# AI_DL_tensorflow_basics_session

# TensorFlow Sessions

### NOTES about Session: Block of Code

#### IMPORTANT NOTES

Initialization & Following Operations MUST reside in the same session,
i.e., in the same block of code of a tf session.

## with tf.Session as ...

In [31]:

```
import tensorflow as tf
```

In [32]:

```
# Create a contant tensor: a matrix of (3. 3)
# The values are random between 0 and 1

aTensor = tf.random_uniform ((3, 3), 0, 1)
```

In [33]:

```
# Declare a variable initialized with aTensor

aTfVar = tf.Variable(initial_value=aTensor)
```

In [36]:

```
# Get an Initializer operation to initialize the variable

anInitializer = tf.global_variables_initializer()
```

In [35]:

```
# Initialize the variable and prepare the print of the contents of

with tf.Session() as sess:
    sess.run(anInitializer)
    results = sess.run(aTfVar)
```

In [18]:

```
print (results)
```

```
[[ 0.42441201  0.80258191  0.97953629]
 [ 0.28094065  0.44814134  0.60620618]
 [ 0.71110785  0.47273791  0.07188153]]
```

**NOTES: Combine 2 statements into 1 for printing**

In [37]:

```
with tf.Session() as sess:
    sess.run(anInitializer)
    print(sess.run(aTfVar))
```

```
[[ 0.3196888   0.73644876  0.12325013]
 [ 0.30934882  0.55409777  0.25523889]
 [ 0.45751989  0.46309483  0.72872865]]
```

**NOTES: ERROR if without initialization**

In [26]:

```
import tensorflow as tf

aTensor = tf.random_uniform ((3, 3), 0, 1)

aTfVar_2 = tf.Variable(initial_value=aTensor)

anInitializer = tf.global_variables_initializer()

""" CAUSE ERROR
with tf.Session() as sess:
    results = sess.run(aTfVar_2)
"""
```

Out[26]:

```
' CAUSE ERROR\nwith tf.Session() as sess:\n    results = sess.run(aTfVar_2)
\n'
```

**NOTES: ERROR if not in the same Session block of code**

In [27]:

```
import tensorflow as tf

aTensor = tf.random_uniform ((3, 3), 0, 1)

aTfVar_2 = tf.Variable(initial_value=aTensor)

anInitializer = tf.global_variables_initializer()

with tf.Session() as sess:
    sess.run(anInitializer)
```

In [29]:

```
""" CAUSE ERROR
results = sess.run(aTfVar_2)
"""
```

Out[29]:

```
' CAUSE ERROR\nresults = sess.run(aTfVar_2)\n'
```

**NOTES: DO NOTHING if without "results = ... " and "print (...)"**

In [30]:

```
import tensorflow as tf

aTensor = tf.random_uniform ((3, 3), 0, 1)

aTfVar_2 = tf.Variable(initial_value=aTensor)

anInitializer = tf.global_variables_initializer()

with tf.Session() as sess:
    sess.run(anInitializer)
    sess.run(aTfVar_2)
```

# Using Interactive Sessions

**NOTES:**

**All the statement between the start and the close of a tf.InteractiveSession() are included in the same session.**

## EXAMPLE 1

In [44]:

```
import tensorflow as tf
```

In [61]:

```
# Start a tf.InteractiveSession

sess = tf.InteractiveSession()
```

In [46]:

```
aTensor = tf.random_uniform ((3, 3), 0, 1)
```

In [47]:

```
aTfVar_3 = tf.Variable(initial_value=aTensor)
```

In [48]:

```
# Get an Initializer operation to initialize the variable

anInitializer = tf.global_variables_initializer()
```

In [49]:

```
# Run the operation initializer to actually initialize the variable

sess.run(anInitializer)
```

In [50]:

```
# Display the values of the initialized variable

sess.run(aTfVar_3)
```

Out[50]:

```
array([[ 0.57438934,  0.70639229,  0.68839812],
       [ 0.74204791,  0.26086426,  0.48703289],
       [ 0.21820045,  0.8931787 ,  0.89574265]], dtype=float32)
```

In [52]:

```
# MUST CLOSE THE SESSION

sess.close()
```

**NOTES: Display the values without "results = ..." and "print (...)"**

In [54]:

```
sess = tf.InteractiveSession()

aTfVar_4 = tf.Variable(initial_value=aTensor)

anInitializer = tf.global_variables_initializer()

sess.run(anInitializer)
```

In [55]:

```
# Display the values of the initialized variable
# WITHOUT "results = ..." and "print(...)

sess.run(aTfVar_4)
```

Out[55]:

```
array([[ 0.5090915 ,  0.24371791,  0.52474129],
       [ 0.6548481 ,  0.3610034 ,  0.06718826],
       [ 0.01342487,  0.81748867,  0.95783019]], dtype=float32)
```

In [56]:

```
sess.close()
```

**ANOTHER EXAMPLE**

In [ ]:

```
import tensorflow as tf
```

In [57]:

```
sess = tf.InteractiveSession()
```

In [58]:

```
a = tf.constant(5.0)
b = tf.constant(6.0)

c = a * b
```

In [59]:

```
# We can just use 'c.eval()' without passing 'sess'
print(c.eval())
```

```
30.0
```

In [60]:

```
sess.close()
```

In [ ]:

```

```