

深度学习基本环境配置

详细说明与指导

王浩辰

DanferWang

目录

- Python-实现深度学习算法的编程语言
 - 下载python3.7
 - 安装python3.7
- PyCharm-强大的python集成开发环境
 - PyCharm的简单配置
- Anaconda-最著名的数据科学平台
 - Anaconda安装
 - Anaconda配置
- PyCharm的完成配置

Python-实现深度学习算法的编程语言

- Python 是近乎自然语言的解释型程序设计语言，鉴于他的可读性强，规范约束较少，丰富的第三方库，参考资源广泛等显著优势，python成为了学习和研究人工智能，特别是深度学习算法中最基础也是最热门的工具。
- Python 的最新稳定大版本是3.8，但是鉴于TensorFlow、Keras、Torch等常见深度学习框架主流支持稳定版本为python 3.7，因此以python3.7作为基础语言配置

下载python3.7

- Python 的官网: <https://www.python.org/>, 是被广泛学习python语言的同学们忽视的超级超级强大的网站, 上面有下载、文档、入门学习、社区问答、甚至是著名工程、下载安装第三方库也可以在这里进行。
- 有不同的版本和操作系统可以选择下载安装:
- Windows (env中有) Linux Mac OS

- [Python 3.7.8 - June 27, 2020](#)

Note that Python 3.7.8 *cannot* be used on Windows XP or earlier.

- Download [Windows help file](#)
- Download [Windows x86-64 embeddable zip file](#)
- Download [Windows x86-64 executable installer](#)
- Download [Windows x86-64 web-based installer](#)
- Download [Windows x86 embeddable zip file](#)
- Download [Windows x86 executable installer](#)
- Download [Windows x86 web-based installer](#)

- [Python 3.7.8 - June 27, 2020](#)

- Download [Gzipped source tarball](#)
- Download [XZ compressed source tarball](#)

- [Python 3.7.8 - June 27, 2020](#)

- Download [macOS 64-bit installer](#)

安装python3.7

- 执行python-3.7.8-amd64.exe
- 注意一些选项： 以下两个必须一定选上， 其他的无所谓， 想点就点吧
 - add python to PATH （将python加入环境变量）
 - Install pip （安装pip）
- 选择一个你记得住的安装路径， 方便开发环境的配置， 建议纯英文路径

PyCharm-强大的python集成开发环境

- 官方网站（科学上网）：<https://www.jetbrains.com/pycharm/>
- PyCharm是python程序开发最友好的、功能最强大的IDE之一，这家公司JetBrains的其他产品在业界也有着非常好的口碑，java开发、html设计与开发都会用到这家的东西。
- 选用PyCharm最主要的原因除了便捷的环境配置和友好的开发界面，还有一点是他兼容匹配了Anaconda（下一步要安装的）管理方案。
- 安装包在env中也有了，执行即可。

PyCharm的简单配置

- 为了方便（偷懒），在配置PyCharm之前，我们先来安装Anaconda。
- 实际上，PyCharm为我们提供一个虚拟的python开发环境，只不过这个环境过于原始，太过简单，很多东西需要我们接着手动配，大家闲来可以自己学习一下。
- 直接用Anaconda的环境省去了很多数据科学和深度学习统一的配置步骤。

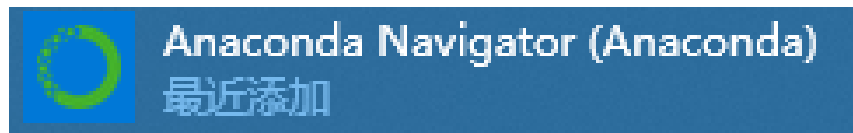
Anaconda-最著名的数据科学平台

- 官方网站（科学上网）：<https://www.anaconda.com/>
- env中有安装包
- Anaconda是致力于数据科学的一整套服务体系平台，（红色我们要用；绿色我们暂时用不到）包括开发环境、第三方库管理、开源项目管理、企业级解决方案，提供了python、R语言支持，包含了这么多机器学习相关工具：



