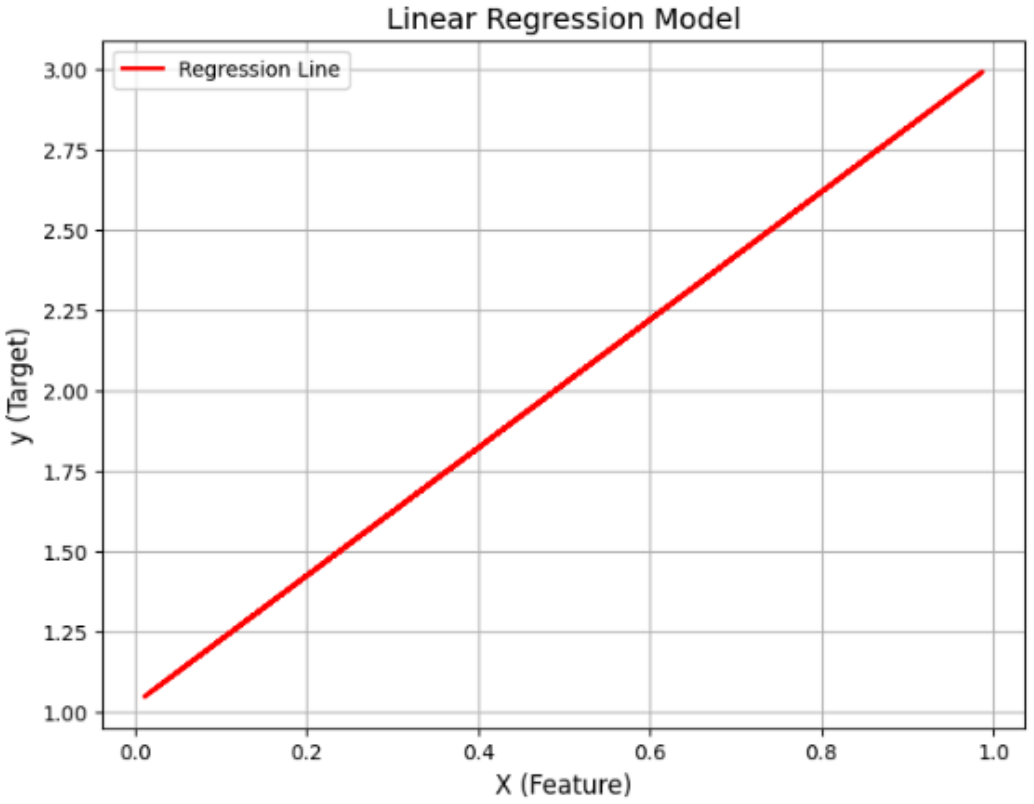
```python
import tensorflow as tf
import numpy as np
import matplotlib.pyplot as plt

scalar = tf.constant(3)
vector = tf.constant([1, 2, 3, 4, 5])
matrix = tf.constant([[1, 2], [3, 4], [5, 6]])
tensor_3d = tf.constant([[[1, 2], [3, 4]], [[5, 6], [7, 8]]])

print("Scalar:", scalar)
print("Vector:", vector)
print("Matrix:", matrix)
print("Tensor 3D:", tensor_3d)

X = np.random.rand(100, 1)
y = 2 * X + 1 + 0.1 * np.random.randn(100, 1)

X_train = tf.constant(X, dtype=tf.float32)
y_train = tf.constant(y, dtype=tf.float32)

weights = tf.Variable(tf.random.normal([1, 1]))
bias = tf.Variable(tf.zeros([1]))

def model(X):
    return tf.matmul(X, weights) + bias

def loss_fn(y_true, y_pred):
    return tf.reduce_mean(tf.square(y_true - y_pred))

optimizer = tf.optimizers.SGD(learning_rate=0.1)

for epoch in range(1000):
    with tf.GradientTape() as tape:
        y_pred = model(X_train)
        loss = loss_fn(y_train, y_pred)

    gradients = tape.gradient(loss, [weights, bias])
    optimizer.apply_gradients(zip(gradients, [weights, bias]))

    if epoch % 100 == 0:
        print(f"Epoch {epoch}, Loss: {loss.numpy()}")

plt.figure(figsize=(8, 6))
plt.plot(X, model(X_train).numpy(), color='red', label="Regression Line", linewidth=2)
plt.xlabel("X (Feature)", fontsize=12)
plt.ylabel("y (Target)", fontsize=12)
plt.title("Linear Regression Model", fontsize=14)
plt.legend()
plt.grid(True)
plt.show()
```

```
Scalar: tf.Tensor(3, shape=(), dtype=int32)
Vector: tf.Tensor([1 2 3 4 5], shape=(5,), dtype=int32)
Matrix: tf.Tensor(
[[1 2]
 [3 4]
 [5 6]], shape=(3, 2), dtype=int32)
Tensor 3D: tf.Tensor(
[[[1 2]
  [3 4]]

 [[5 6]
  [7 8]]], shape=(2, 2, 2), dtype=int32)
Epoch 0, Loss: 2.39792799949646
Epoch 100, Loss: 0.009327584877610207
Epoch 200, Loss: 0.008639639243483543
Epoch 300, Loss: 0.008591435849666595
Epoch 400, Loss: 0.008588059805333614
Epoch 500, Loss: 0.008587821386754513
Epoch 600, Loss: 0.008587806485593319
Epoch 700, Loss: 0.00858780462294817
Epoch 800, Loss: 0.008587806485593319
Epoch 900, Loss: 0.00858780462294817
```



Tensors are just multi-dimensional arrays (like a list or matrix) that hold data.
Scalar-This is just a single number (e.g., 3).
Vector- This is a list of numbers (e.g., [1, 2, 3, 4, 5]).
Matrix- A 2D array (like a grid) of numbers (e.g., [[1, 2], [3, 4], [5, 6]]).
3D Tensor- A more complex 3D array (e.g., [[[1, 2], [3, 4]], [[5, 6], [7, 8]]]). These tensors are just examples to show different shapes of data.