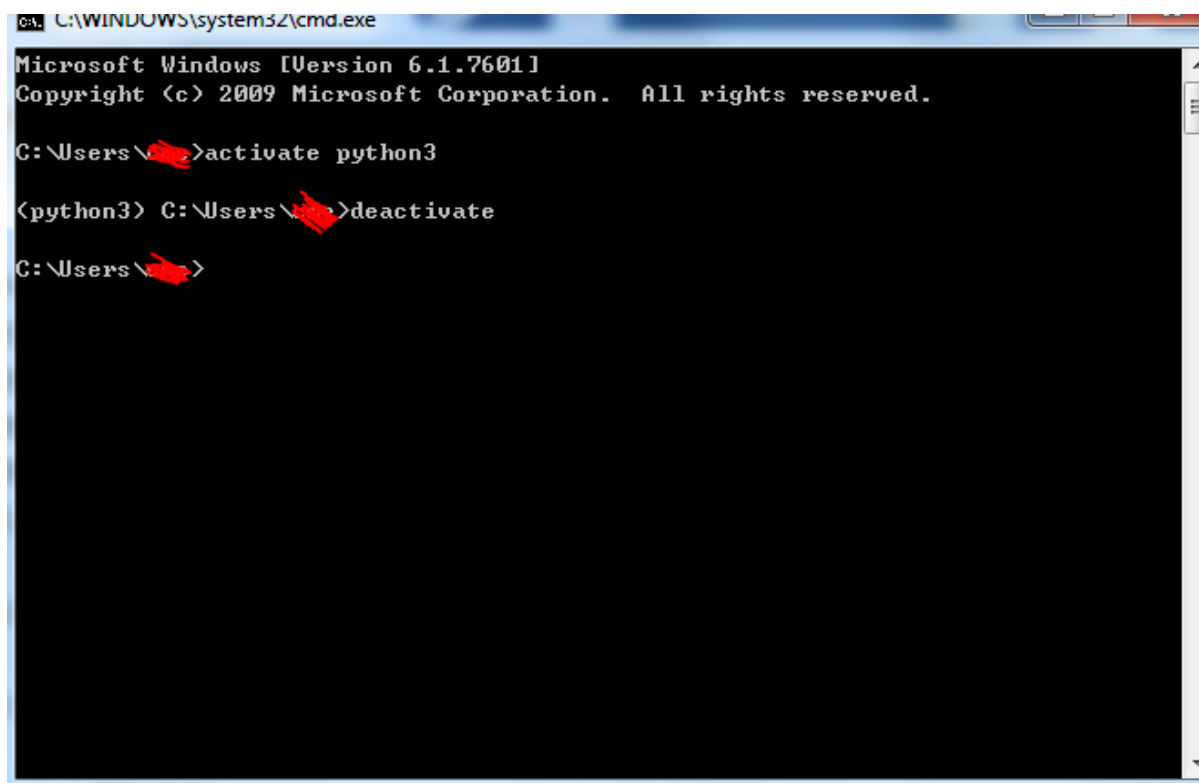
```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\>python
Python 2.7.15 |Anaconda, Inc.| (default, May 1 2018, 18:37:09) [MSC v.1500 64 b
it (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> exit()

C:\Users\>activate python3

[python3] C:\Users\>
```

使用 `deactivate` 命令可取消激活 Python3



```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\>activate python3

[python3] C:\Users\>deactivate

C:\Users\>
```

2.关于 Pycharm 的版本下载与安装

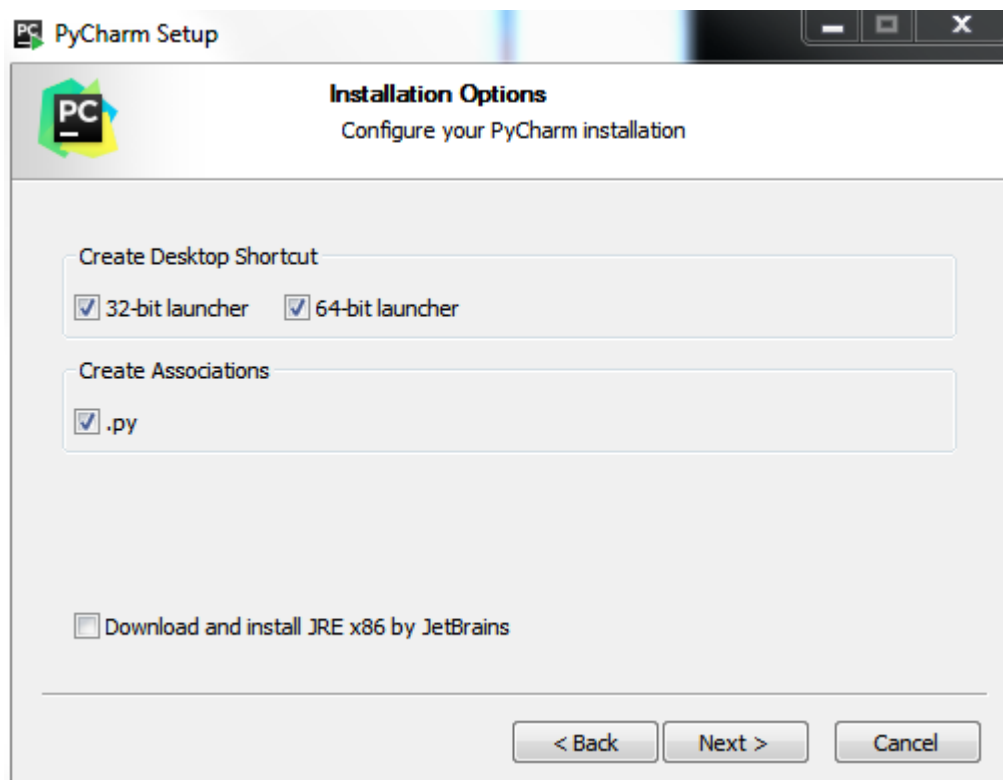
PyCharm 是一种 Python IDE，带有一整套可以帮助用户在使用 Python 语言开发时提高其效率的工具，比如调试、语法高亮、Project 管理、代码跳转、智能提示、自动完成、单元测试、版本控制。此外，该 IDE 提供了一些高级功能，以用于支持 Django 框架下的专业 Web 开发，但是如果 python 脚本中的代码是基于控制台操作的，那么在 Pycharm 中运行时会报错，这一点注意。最后，尊重版权，有钱就买个正版的吧，没钱用社区版的也可以的。

