Writing Neural Network Code: Introduction to TensorFlow, Hands-On

Class 3: TensorFlow Hands-On
Part 1: Hello World!

May 13, 2020

Charles J. Lord, PE
President, Consultant, Trainer
Blue Ridge Advanced Design and Automation

Presented by:

This Week's Agenda

5/11 A Brief History of Artificial Neural Networks

5/12 Neural Network Simulation and Programming

5/13 TensorFlow Hands-On Part 1: Hello World!

5/14 TensorFlow Hands-On Part 2: Defining and Building Your Network

5/15 TensorFlow Hands-On Part 3: Teaching and Testing and Conclusion

This Week's Agenda

5/11 A Brief History of Artificial Neural Networks

5/12 Neural Network Simulation and Programming

5/13 TensorFlow Hands-On Part 1: Hello World!

5/14 TensorFlow Hands-On Part 2: Defining and Building Your Network

5/15 TensorFlow Hands-On Part 3: Teaching and Testing and Conclusion

Lessons Learned – Version Updates

Life comes at you fast.

Most Literature on TensorFlow is Out of Date

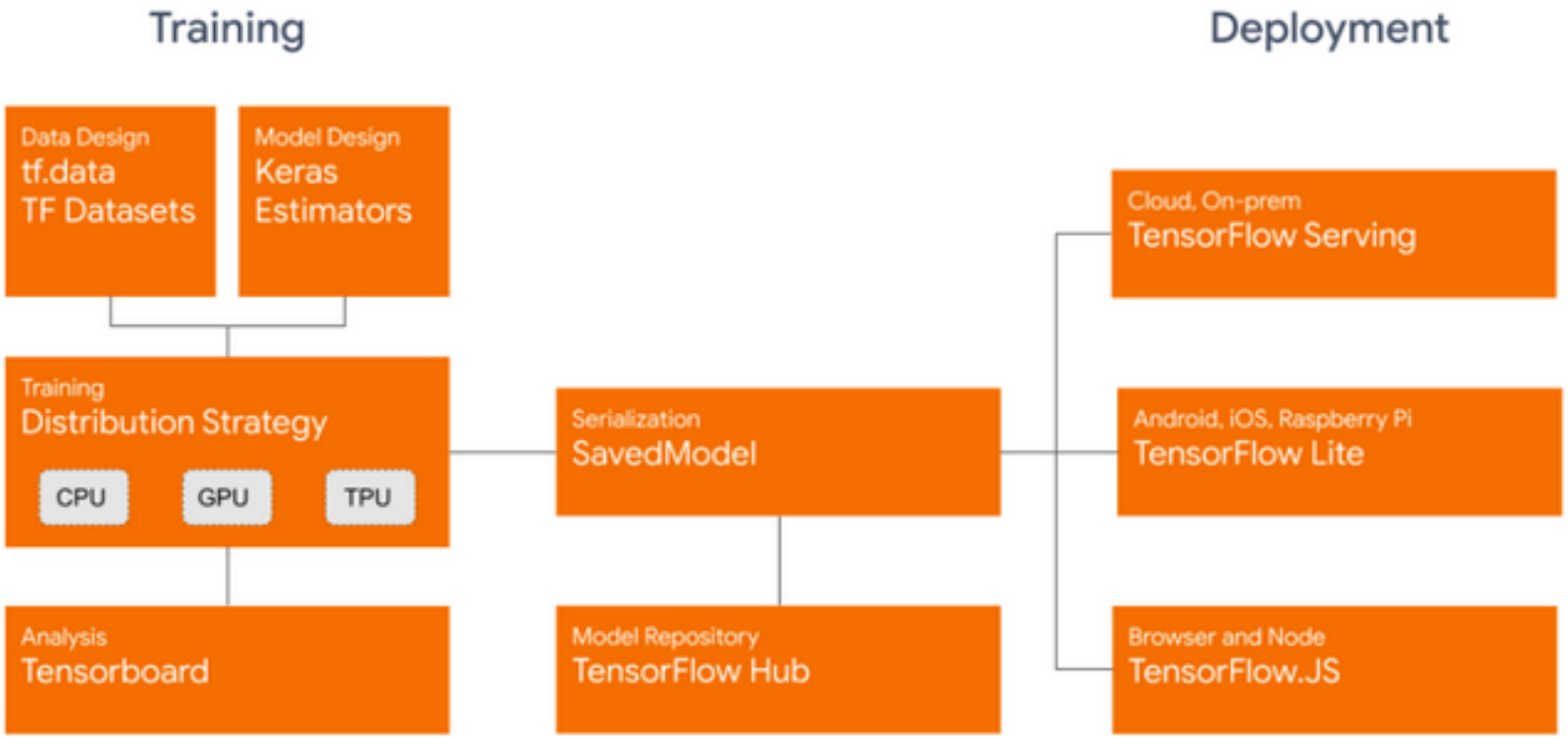
- TensorFlow made major changes from version 1.x to 2.x in late October 2019.
- This class was originally designed in 1.x
- All information here should be correct for v2
  - TRUST YET VERIFY
- If a book, article, or website mentions sessions, disregard it.

Question 1 – Experience with TensorFlow (including lite)?

Major Updates

- “tf.contrib” is gone and mostly integrated
- Easy model building with Keras (now fully integrated) and *eager execution* - no more build then run sessions.
- Robust model deployment in production on any platform.
- Standard databases are integral
- Powerful experimentation for research.
- API simplification by reducing duplication and removing deprecated endpoints.

