Homework -1:

Question: 3

Create a linear model using tensor flow for dynamic line equation y=mx +c. (your learning from Programming with Python to be applied as well)

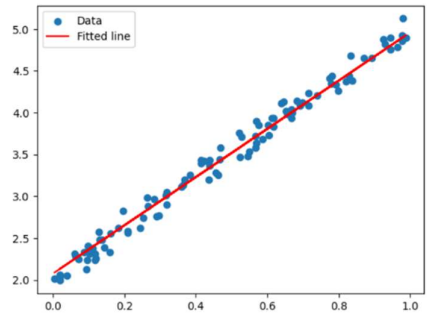
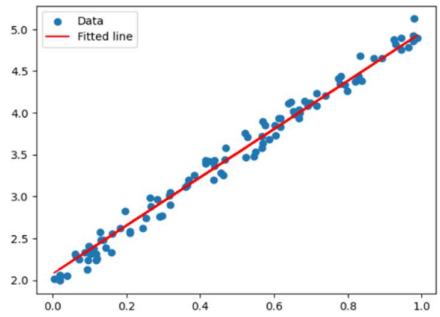
Solution:

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

# Generate some random data
np.random.seed(0)
X_train = np.random.rand(100, 1)
y_train = 3 * X_train + 2 + np.random.randn(100, 1) * 0.1

# Create a linear model
model = tf.keras.models.Sequential([
    tf.keras.layers.Dense(units=1, input_shape=[1])
])

# Compile the model
model.compile(optimizer='sgd', loss='mean_squared_error')

# Train the model
model.fit(X_train, y_train, epochs=500, verbose=0)

# Get the weights
weights = model.layers[0].get_weights()
m = weights[0][0][0]
c = weights[1][0]

print(f"y = {m:.2f}x + {c:.2f}")
```

```python
# Plot the data and the fitted line
plt.scatter(X_train, y_train, label='Data')
plt.plot(X_train, model.predict(X_train), color='red', label='Fitted line')
plt.legend()
plt.show()
```

Out put:

```
/usr/local/lib/python3.12/dist-packages/keras/src/layers/core/dense.py:93: UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When using Sequential models, pre
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
y = 2.88x + 2.07
4/4 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step
```



```
/usr/local/lib/python3.12/dist-packages/keras/src/layers/core/dense.py:93: UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When using Sequential models, pre
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
y = 2.88x + 2.07
4/4 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step
```