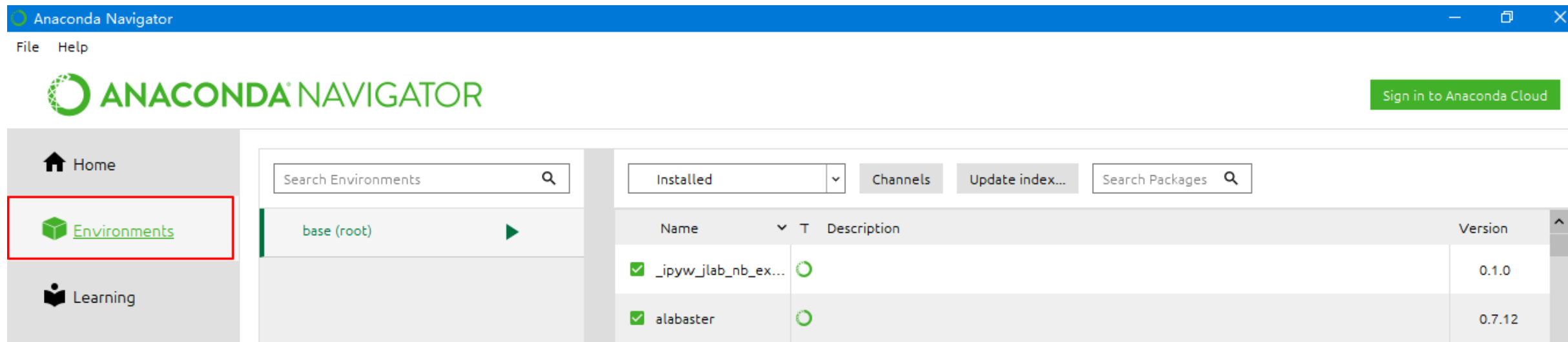
Anaconda安装

- 实际上，安装Anaconda的同时，会附带安装Python，也就是说之前安装的Python3.7的多余的，那当然不是了！！！如果仔细观察可以发现，从安装Anaconda的详细信息中得知，Anaconda会在计算机系统环境变量中找寻python的路径，会在python解释器同目录下安装conda命令，这也就是在安装python3.7时必须一定要add python to PATH的原因。
- 安装好后，其实就有一个这个：
- 点击这个进入配置界面：



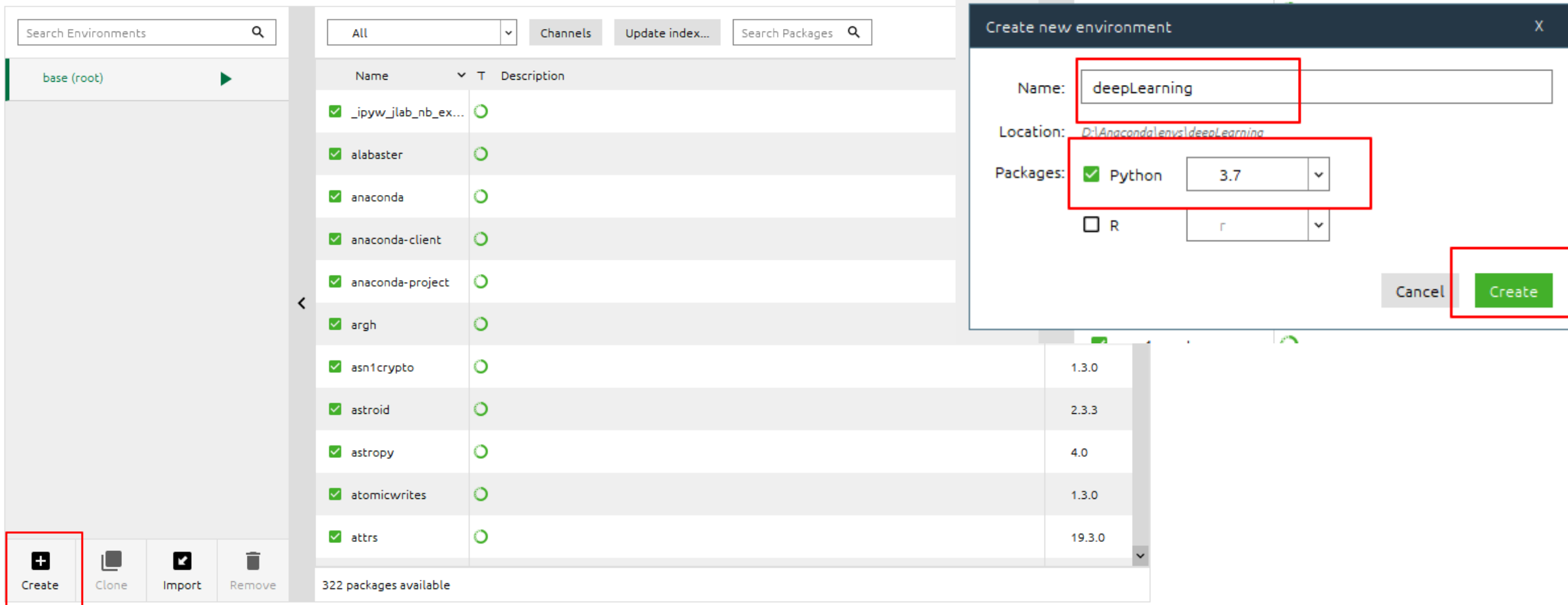
Anaconda配置

- 点击执行后，会不断弹出黑底命令提示符界面，不要管他
- 然后就能看到非常好看的用户界面了，其中jupyter notebook是自带安装好的，这也是一款非常好用的python编写与执行环境，是佳宝同学推荐过的，可以学一手。
- 其实我们用Anaconda最主要的还是虚拟环境支持和包管理，此时默认只有base（root）环境，我们应当创建一个适合我们的环境：



