Week 8

TensorFlow

1927 USTSouthernP
UNIVERSITY OF
SCIENCE AND TECHNOLOGY
OF SOUTHERN PHILIPPINES

Key Features of TensorFlow

1. **Open Source:** Free to use and modify.
2. **Flexibility:** Works with CPUs, GPUs, and TPUs.
3. **Scalability:** From small devices to large distributed systems.
4. **Pre-built Models:** Access to pre-trained models (e.g., TensorFlow Hub).
5. **Visualization:** TensorBoard for visualizing models and metrics.

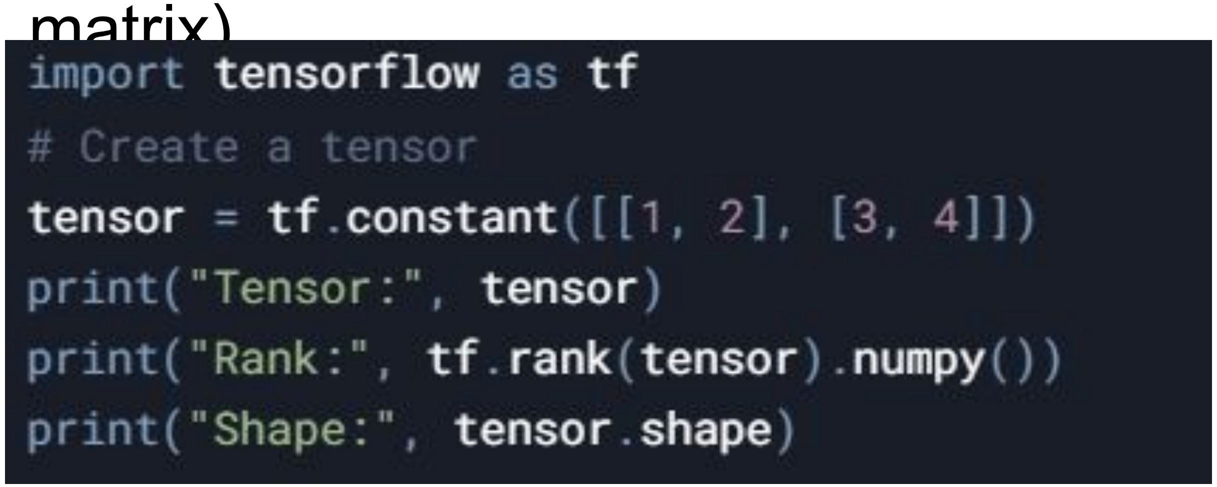
TensorFlow Ranks and Tensors

- **What are Tensors?**

✔ Tensors are multi-dimensional arrays with a uniform type.

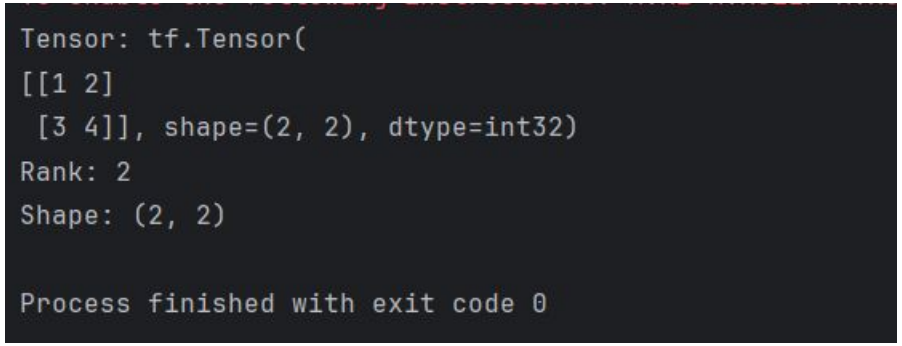
✔ **Rank:** Number of dimensions (e.g., scalar = 0D, vector = 1D, matrix = 2D).

