# 课程实践作业一

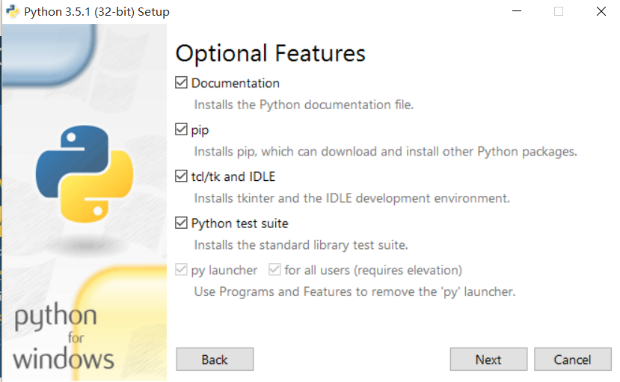
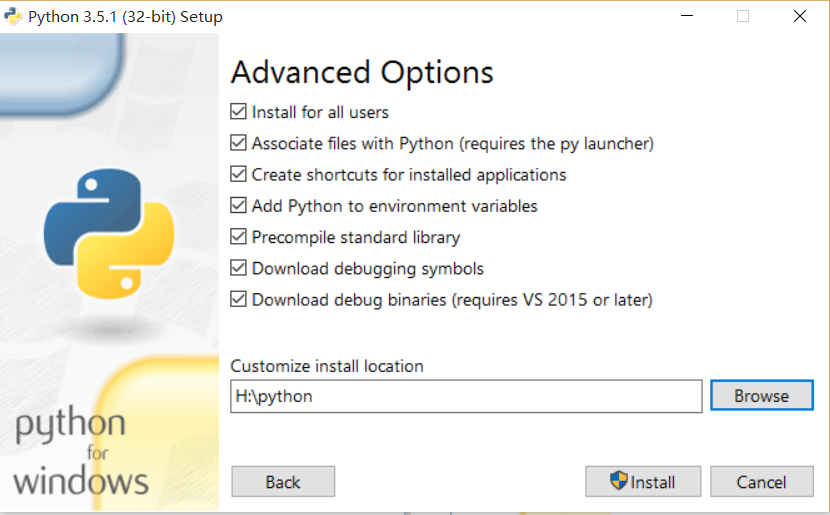
Python在设计上坚持了清晰划一的风格，这使得Python成为一门易读、易维护，并且被大量用户所欢迎的、用途广泛的语言。本次软件工程基础与实践课程也选用Python作为编程语言。本次作业的目的是编写建立Python开发环境。根据其过程，写一个环境建立指导手册。