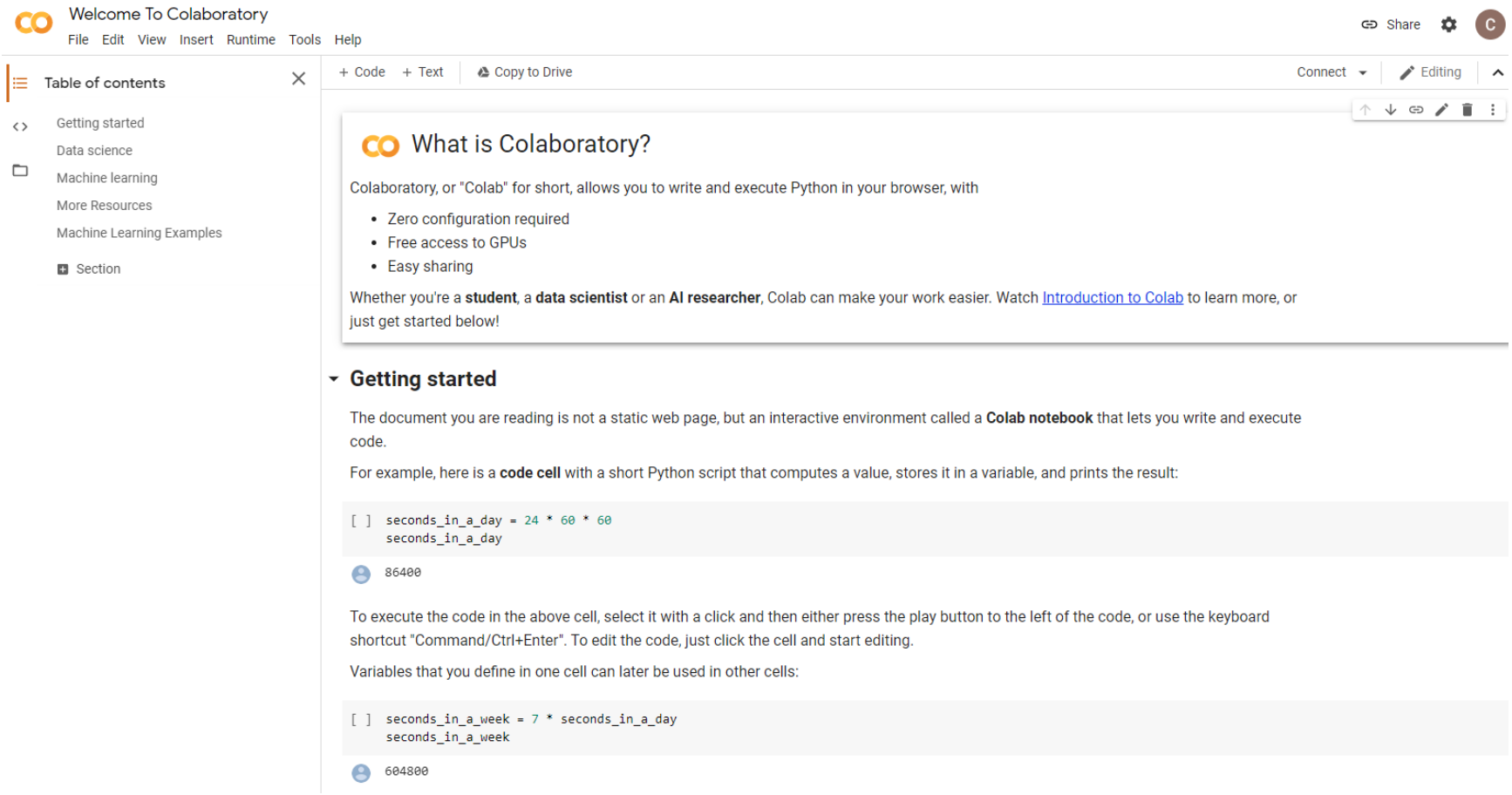
TensorFlow 2.x Framework



Installing TensorFlow 2

- There are many ways to install (optional) and run TensorFlow
- Simplest for just testing out a few simple concepts – Google Colaboratory (Colab)
- Direct on a python installation (command line or IDE) on PC/Mac/Linux
- In a development environment, typically that runs in a virtual machine (Docker, PyCharm, etc)

https://colab.research.google.com/notebooks/welcome.ipynb



Click on the [ ] to run the code

Getting started

The document you are reading is not a static web page, but an interactive environment called a **Colab notebook** that lets you write and execute code.

For example, here is a **code cell** with a short Python script that computes a value, stores it in a variable, and prints the result:

```
[ ]  seconds_in_a_day = 24 * 60 * 60
     seconds_in_a_day
```

```
86400
```

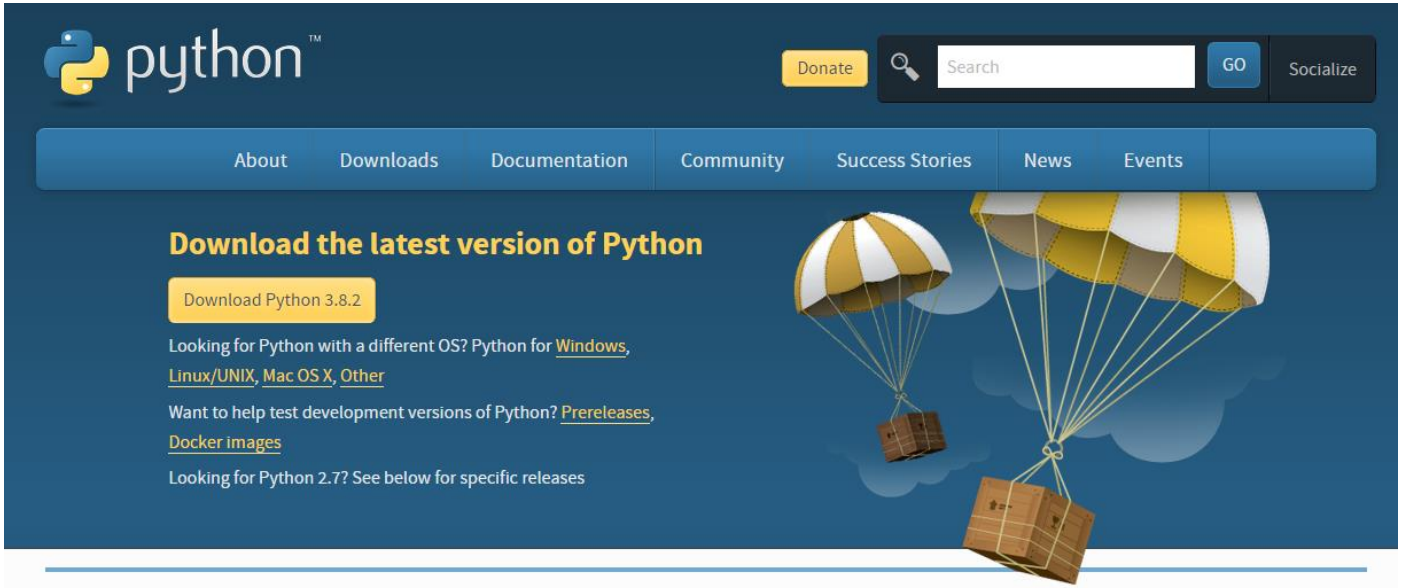
To execute the code in the above cell, select it with a click and then either press the play button to the left of the code, or use the keyboard shortcut "Command/Ctrl+Enter". To edit the code, just click the cell and start editing.

Variables that you define in one cell can later be used in other cells:

```
[ ]  seconds_in_a_week = 7 * seconds_in_a_day
     seconds_in_a_week
```

```
604800
```

https://www.python.org/downloads/

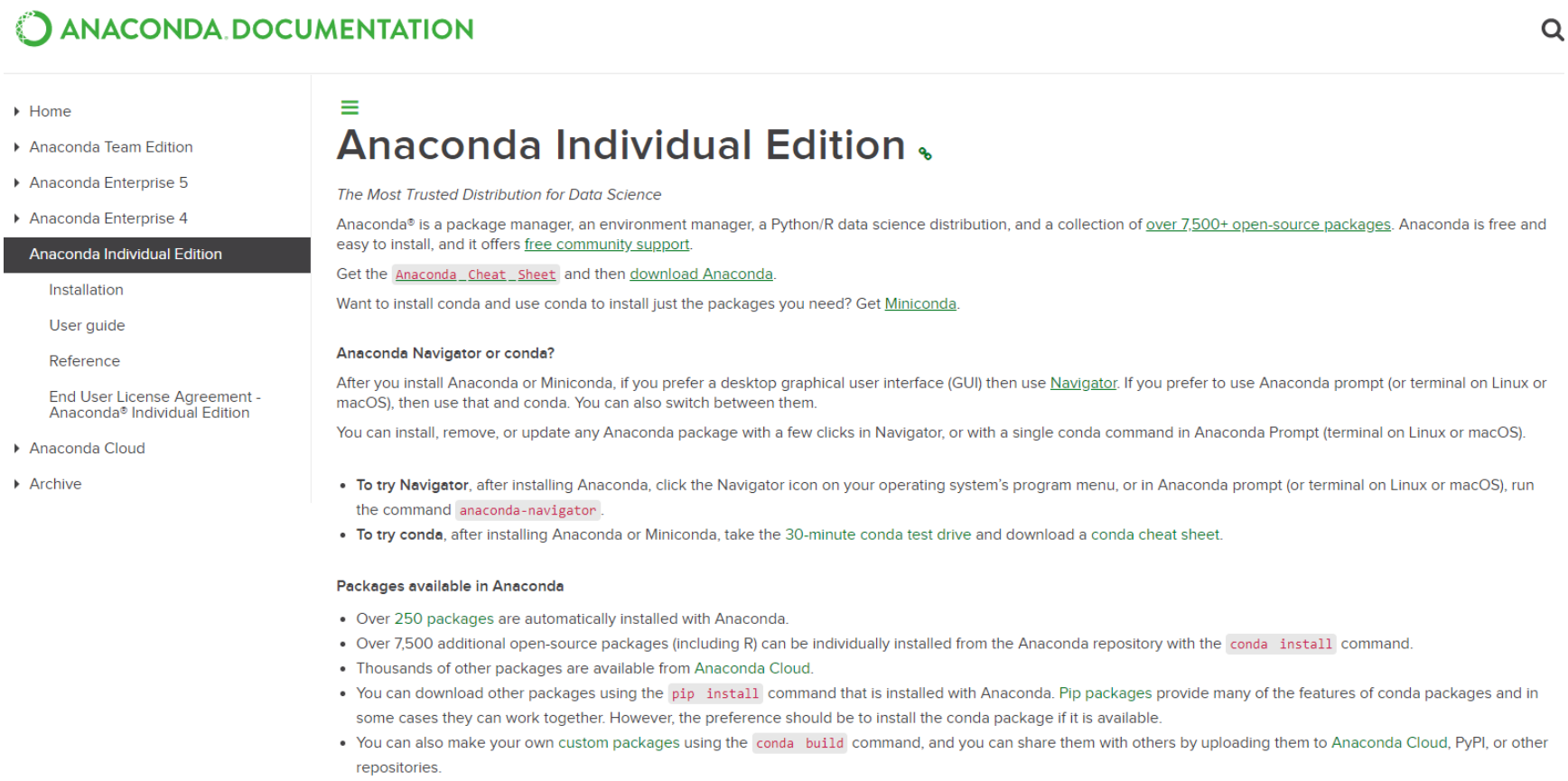


Active Python Releases

For more information visit the Python Developer's Guide.

