Final Year B. Tech. (CSE) – I : 2021-22
4CS462 : PE2 - Data Mining Lab
Assignment No. 1

Group id: DM21G12

Group members:
Abhishek More(2018BTECS00037)
Sushil Wagh(2018BTECS00031)

Title : TensorFlow python module

Objective/Aim : To study Tensorflow python module

Introduction:

TensorFlow is one of the famous deep learning frameworks, developed by Google Brain Team. It is a free and open source software library and designed in the Python programming language. It is entirely based on the Python programming language and used for numerical computation and data flow, which makes machine learning faster and easier.

Theory/Algorithms:

- Components of TensorFlow

1. Graphs: TensorFlow makes use of a graph framework. The chart gathers and describes all the computations done during the training.

2. Tensor: The name TensorFlow is derived from its core framework, "Tensor." A tensor is a vector or a matrix of n-dimensional that represents all type of data. All values in a tensor hold similar data type with a known shape. The shape of the data is the dimension of the matrix or an array.

   A tensor can be generated from the input data or the result of a computation. In TensorFlow, all operations are conducted inside a graph. The group is a set of calculation that takes place successively. Each transaction is called an op node are connected.

3. Session: A session can execute the operation from the graph. To feed the graph with the value of a tensor, we need to open a session. Inside a session, we must run an operator to create an output.

- Why is TensorFlow popular?

  TensorFlow is the better library for all because it is accessible to everyone. TensorFlow library integrates different API to create a scale deep learning architecture like CNN (Convolutional Neural Network) or RNN (Recurrent Neural Network).

TensorFlow is based on graph computation; it can allow the developer to create the construction of the neural network with Tensorboard. This tool helps debug our program. It runs on CPU (Central Processing Unit) and GPU (Graphical Processing Unit).

Documentation:

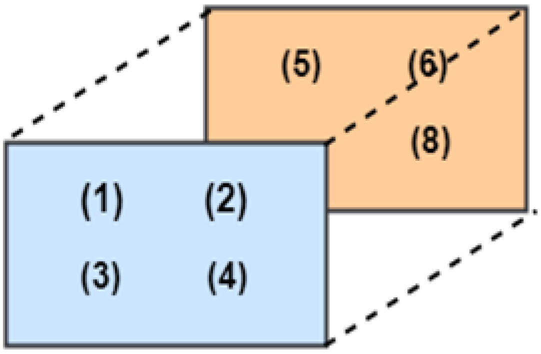
In TensorFlow, a tensor is the collection of feature vector (Like, array) of n-dimension. For instance, if we have any 2x3 matrix with values 1 to 6, we write:

$$\begin{bmatrix} 1 & 3 & 5 \\ 2 & 4 & 6 \end{bmatrix}$$

TensorFlow represents this matrix as:

1. [[1, 3, 5],
2. [2, 4, 6]]

If we create any three-dimensional matrix with values 1 to 8, we have:



TensorFlow represents this matrix as:

[[[1, 2], [3, 4]], [[5, 6], [7, 8]]]

Types of Tensor

All computations pass through one or more Tensors in TensorFlow. A tensor is an object which has three properties which are as follows:

- A unique label (name)
- A dimension (shape)
- A data type (dtype)

Each operation we will TensorFlow involves the manipulation of a tensor. There are four main tensors we can create:

- tf.Variable
- tf.constant
- tf.placeholder
- tf.SparseTensor

Create a tensor of n-dimension

We begin with the creation of a tensor with one dimension, namely a scalar.

To create a tensor, we can use tf.constant ()

1. tf.constant(value, dtype, name = "")
2. arguments
3. `Value`: It is the Value of n dimension to define the tensor. And it is Optional.
4. `dtype`: Define the type of data:
5. `tf.string`: String variable
6. `tf.float32`: Float variable
7. `tf.int16`: Integer variable
8. "name": Name of the tensor. Optional. By default, `Const_1:0`

To create a tensor of dimension 0, We have to run below code.

```
## rank 0
## Default name
r1=tf.constant (1, tf.int18)
print (r1)
```

Output:

```
Tensor ("Const: 0", shape= (), dtype=int18
```

tf.constant() => converts given data into constant tensors for numerical operations
tf.compat.v1.Session() => returns an instance of session for allowing execution of numerical operations
sess.close() => terminates the session
tf.matmul()   => performs multiplication on provided tensors
tf.add()   => performs addition operation on given tensors

Procedure:

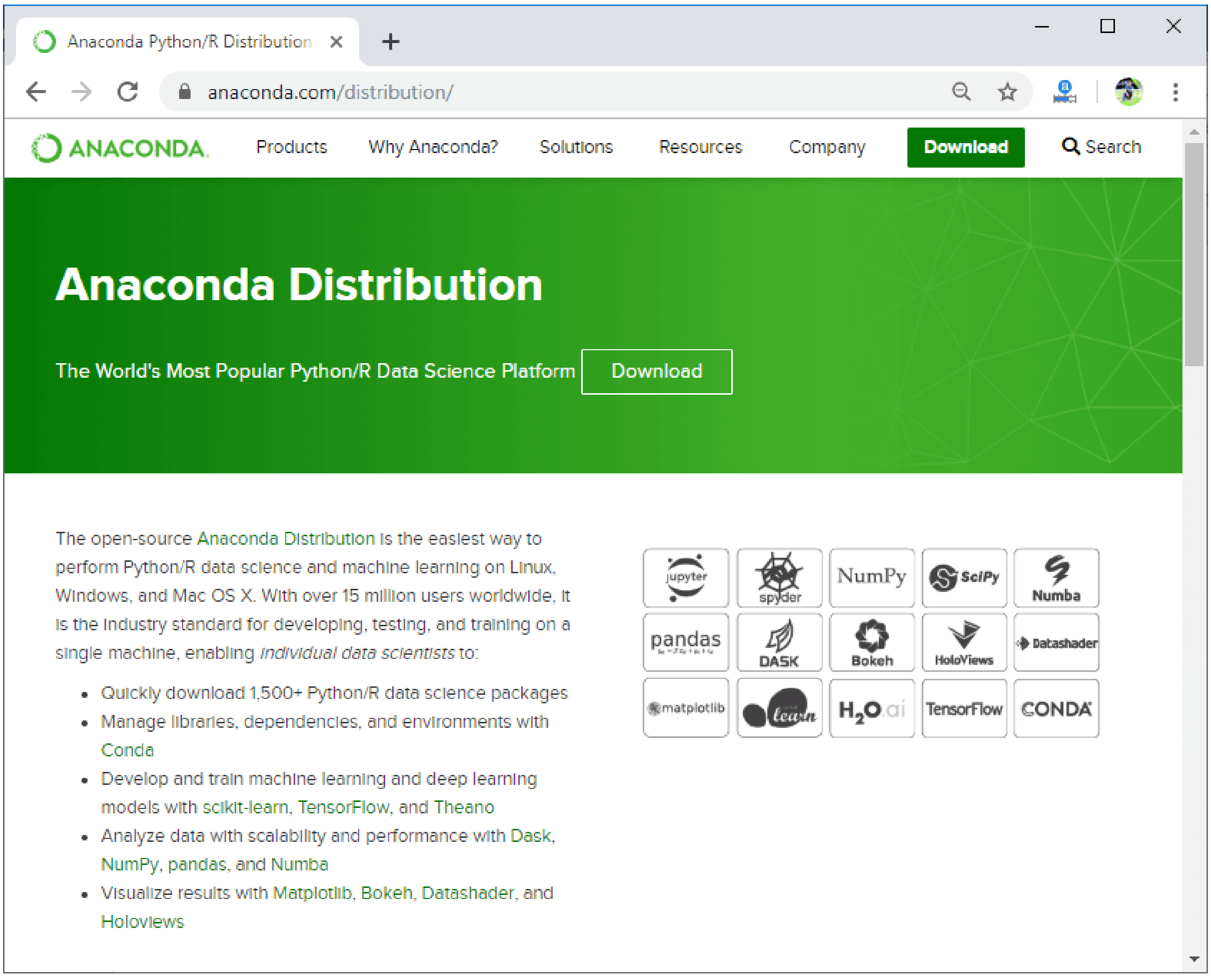
As tensorflow is inherently not compatible with Windows operating system. We have used Anaconda Distribution.