2.1 下载 Pycharm

下载官方 Pycharm 2018.2.4（专业版）：<https://www.jetbrains.com/pycharm>

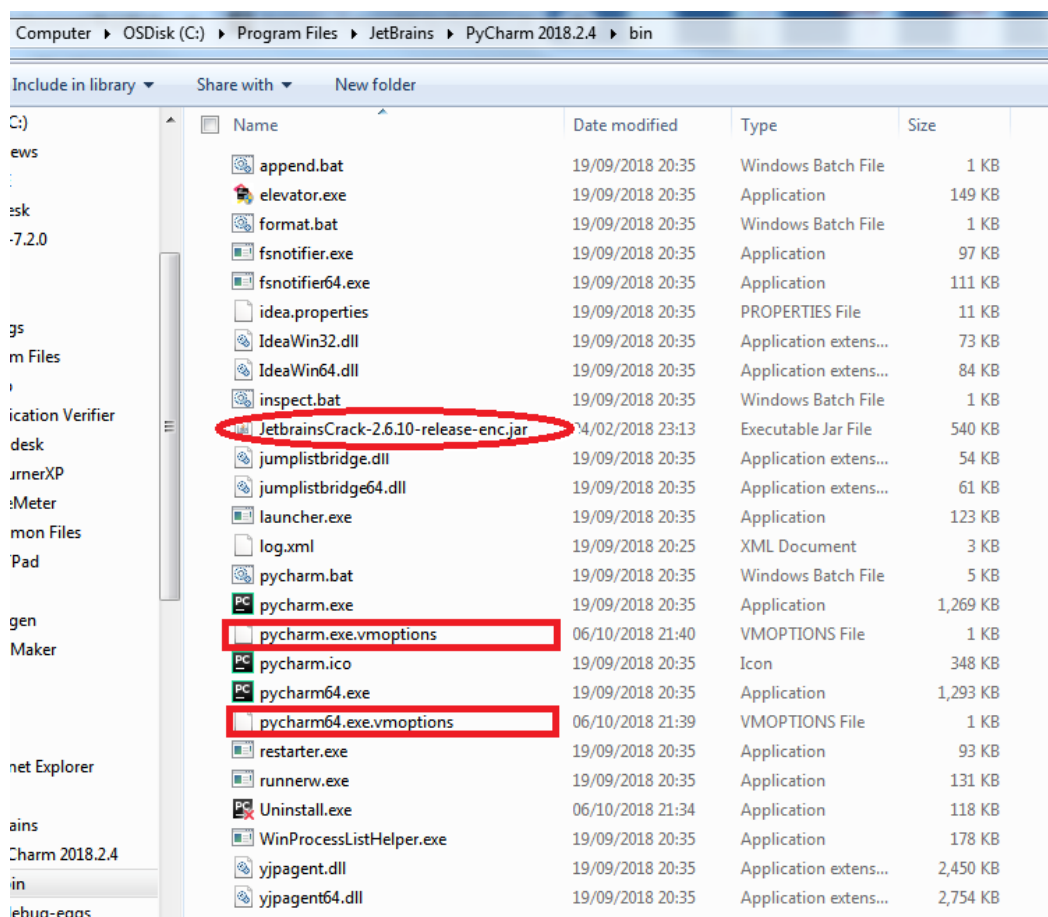
提供的压缩包 AnacondaAndPycharm.7z 中有两个版本。

这里的安装依然推荐默认路径，这里选择 All Users 或是 Just Me 都可以，但是一定要与上面安装 Anaconda 的选择一致，在 Installation Options 中选择前三个选项，最后一个不要选，安装中下载很慢。安装完成后先不要打开。

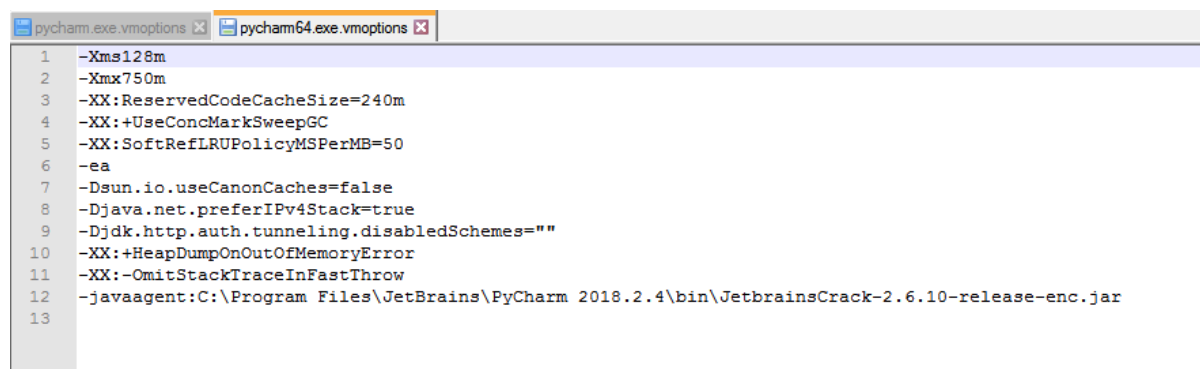


2.2 Pycharm 的安装

将 Pycharm 破解文件夹中的 JetBrainsCrack-2.6.10-release-enc.Jar 放入 pycharm 安装目录的 bin 目录下：C:/Program Files/JetBrains/PyCharm 2018.2.4/bin



在 pycharm.exe.vmoptions, pycharm64.exe.vmoptions 两文件里加入-javaagent:C:\Program Files\JetBrains\PyCharm 2018.2.4\bin\JetbrainsCrack-2.6.10-release-enc.jar



打开 Pycharm, 开始注册, 如果没有自己的注册码, 可以填写破解文件夹中的注册码。可以通过 Help->Register 来查看注册信息。该注册码只到 2018 年 11 月。

The screenshot shows the "PyCharm License Activation" window. It has a title bar with a PC icon and a close button. The main area contains the following text:

Licensed to 彼岸 software1.

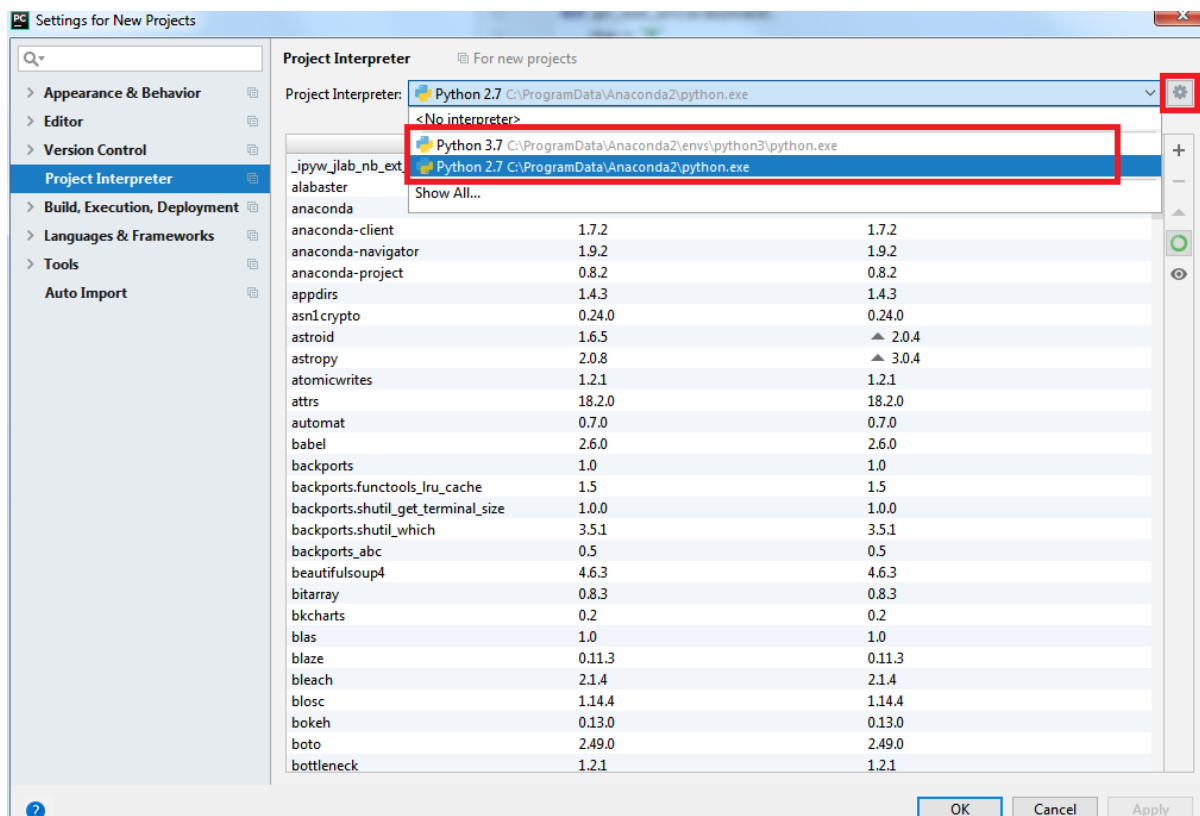
Subscription is active until November 5, 2018.