✔ **Shape:** Number of elements in each dimension (e.g., [2, 3] for a 2x3 matrix)



```
import tensorflow as tf
# Create a tensor
tensor = tf.constant([[1, 2], [3, 4]])
print("Tensor:", tensor)
print("Rank:", tf.rank(tensor).numpy())
print("Shape:", tensor.shape)
```

OUTPUT:



```
Tensor: tf.Tensor(
[[1 2]
 [3 4]], shape=(2, 2), dtype=int32)
Rank: 2
Shape: (2, 2)

Process finished with exit code 0
```

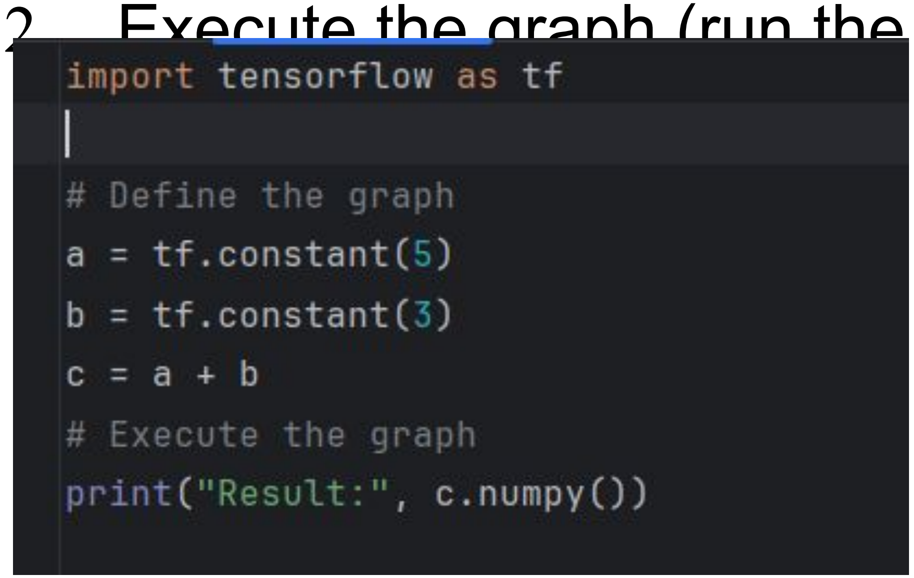
TensorFlow Computation Graphs

- **What is a Computation Graph?**

✔ A graph where nodes represent operations, and edges represent data (tensors).

✔ **Two Steps:**

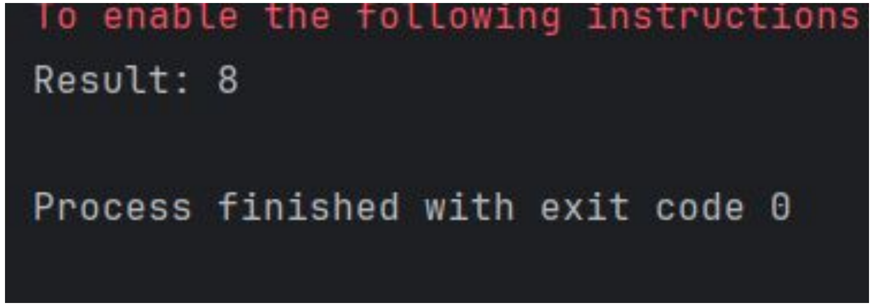
1. Define the graph (build the model).
2. Execute the graph (run the session).



```
import tensorflow as tf

# Define the graph
a = tf.constant(5)
b = tf.constant(3)
c = a + b
# Execute the graph
print("Result:", c.numpy())
```

OUTPUT:



```
To enable the following instructions
Result: 8

Process finished with exit code 0
```