Installation of TensorFlow through conda

In our previous tutorial of TensorFlow, we learn how to install TensorFlow through pip. In this tutorial, we understand that how to install TensorFlow through Conda. Here, we need anaconda Navigator to set-up the platform.

These are the following steps which are given below:

Firstly, we have to open the official site of Anaconda and download Anaconda from the below link: https://www.anaconda.com/distribution/



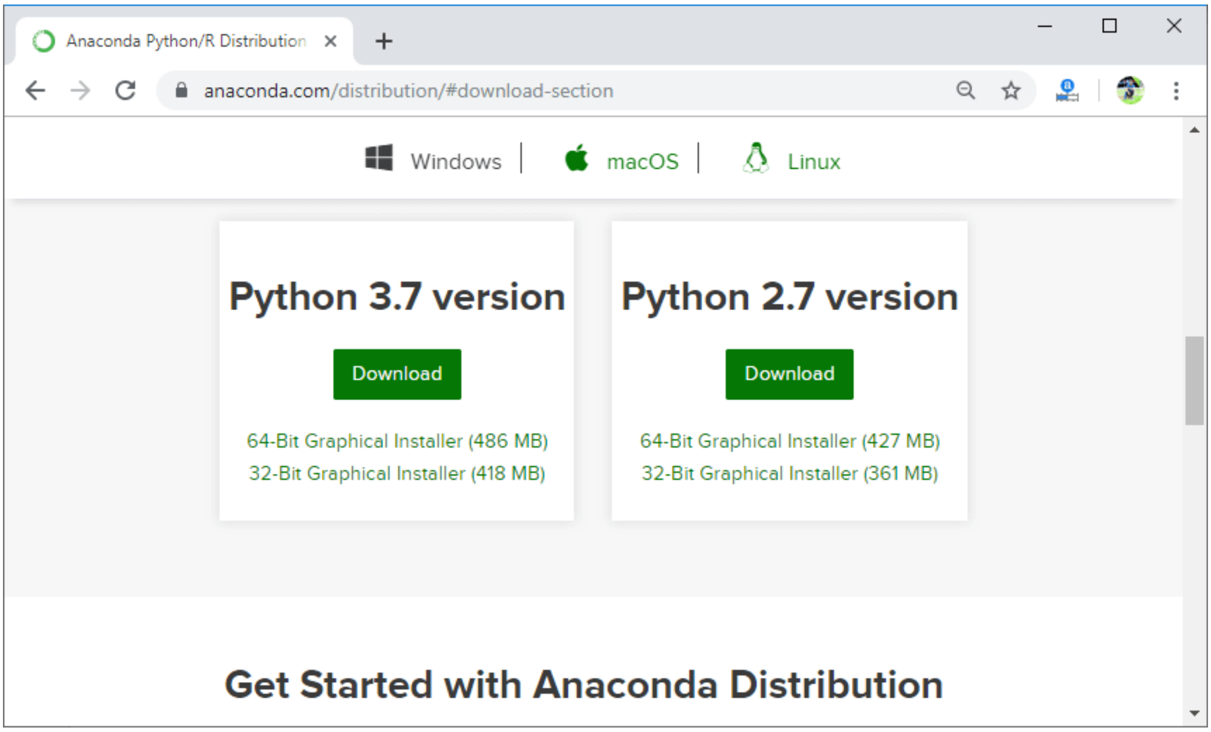
Anaconda Distribution

The World's Most Popular Python/R Data Science Platform Download

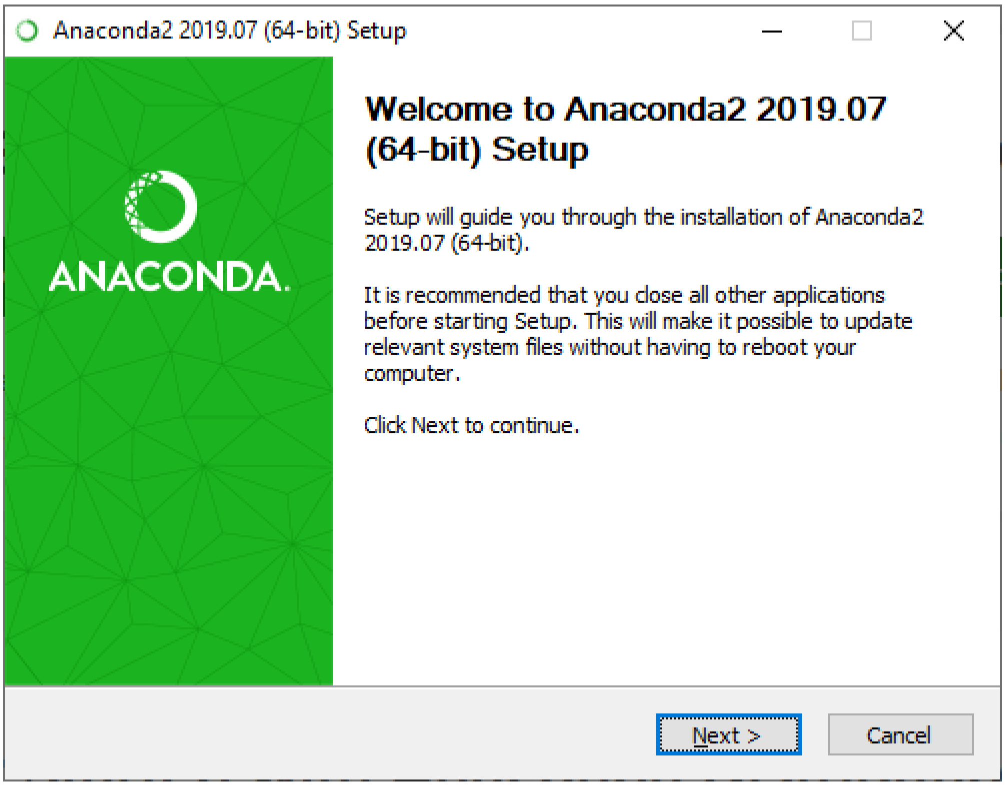
The open-source Anaconda Distribution is the easiest way to perform Python/R data science and machine learning on Linux, Windows, and Mac OS X. With over 15 million users worldwide, it is the industry standard for developing, testing, and training on a single machine, enabling *individual data scientists* to:

- Quickly download 1,500+ Python/R data science packages
- Manage libraries, dependencies, and environments with Conda
- Develop and train machine learning and deep learning models with scikit-learn, TensorFlow, and Theano
- Analyze data with scalability and performance with Dask, NumPy, pandas, and Numba
- Visualize results with Matplotlib, Bokeh, Datashader, and Holoviews

After that, we have to download Anaconda from below highlighted Python 2.7 version.


