In [ ]:
```
import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)
import matplotlib.pyplot as plt

# Input data files are available in the "../input/" directory.
# For example, running this (by clicking run or pressing Shift+Enter) will list the

import os
print(os.listdir("../input"))

# Any results you write to the current directory are saved as output.
```

```
['testset.csv', 'trainset.csv']
```

In [ ]:
```
dataset_train = pd.read_csv("../input/trainset.csv")
```

In [ ]:
```
dataset_train
```

Out[ ]:

	Date	Open	High	Low	Close	Adj Close	Volume
0	2013-01-02	357.385559	361.151062	355.959839	359.288177	359.288177	5115500
1	2013-01-03	360.122742	363.600128	358.031342	359.496826	359.496826	4666500
2	2013-01-04	362.313507	368.339294	361.488861	366.600616	366.600616	5562800
3	2013-01-07	365.348755	367.301056	362.929504	365.001007	365.001007	3332900
4	2013-01-08	365.393463	365.771027	359.874359	364.280701	364.280701	3373900
5	2013-01-09	363.769043	366.789398	361.945892	366.675140	366.675140	4075700
6	2013-01-10	369.014923	370.092896	364.380066	368.344269	368.344269	3695100
7	2013-01-11	368.602600	368.816193	365.771027	367.604095	367.604095	2587000
8	2013-01-14	366.118744	368.701935	358.841095	359.288177	359.288177	5765000
9	2013-01-15	357.340851	365.125214	353.749207	360.122742	360.122742	7906300
10	2013-01-16	358.865936	359.829651	354.529144	355.284210	355.284210	4073100
11	2013-01-17	356.536072	357.494843	353.212677	353.361725	353.361725	4451700
12	2013-01-18	352.884827	354.082031	348.398987	349.978729	349.978729	6495500
13	2013-01-22	350.053253	350.391052	345.512787	349.164032	349.164032	7634000
14	2013-01-23	365.617004	372.079987	365.517670	368.354218	368.354218	11895000
15	2013-01-24	368.225037	375.969666	367.862396	374.668152	374.668152	6809200
16	2013-01-25	372.959259	376.789337	372.700928	374.399902	374.399902	4480700
17	2013-01-28	373.451050	375.358643	371.528564	372.939392	372.939392	3275300
18	2013-01-29	370.962250	376.029297	370.857941	374.404846	374.404846	3516800
19	2013-01-30	374.434662	378.016357	374.022339	374.479370	374.479370	3488500
20	2013-01-31	372.830109	376.362122	372.700928	375.403351	375.403351	3289500
21	2013-02-01	376.650238	385.790802	376.600586	385.294037	385.294037	7540700
22	2013-02-04	381.364594	382.745605	376.685028	377.057617	377.057617	6120500
23	2013-02-05	378.105774	383.063538	377.281158	380.395905	380.395905	3765600
24	2013-02-06	377.082428	383.982574	376.799286	382.596588	382.596588	4183200
25	2013-02-07	382.363098	386.888672	380.276672	384.474365	384.474365	5717300
26	2013-02-08	387.544403	390.793274	387.261230	390.147461	390.147461	6079300
27	2013-02-11	386.684998	388.970123	384.375000	388.682007	388.682007	4363700
28	2013-02-12	388.349152	391.404297	387.166840	387.827545	387.827545	3742100
29	2013-02-13	387.544403	390.137543	387.464905	388.900574	388.900574	2411800
...	...	...	...	...	...	...	...
1229	2017-11-16	1022.520020	1035.920044	1022.520020	1032.500000	1032.500000	1129700
1230	2017-11-17	1034.010010	1034.420044	1017.750000	1019.090027	1019.090027	1397100
1231	2017-11-20	1020.260010	1022.609985	1017.500000	1018.380005	1018.380005	953500
1232	2017-11-21	1023.309998	1035.109985	1022.655029	1034.489990	1034.489990	1097000
1233	2017-11-22	1035.000000	1039.706055	1031.430054	1035.959961	1035.959961	746300

	Date	Open	High	Low	Close	Adj Close	Volume
1234	2017-11-24	1035.869995	1043.177979	1035.000000	1040.609985	1040.609985	537000
1235	2017-11-27	1040.000000	1055.459961	1038.439941	1054.209961	1054.209961	1307900
1236	2017-11-28	1055.089966	1062.375000	1040.000000	1047.410034	1047.410034	1424400
1237	2017-11-29	1042.680054	1044.079956	1015.650024	1021.659973	1021.659973	2459400
1238	2017-11-30	1022.369995	1028.489990	1015.000000	1021.409973	1021.409973	1724000
1239	2017-12-01	1015.799988	1022.489990	1002.020020	1010.169983	1010.169983	1909600
1240	2017-12-04	1012.659973	1016.099976	995.570007	998.679993	998.679993	1906400
1241	2017-12-05	995.940002	1020.609985	988.280029	1005.150024	1005.150024	2067300
1242	2017-12-06	1001.500000	1024.969971	1001.140015	1018.380005	1018.380005	1272000
1243	2017-12-07	1020.429993	1034.239990	1018.070984	1030.930054	1030.930054	1458200
1244	2017-12-08	1037.489990	1042.050049	1032.521973	1037.050049	1037.050049	1290800
1245	2017-12-11	1035.500000	1043.800049	1032.050049	1041.099976	1041.099976	1192800
1246	2017-12-12	1039.630005	1050.310059	1033.689941	1040.479980	1040.479980	1279500
1247	2017-12-13	1046.119995	1046.665039	1038.380005	1040.609985	1040.609985	1282700
1248	2017-12-14	1045.000000	1058.500000	1043.109985	1049.150024	1049.150024	1558700
1249	2017-12-15	1054.609985	1067.619995	1049.500000	1064.189941	1064.189941	3275900
1250	2017-12-18	1066.079956	1078.489990	1062.000000	1077.140015	1077.140015	1554600
1251	2017-12-19	1075.199951	1076.839966	1063.550049	1070.680054	1070.680054	1338700
1252	2017-12-20	1071.780029	1073.380005	1061.520020	1064.949951	1064.949951	1268600
1253	2017-12-21	1064.949951	1069.329956	1061.793945	1063.630005	1063.630005	995700
1254	2017-12-22	1061.109985	1064.199951	1059.439941	1060.119995	1060.119995	755100
1255	2017-12-26	1058.069946	1060.119995	1050.199951	1056.739990	1056.739990	760600
1256	2017-12-27	1057.390015	1058.369995	1048.050049	1049.369995	1049.369995	1271900
1257	2017-12-28	1051.599976	1054.750000	1044.770020	1048.140015	1048.140015	837100
1258	2017-12-29	1046.719971	1049.699951	1044.900024	1046.400024	1046.400024	887500