Python version	Maintenance status	First released	End of support	Release schedule
3.8	bugfix	2019-10-14	2024-10	PEP 569
3.7	bugfix	2018-06-27	2023-06-27	PEP 537
3.6	security	2016-12-23	2021-12-23	PEP 494
3.5	security	2015-09-13	2020-09-13	PEP 478
2.7	end-of-life	2010-07-03	2020-01-01	PEP 373

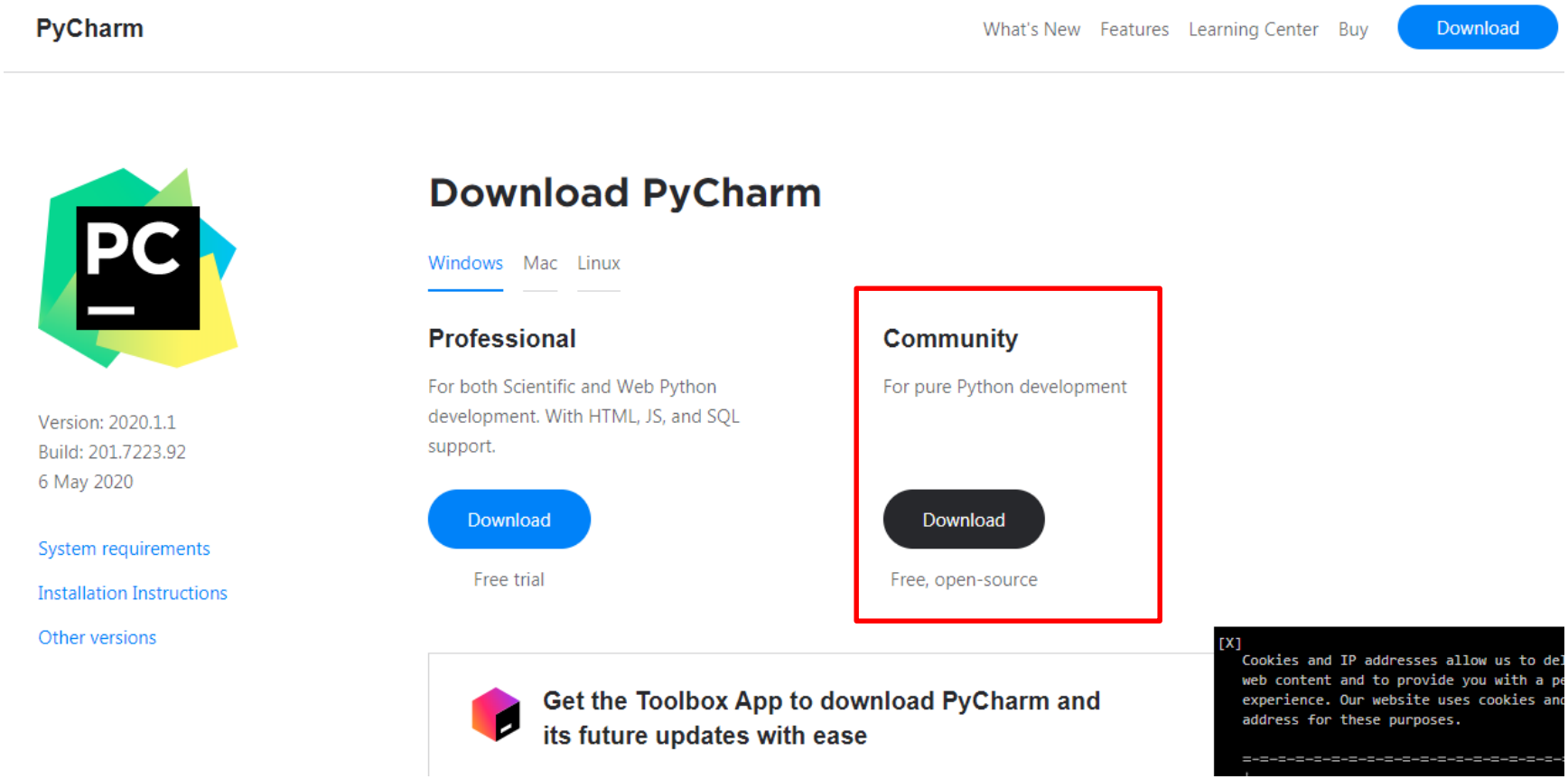
https://docs.anaconda.com/anaconda/



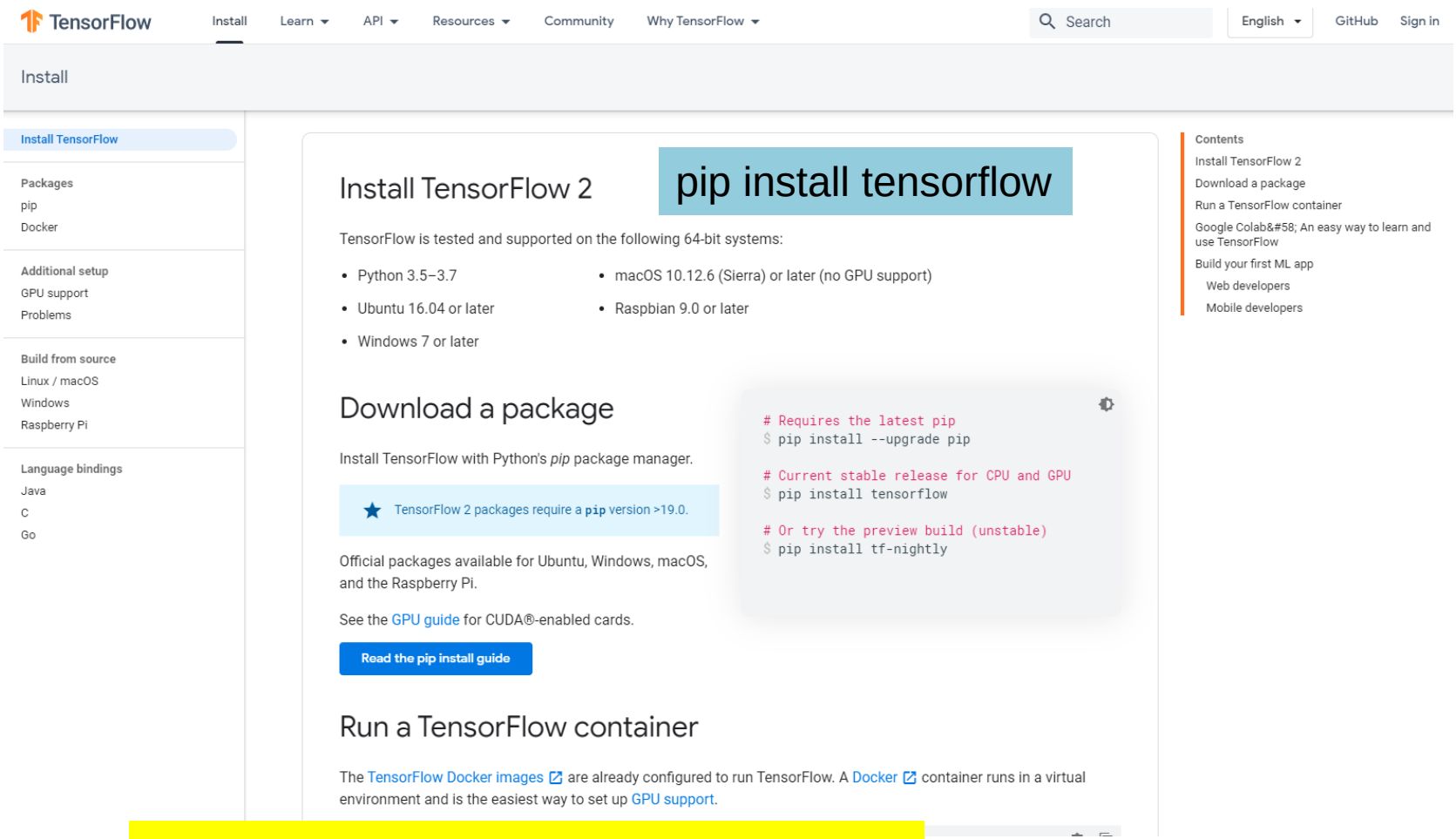
Our Setup

- Python 3.8.3 from python.org
- PyCharm IDE and virtual environments
- Will add numpy and TensorFlow elements
- We will also look at Colab
- We will also set up the environment for the NVIDIA Jetson NANO

https://www.jetbrains.com/pycharm/download/#section=windows



https://www.tensorflow.org/install



Question 2 – What does “PIP” stand for?