Anaconda Python/R Distribution
anaconda.com/distribution/#download-section
Windows
macOS
Linux
Python 3.7 version
Download
64-Bit Graphical Installer (486 MB)
32-Bit Graphical Installer (418 MB)
Python 2.7 version
Download
64-Bit Graphical Installer (427 MB)
32-Bit Graphical Installer (361 MB)
Get Started with Anaconda Distribution

It will successfully be downloaded in our system. After that, we have to install **Anaconda** in our system.



Anaconda2 2019.07 (64-bit) Setup

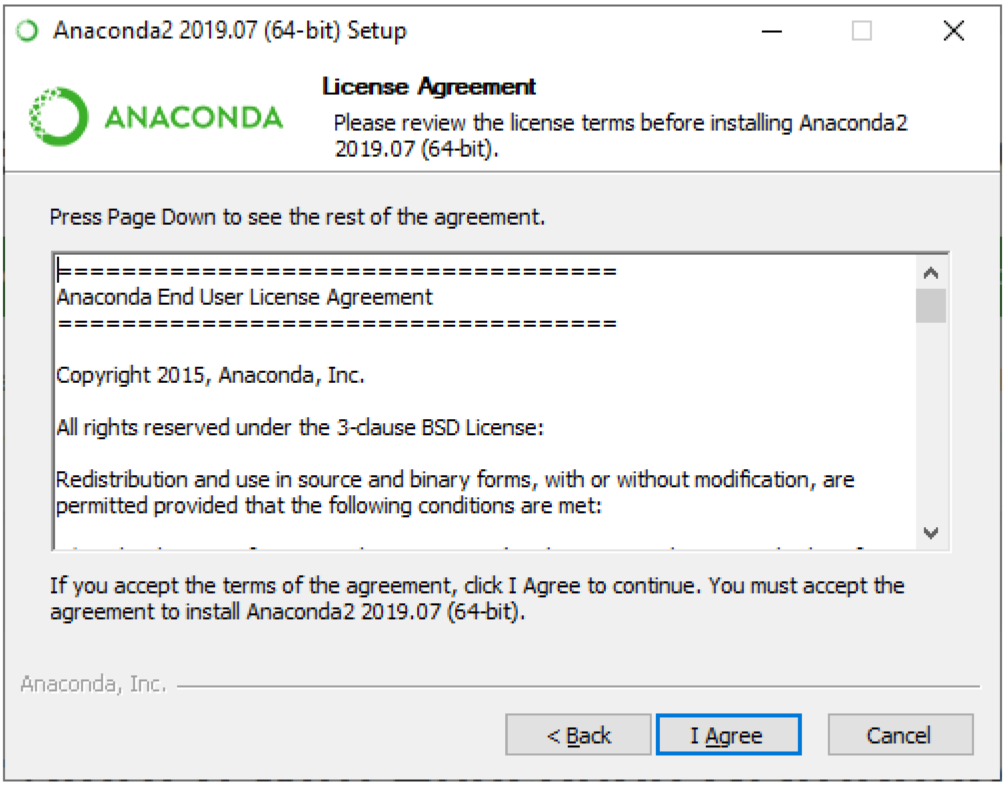
Welcome to Anaconda2 2019.07 (64-bit) Setup

Setup will guide you through the installation of Anaconda2 2019.07 (64-bit).

It is recommended that you close all other applications before starting Setup. This will make it possible to update relevant system files without having to reboot your computer.

Click Next to continue.

Click on "**Next**."



License Agreement

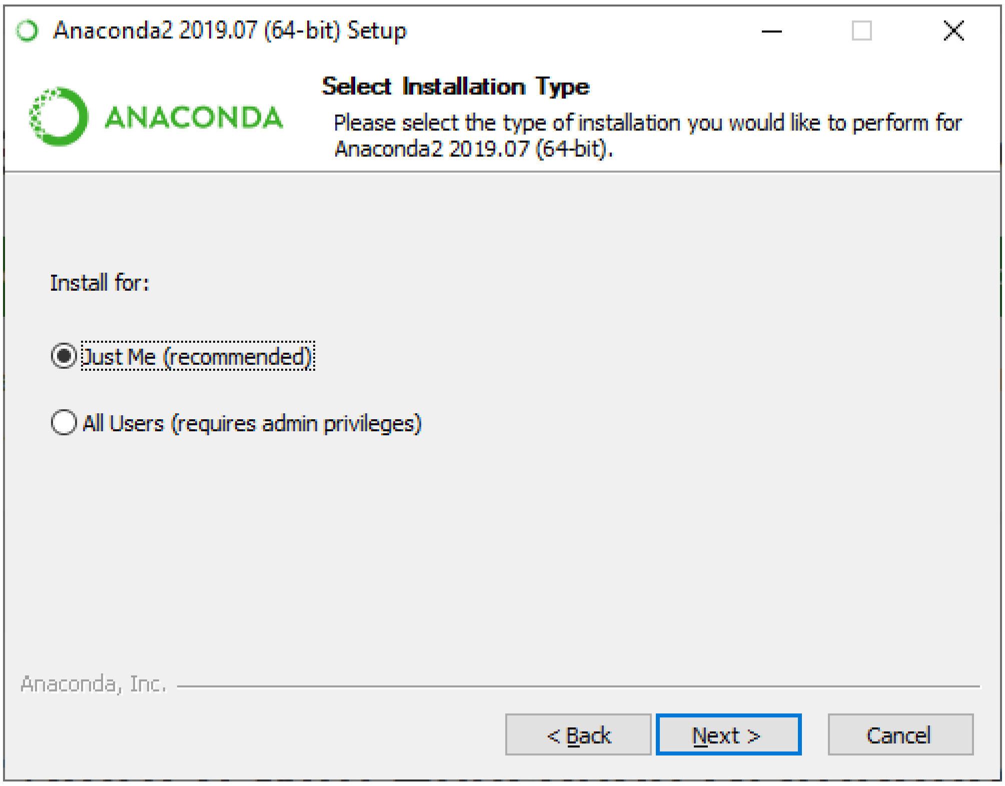
Please review the license terms before installing Anaconda2 2019.07 (64-bit).

