In [ ]:
```
# import required packages
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
from sklearn.preprocessing import MinMaxScaler
from keras.models import Sequential
from keras.layers import LSTM, Dropout, Dense

# YOUR IMPLEMENTATION
# Thoroughly comment your code to make it easy to follow

if __name__ == "__main__":
    data=pd.read_csv('/Users/eberechukwukathomas/Desktop/Thomas_20868189/data/q2_dataset.csv')#Loading
 the dataset from the directory
    data=data.drop(['Date'],axis=1)#dropping the date column
    data=data.reindex(columns=[' Open',' Close/Last',' Volume', ' High', ' Low'])#reshuffling the colum
ns
    data=data.replace('\$', '', regex=True).astype(float)#Removing the $ sign from the dataset
    data.isna().sum()#Checking for NULL values
    from sklearn.preprocessing import MinMaxScaler#Performing Normalization
    sc = MinMaxScaler(feature_range=(0,1))
    scaled_data= sc.fit_transform(data)
    #creating the dataset from the last 3 days
    X=[]
    y=[]
    for i in range(3, 1259):
        X.append(scaled_data[i-3:i,0])
        y.append(scaled_data[i,0])
        #Splitting the dataset into testing and train sets
    from sklearn.model_selection import train_test_split
    X_train, X_test, y_train, y_test=train_test_split(X, y, test_size=0.3, random_state=42)


        # 1. load your training data
    X_train, y_train=np.array(X_train), np.array(y_train)
    X_test,y_test=np.array(X_test), np.array(y_test)
    y_train=y_train.reshape(-1,1)
    y_test=y_test.reshape(-1,1)
    X_train=pd.DataFrame(X_train)
    y_train=pd.DataFrame(y_train)
    X_test=pd.DataFrame(X_test)
    y_test=pd.DataFrame(y_test)
    frames=[X_train,y_train]
    frameess=[X_test, y_test]
    result=pd.concat(frames, keys=['X_train', 'y_train'])
    resultsss=pd.concat(frameess,keys=['X_test', 'y_test'])
    result.to_csv('train_data_RNN.csv')
    resultsss.to_csv('test_data_RNN.csv')
    datal=pd.read_csv('train_data_RNN.csv')
    X_train=datal[0:879:]
    X_train=X_train.drop(['Unnamed: 0','Unnamed: 1'], axis=1)
    y_train=datal[879:1759:]
    y_train=y_train.drop(['Unnamed: 0','1','2','Unnamed: 1'], axis=1)
    X_train, y_train=np.array(X_train), np.array(y_train)
    X_train=np.reshape(X_train, (X_train.shape[0],
    X_train.shape[1],1))
        # 2. Train your network
    model = Sequential()
    model.add(LSTM(units=512,return_sequences=True,input_shape=(X_train.shape[1], 1)))
    model.add(Dropout(0.2))
    model.add(LSTM(units=512,return_sequences=True))
    model.add(Dropout(0.2))
    model.add(LSTM(units=512,return_sequences=True))
    model.add(Dropout(0.2))
    model.add(LSTM(units=512))
    model.add(Dropout(0.2))
    model.add(Dense(units=1))
    model.compile(optimizer='adam',loss='mean_squared_error', metrics=['accuracy'])
    model.fit(X_train,y_train,epochs=50,batch_size=70)
    sc=sc.scale_
    print(sc)
        #               Make sure to print your training loss within training to show progress
        #               Make sure you print the final training loss

        # 3. Save your model
    model.save('20868189_RNN_model')
```

```
Epoch 1/50
13/13 [==============================] - 1s 105ms/step - loss: 0.0500 - accuracy: 0.0011
Epoch 2/50
13/13 [==============================] - 1s 112ms/step - loss: 0.0045 - accuracy: 0.0011
Epoch 3/50
13/13 [==============================] - 1s 112ms/step - loss: 0.0018 - accuracy: 0.0011
Epoch 4/50
13/13 [==============================] - 1s 114ms/step - loss: 0.0010 - accuracy: 0.0011
Epoch 5/50
13/13 [==============================] - 2s 122ms/step - loss: 9.1131e-04 - accuracy: 0.0011
Epoch 6/50
13/13 [==============================] - 2s 115ms/step - loss: 6.8743e-04 - accuracy: 0.0011
Epoch 7/50
13/13 [==============================] - 2s 120ms/step - loss: 6.0996e-04 - accuracy: 0.0011
Epoch 8/50
13/13 [==============================] - 2s 126ms/step - loss: 6.3321e-04 - accuracy: 0.0011
Epoch 9/50
13/13 [==============================] - 1s 114ms/step - loss: 7.5887e-04 - accuracy: 0.0011
Epoch 10/50
13/13 [==============================] - 1s 113ms/step - loss: 0.0012 - accuracy: 0.0011
Epoch 11/50
13/13 [==============================] - 1s 113ms/step - loss: 0.0014 - accuracy: 0.0011
