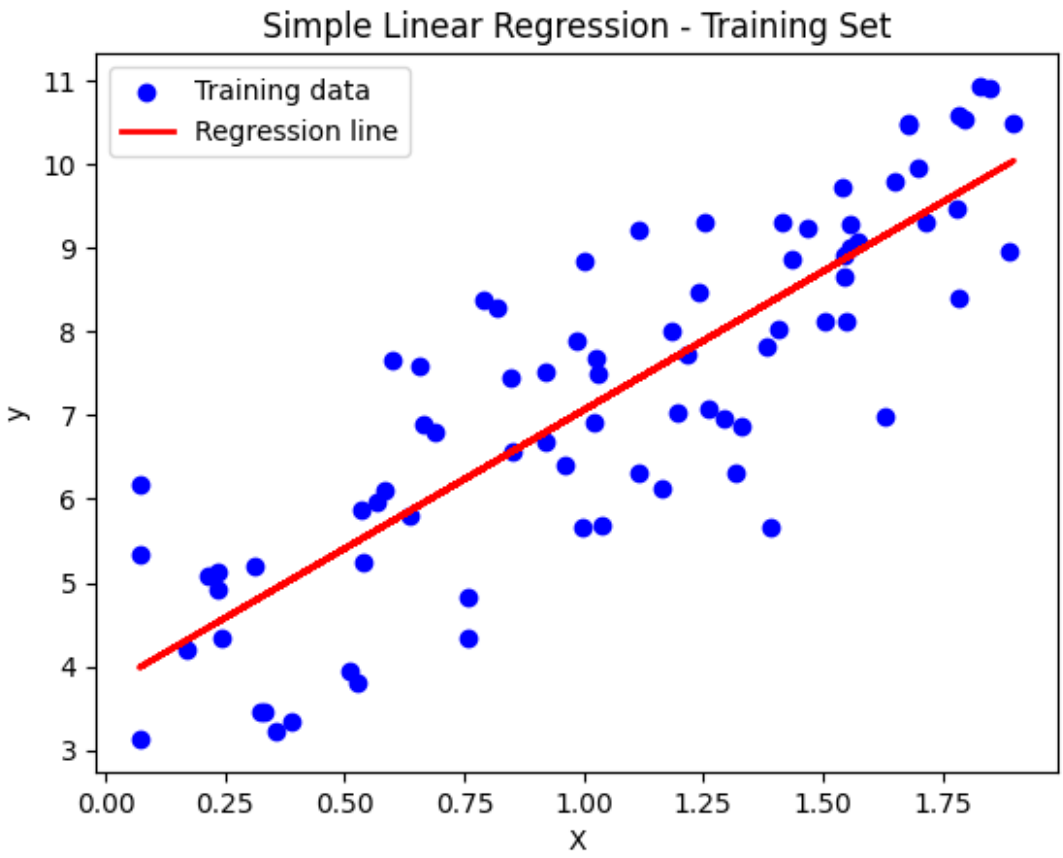
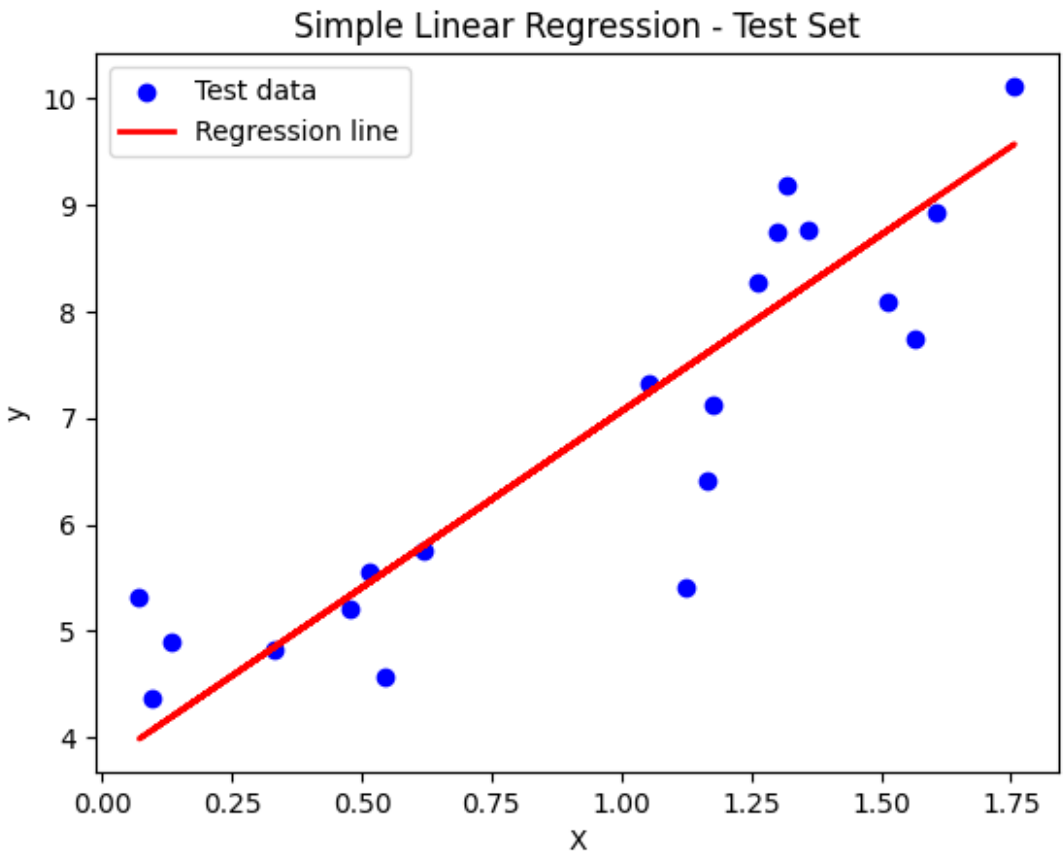
```
#Jayesh Mali T084
import numpy as np
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression

# Generate  data
np.random.seed(20)
X = 2 * np.random.rand(100, 1)  # Generate  values for the predictor
variable
y = 4 + 3 * X + np.random.randn(100, 1)  # Generate response variable
with noise

# Split the data into training and test sets
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.2, random_state=42)

# Train the linear regression model
model = LinearRegression()
model.fit(X_train, y_train)

# Predictions on the training and test sets
y_train_pred = model.predict(X_train)
y_test_pred = model.predict(X_test)

# Plot the training data and regression line
plt.scatter(X_train, y_train, color='blue', label='Training data')
plt.plot(X_train, y_train_pred, color='red', linewidth=2,
label='Regression line')
plt.xlabel('X')
plt.ylabel('y')
plt.title('Simple Linear Regression - Training Set')
plt.legend()
plt.show()

# Plot the test data and regression line
plt.scatter(X_test, y_test, color='blue', label='Test data')
plt.plot(X_test, y_test_pred, color='red', linewidth=2,
label='Regression line')
plt.xlabel('X')
plt.ylabel('y')
plt.title('Simple Linear Regression - Test Set')
plt.legend()
plt.show()
```


Simple Linear Regression - Training Set
Training data
Regression line
y
X
11
10
9
8
7
6
5
4
3
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75


Simple Linear Regression - Test Set
Test data
Regression line
y
X
10
9
8
7
6
5
4
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75