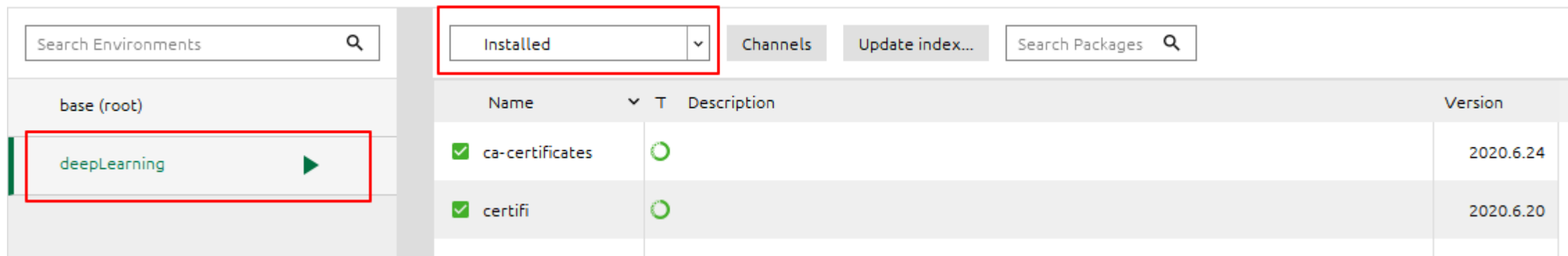
Anaconda配置

- base (root) 一般是不允许直接更改的，为了方便管理我们所需要的库，创建一个新的环境deepLearning，这可能需要一段时间：

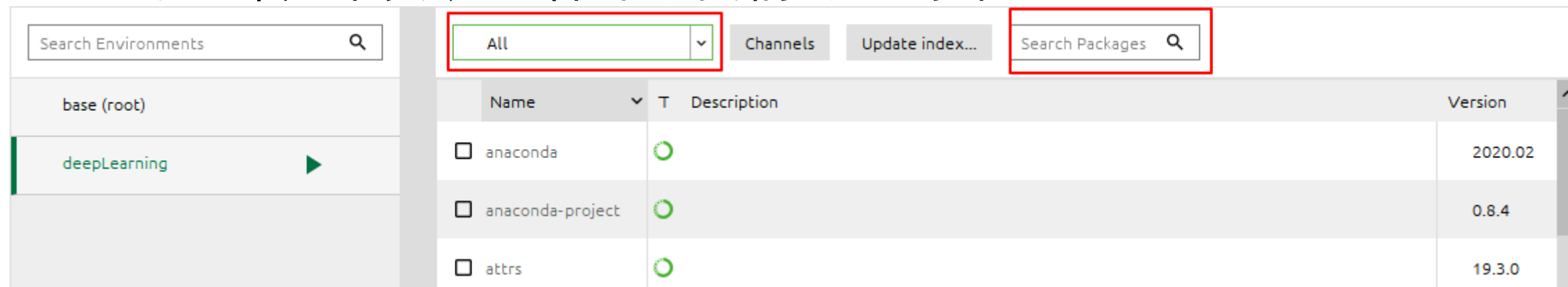


Anaconda配置

- 我们常用的机器学习深度学习的库（也就是包），通常在上张片子的图片里的Environment中管理。
- 这显示的是已安装的：



- 点到全部，来安装或者升级我们需要的库：



Anaconda配置-TensorFlow

- 我们需要的库，请搜索安装，可能会出现搜索不到的问题，请科学上网：首先安装TensorFlow，说真的2.0不好用///

Search Environments

base (root)

deepLearning

All Channels Update index... tensorflow X

Name	T	Description
☐ r-tensorflow		
☒ tensorflow		Tensorflow is a machine learning library.
☐ tensorflow-base		Tensorflow is a machine learning library, base gpu package, tensorflow only.
☐ tensorflow-datasets		Tensorflow/datasets is a library of datasets ready to use with tensorflow.
☐ tensorflow-eigen		Metapackage for selecting a tensorflow variant.
☐ tensorflow-estimator		Tensorflow estimator
☐ tensorflow-gpu		Metapackage for selecting a tensorflow variant.
☐ tensorflow-gpu-base		Tensorflow is a machine learning library, base gpu package, tensorflow only.
☐ tensorflow-hub		A library for transfer learning by reusing parts of tensorflow models.
☐ tensorflow-metadata		Library and standards for schema and statistics.
☐ tensorflow-mkl		Metapackage for selecting a tensorflow variant.
☐ tensorflow-probability		Tensorflow probability is a library for probabilistic reasoning and statistical analysis in tensorflow.