Press Page Down to see the rest of the agreement.

```
==================================
Anaconda End User License Agreement
==================================

Copyright 2015, Anaconda, Inc.

All rights reserved under the 3-clause BSD License:

Redistribution and use in source and binary forms, with or without modification, are
permitted provided that the following conditions are met:
```

If you accept the terms of the agreement, click I Agree to continue. You must accept the agreement to install Anaconda2 2019.07 (64-bit).

Click on "**I Agree**."



Select Installation Type

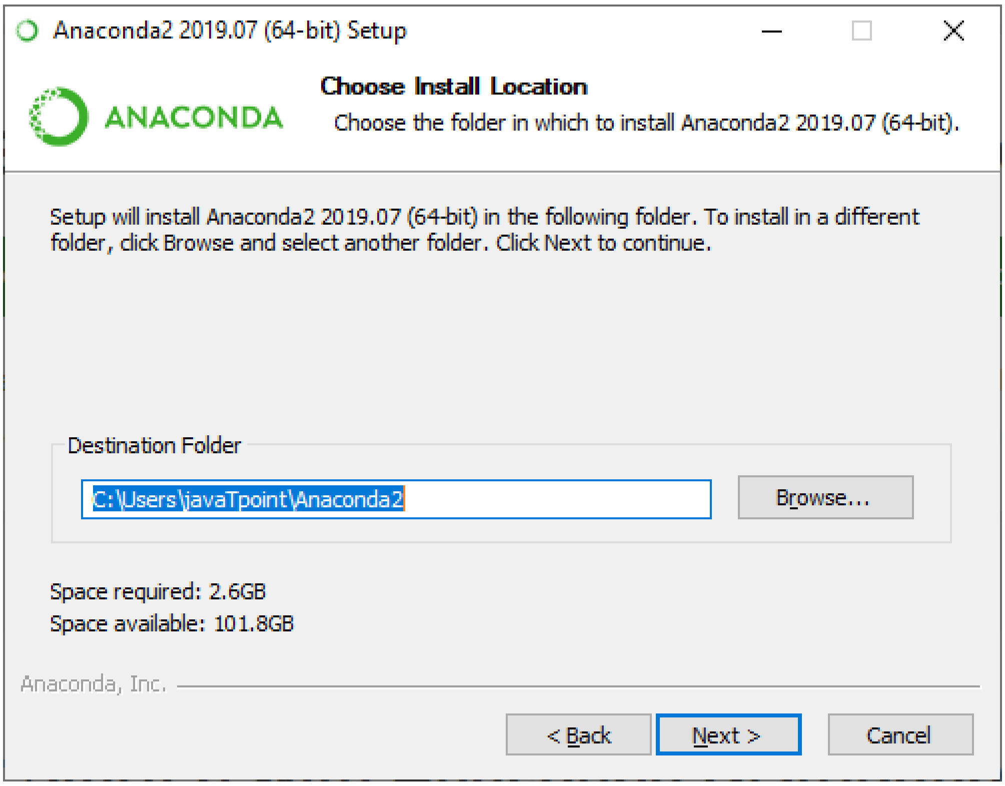
Please select the type of installation you would like to perform for Anaconda2 2019.07 (64-bit).

Install for:

● Just Me (recommended)

○ All Users (requires admin privileges)

Again click on "**Next**."



Anaconda2 2019.07 (64-bit) Setup

Choose Install Location

Choose the folder in which to install Anaconda2 2019.07 (64-bit).

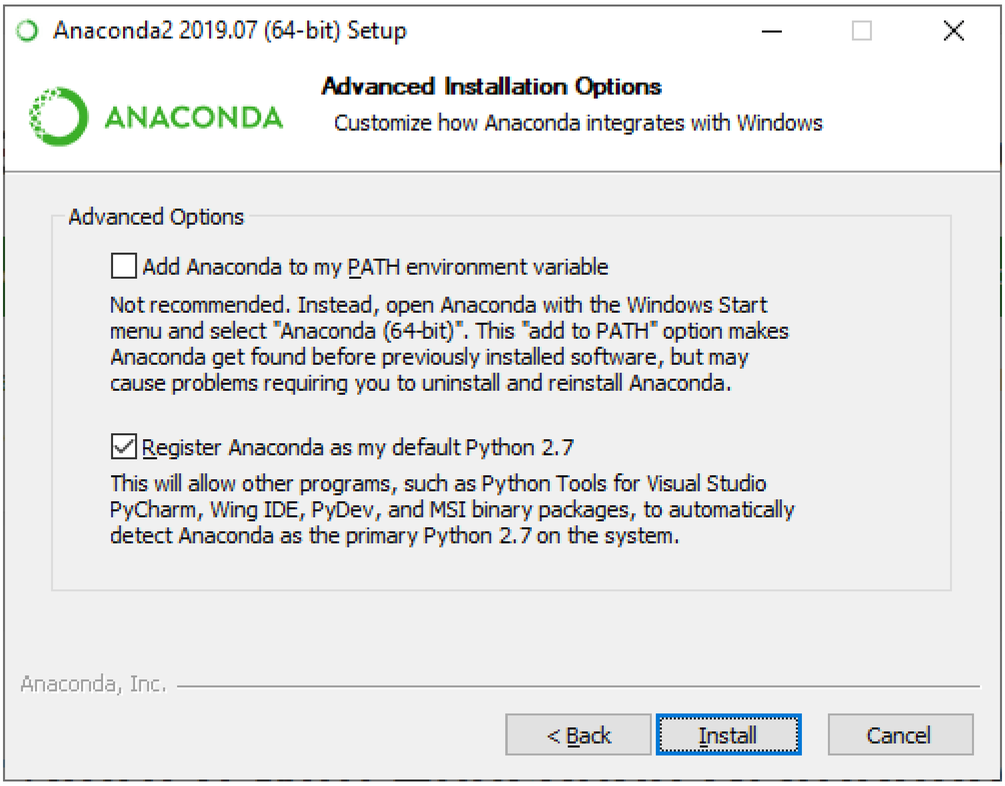
Setup will install Anaconda2 2019.07 (64-bit) in the following folder. To install in a different folder, click Browse and select another folder. Click Next to continue.

Destination Folder

C:\Users\javaTpoint\Anaconda2

Space required: 2.6GB
Space available: 101.8GB

Click on "**Next**" again.



