Tensorflow Test

In [1]:
```
import tensorflow as ts
```

In [2]:
```
print(ts.__version__)
```

```
2.9.1
```

In [3]:
```
print(ts.reduce_sum(ts.random.normal([1000, 1000])))
```

```
tf.Tensor(-275.37915, shape=(), dtype=float32)
```

Keras Test

In [4]:
```
from tensorflow import keras
from keras import datasets
```

In [5]:
```
#MNIST Dataset
(train_images, train_labels), (test_images, test_labels) = datasets.mnist.load_data()
```

In [6]:
```
# Check the dataset loaded
train_images.shape, test_images.shape
```

Out[6]:
```
((60000, 28, 28), (10000, 28, 28))
```

Theano Test

In [7]:
```
# Python program showing
# addition of two scalars
# Addition of two scalars
import numpy
import theano.tensor as T
from theano import function
```

```
WARNING (theano.configdefaults): g++ not available, if using conda: `conda install m2
w64-toolchain`
D:\Anaconda\lib\site-packages\theano\configdefaults.py:560: UserWarning: DeprecationW
arning: there is no c++ compiler.This is deprecated and with Theano 0.11 a c++ compil
er will be mandatory
  warnings.warn("DeprecationWarning: there is no c++ compiler."
WARNING (theano.configdefaults): g++ not detected ! Theano will be unable to execute
optimized C-implementations (for both CPU and GPU) and will default to Python impleme
ntations. Performance will be severely degraded. To remove this warning, set Theano f
lags cxx to an empty string.
WARNING (theano.tensor.blas): Using NumPy C-API based implementation for BLAS functio
ns.
```

In [8]:
```
# Declaring two variables
x = T.dscalar('x')
y = T.dscalar('y')
# Summing up the two numbers
z = x + y
# Converting it to a callable object
# so that it takes matrix as parameters
f = function([x, y], z)
f(5, 7)
```

Out[8]:
```
array(12.)
```

Torch Test

In [1]:
```
import torch
import torch.nn as nn
```

In [2]:
```
print(torch.__version__)
```

```
1.13.0
```

In [ ]:
```

```