## 一·Python基本开发环境建立

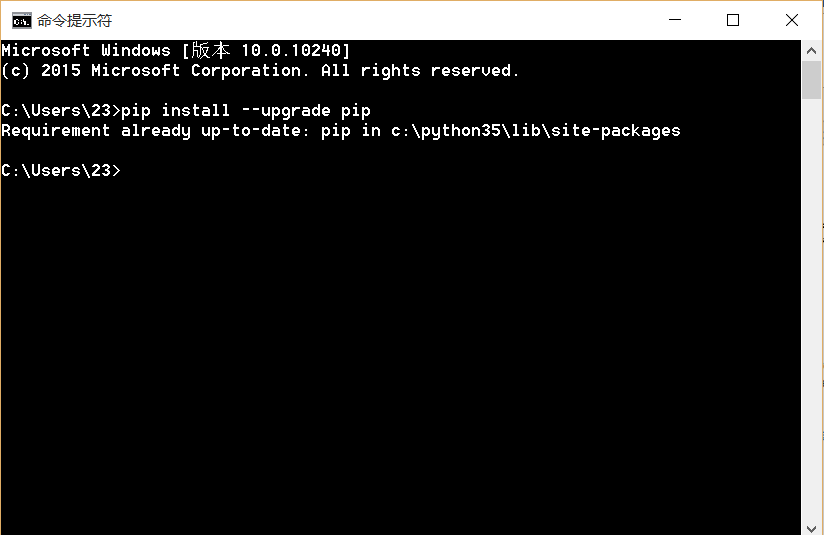
从官网：<http://www.python.org>

 下载和操作系统（32/64位）对应版本的Python3软件。下面以Python3.5.1版本安装为例做简单说明



在官网下载Python3.5.1之后，运行安装包。勾选加环境变量PATH并自定义安装。

自定义安装目录： H:\python，勾选如图所示的选项。

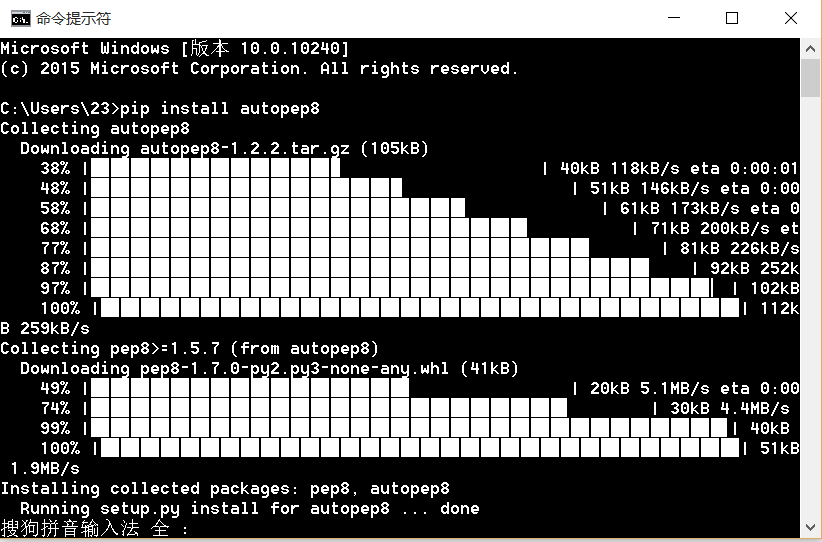
 

安装完成后，在命令行中输入命令行下更新pip到新版本：>pip install --upgrade pip

## 二 Python扩展包安装

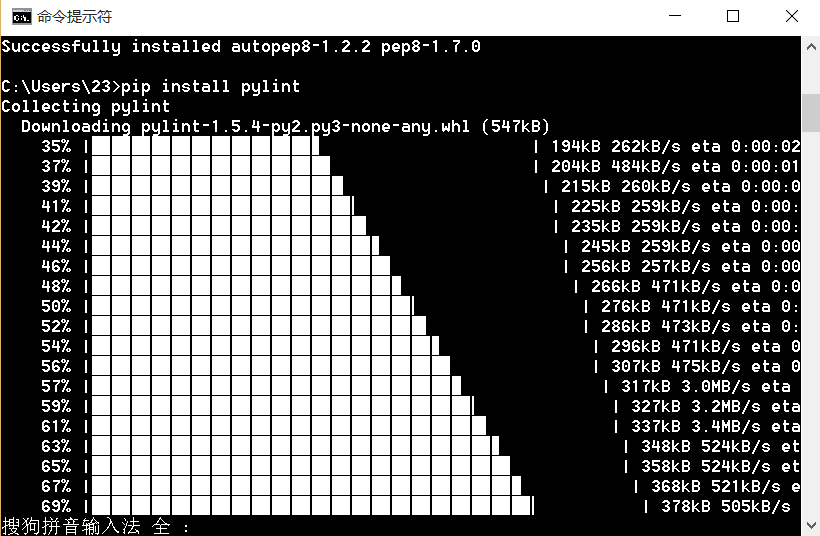
安装格式规范扩展包autopep8

Autopep8是一个将Python代码自动排版为PEP8风格的小工具。它使用pep8工具来决定代码中的哪部分需要被排版。Autopep8可以修复大部分pep8工具中报告的排版问题。



安装pylint

Pylint 是一个 Python 工具，除了平常代码分析工具的作用之外，它提供了更多的功能：如检查一行代码的长度，变量名是否符合命名标准，一个声明过的接口是否被真正实现等等。



1） 安装Jupiter ：

>pip install jupyter 在线安装

2） 安装Python语言内核

>pip install ipython ，支持Python语言

3） 安装依赖包

>pip install pyreadline

>pip install sympy

4）支持显示数学符号、公式，安装MathJax：

命令行下，

>IPython

打开一个IPython的shell，然后，在其中键入如下代码：

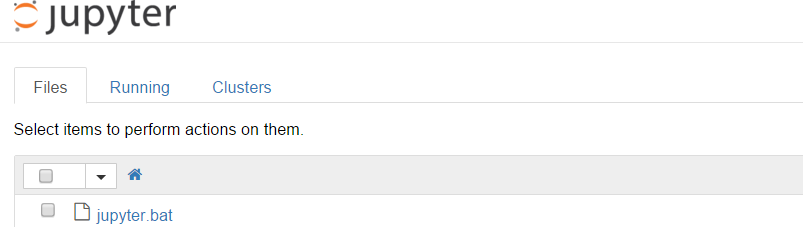
from IPython.external.mathjax import install\_mathjax

install\_mathjax()