Epoch 12/50
13/13 [==============================] - 1s 115ms/step - loss: 9.4411e-04 - accuracy: 0.0011
Epoch 13/50
13/13 [==============================] - 2s 124ms/step - loss: 7.2554e-04 - accuracy: 0.0011
Epoch 14/50
13/13 [==============================] - 2s 121ms/step - loss: 6.0649e-04 - accuracy: 0.0011
Epoch 15/50
13/13 [==============================] - 1s 109ms/step - loss: 6.9816e-04 - accuracy: 0.0011
Epoch 16/50
13/13 [==============================] - 1s 108ms/step - loss: 5.3817e-04 - accuracy: 0.0011
Epoch 17/50
13/13 [==============================] - 1s 109ms/step - loss: 4.9802e-04 - accuracy: 0.0011
Epoch 18/50
13/13 [==============================] - 1s 110ms/step - loss: 5.1822e-04 - accuracy: 0.0011
Epoch 19/50
13/13 [==============================] - 2s 119ms/step - loss: 5.3725e-04 - accuracy: 0.0011
Epoch 20/50
13/13 [==============================] - 1s 109ms/step - loss: 7.2664e-04 - accuracy: 0.0011
Epoch 21/50
13/13 [==============================] - 1s 110ms/step - loss: 8.6338e-04 - accuracy: 0.0011
Epoch 22/50
13/13 [==============================] - 1s 109ms/step - loss: 5.1314e-04 - accuracy: 0.0011
Epoch 23/50
13/13 [==============================] - 1s 113ms/step - loss: 5.0350e-04 - accuracy: 0.0011
Epoch 24/50
13/13 [==============================] - 1s 114ms/step - loss: 8.1291e-04 - accuracy: 0.0011
Epoch 25/50
13/13 [==============================] - 1s 112ms/step - loss: 5.9629e-04 - accuracy: 0.0011
Epoch 26/50
13/13 [==============================] - 1s 114ms/step - loss: 5.8152e-04 - accuracy: 0.0011
Epoch 27/50
13/13 [==============================] - 2s 123ms/step - loss: 5.1854e-04 - accuracy: 0.0011
Epoch 28/50
13/13 [==============================] - 2s 121ms/step - loss: 5.2168e-04 - accuracy: 0.0011
Epoch 29/50
13/13 [==============================] - 2s 116ms/step - loss: 4.9952e-04 - accuracy: 0.0011
Epoch 30/50
13/13 [==============================] - 1s 115ms/step - loss: 4.7706e-04 - accuracy: 0.0011
Epoch 31/50
13/13 [==============================] - 2s 124ms/step - loss: 5.3029e-04 - accuracy: 0.0011
Epoch 32/50
13/13 [==============================] - 1s 115ms/step - loss: 5.2752e-04 - accuracy: 0.0011
Epoch 33/50
13/13 [==============================] - 2s 116ms/step - loss: 4.5975e-04 - accuracy: 0.0011
Epoch 34/50
13/13 [==============================] - 1s 115ms/step - loss: 5.6496e-04 - accuracy: 0.0011
Epoch 35/50
13/13 [==============================] - 2s 124ms/step - loss: 5.8367e-04 - accuracy: 0.0011
Epoch 36/50
13/13 [==============================] - 2s 120ms/step - loss: 4.3594e-04 - accuracy: 0.0011
Epoch 37/50
13/13 [==============================] - 1s 114ms/step - loss: 5.0107e-04 - accuracy: 0.0011
Epoch 38/50
13/13 [==============================] - 1s 115ms/step - loss: 5.0490e-04 - accuracy: 0.0011
Epoch 39/50
13/13 [==============================] - 2s 127ms/step - loss: 6.0985e-04 - accuracy: 0.0011
Epoch 40/50
13/13 [==============================] - 2s 116ms/step - loss: 4.8573e-04 - accuracy: 0.0011
Epoch 41/50
13/13 [==============================] - 1s 115ms/step - loss: 5.8727e-04 - accuracy: 0.0011
Epoch 42/50
13/13 [==============================] - 1s 112ms/step - loss: 7.8116e-04 - accuracy: 0.0011
Epoch 43/50
13/13 [==============================] - 1s 113ms/step - loss: 6.0617e-04 - accuracy: 0.0011
Epoch 44/50
13/13 [==============================] - 1s 113ms/step - loss: 5.7805e-04 - accuracy: 0.0011
Epoch 45/50
13/13 [==============================] - 1s 109ms/step - loss: 5.3229e-04 - accuracy: 0.0011
Epoch 46/50
13/13 [==============================] - 1s 111ms/step - loss: 6.8701e-04 - accuracy: 0.0011
Epoch 47/50
13/13 [==============================] - 2s 119ms/step - loss: 4.7513e-04 - accuracy: 0.0011
Epoch 48/50
13/13 [==============================] - 1s 110ms/step - loss: 4.7515e-04 - accuracy: 0.0011
Epoch 49/50
13/13 [==============================] - 1s 110ms/step - loss: 4.4704e-04 - accuracy: 0.0011
Epoch 50/50
13/13 [==============================] - 1s 110ms/step - loss: 4.6660e-04 - accuracy: 0.0011
[3.48772321e-03 3.43607188e-03 6.64361554e-09 3.45029845e-03
 3.48565652e-03]
```

In [ ]:

In [ ]:

In [ ]: