In-Lab

Task 1:
Decision tree:

```
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 1.4739259778473743e-36
Decision Tree Test set: RMSE 0.008436
========================================
```

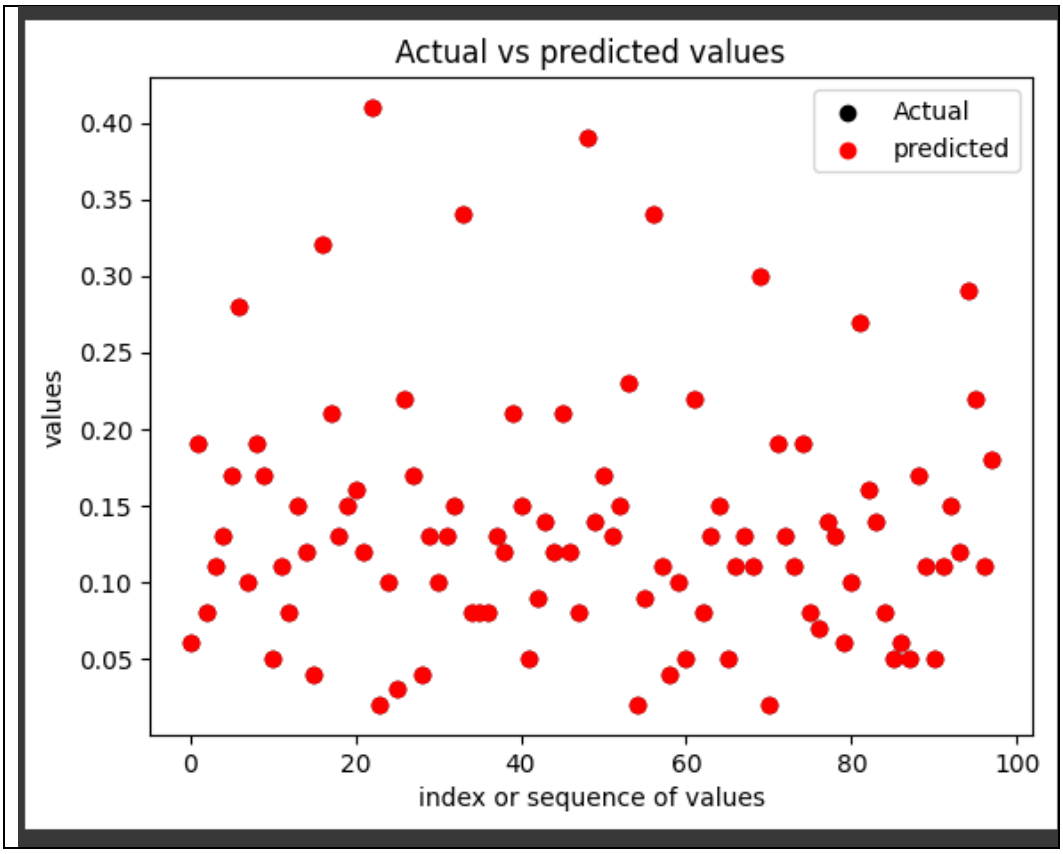
Task 2:

Load and show the data set in the given below section:

```
x_values = np.arange(len(y_train))
plt.scatter(x_values,y_train,color = 'black',label = 'Actual')
plt.scatter(x_values,y_pred_train2,color = 'red',label = 'predicted')
plt.xlabel('index or sequence of values')
plt.ylabel('values')
```

```
plt.title('Actual vs predicted values')
plt.legend()
plt.show()
```

Output:

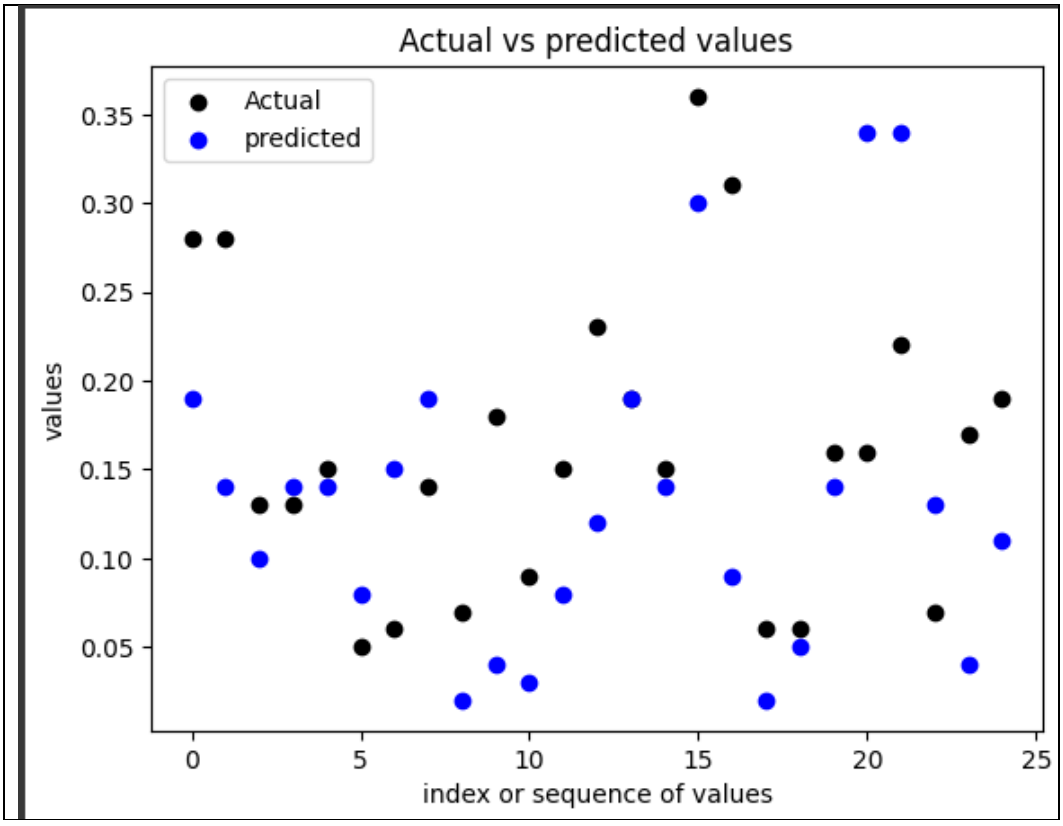


Task 3:

```
x_values = np.arange(len(y_test))
plt.scatter(x_values,y_test,color = 'black',label = 'Actual')
plt.scatter(x_values,y_pred_test2,color = 'blue',label = 'predicted')
plt.xlabel('index or sequence of values')
```

```
plt.ylabel('values')
plt.title('Actual vs predicted values')
plt.legend()
plt.show()
```

Output:



Task 4:

In this section we will calculate the correlation as given below:

```
mode13 = RandomForestRegressor()

mode13.fit(X_train, y_train)

print("Random Forest Regressor")

print('====================================')
y_pred_train3 = mode13.predict(X_train)

RMSE_train3 = mean_squared_error(y_train,y_pred_train3)
print("Random Forest Regressor TrainSet: RMSE {}".format(RMSE_train3))

print('====================================')

y_pred_test3 = mode13.predict(X_test)

RMSE_test3 = mean_squared_error(y_test,y_pred_test3)
print("Random Forest Regressor TestSet: RMSE {}".format(RMSE_test3))

print('====================================')
```

Output:

```
Random Forest Regressor
==================================
Random Forest Regressor TrainSet: RMSE 0.0004986999999999998
==================================
Random Forest Regressor TestSet: RMSE 0.004825038399999996
==================================
```