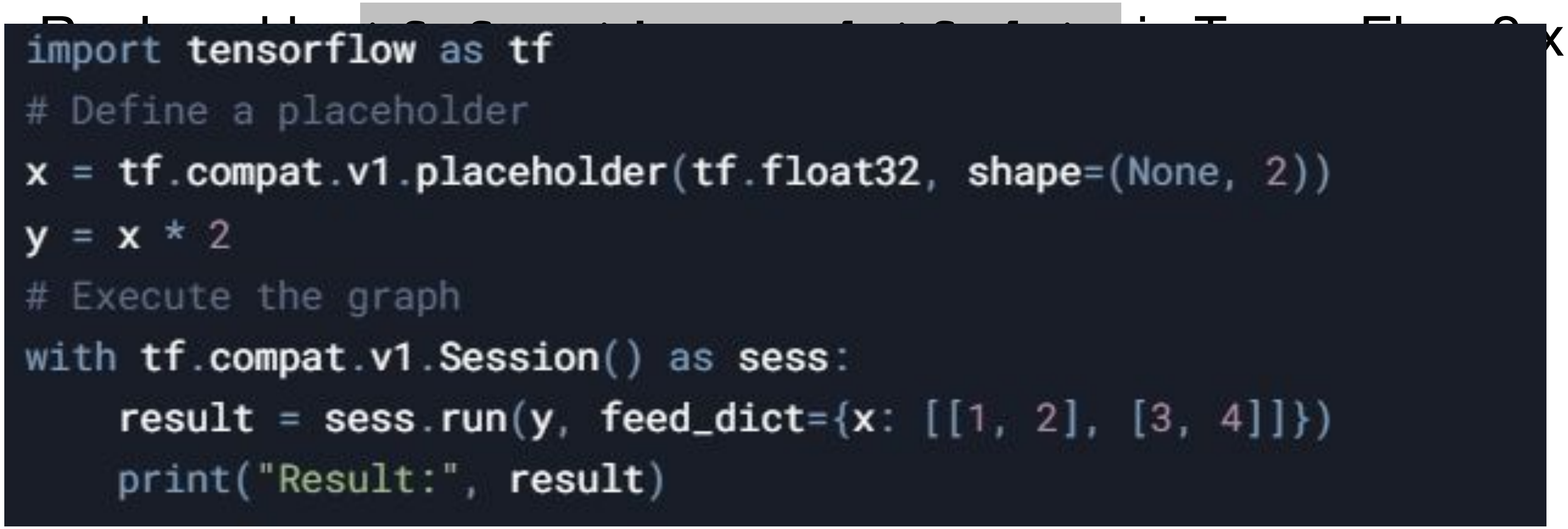
Placeholders in TensorFlow

- **What are Placeholders?**

✔ Placeholders are used to feed data into a TensorFlow graph during execution.

✔ **Key Points:**

1. Defined using `tf.compat.v1.placeholder` (in TensorFlow 1.x).
2. Replaced by `tf.function and tf.data` in TensorFlow 2.x.



```
import tensorflow as tf
# Define a placeholder
x = tf.compat.v1.placeholder(tf.float32, shape=(None, 2))
y = x * 2
# Execute the graph
with tf.compat.v1.Session() as sess:
    result = sess.run(y, feed_dict={x: [[1, 2], [3, 4]]})
    print("Result:", result)
```