Activate new license with:

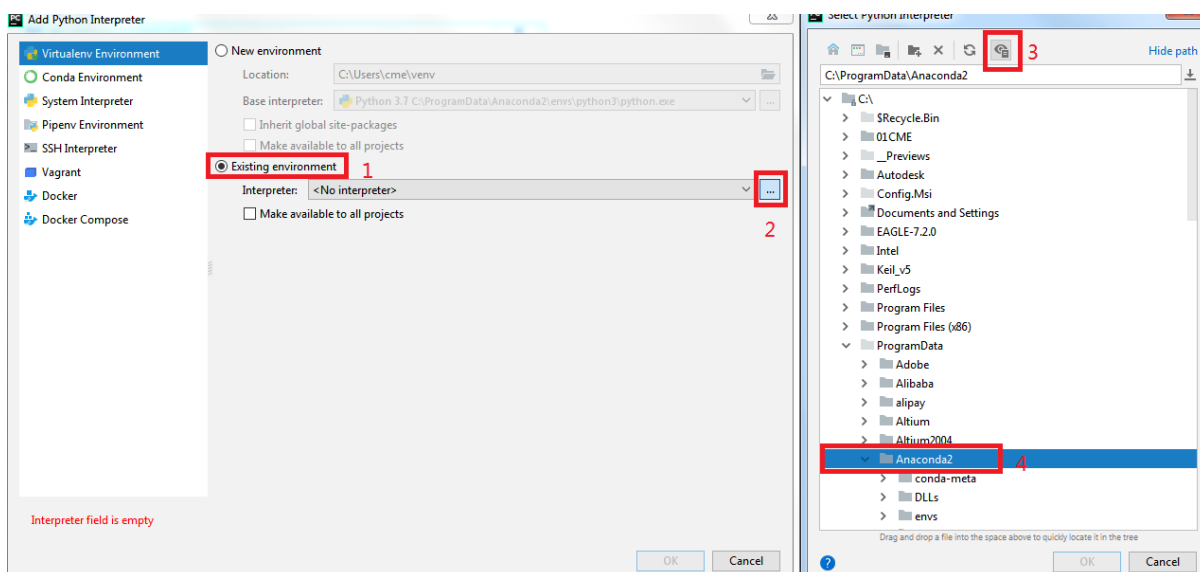
☐ JetBrains Account ☒ Activation code ☐ License server

A text field below contains a long alphanumeric activation code: 9DK8FDSOS2-eyJsaWNlbnNISWQiOiI5REs4RkRTT1MyIiwibGljZW5zZWVOYWYwIljo... (truncated). At the bottom are three buttons: "Remove License", "OK", and "Cancel".

现在我们的电脑上安装了 Python2.7 和 Python3 两个版本，可以通过 File->Settings for New Projects 来设定选择哪个：点击右侧的按钮并选择 Add，之后开始添加 python 环境



按下图所示依次在 Anaconda2 中找到 python.exe.最后点击 ok，完成 python2.7 添加。



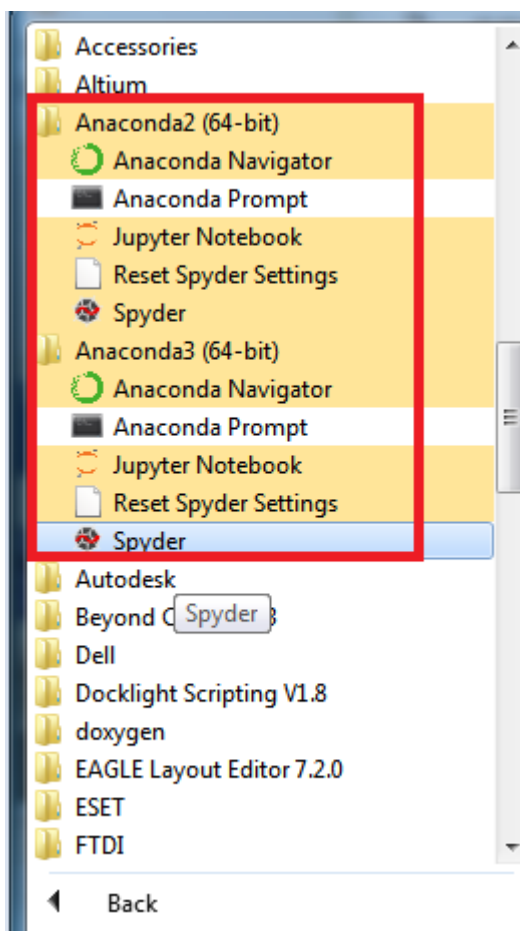
按照同样的方法添加 python3 的环境到 pycharm，注意 python3 的目录在 Anaconda2 下的 envs 子文件夹下。到此，我们完成了基本的环境配置，真的不同的 Python 版本选择不同的环境。

4.配置和使用 conda 来管理需要的包

4.1 anaconda 的配置

在安装了 anaconda 后，我们也可以使用 anaconda 来进行 Python 库的安装，同样的也需要进行源的配置。

在开始菜单中找到 anaconda2 和 anaconda3 的 Anaconda Prompt 并打开（这里只配置 anaconda2 或 anaconda3 其中一个就可以）。



将下列命令拷贝到终端中并回车：

```
conda config --add channels https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkg/free/  
conda config --add channels https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkg/main/  
conda config --set show_channel_urls yes  
conda config --add channels https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud/conda-forge/  
conda config --add channels https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud/msys2/  
conda config --add channels https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud/bioconda/  
conda config --add channels https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud/menpo/  
conda config --add channels https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud/pytorch/  
conda config --add channels https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud/peterjc123/
```

4.2 anaconda 的包管理

anaconda 为我们提供方便的包管理命令——conda

查看已经安装的 packages

```
conda list
```

查看某个指定环境的已安装包

```
conda list -n python34
```

查找 package 信息

```
conda search numpy
```

安装 scipy

```
conda install scipy
```

安装 package

如果不用 -n 指定环境名称, 则被安装在当前活跃环境

```
conda install -n python34 numpy
```

更新 package

```
conda update -n python34 numpy
```

删除 package

```
conda remove -n python34 numpy123456789101112131415161718192021
```

由于 conda 将 conda、python 等都视为 package, 因此, 完全可以使用 conda 来管理 conda 和 python 的版本, 例如