15 packages available matching "tensorflow" 1 package selected

Install Packages

56 packages will be installed

	Name	Unlink	Link	Channel
1	tensorflow	-	2.1.0	pkgs/main
2	*_tflow_select	-	2.2.0	pkgs/main
3	*_absl-py	-	0.9.0	pkgs/main
4	*_astor	-	0.8.0	pkgs/main
5	*_blas	-	1.0	pkgs/main
6	*_blinker	-	1.4	pkgs/main

* indicates the package is a dependency of a selected package

Cancel Apply

0.8.0

Apply Clear

Anaconda配置-Keras、PyTorch

- 其次，相同的方法安装Keras、PyTorch。
- 如果仔细观察可以发现，Keras有些应用已经被安装了，这是TensorFlow的功劳。

All Channels Update index... keras X

Name	T	Description	Version
☒ keras	🟢	Deep learning library for theano and tensorflow	2.4.3
☒ keras-applications	🟢	Applications module of the keras deep learning library.	1.0.8
☐ keras-base	🟢		2.4.3
☐ keras-gpu	🟢	Deep learning library for theano and tensorflow	2.4.3
☒ keras-preprocessing	🟢	Data preprocessing and data augmentation module of the keras deep learning library	1.1.0
☐ r-keras	🟢		2.2.4.1
☐ r-kerasr	🟢		0.6.1

7 packages available matching "keras" 1 package selected

Apply Clear

All Channels Update index... pytorch X

Name	T	Description	Version
☐ _pytorch_select	🟢	Metapackage for establishing variant priority in pytorch variants.	1.2.0
☒ pytorch	🟢	Pytorch is an optimized tensor library for deep learning using gpus and cpus.	1.5.0

2 packages available matching "pytorch" 2 packages selected

Apply Clear

深度学习库-方便我们实现复杂算法

- 当今深度学习非常热门，应运而生了许多方便人们开发的工具集（kits），也已进一步的拉低人工智能入门门槛：会英语的都会搭深度神经网络。
- 那么我们做深度学习，就是要在掌握这些工具方法的基础上，深究算法本质，才能更好的调整参数、优化模型。那么这个片子的目的就是先学会安装、配置、简单了解这几个我们用的包都是干嘛的。
- 接下来的片子里介绍的就是最常用三个库，如要安装别的库，方法都教给你了。

TensorFlow-Google敲响深度学习大门

- TensorFlow是由Google开发的深度学习框架，他是现在最流行的一种深度学习工具，我们后续的研究都是在以TensorFlow为开发工具为基础的。
- 他是众多更高层封装工具集（库、包）的后端实现。
- 推荐关注TensorFlow的官方微信公众号。
- TensorFlow官方网站：<https://www.tensorflow.org/?hl=zh-cn>

Keras、PyTorch-高级封装

- 这两个库都是进一步封装TensorFlow的深度学习工具，这两个实际上不分伯仲，都是简单易上手的构建网络、调整参数的工具。
- Keras 官方网站: <https://keras.io/>
- PyTorch官方网站: <https://pytorch.org/>