5） 运行notebook:

在iPython notebook文件所在目录下，打开命令行窗口：

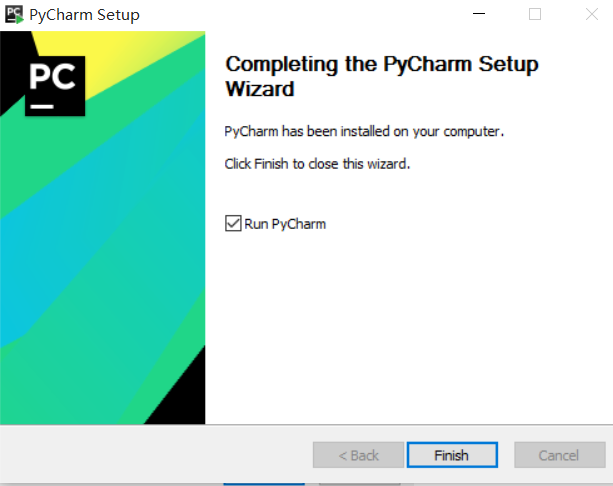
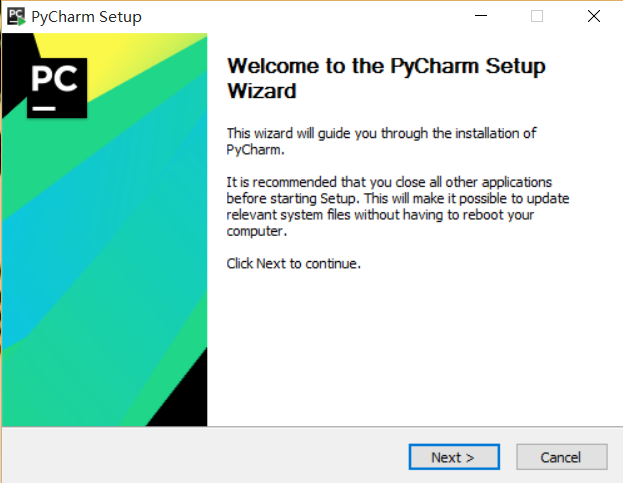
>jupyter notebook