更新 conda, 保持 conda 最新

```
conda update conda
```

更新 anaconda

```
conda update anaconda
```

更新 python

```
conda update python
```

到这里, 你可以使用 Pycharm 来编辑相对大的工程。如使用 TENSORFLOW

但是, 对于需要使用控制台来显示的工程, 在 Pycharm 中不能正常工作。

对于 2048.py, 你可以在 pycharm 中进行编辑, 但是如果在 pycharm 中运行就会报错, 打开 conda, 在命令行中输入 `python 2048.py` 就会运行。

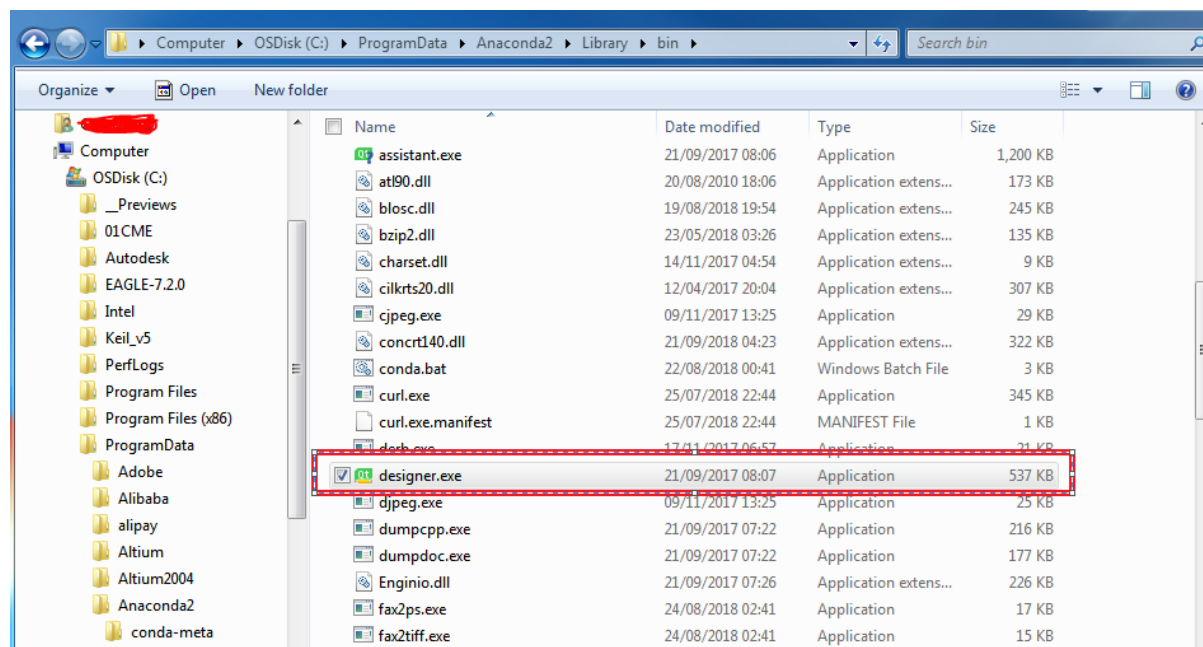
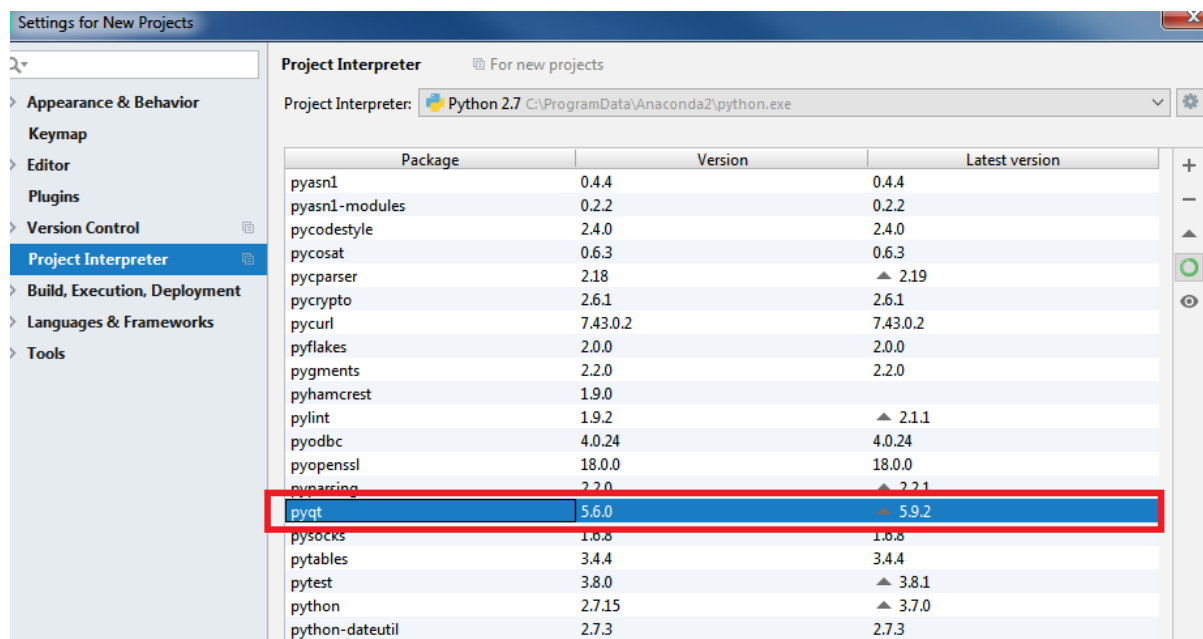
5.python+pycharm+pyqt5+QT Designer 前端开发完整配置