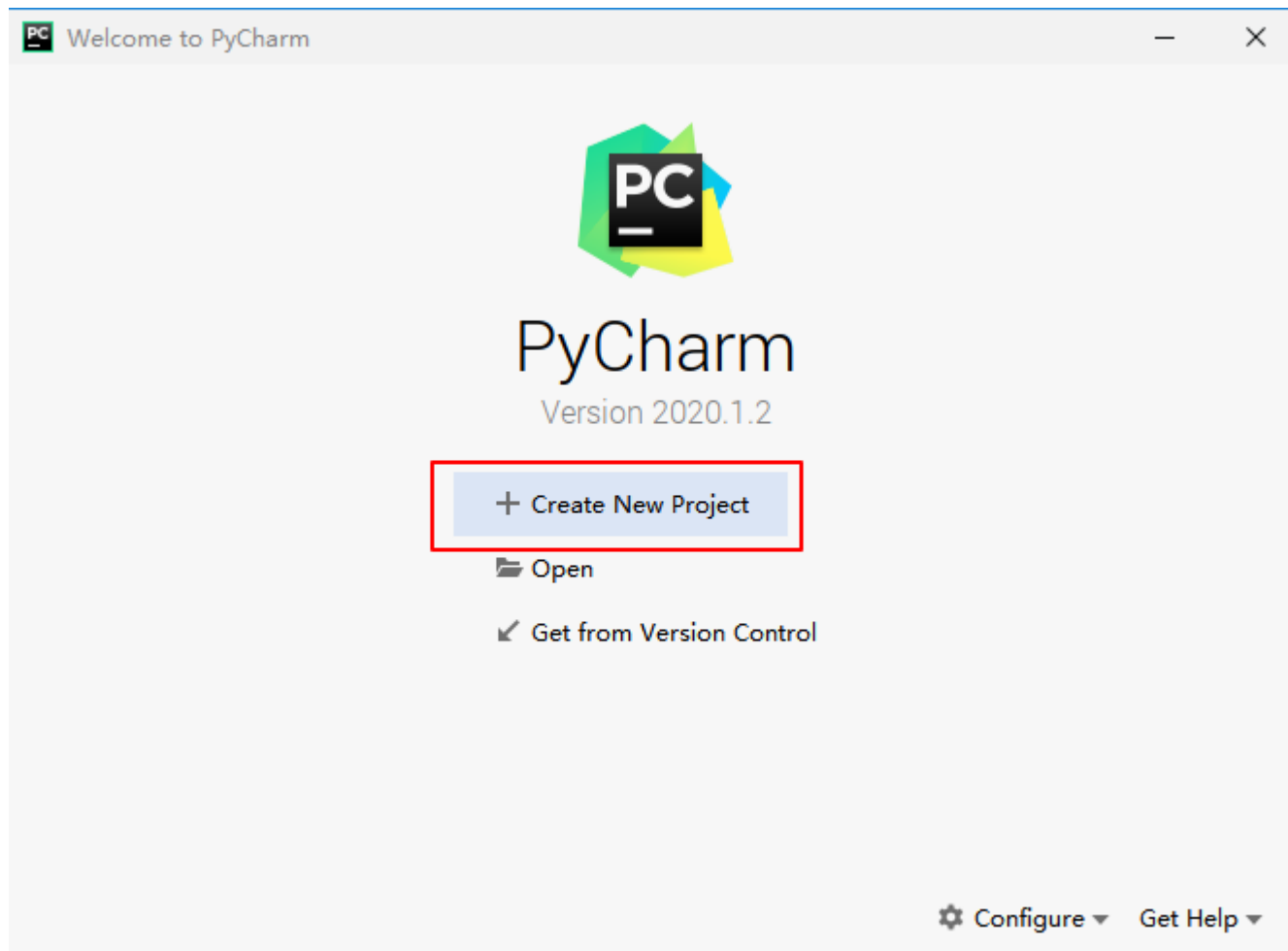
PyCharm的完成配置

- 最后，我们回到集成开发环境PyCharm，完成最终配置，开始深度学习模型构建。
- 其实，使用输入pip命令安装库并配置的方法也是一种方法，你也要通过查看相关教程学会。
- 执行PyCharm：