## 三 基于Pycharm的Python开发环境

### 3.1安装Pycharm

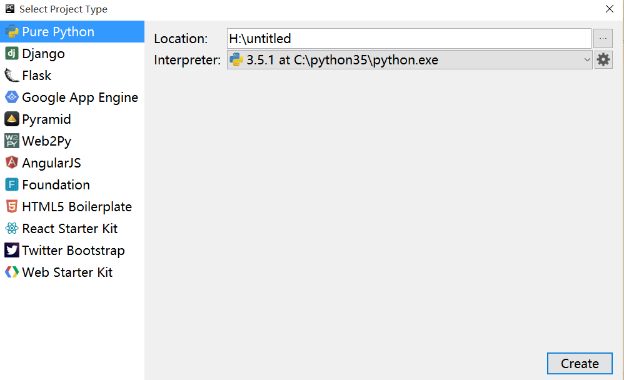
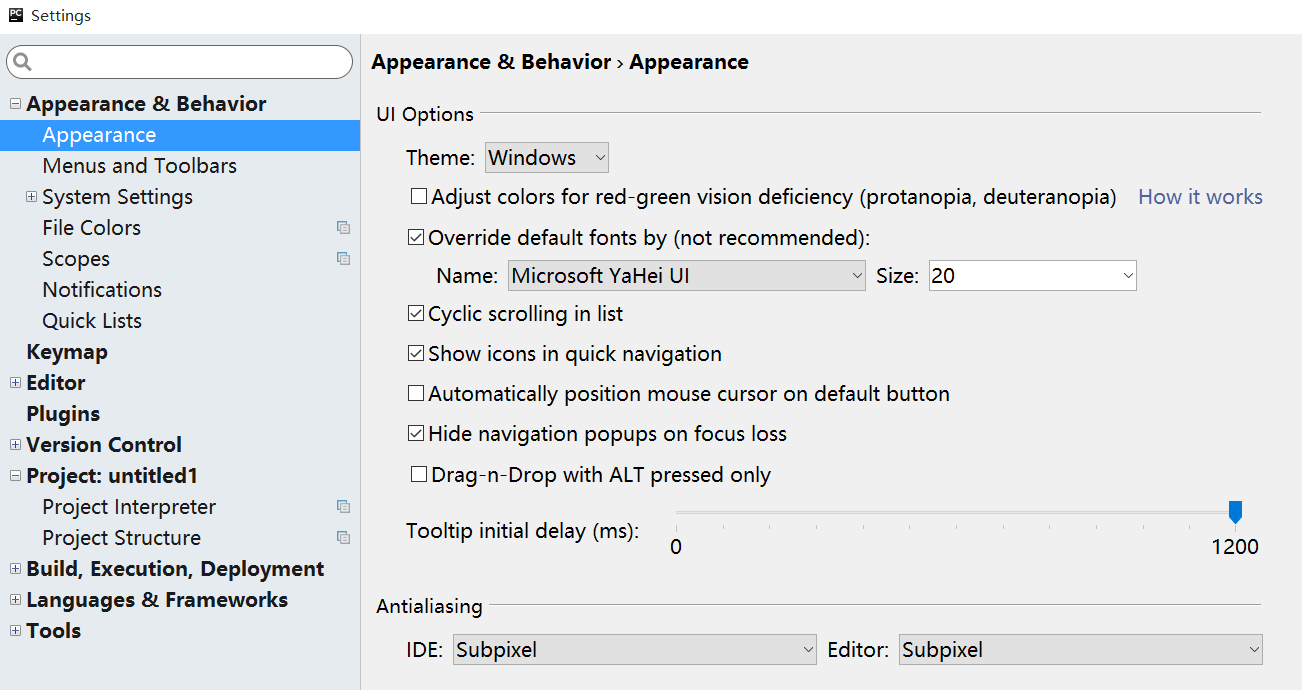
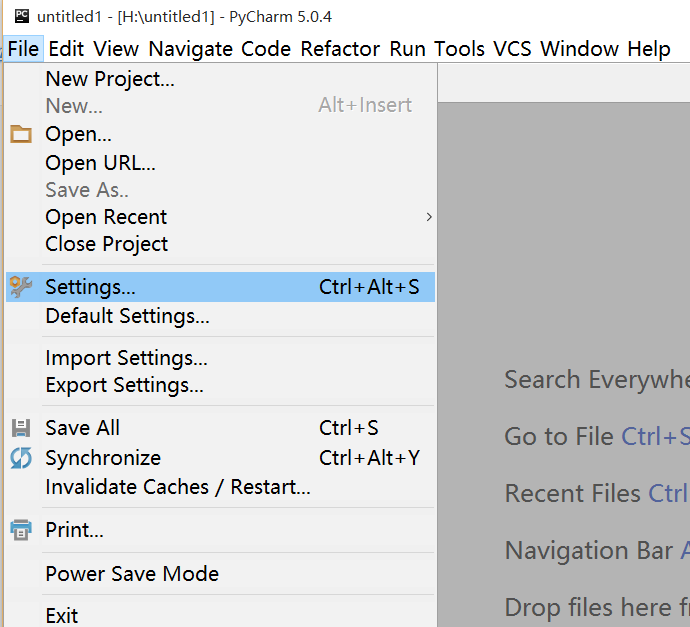
PyCharm是一种Python IDE，带有一整套可以帮助用户在使用Python语言开发时提高其效率的工具，比如调试、语法高亮、Project管理、代码跳转、智能提示、自动完成、单元测试、版本控制。此外，该IDE提供了一些高级功能，以用于支持Django框架下的专业Web开发。