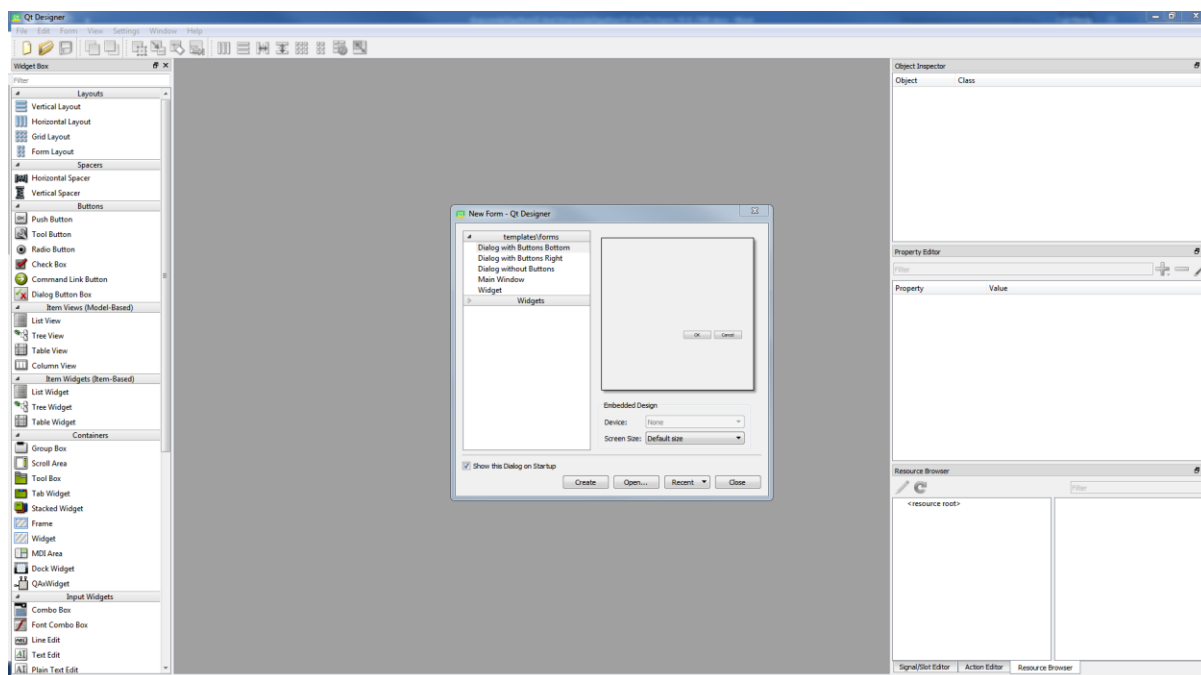
5.1 环境安装

为了实现快速开发前端，推荐是用 pyqt 框架，因为这个框架有 QT Designer 作为辅助，可以加快开发效率，而且，来自于 C++ 的积淀，框架的稳定性应该也令人放心。

通过 Anaconda 安装 python，同时安装了 pyqt，qtdesigner 等。这点不错。

同时，在 C:\.....（你的目录）\Anaconda2\Library\bin 目录下，如果能够找到：designer.exe 这个文件，并且双击出现下图界面，那么你的 qtdesigner 也是可以使用的。





5.2 配置 Pycharm

5.2.1 配置 PyQt5

安装 Anaconda 时候自动装的是 pyqt, 还需要一个 PyQt5 和 pyqt5-tools 的第三方库, 如果不安装不能正常运行。

1. 以管理员权限打开 Anaconda3 下的 Anaconda Prompt.

2. 安装 PyQt5 和 pyqt5-tools:

```
pip install PyQt5
```

```
pip install pyqt5-tools
```

5.2.2 配置 QtDesigner 以便快速打开 QtDesigner, 可视化地拖控件写 UI

打开 PyCharm, 打开 File—>Settings—>External Tools, 点击加号来添加自己的工具, 做如下配置:

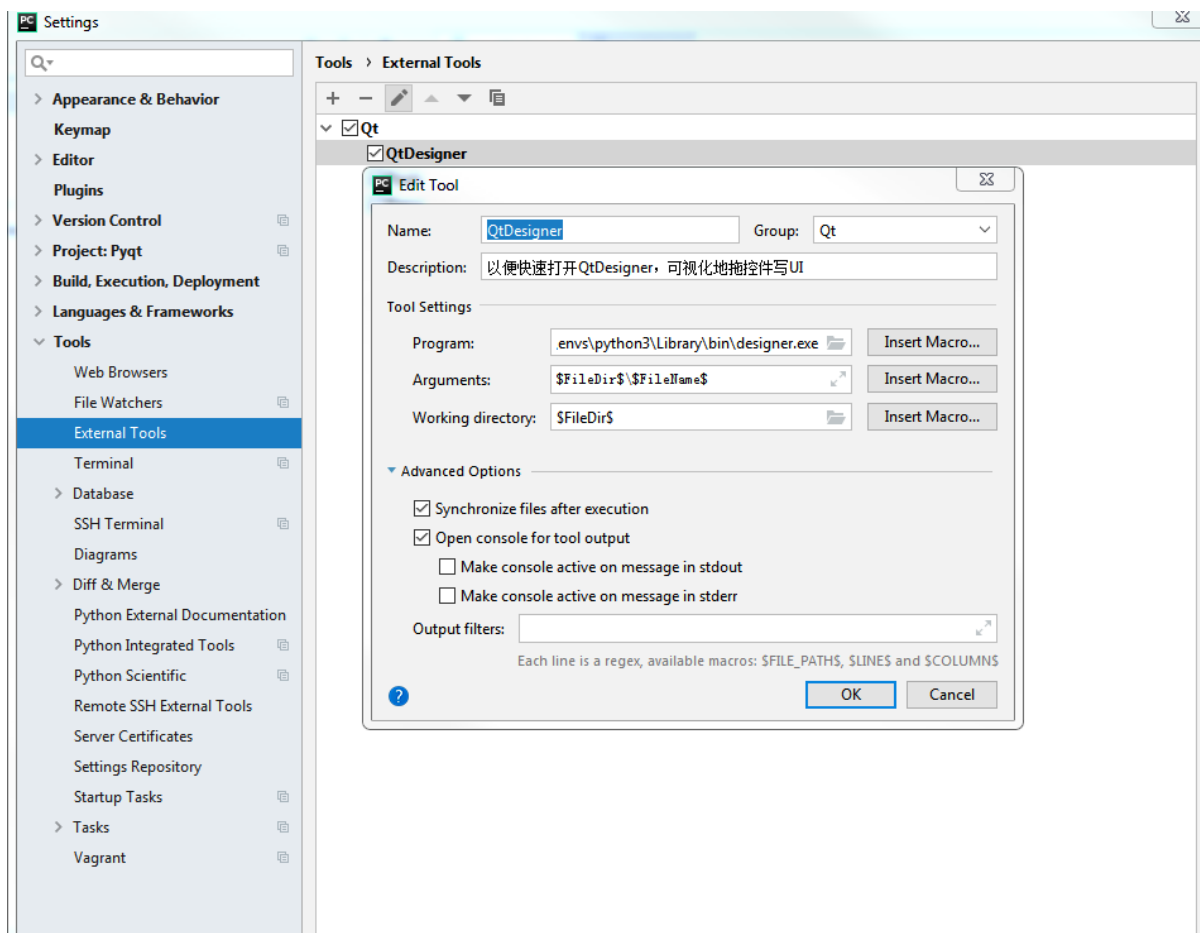
Name: QtDesigner

Group: Qt

Program: C:\ProgramData\Anaconda2\envs\python3\Library\bin\designer.exe(路径)

Arguments: \$FileDir\$\\$FileName\$

Working directory: Working directory: \$FileDir\$



5.2.3 配置 Pyuic 用来把 QtDesigner 生成的 UI 文件转换成.py 文件.

打开 PyCharm, 打开 File—>Settings—>External Tools, 点击加号来添加自己的工具, 做如下配置:

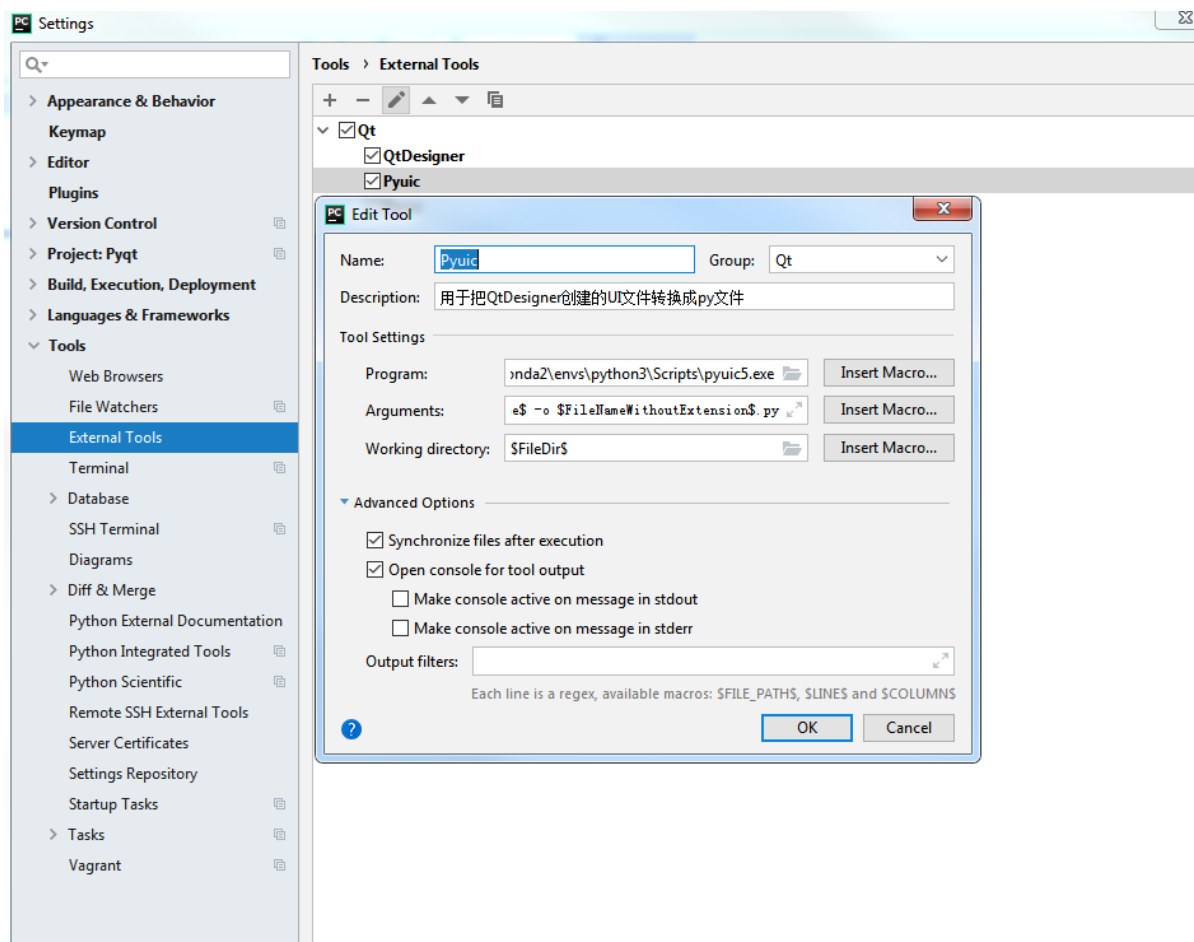
Name: Pyuic

Group: Qt

Program: C:\ProgramData\Anaconda2\envs\python3\Scripts\pyuic5.exe (pyuic5 路径)

Arguments: \$FileName\$ -o \$FileNameWithoutExtension\$.py

Working directory: Working directory: \$FileDir\$



5.2.3 配置 Pyrc 用来将资源文件如图片等转成 python 代码能识别的文件

打开 PyCharm，打开 File—>Settings—>External Tools,点击加号来添加自己的工具，做如下配置:

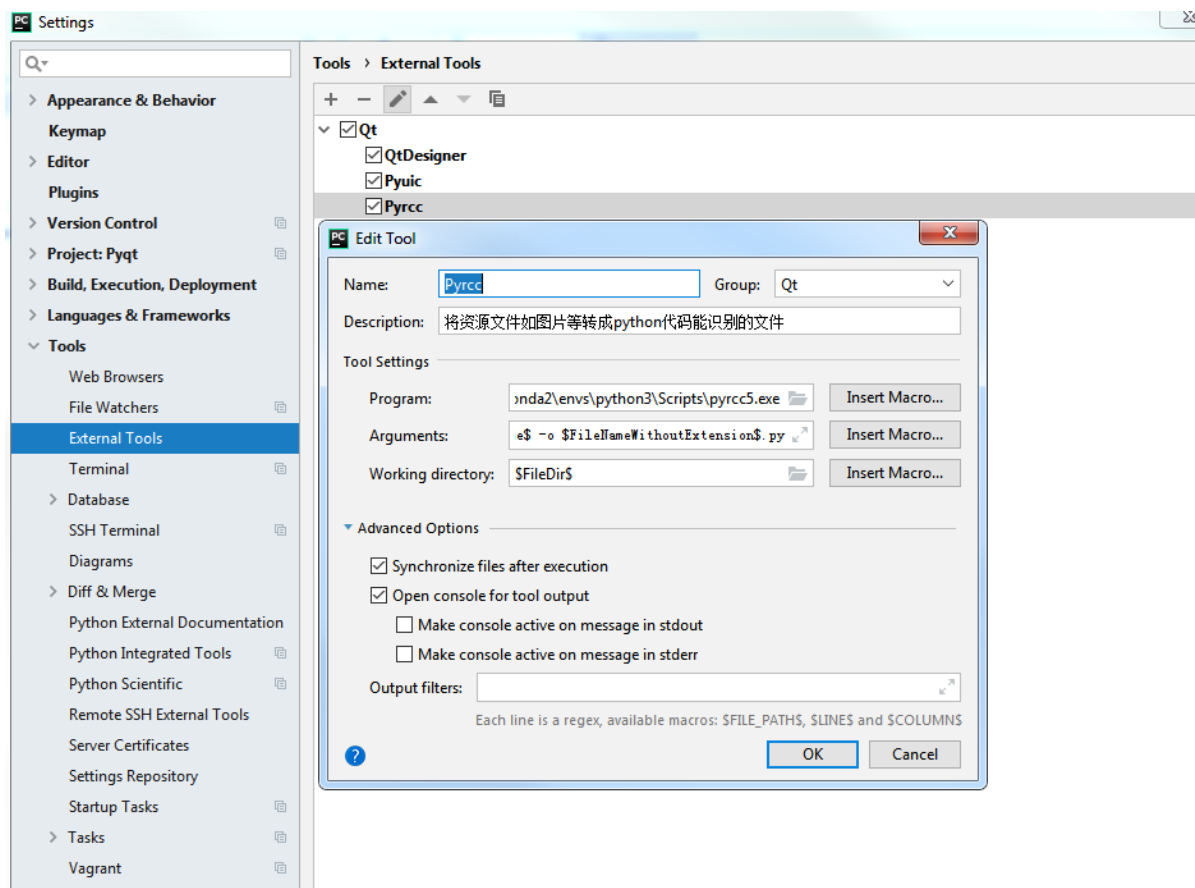
Name: Pyrc

Group: Qt

Program: C:\ProgramData\Anaconda2\envs\python3\Scripts\pyrc5.exe (pyuic5 路径)