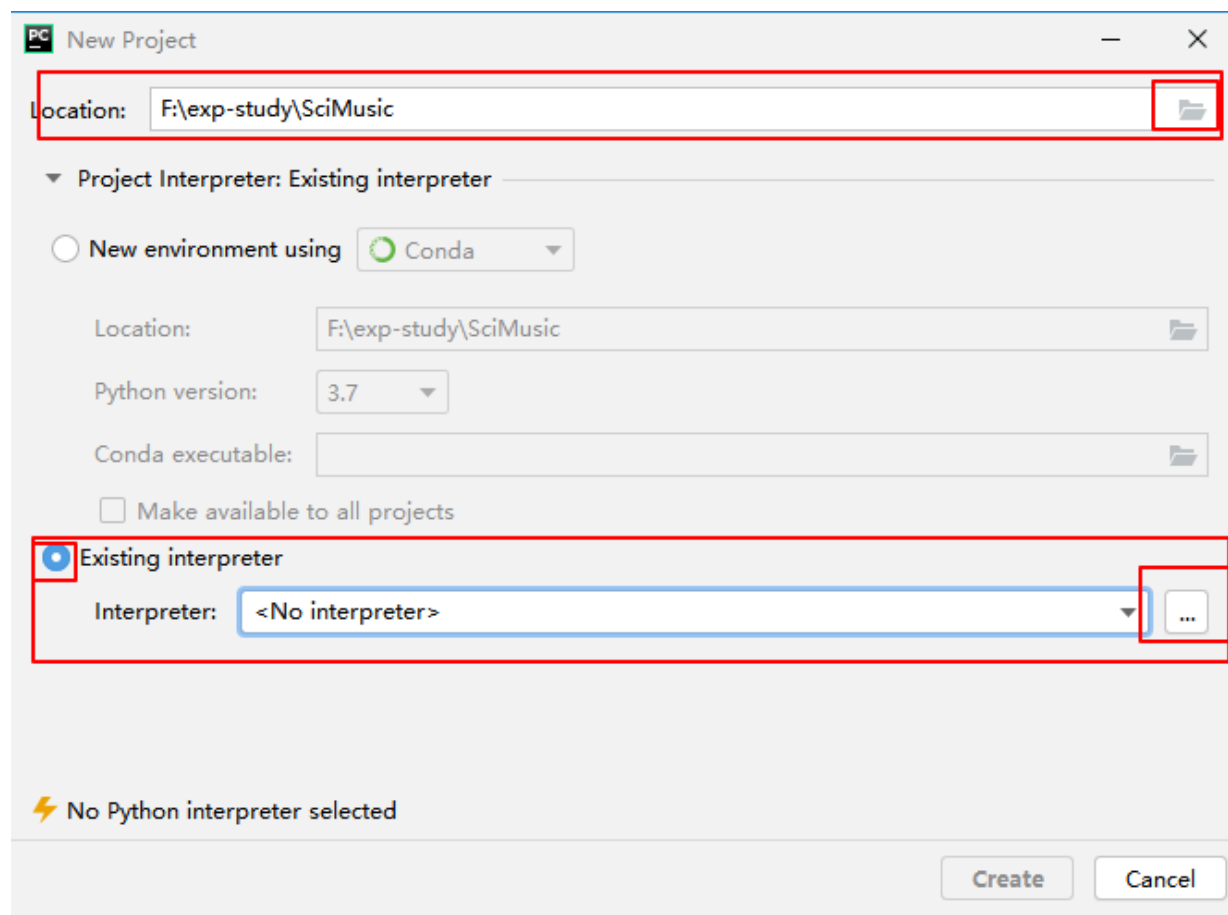
PyCharm的完成配置

- 创建一个工作项目：



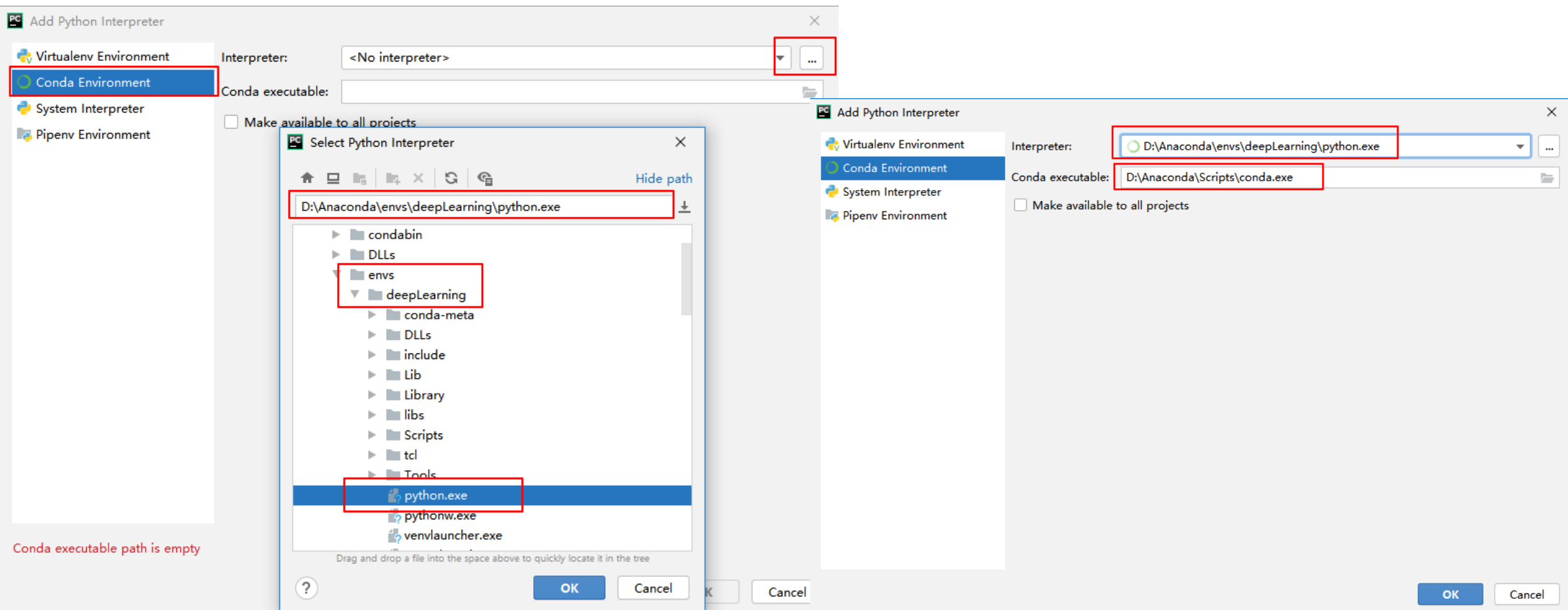
PyCharm的完成配置

- 选择项目保存路径，建议全英文无空格路径，且空文件夹