Anaconda2 2019.07 (64-bit) Setup

Advanced Installation Options

Customize how Anaconda integrates with Windows

Advanced Options

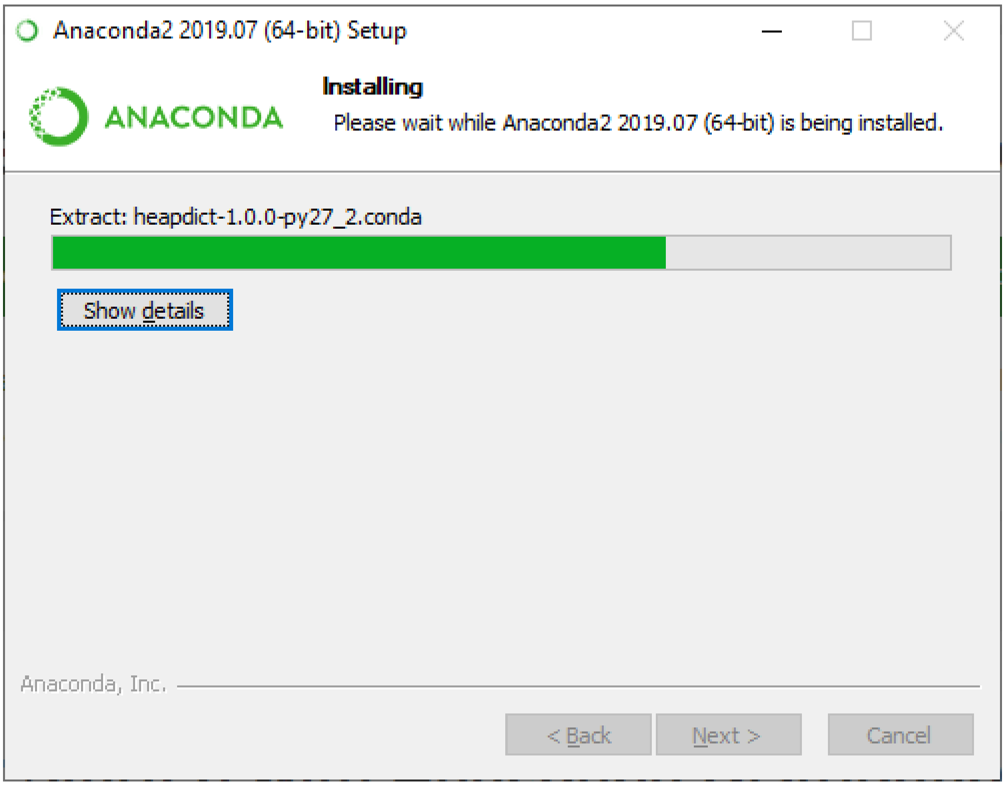
□ Add Anaconda to my PATH environment variable

Not recommended. Instead, open Anaconda with the Windows Start menu and select "Anaconda (64-bit)". This "add to PATH" option makes Anaconda get found before previously installed software, but may cause problems requiring you to uninstall and reinstall Anaconda.

☑ Register Anaconda as my default Python 2.7

This will allow other programs, such as Python Tools for Visual Studio PyCharm, Wing IDE, PyDev, and MSI binary packages, to automatically detect Anaconda as the primary Python 2.7 on the system.

Click on "**Install**."



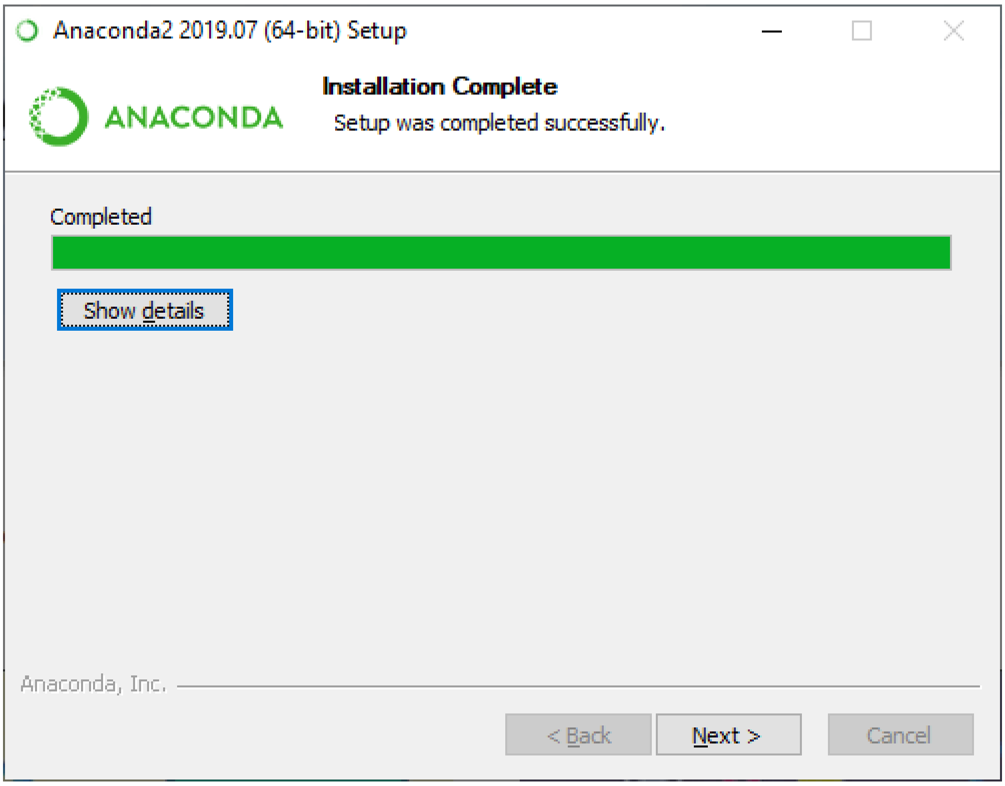
Anaconda2 2019.07 (64-bit) Setup

Installing

Please wait while Anaconda2 2019.07 (64-bit) is being installed.

Extract: heapdict-1.0.0-py27_2.conda

Click on "**Next**."



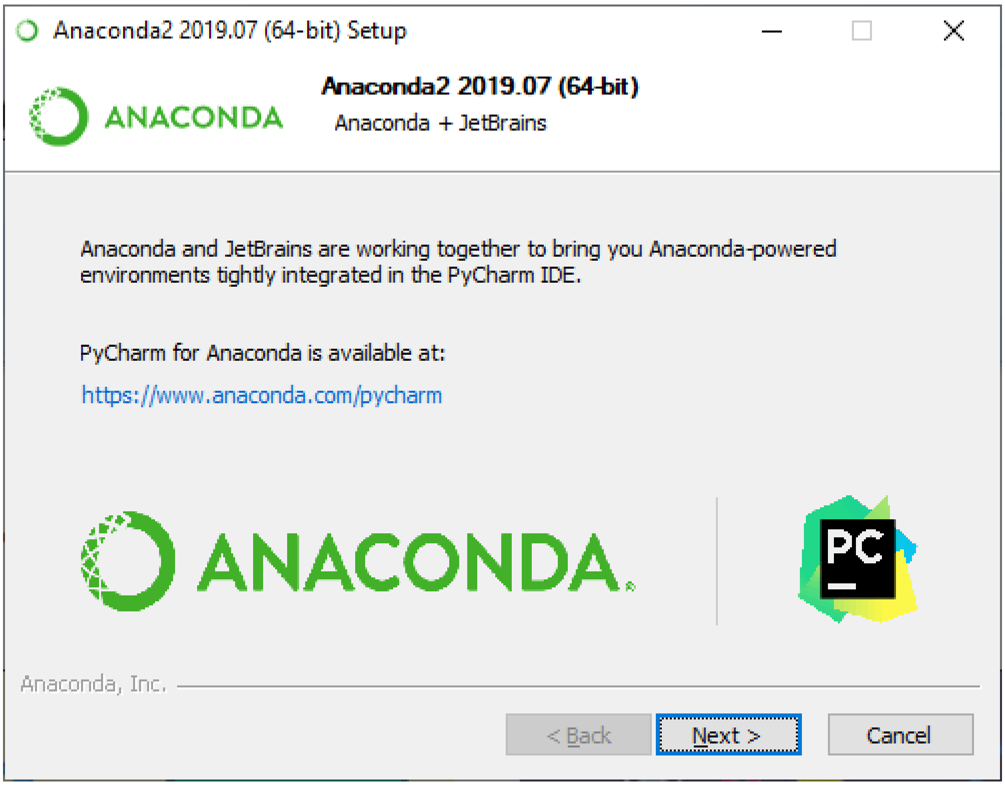
Anaconda2 2019.07 (64-bit) Setup

Installation Complete

Setup was completed successfully.