API Documentation

TensorFlow has APIs available in several languages both for constructing and executing a TensorFlow graph. The Python API is at present the most complete and the easiest to use, but other language APIs may be easier to integrate into projects and may offer some performance advantages in graph execution.

A word of caution: the APIs in languages other than Python are not yet covered by the API stability promises.

- Python
- JavaScript
- C++
- Java
- Go
- Swift (Early Release)

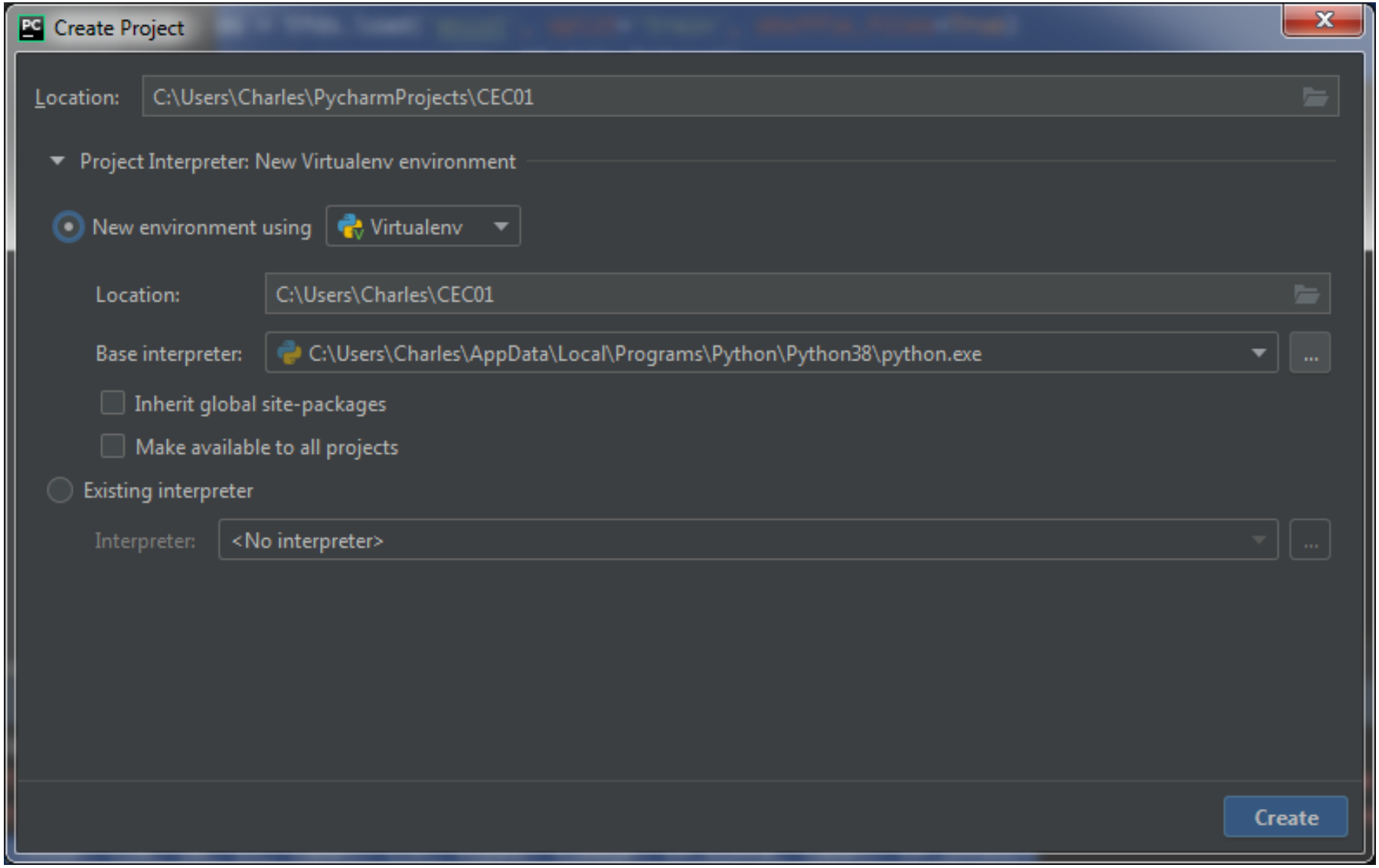
We encourage the community to develop and maintain support for other languages with the approach recommended by the TensorFlow maintainers. For example, see the bindings for:

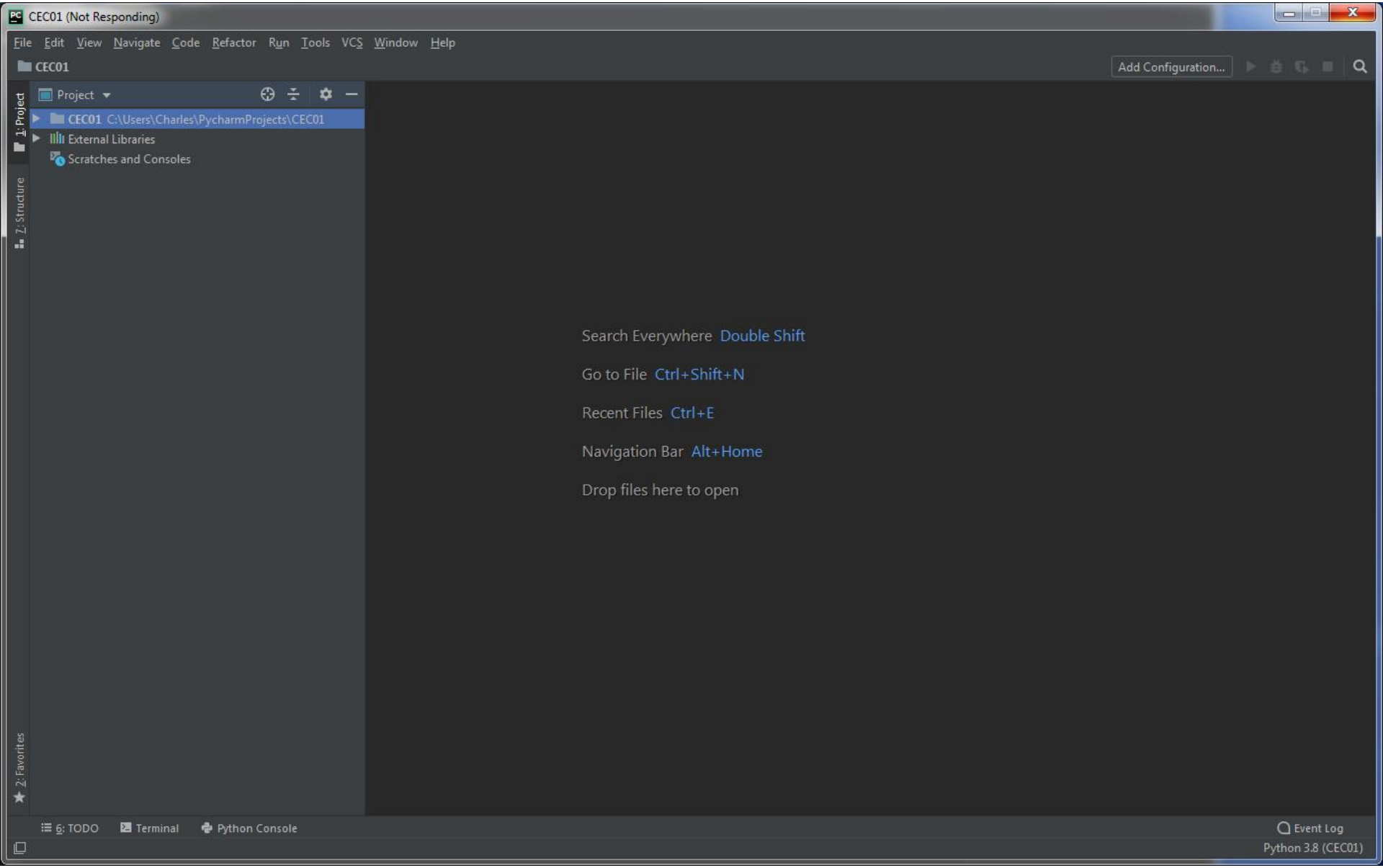
- C#: TensorFlowSharp and TensorFlow.NET,
- Haskell,
- Julia,
- MATLAB,
- R,
- Ruby,
- Rust, and
- Scala.

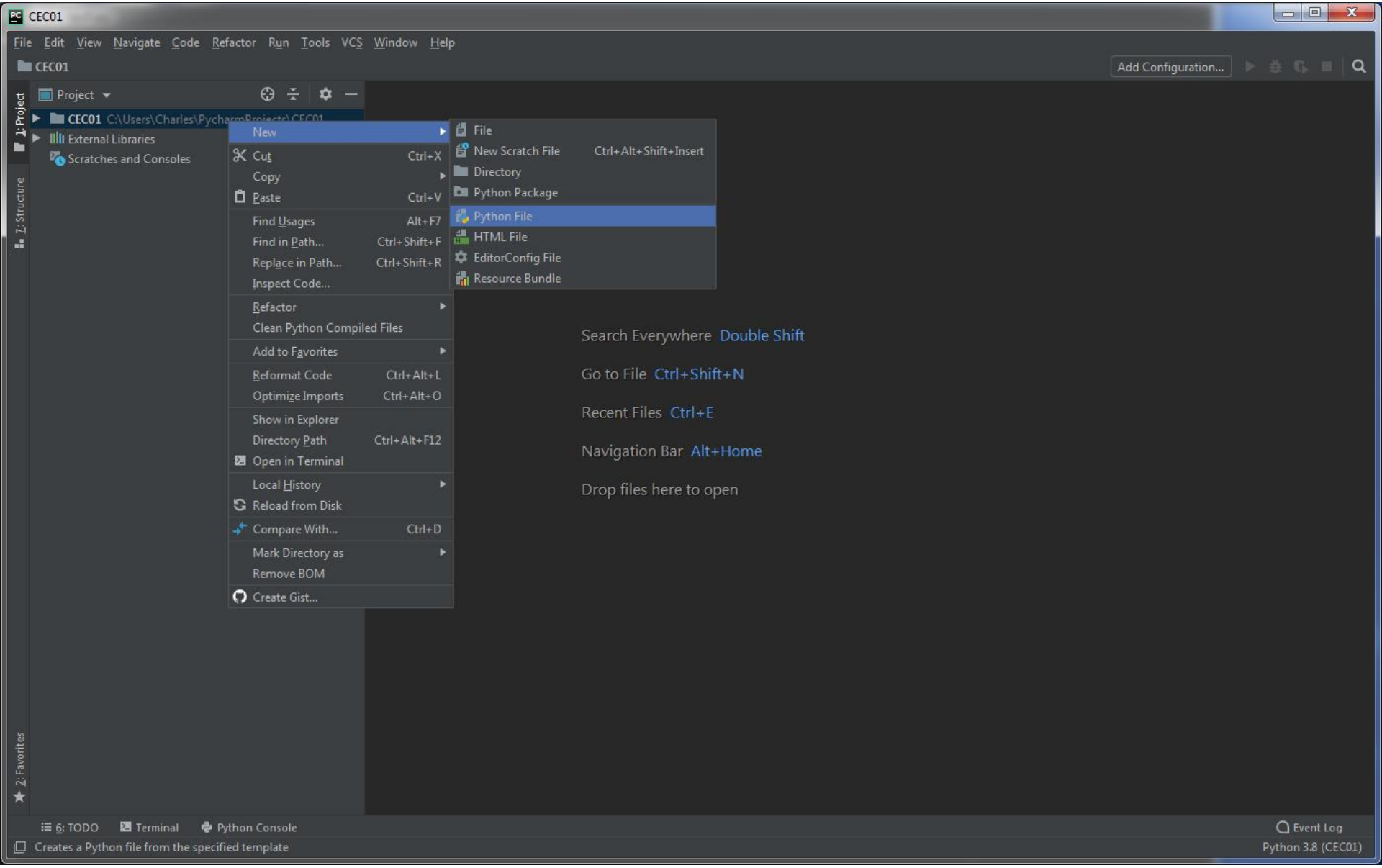
https://www.tensorflow.org/api_docs

We also provide the C++ API reference for TensorFlow Serving:

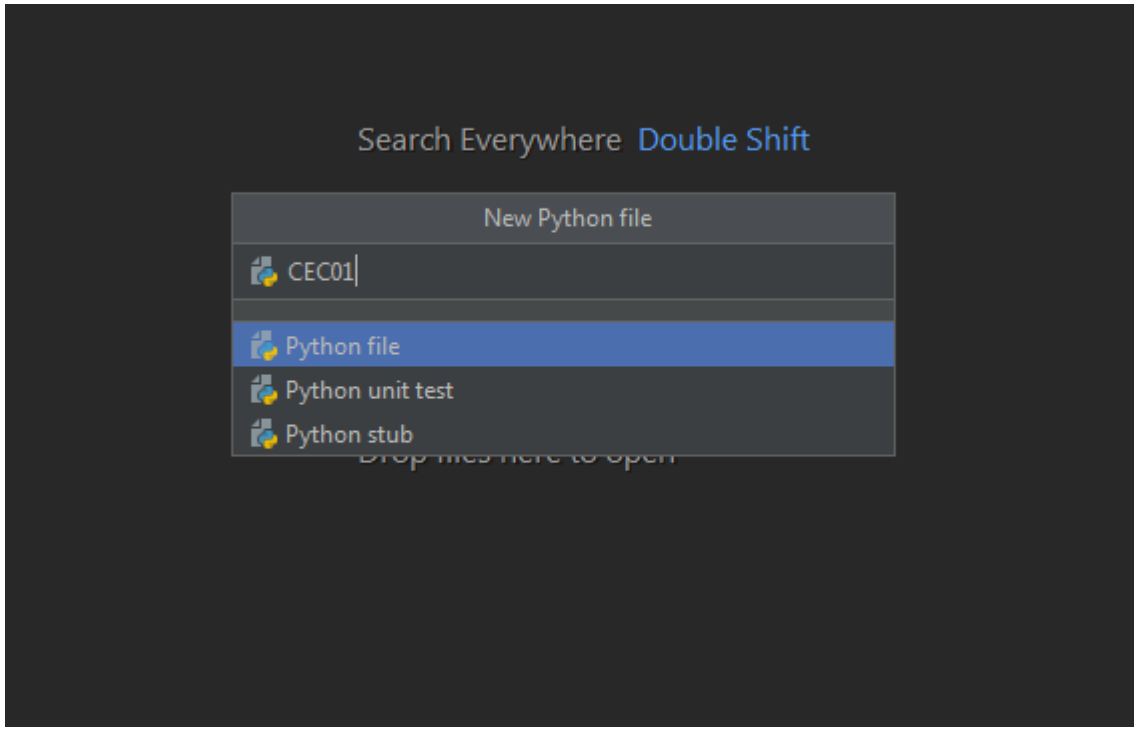
- TensorFlow Serving


Create Project
Location: C:\Users\Charles\PycharmProjects\CEC01
Project Interpreter: New Virtualenv environment
New environment using Virtualenv
Location: C:\Users\Charles\CEC01
Base interpreter: C:\Users\Charles\AppData\Local\Programs\Python\Python38\python.exe
Inherit global site-packages
Make available to all projects
Existing interpreter
Interpreter: <No interpreter>
Create

