In-Lab

Task 1:

```
model2 = tree. DecisionTreeRegressor()
model2.fit(X_train, y_train)
print("Decision Tree")
print("==========")
y_pred_train2 = model2.predict(X_train)
RMSE_train2 = mean_squared_error (y_train, y_pred_train2)
print("Decision Tree Train set: RMSE {}".format(RMSE_train2))
y_pred_test2 = model2.predict(X_test)
RMSE_test2 = mean_squared_error(y_test,y_pred_test2)
print("Decision Tree Test set: RMSE {}".format(RMSE_test2))
print("================")
```

Output:

```
Decision Tree
==========
Decision Tree Train set: RMSE 0.0
Decision Tree Test set: RMSE 0.7583333361111111
================
```

Task 2:

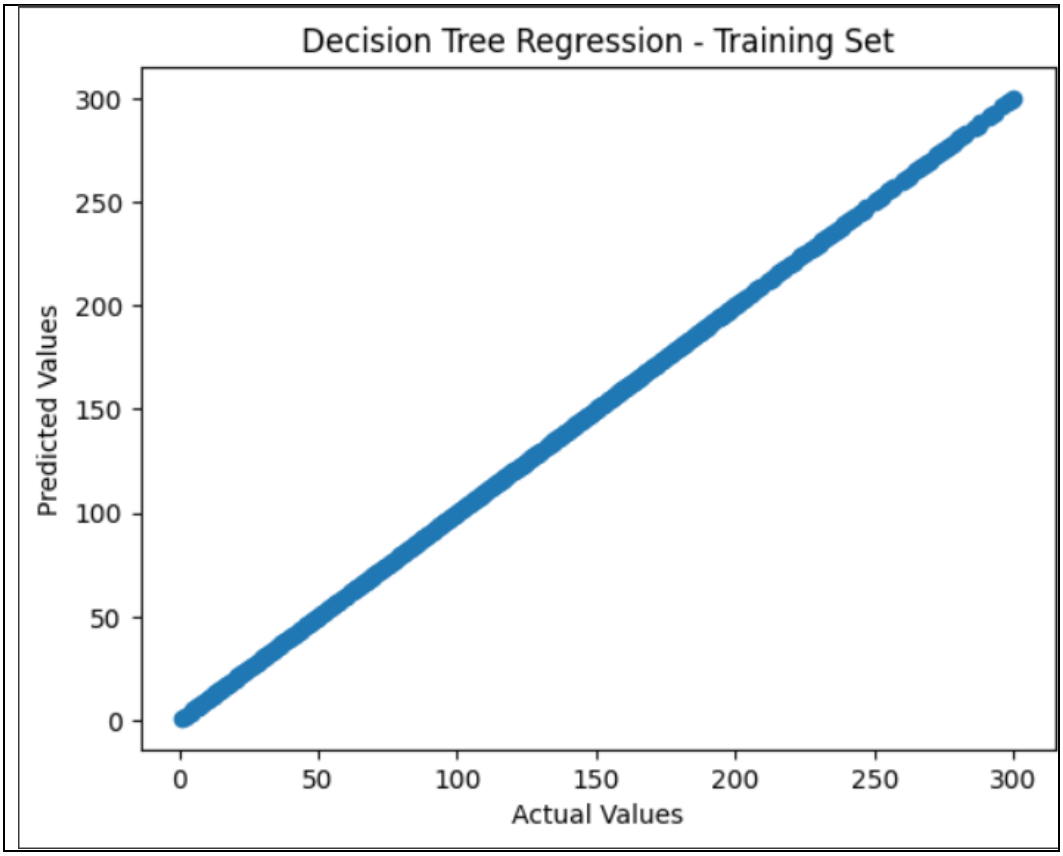
```
import matplotlib.pyplot as plt

# Your code for training the Decision Tree model goes here
model2 = tree.DecisionTreeRegressor()
model2.fit(X_train, y_train)

# Predictions on the training set
y_pred_train2 = model2.predict(X_train)
```

```
# Scatter plot for actual vs. predicted values in the
training set
plt.scatter(y_train, y_pred_train2)
plt.title('Decision Tree Regression - Training Set')
plt.xlabel('Actual Values')
plt.ylabel('Predicted Values')
plt.show()
```

Output:



Task 3:

```
# Your code for training the Decision Tree model goes here
model2 = tree.DecisionTreeRegressor()
model2.fit(X_train, y_train)

# Predictions on the test set
y_pred_test2 = model2.predict(X_test)

# Print results for the test set
print("Decision Tree")
print("==========")
RMSE_test2 = mean_squared_error(y_test, y_pred_test2)
print("Decision Tree Test set: RMSE {}".format(RMSE_test2))

# Scatter plot for actual vs. predicted values in the test
set
plt.scatter(y_test, y_pred_test2)
plt.title('Decision Tree Regression - Test Set')
plt.xlabel('Actual Values')
plt.ylabel('Predicted Values')
plt.show()
```

Output:

```
Decision Tree
==========
Decision Tree Test set: RMSE 0.7583333327777781
```