- 选择现有Anaconda环境，选择解释器



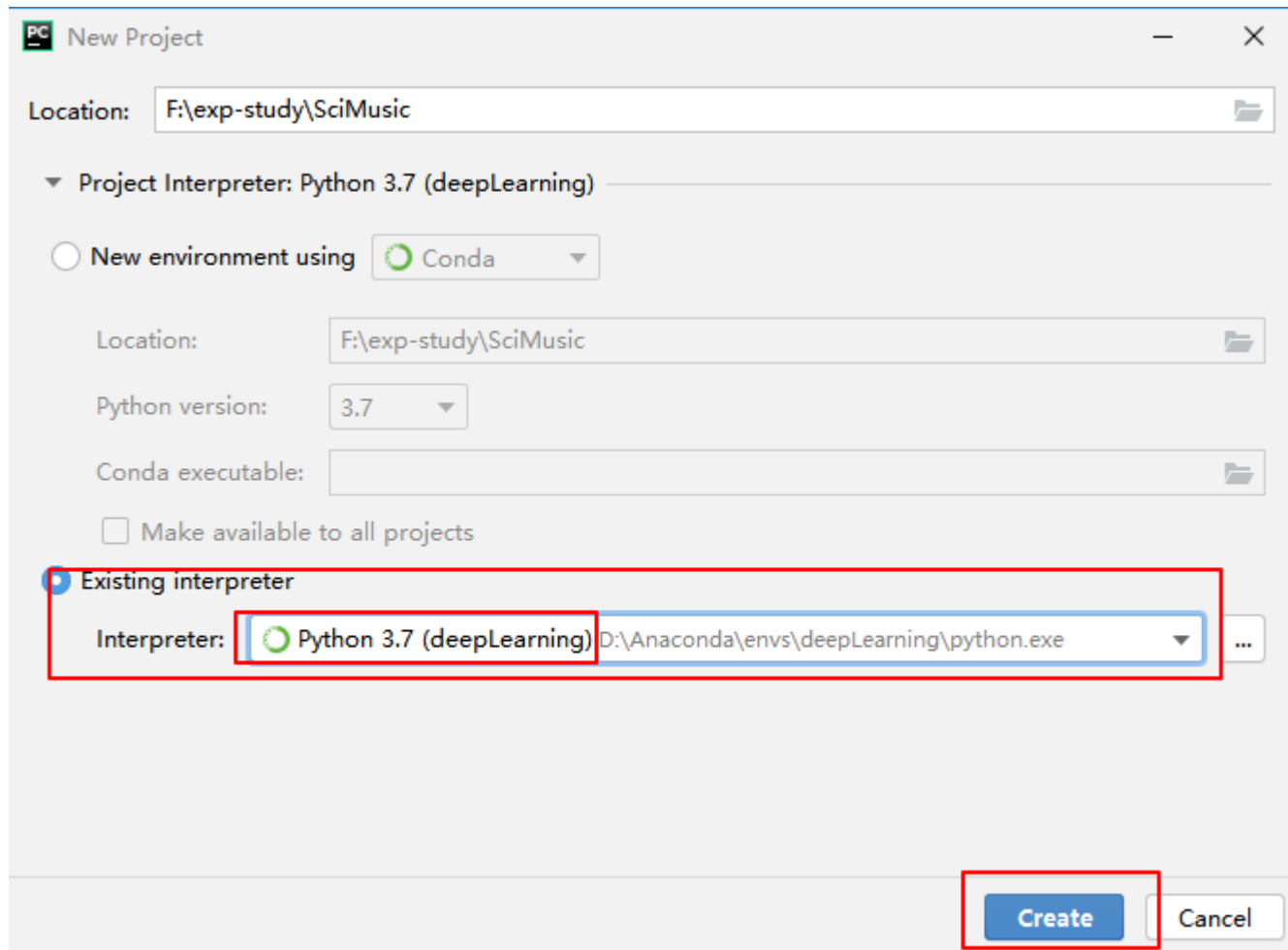
PyCharm的完成配置

- 选择我们刚刚创建好的Anaconda环境，找到python.exe解释器的位置，然后会发现，Anaconda自动定位可执行conda