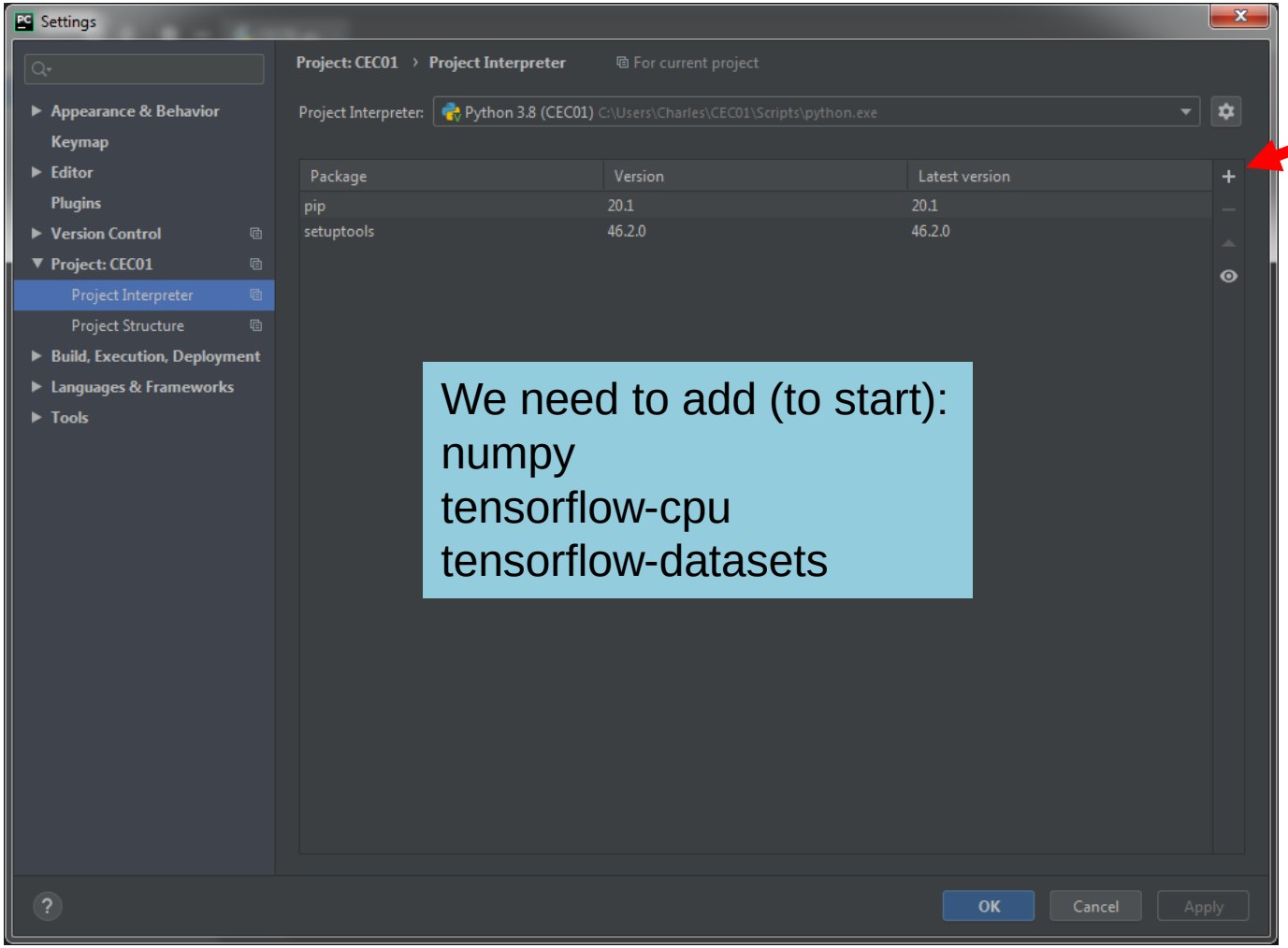

CEC01 (Not Responding)
File Edit View Navigate Code Refactor Run Tools VCS Window Help
CEC01
Add Configuration...
Project
CEC01 C:\Users\Charles\PycharmProjects\CEC01
External Libraries
Scratches and Consoles
1: Project
7: Structure
2: Favorites
Search Everywhere Double Shift
Go to File Ctrl+Shift+N
Recent Files Ctrl+E
Navigation Bar Alt+Home
Drop files here to open
6: TODO
Terminal
Python Console
Event Log
Python 3.8 (CEC01)

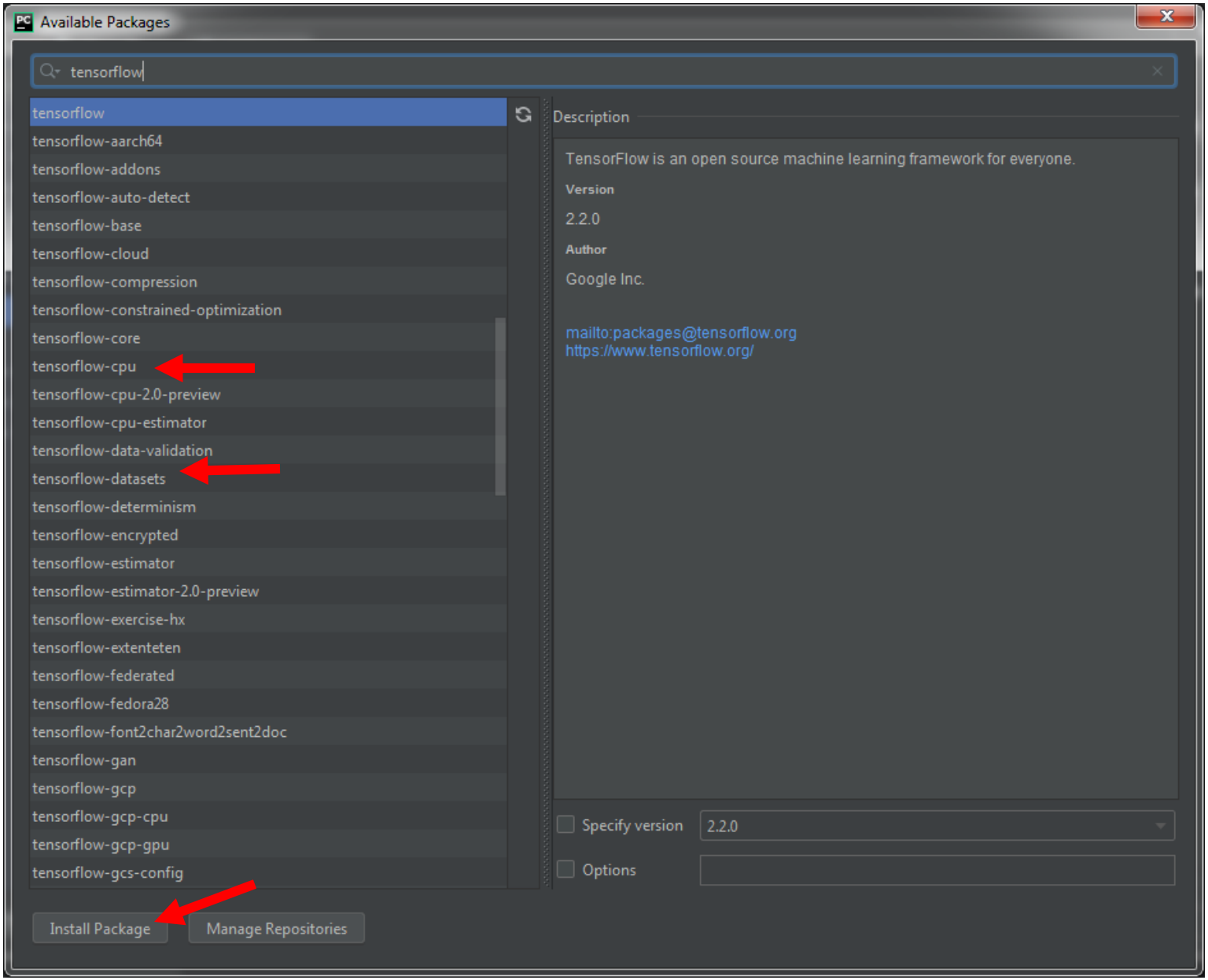

CEC01
File Edit View Navigate Code Refactor Run Tools VCS Window Help
CEC01
Add Configuration...
Project
CEC01 C:\Users\Charles\PycharmProjects\CEC01
External Libraries
Scratches and Consoles
New
Cut Ctrl+X
Copy
Paste Ctrl+V
Find Usages Alt+F7
Find in Path... Ctrl+Shift+F
Replace in Path... Ctrl+Shift+R
Inspect Code...
Refactor
Clean Python Compiled Files
Add to Favorites
Reformat Code Ctrl+Alt+L
Optimize Imports Ctrl+Alt+O
Show in Explorer
Directory Path Ctrl+Alt+F12
Open in Terminal
Local History
Reload from Disk
Compare With... Ctrl+D
Mark Directory as
Remove BOM
Create Gist...
File
New Scratch File Ctrl+Alt+Shift+Insert
Directory
Python Package
Python File
HTML File
EditorConfig File
Resource Bundle
Search Everywhere Double Shift
Go to File Ctrl+Shift+N
Recent Files Ctrl+E
Navigation Bar Alt+Home
Drop files here to open
1: Project
7: Structure
2: Favorites
6: TODO
Terminal
Python Console
Event Log
Creates a Python file from the specified template
Python 3.8 (CEC01)

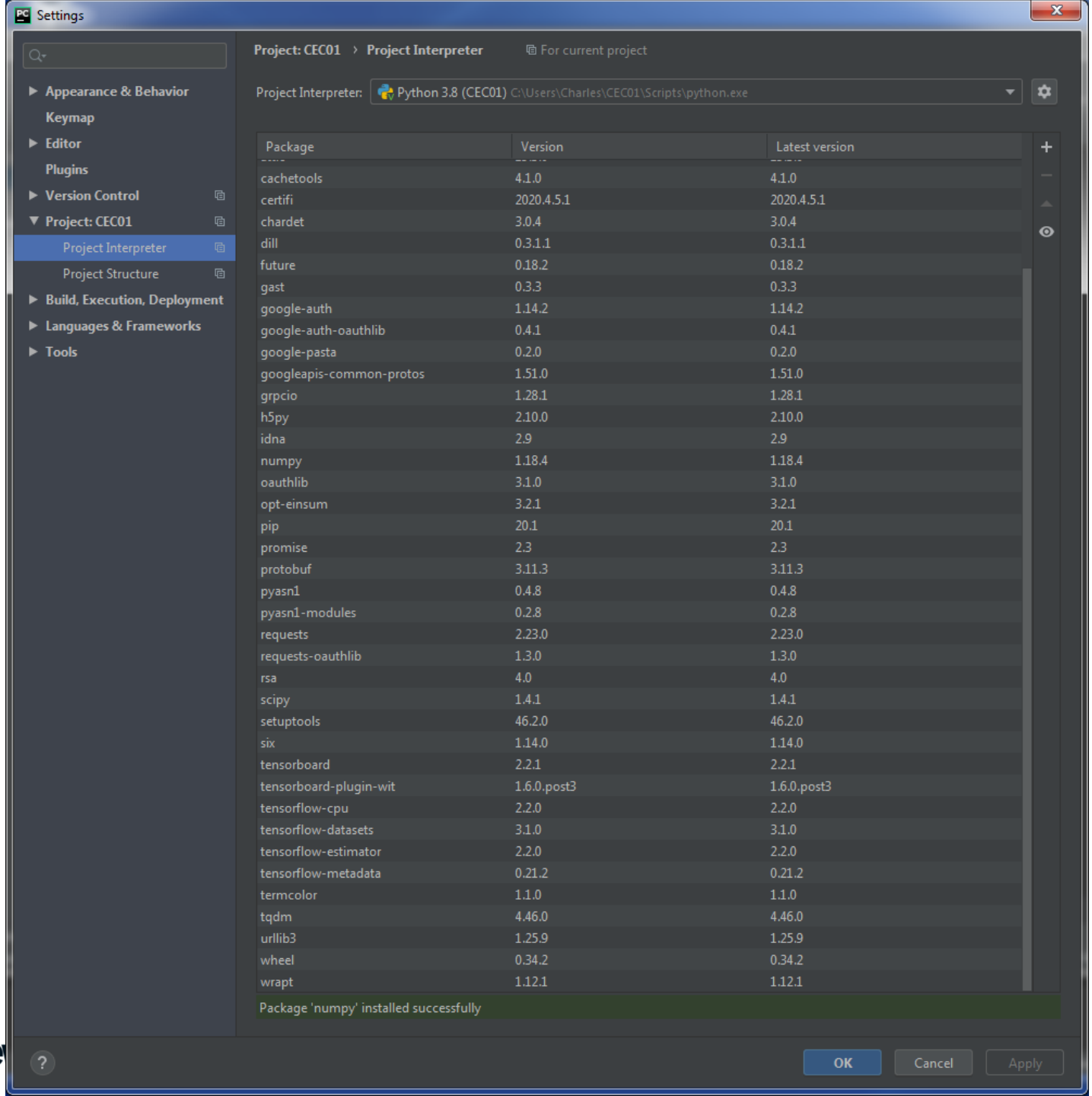
Name Our File
(.py added automatically)



We Need Our Environment!
Click on file, settings




Available Packages
tensorflow
tensorflow
tensorflow-aarch64
tensorflow-addons
tensorflow-auto-detect
tensorflow-base
tensorflow-cloud
tensorflow-compression
tensorflow-constrained-optimization
tensorflow-core
tensorflow-cpu
tensorflow-cpu-2.0-preview
tensorflow-cpu-estimator
tensorflow-data-validation
tensorflow-datasets
tensorflow-determinism
tensorflow-encrypted
tensorflow-estimator
tensorflow-estimator-2.0-preview
tensorflow-exercise-hx
tensorflow-extenteten
tensorflow-federated
tensorflow-fedora28
tensorflow-font2char2word2sent2doc
tensorflow-gan
tensorflow-gcp
tensorflow-gcp-cpu
tensorflow-gcp-gpu
tensorflow-gcs-config
Description
TensorFlow is an open source machine learning framework for everyone.
Version
2.2.0
Author
Google Inc.
mailto:packages@tensorflow.org
https://www.tensorflow.org/
Specify version
2.2.0
Options
Install Package
Manage Repositories

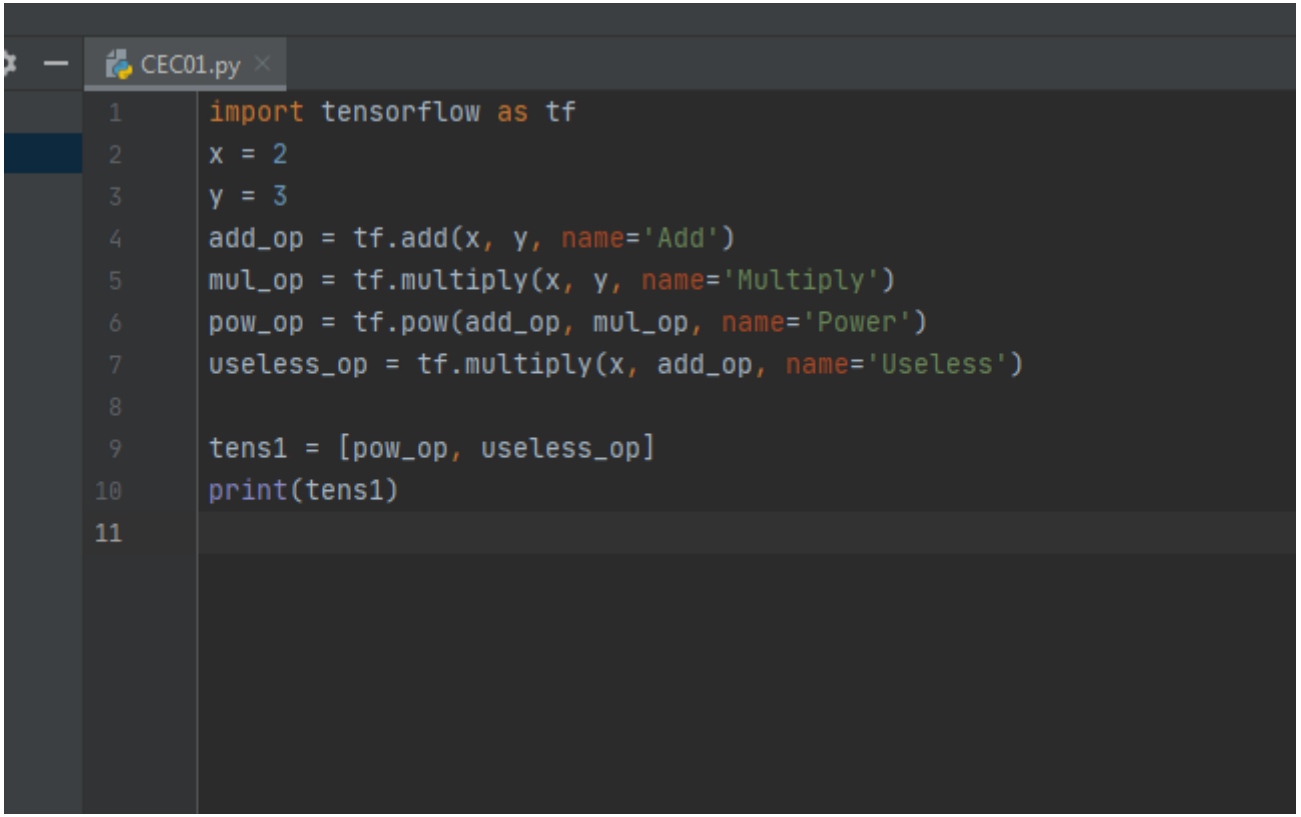


Short Sample Code

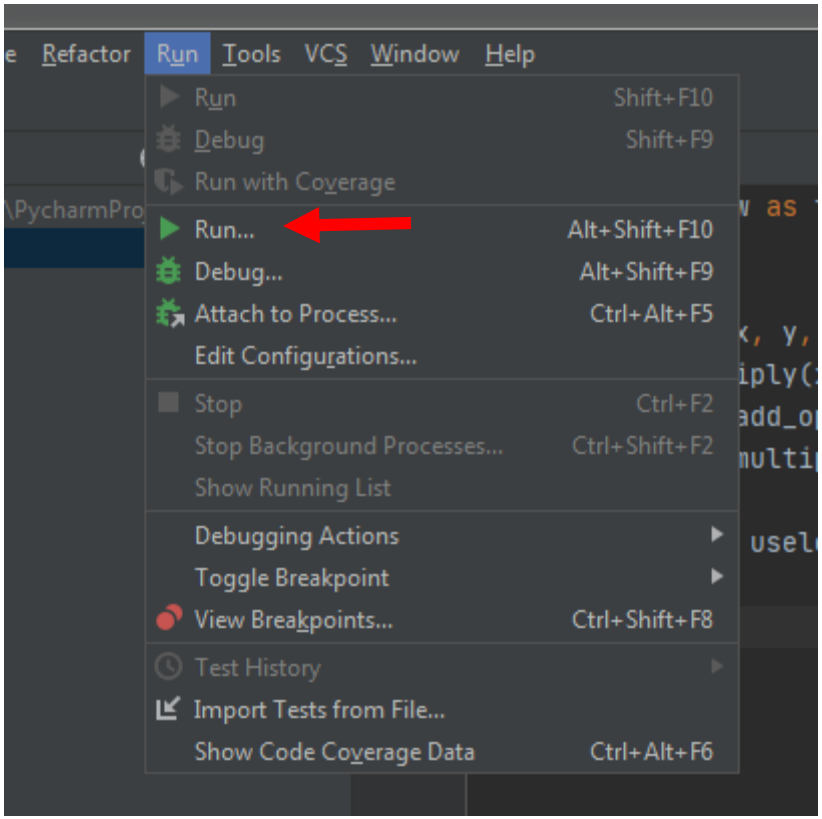
```
import tensorflow as tf
x = 2
y = 3
add_op = tf.add(x, y, name='Add')
mul_op = tf.multiply(x, y, name='Multiply')
pow_op = tf.pow(add_op, mul_op, name='Power')
useless_op = tf.multiply(x, add_op, name='Useless')

tens1 = [pow_op, useless_op]
print(tens1)
```

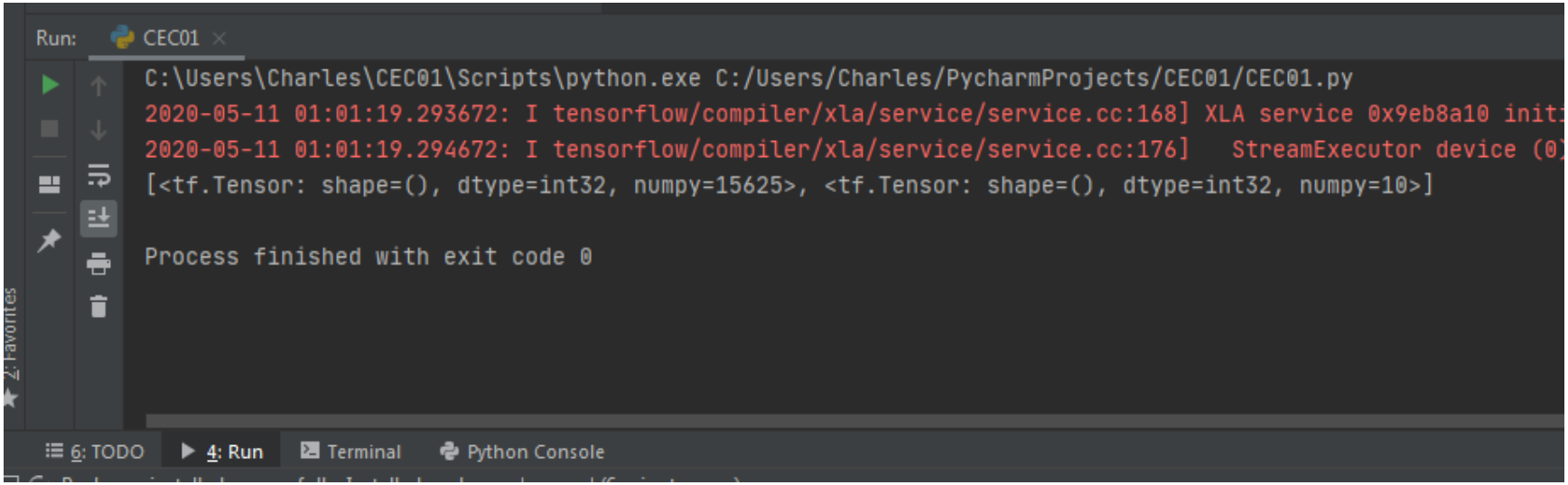
Color Coded!


CEC01.py
1 import tensorflow as tf
2 x = 2
3 y = 3
4 add_op = tf.add(x, y, name='Add')
5 mul_op = tf.multiply(x, y, name='Multiply')
6 pow_op = tf.pow(add_op, mul_op, name='Power')
7 useless_op = tf.multiply(x, add_op, name='Useless')
8
9 tens1 = [pow_op, useless_op]
10 print(tens1)
11

Run , Run


Refactor
Run
Tools
VCS
Window
Help
Run Shift+F10
Debug Shift+F9
Run with Coverage
Run... Alt+Shift+F10
Debug... Alt+Shift+F9
Attach to Process... Ctrl+Alt+F5
Edit Configurations...
Stop Ctrl+F2
Stop Background Processes... Ctrl+Shift+F2
Show Running List
Debugging Actions
Toggle Breakpoint
View Breakpoints... Ctrl+Shift+F8
Test History
Import Tests from File...
Show Code Coverage Data Ctrl+Alt+F6

Our Output



Question 3 – What is this output called and what is its shape?

Tomorrow!

- We will take what we learned on Tuesday about the mathematics of a CNN and apply it to TensorFlow
- We will also look at how the new enhancements to TF have sped up and simplified the definition of our network definition

This Week's Agenda

5/11 A Brief History of Artificial Neural Networks

5/12 Neural Network Simulation and Programming

5/13 TensorFlow Hands-On Part 1: Hello World!

5/14 TensorFlow Hands-On Part 2: Defining and Building Your Network

5/15 TensorFlow Hands-On Part 3: Teaching and Testing and Conclusion

Please stick around as I answer your questions!

- Please give me a moment to scroll back through the chat window to find your questions
- I will stay on chat as long as it takes to answer!
- I am available to answer simple questions or to consult (or offer in-house training for your company)
  c.j.lord@ieee.org
  http://www.blueridgetechnc.com
  http://www.linkedin.com/in/charleslord
  Twitter: @charleslord
  https://www.github.com/bradatraining