安装过程中全部勾选默认选项。

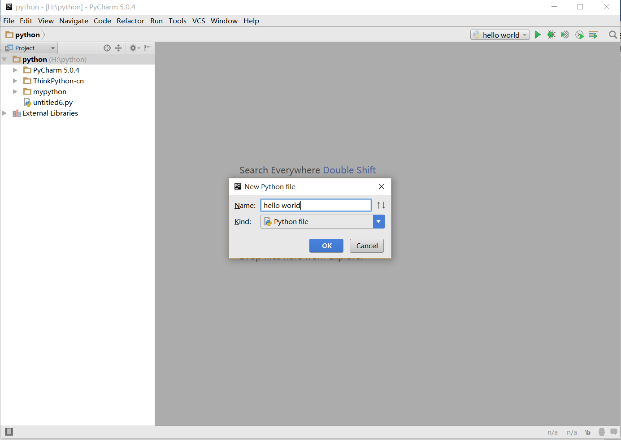
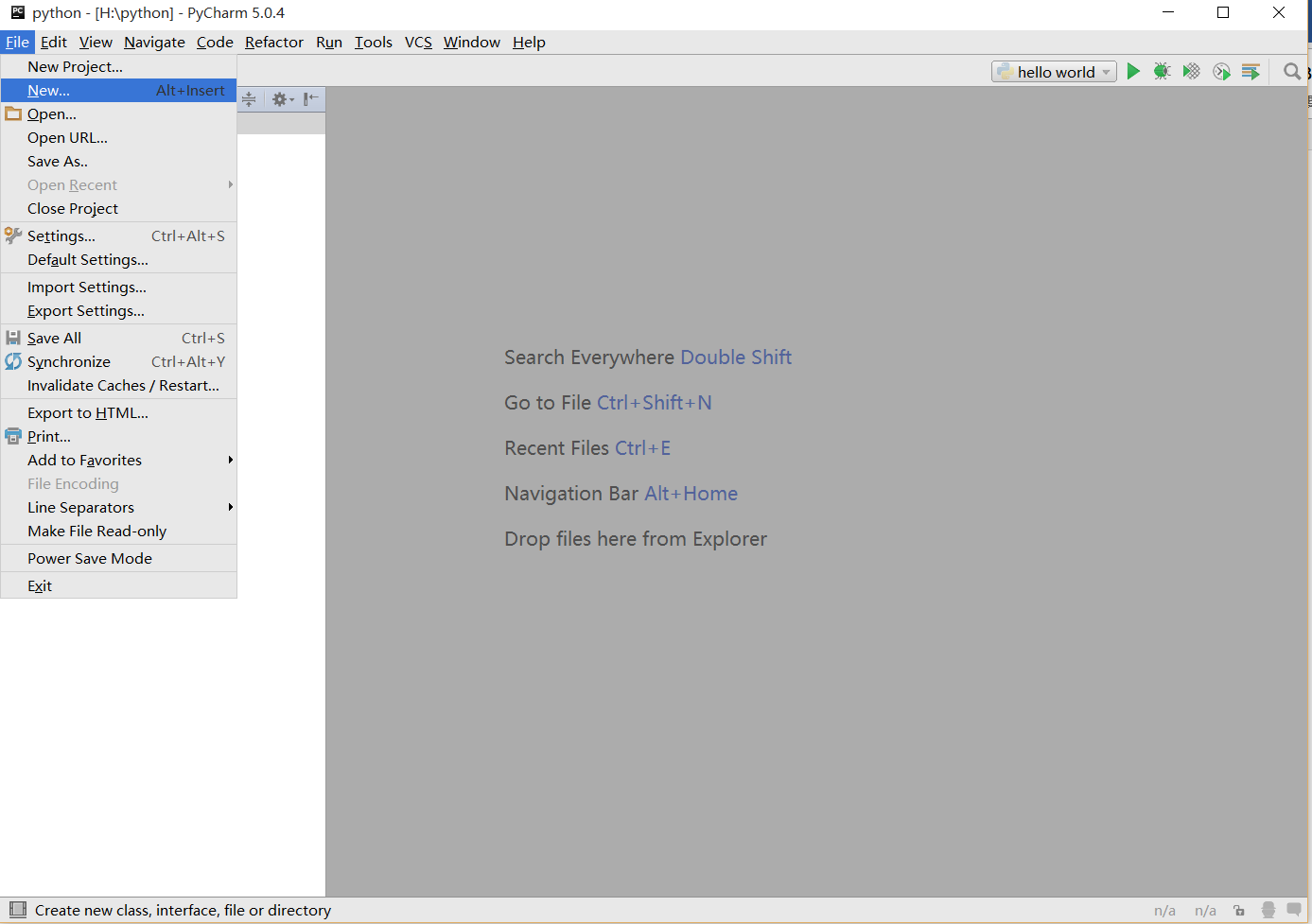
### 3.2配置Pycharm开发环境