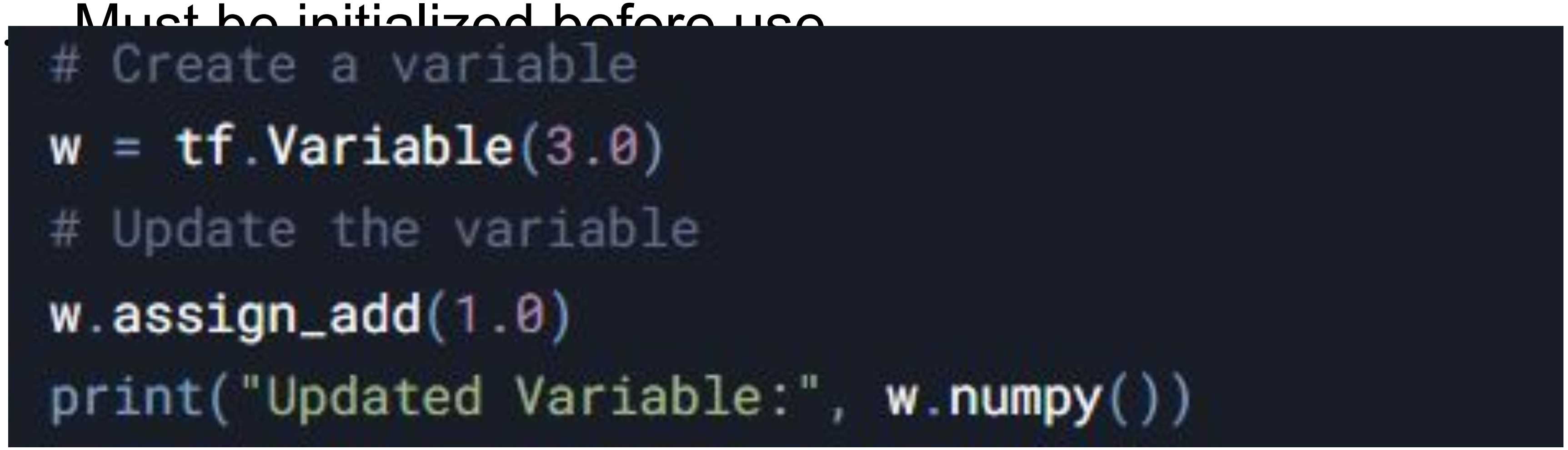
Variables in TensorFlow

- **What are Variables?**

✔ Variables are used to store and update parameters (e.g., weights and biases) during training.

✔ **Key Points:**

1. Created using `tf.Variable`.
2. Must be initialized before use.



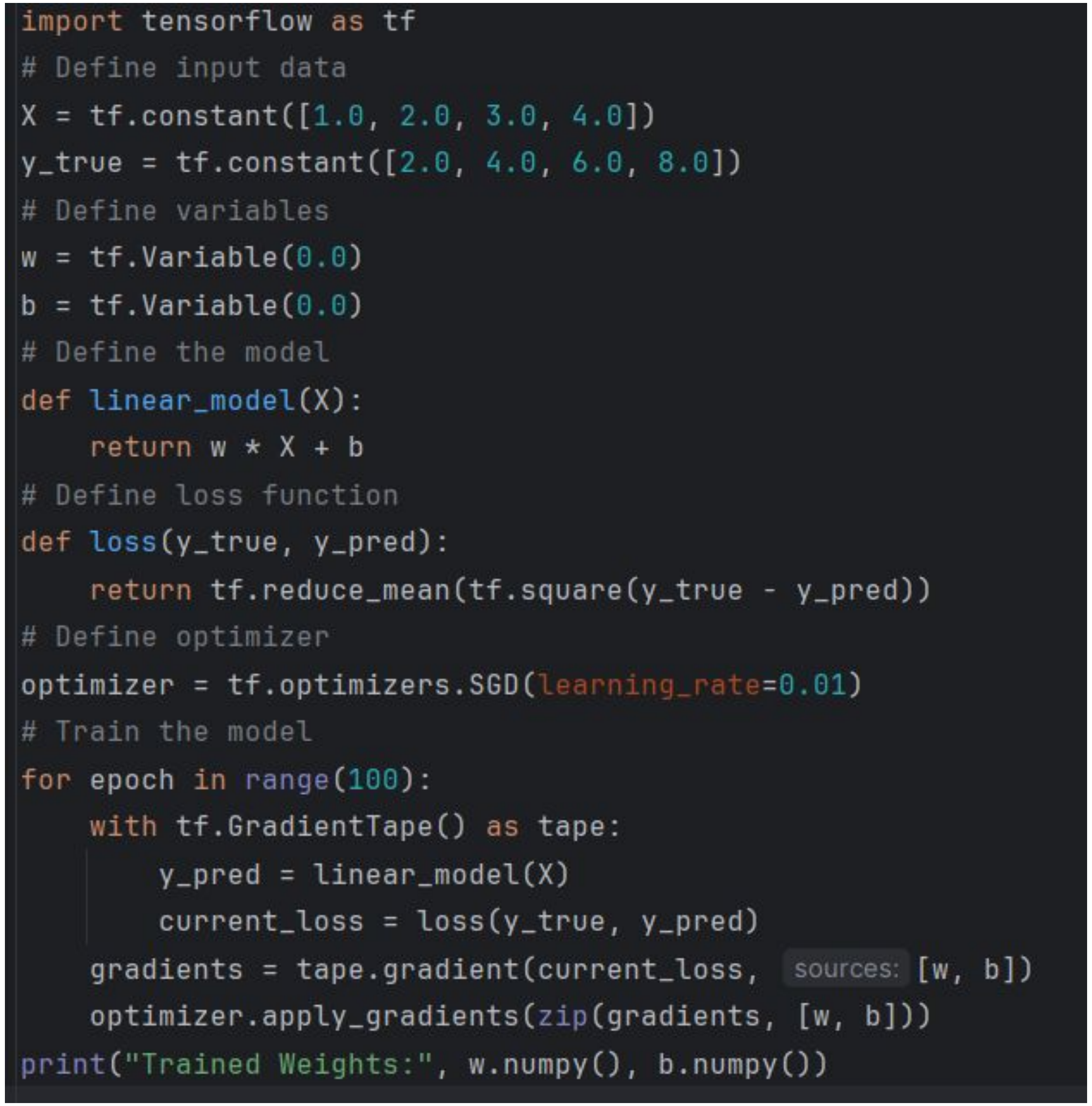
```
# Create a variable
w = tf.Variable(3.0)
# Update the variable
w.assign_add(1.0)
print("Updated Variable:", w.numpy())
```