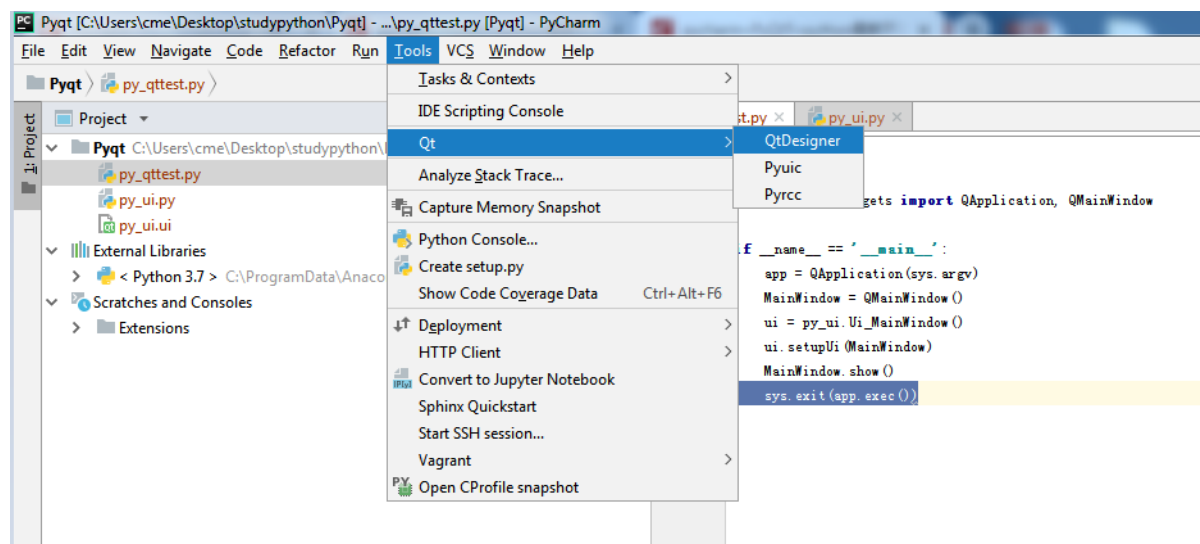
Arguments: \$FileName\$ -o \$FileNameWithoutExtension\$.py