在使用Pycharm编写程序之前需要配置python解释器，并配置开发环境以提高效率。

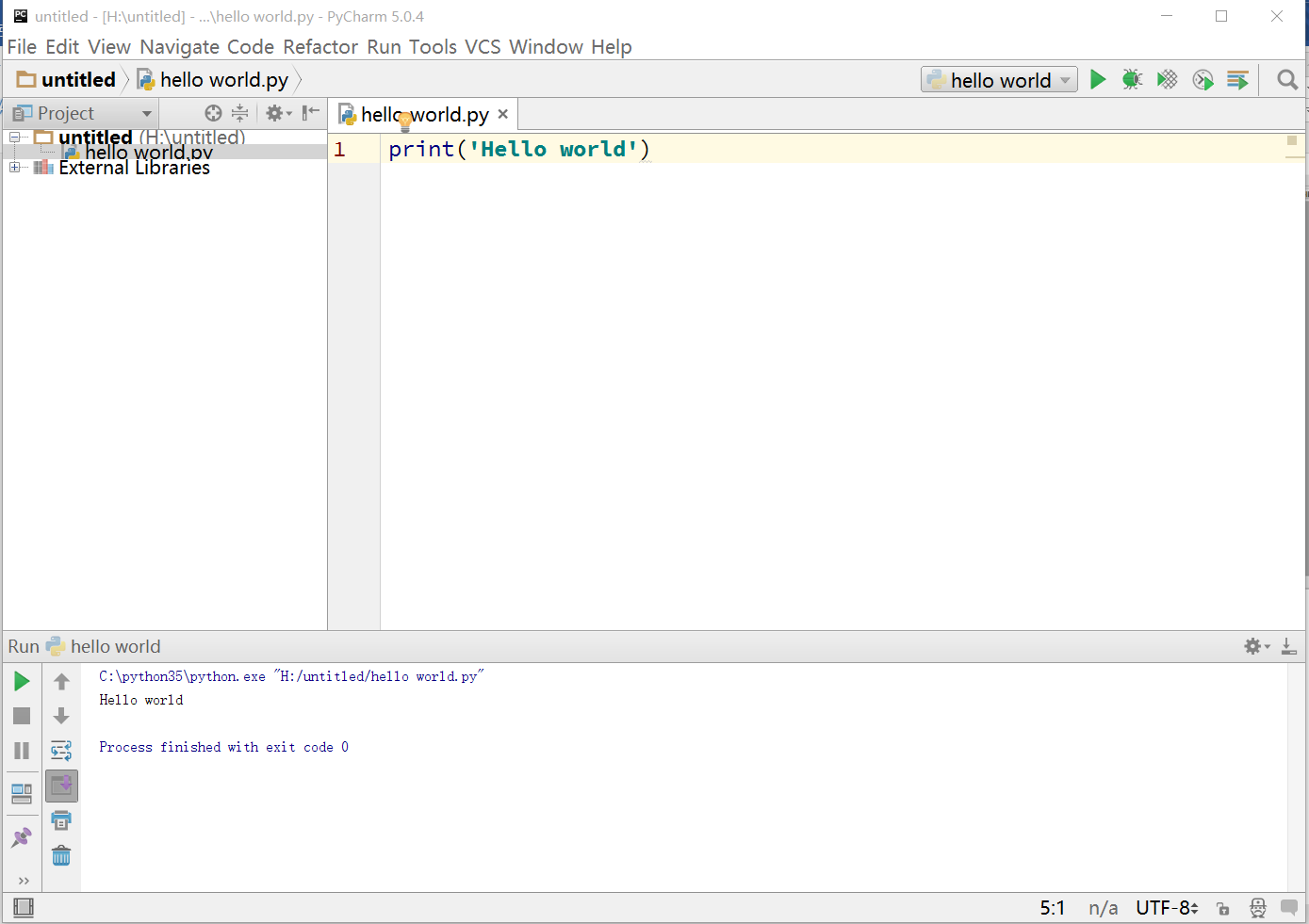
本地添加python解释器，找到之前安装的python3.5.1，并加入到pycharm中。

打开File-setting 对界面字体等选项进行调整。

### 3.3建立一个新程序

点击File –>New file创建一个Python file并命名。



编写代码并运行。

## 四 小结

工欲善其事必先利其器。搭建好良好的编译环境才能使编码效率提升。不能过分重视算法而忽略了基本环境的搭建。通过此次作业，我对python有了一定的了解，也学会了用互联网解决不会的问题。