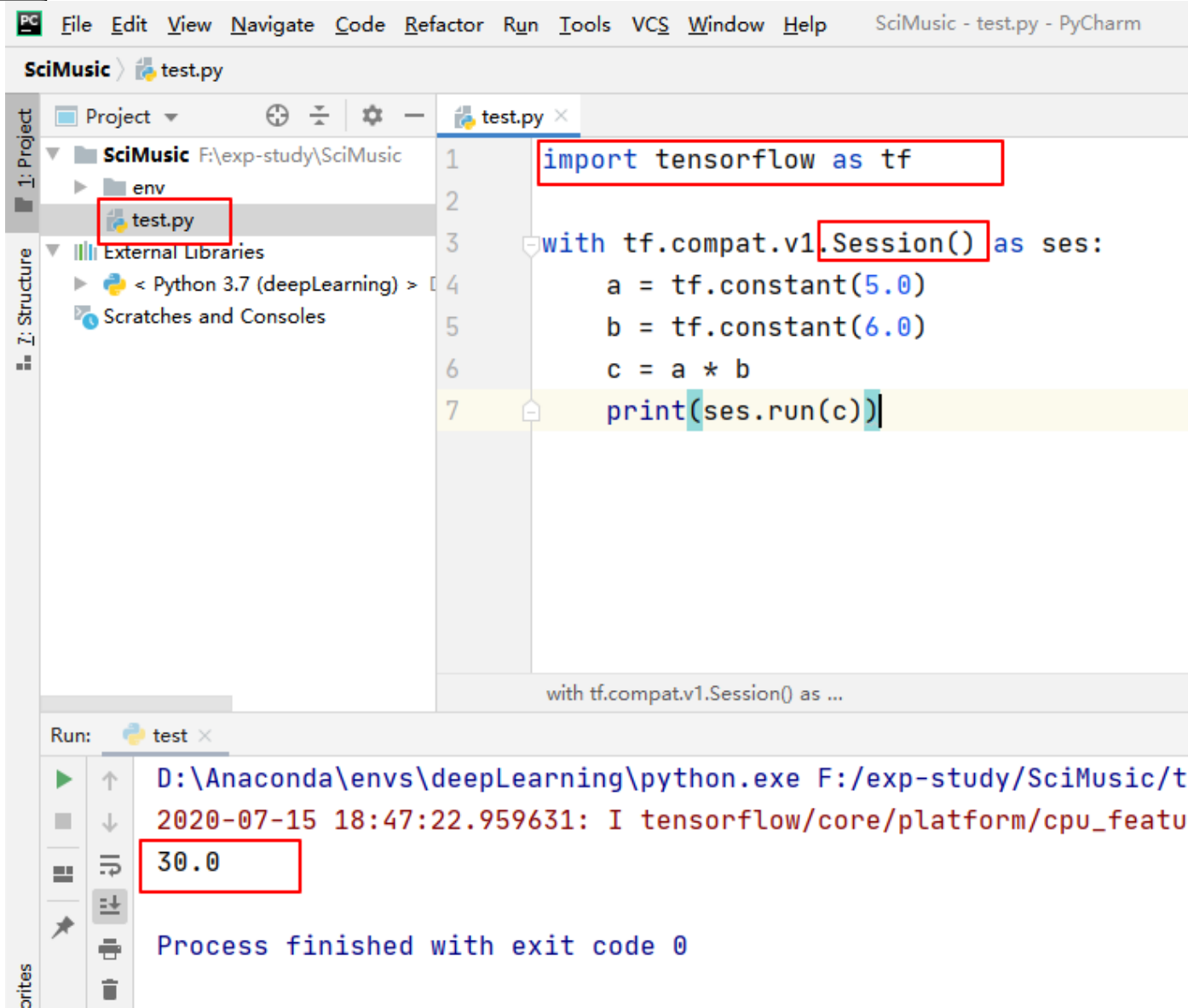
PyCharm的完成配置

- 创建配置好的环境



PyCharm的完成配置

- 测试TensorFlow配置完成, 并成功运行



完了

操作过程中有任何问题，随时找我 

Tel: 18729002629

QQ: 1535367354