Building a Regression Model

- **Steps to Build a Regression Model:**

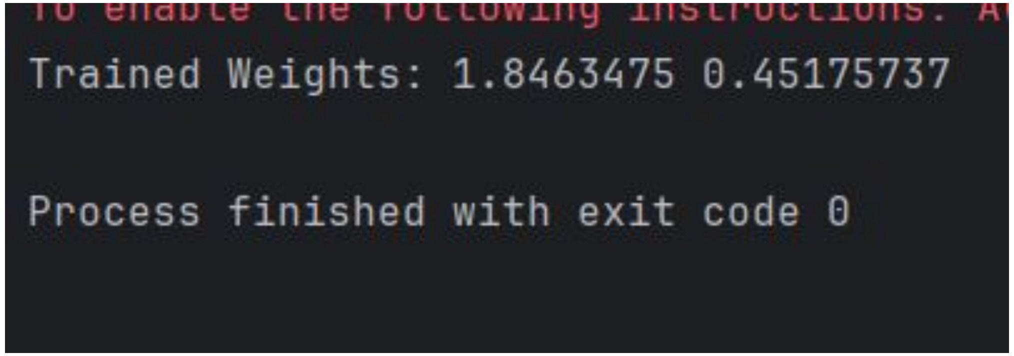
1. Define input data and labels.
2. Create variables for weights and biases.
3. Define the model (e.g., `y = wx + b`).
4. Define a loss function (e.g., Mean Squared Error).
5. Use an optimizer (e.g., Gradient Descent).
6. Train the model.

Building a Regression Model



```
import tensorflow as tf
# Define input data
X = tf.constant([1.0, 2.0, 3.0, 4.0])
y_true = tf.constant([2.0, 4.0, 6.0, 8.0])
# Define variables
w = tf.Variable(0.0)
b = tf.Variable(0.0)
# Define the model
def linear_model(X):
    return w * X + b
# Define loss function
def loss(y_true, y_pred):
    return tf.reduce_mean(tf.square(y_true - y_pred))
# Define optimizer
optimizer = tf.optimizers.SGD(learning_rate=0.01)
# Train the model
for epoch in range(100):
    with tf.GradientTape() as tape:
        y_pred = linear_model(X)
        current_loss = loss(y_true, y_pred)
    gradients = tape.gradient(current_loss, sources: [w, b])
    optimizer.apply_gradients(zip(gradients, [w, b]))
print("Trained Weights:", w.numpy(), b.numpy())
```

Output:



```
To enable the following instructions: A
Trained Weights: 1.8463475 0.45175737

Process finished with exit code 0
```

TensorFlow Activity

- **Activity Title:** Hands-On with Tensors and Regression Models

- **Instructions :**
  1. Create a Python script to:
     - Define and print tensors of different ranks and shapes.
     - Build a simple linear regression model using TensorFlow.
  2. Share your code and results in the next class.

- **Deliverable:** Submit your Python script and a short explanation of your code:

References

- **Books:** Hands-On with TensorFlow Basics
  - *Python Machine Learning for Beginners* by AI Publishing.
  - *Internet-of-Things* by Dimitrios Serpanos and Marilyn Wolf.

- **Websites:**
  - TensorFlow Official Documentation: https://www.tensorflow.org
  - TensorFlow Tutorials: https://www.tensorflow.org/tutorials

1927

USTP

Southern

UNIVERSITY OF
SCIENCE AND TECHNOLOGY
OF SOUTHERN PHILIPPINES

Alubijid | Balubal | Cagayan de Oro | Claveria | Jasaan | Oroquieta | Panaon | Villanueva

Home of the Trailblazers