Completed

Click on "**Next**."



Anaconda2 2019.07 (64-bit) Setup

Anaconda2 2019.07 (64-bit)

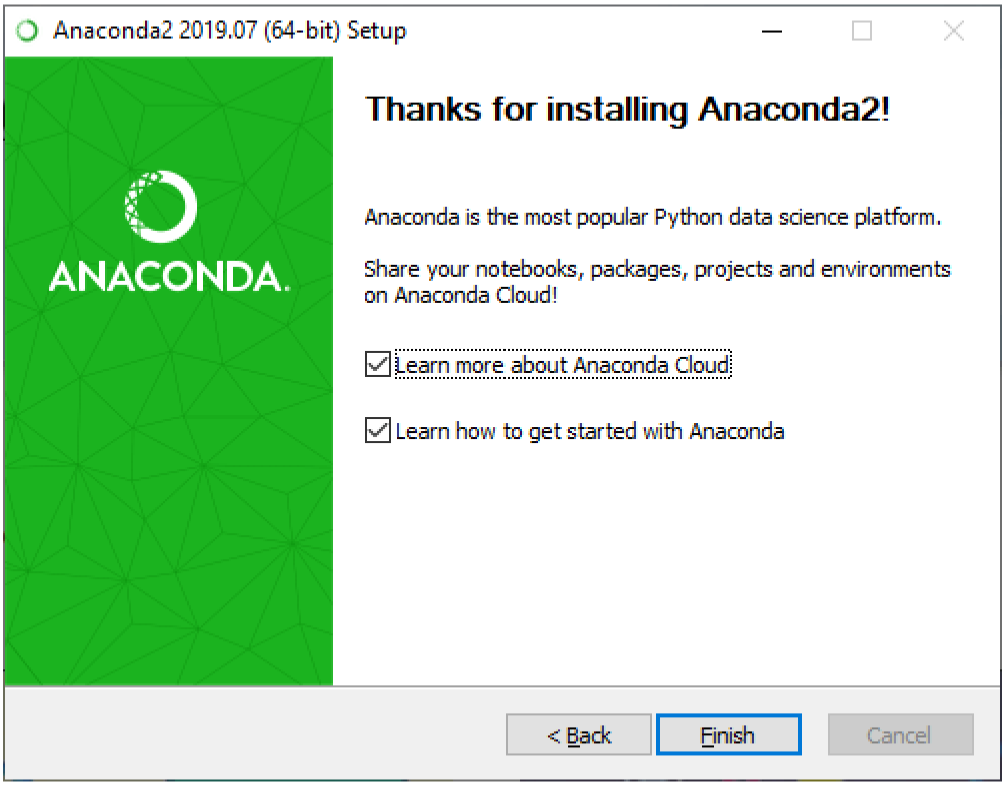
Anaconda + JetBrains

Anaconda and JetBrains are working together to bring you Anaconda-powered environments tightly integrated in the PyCharm IDE.

PyCharm for Anaconda is available at:

https://www.anaconda.com/pycharm

Click on "**Next**."



Anaconda2 2019.07 (64-bit) Setup

Thanks for installing Anaconda2!

Anaconda is the most popular Python data science platform.

Share your notebooks, packages, projects and environments on Anaconda Cloud!

☑ Learn more about Anaconda Cloud

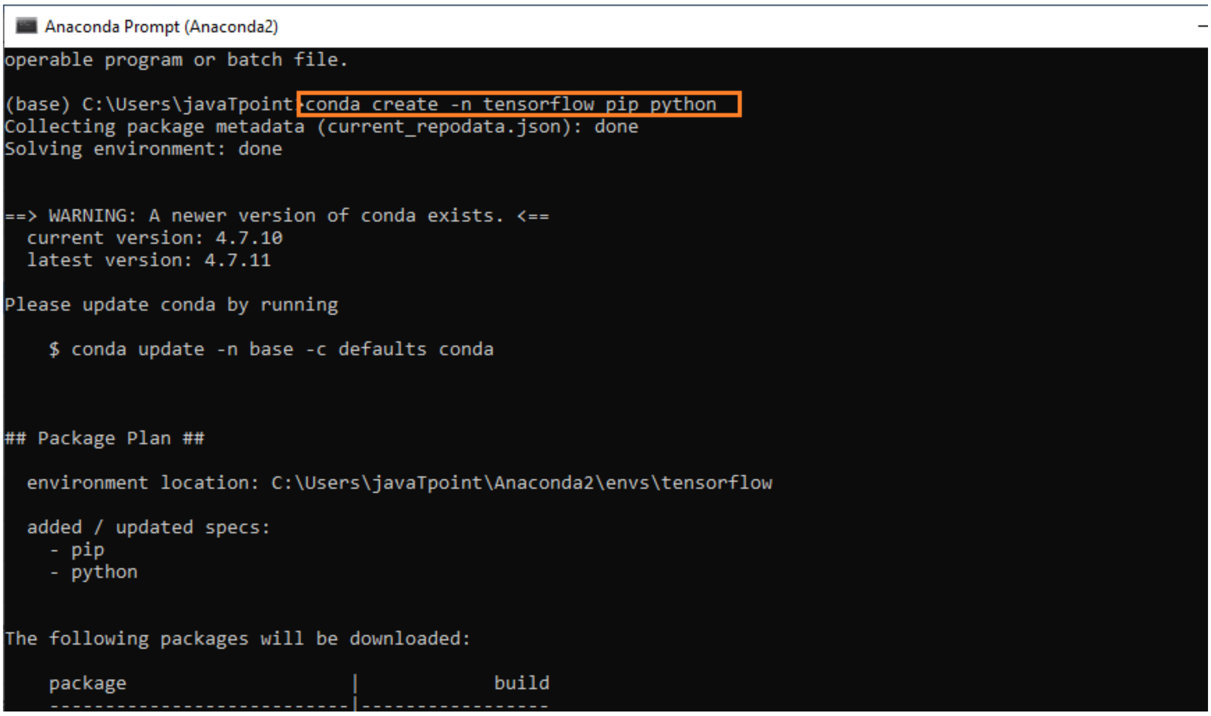
☑ Learn how to get started with Anaconda

After clicking on "**finish**."

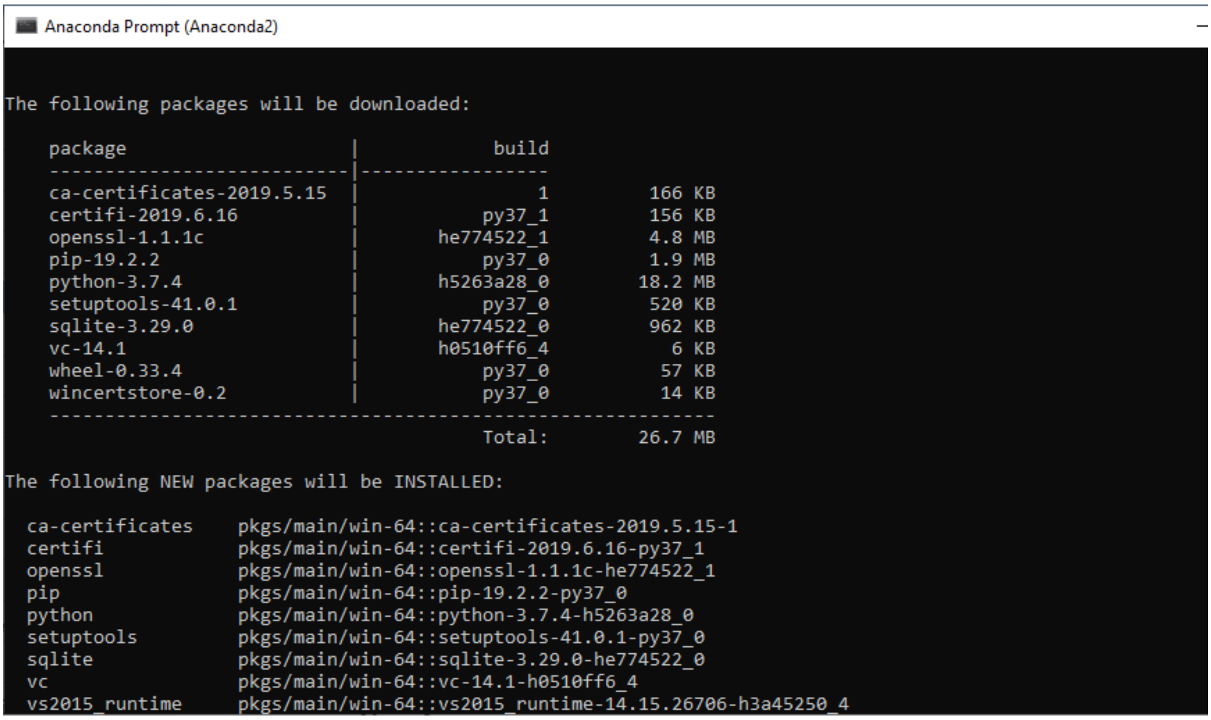
It will successfully installed in our system.

After that, we have to run the given command to set-up the TensorFlow and libraries.

1. Conda create -n tensorflow pip python.

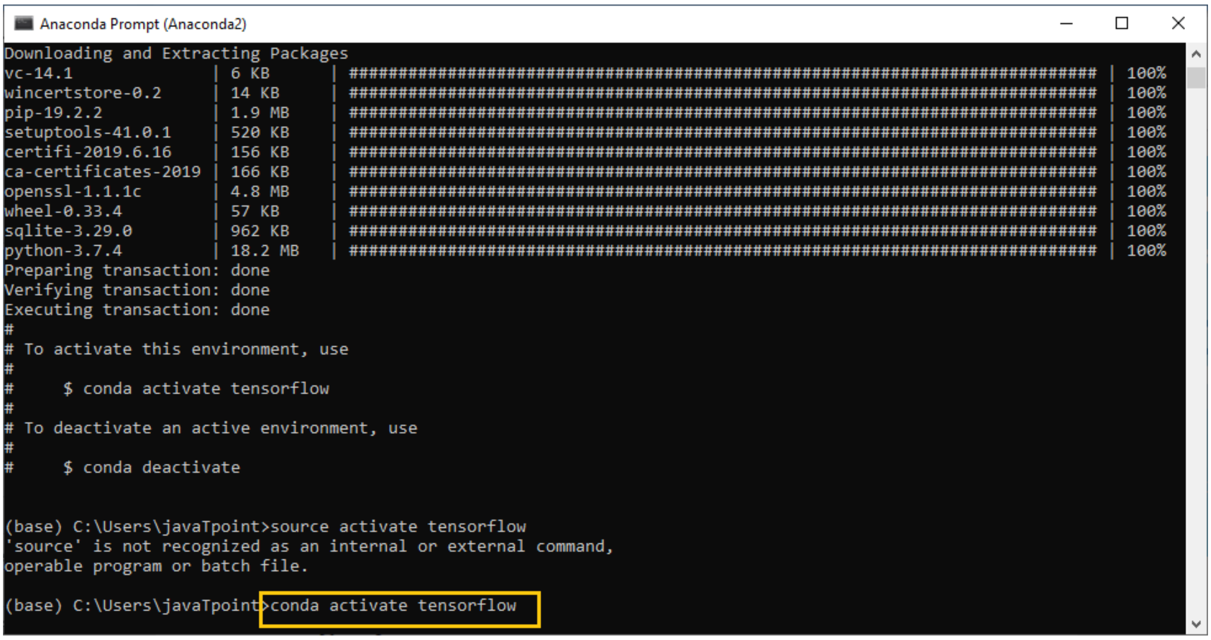


```
operable program or batch file.

(base) C:\Users\javaTpoint>conda create -n tensorflow pip python
Collecting package metadata (current_repodata.json): done
Solving environment: done


==> WARNING: A newer version of conda exists. <==
  current version: 4.7.10
  latest version: 4.7.11

Please update conda by running

    $ conda update -n base -c defaults conda



## Package Plan ##

  environment location: C:\Users\javaTpoint\Anaconda2\envs\tensorflow

  added / updated specs:
    - pip
    - python


The following packages will be downloaded:

    package                    |            build
    ---------------------------|-----------------
```



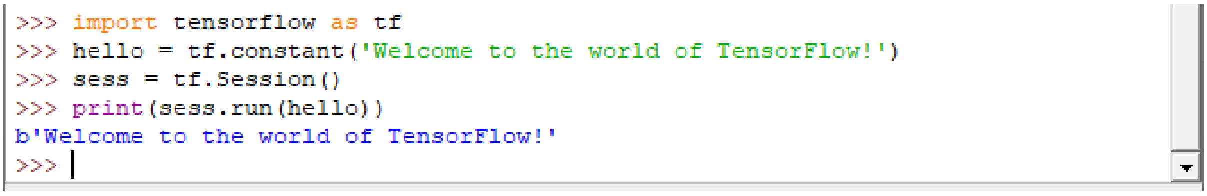
```

The following packages will be downloaded:

    package                    |            build
    ---------------------------|-----------------
    ca-certificates-2019.5.15  |                1         166 KB
    certifi-2019.6.16          |           py37_1         156 KB
    openssl-1.1.1c             |       he774522_1         4.8 MB
    pip-19.2.2                 |           py37_0         1.9 MB
    python-3.7.4               |       h5263a28_0        18.2 MB
    setuptools-41.0.1          |           py37_0         520 KB
    sqlite-3.29.0              |       he774522_0         962 KB
    vc-14.1                    |       h0510ff6_4           6 KB
    wheel-0.33.4               |           py37_0          57 KB
    wincertstore-0.2           |           py37_0          14 KB
    ------------------------------------------------------------
                                           Total:        26.7 MB

The following NEW packages will be INSTALLED:

  ca-certificates    pkgs/main/win-64::ca-certificates-2019.5.15-1
  certifi            pkgs/main/win-64::certifi-2019.6.16-py37_1
  openssl            pkgs/main/win-64::openssl-1.1.1c-he774522_1
  pip                pkgs/main/win-64::pip-19.2.2-py37_0
  python             pkgs/main/win-64::python-3.7.4-h5263a28_0
  setuptools         pkgs/main/win-64::setuptools-41.0.1-py37_0
  sqlite             pkgs/main/win-64::sqlite-3.29.0-he774522_0
  vc                 pkgs/main/win-64::vc-14.1-h0510ff6_4
  vs2015_runtime     pkgs/main/win-64::vs2015_runtime-14.15.26706-h3a45250_4
```

Here, we are downloading and installing the essential things which are used in **TensorFlow** to work.



```
Downloading and Extracting Packages
vc-14.1              | 6 KB      | ############################################################################ | 100%
wincertstore-0.2     | 14 KB     | ############################################################################ | 100%
pip-19.2.2           | 1.9 MB    | ############################################################################ | 100%
setuptools-41.0.1    | 520 KB    | ############################################################################ | 100%
certifi-2019.6.16    | 156 KB    | ############################################################################ | 100%
ca-certificates-2019 | 166 KB    | ############################################################################ | 100%
openssl-1.1.1c       | 4.8 MB    | ############################################################################ | 100%
wheel-0.33.4         | 57 KB     | ############################################################################ | 100%
sqlite-3.29.0        | 962 KB    | ############################################################################ | 100%
python-3.7.4         | 18.2 MB   | ############################################################################ | 100%
Preparing transaction: done
Verifying transaction: done
Executing transaction: done
#
# To activate this environment, use
#
#     $ conda activate tensorflow
#
# To deactivate an active environment, use
#
#     $ conda deactivate


(base) C:\Users\javaTpoint>source activate tensorflow
'source' is not recognized as an internal or external command,
operable program or batch file.

(base) C:\Users\javaTpoint>conda activate tensorflow
```

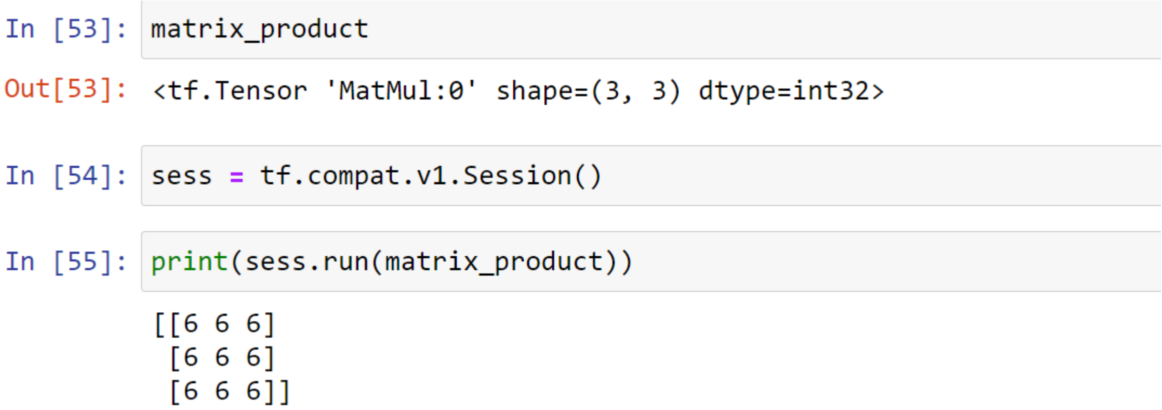
After that, we have to check that TensorFlow is working or not in our system.



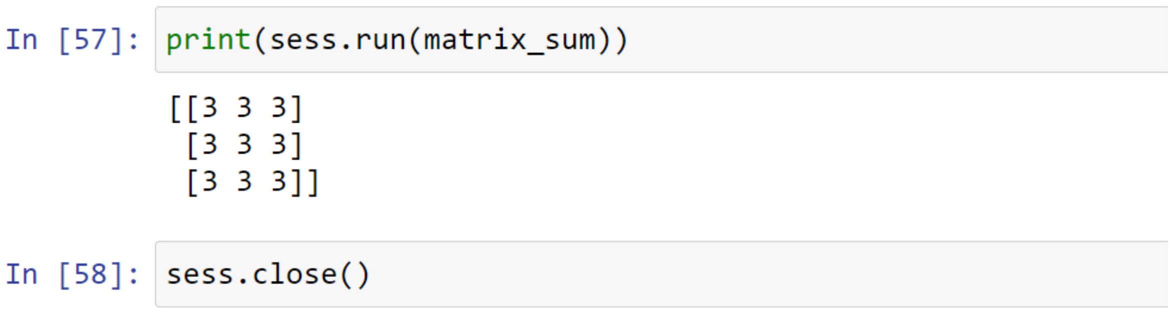
```
>>> import tensorflow as tf
>>> hello = tf.constant('Welcome to the world of TensorFlow!')
>>> sess = tf.Session()
>>> print(sess.run(hello))
b'Welcome to the world of TensorFlow!'
>>> 
```

So, according to the above screenshot, TensorFlow is successfully working in our system.

Result/Observations/Screenshots:



```
In [53]: matrix_product

Out[53]: <tf.Tensor 'MatMul:0' shape=(3, 3) dtype=int32>

In [54]: sess = tf.compat.v1.Session()

In [55]: print(sess.run(matrix_product))

         [[6 6 6]
          [6 6 6]
          [6 6 6]]
```



```
In [57]: print(sess.run(matrix_sum))

[[3 3 3]
 [3 3 3]
 [3 3 3]]

In [58]: sess.close()
```

Conclusion:

Tensorflow uses graph implementation for numerical operations. It can train and run deep neural networks for image recognition, handwritten digit classification, recurrent neural network, word embedding, natural language processing, video detection, and many more. TensorFlow is run on multiple CPUs or GPUs and also mobile operating systems.

References:

- https://www.tensorflow.org/tutorials
- https://towardsdatascience.com/the-complete-tensorflow-tutorial-for-newbies-dc3acc1310f8
- https://towardsdatascience.com/beginners-guide-to-deep-learning-with-tensorflow-ca85969b2f2