Working directory: Working directory: \$FileDir\$

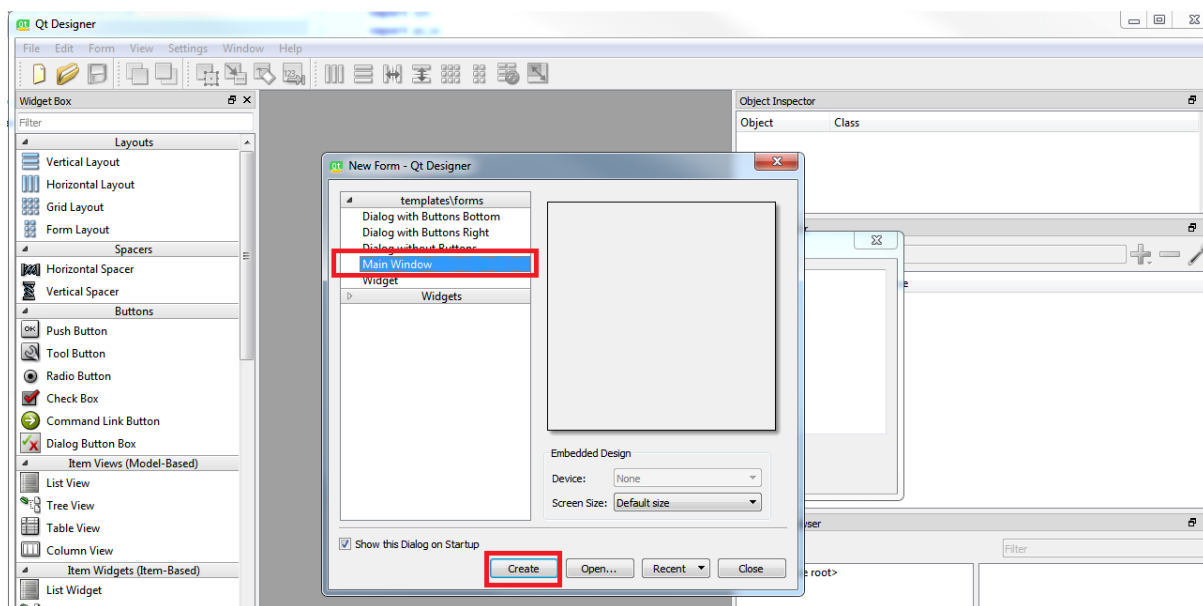


5.2 测试 Pycharm + python3 + QtDesigner

5.2.1 打开 pycharm->Tools->Qt->QtDesigner

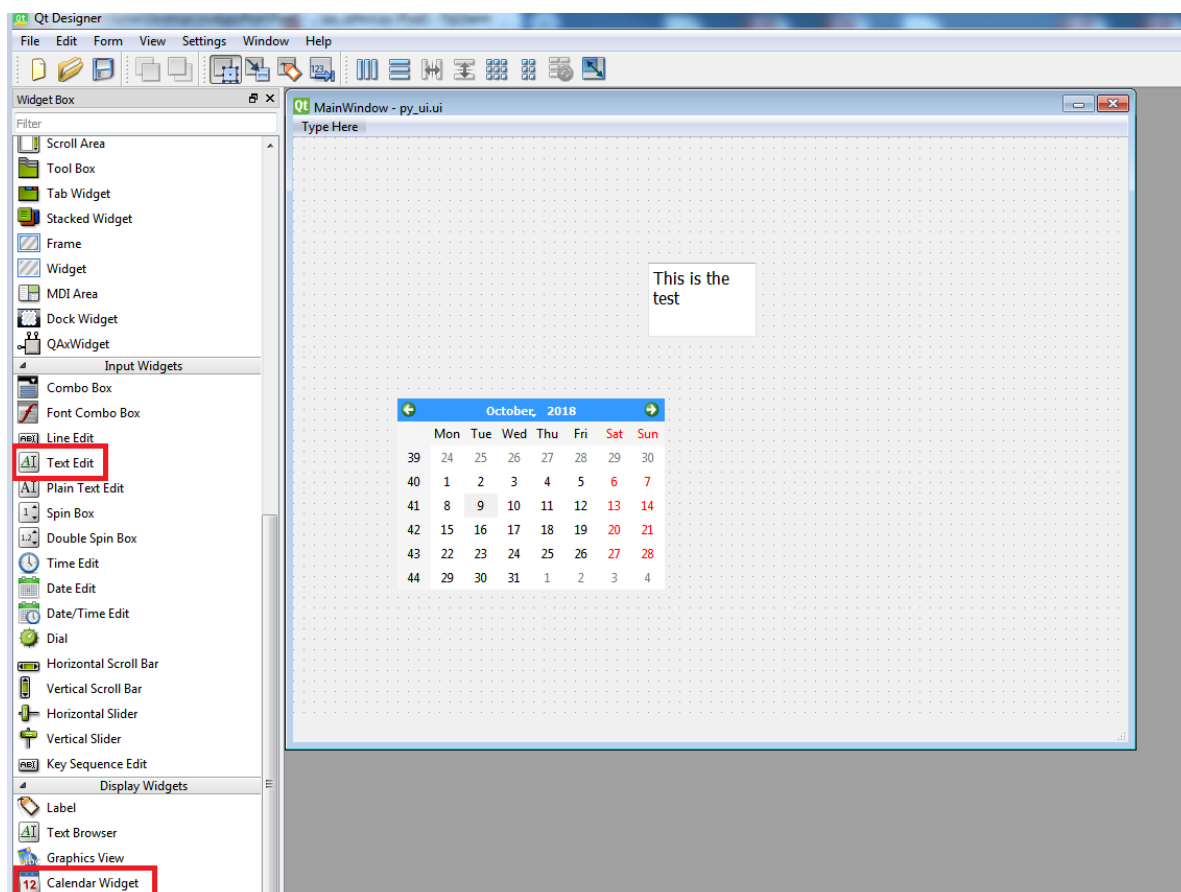


5.2.2 创建以个主窗口



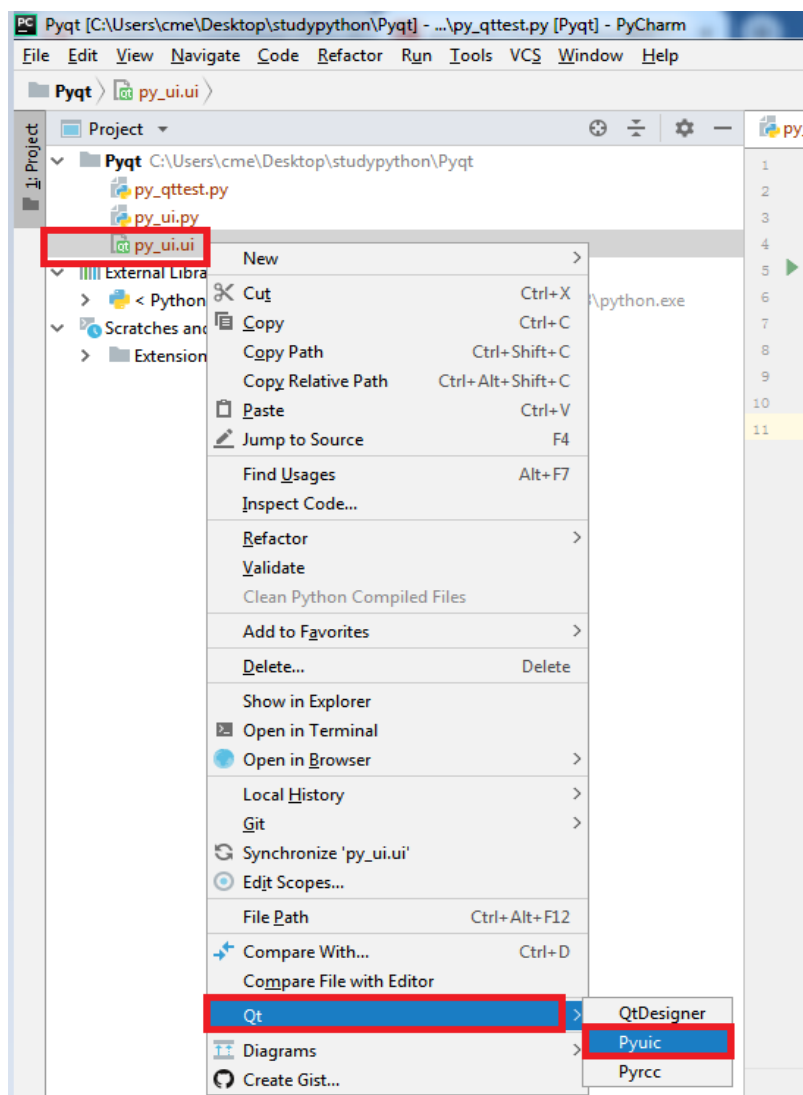
5.2.3 在主窗口中可以拖放控件，如拖放 Text Edit 和 Calendar Widget。可在 Text 中输入字符

5.2.4 点击保存按钮，选择保存到 pycharm 的工程目录，这时在 pycharm 中的 project 中会看到刚刚创建的.ui 文件。



5.2.5 在上一步骤中得到.ui 文件需要通过添加的外部工具将其转换成.py 文件才能进行编写和运行：

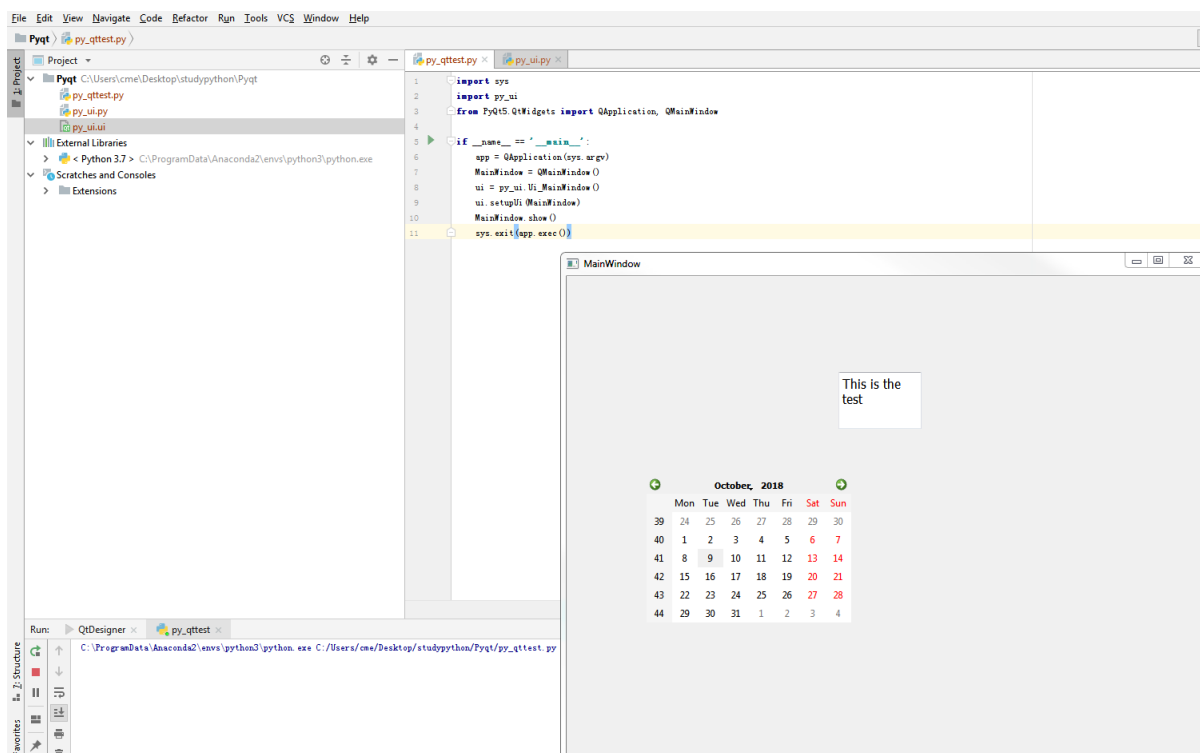
右击 project 中的.ui 文件, 选择 Qt->Pyuic, 单点后会成功会在 project 中出现对应的.py 文件。这里面就是界面的代码。



5.2.6 最后, 我们创建一个 Qt 的入口主文件, 并将上面生成的 py 模块导入运行:

```
import sys
import py_ui
from PyQt5.QtWidgets import QApplication, QMainWindow

if __name__ == '__main__':
    app = QApplication(sys.argv)
    MainWindow = QMainWindow()
    ui = py_ui.Ui_MainWindow()
    ui.setupUi(MainWindow)
    MainWindow.show()
    sys.exit(app.exec())
```



5.2.7 最后附上一些 PyQt5 的资源

<http://code.py40.com/pyqt5/14.html> （快速阅读）

<https://my.oschina.net/wisedream?tab=newest&catalogId=3404569> （中文）

<http://zetcode.com/gui/pyqt5/> （英文）

<https://www.kancloud.cn/digest/py-qt/119453> （中文）