1259 rows × 7 columns

In [ ]:
```
trainset = dataset_train.iloc[:,1:2].values
```

In [ ]:
```
trainset
```

Out[ ]:
```
array([[ 357.385559],
       [ 360.122742],
       [ 362.313507],
       ...,
       [1057.390015],
       [1051.599976],
       [1046.719971]])
```

In [ ]:
```
from sklearn.preprocessing import MinMaxScaler
sc = MinMaxScaler(feature_range = (0,1))
training_scaled = sc.fit_transform(trainset)
```

In [ ]:
```
training_scaled
```

Out[ ]:
```
array([[0.01011148],
       [0.01388614],
       [0.01690727],
       ...,
       [0.97543954],
       [0.9674549 ],
       [0.96072522]])
```

In [ ]:
```
x_train = []
y_train = []
```

In [ ]:
```
for i in range(60,1259):
    x_train.append(training_scaled[i-60:i, 0])
    y_train.append(training_scaled[i,0])
x_train,y_train = np.array(x_train),np.array(y_train)
```

In [ ]:
```
x_train.shape
```

Out[ ]:
```
(1199, 60)
```

In [ ]:
```
x_train = np.reshape(x_train, (x_train.shape[0],x_train.shape[1],1))
```

In [ ]:
```
from keras.models import Sequential
from keras.layers import Dense
from keras.layers import LSTM
from keras.layers import Dropout
```

```
Using TensorFlow backend.
```

In [ ]:
```
regressor = Sequential()
regressor.add(LSTM(units = 50,return_sequences = True,input_shape = (x_train.shape
```

In [ ]:
```
regressor.add(Dropout(0.2))
```

In [ ]:
```
regressor.add(LSTM(units = 50,return_sequences = True))
regressor.add(Dropout(0.2))
```

In [ ]:
```
regressor.add(LSTM(units = 50,return_sequences = True))
regressor.add(Dropout(0.2))
```

In [ ]:
```
regressor.add(LSTM(units = 50))
regressor.add(Dropout(0.2))
```

In [ ]:
```
regressor.add(Dense(units = 1))
```

In [ ]:
```
regressor.compile(optimizer = 'adam',loss = 'mean_squared_error')
```

In [ ]:
```
regressor.fit(x_train,y_train,epochs = 100, batch_size = 32)
```

```
Epoch 1/100
1199/1199 [==============================] - 13s 11ms/step - loss: 0.0295
Epoch 2/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0054
Epoch 3/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0038
Epoch 4/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0040
Epoch 5/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0037
Epoch 6/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0032
Epoch 7/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0035
Epoch 8/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0033
Epoch 9/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0034
Epoch 10/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0029
Epoch 11/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0032
Epoch 12/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0028
Epoch 13/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0030
Epoch 14/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0029
Epoch 15/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0029
Epoch 16/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0026
Epoch 17/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0025
Epoch 18/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0028
Epoch 19/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0023
Epoch 20/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0025
Epoch 21/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0025
Epoch 22/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0024
Epoch 23/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0022
Epoch 24/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0023
Epoch 25/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0022
Epoch 26/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0020
Epoch 27/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0024
Epoch 28/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0023
Epoch 29/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0019
Epoch 30/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0023
Epoch 31/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0021
Epoch 32/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0021
```

```
Epoch 33/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0018
Epoch 34/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0022
Epoch 35/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0020
Epoch 36/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0019
Epoch 37/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0018
Epoch 38/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0017
Epoch 39/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0017
Epoch 40/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0019
Epoch 41/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0018
Epoch 42/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0019
Epoch 43/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0016
Epoch 44/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0021
Epoch 45/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0018
Epoch 46/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0016
Epoch 47/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0016
Epoch 48/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0017
Epoch 49/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0017
Epoch 50/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0015
Epoch 51/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0017
Epoch 52/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0016
Epoch 53/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0015
Epoch 54/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0014
Epoch 55/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0014
Epoch 56/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0014
Epoch 57/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0014
Epoch 58/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0014
Epoch 59/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0014
Epoch 60/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0016
Epoch 61/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0015
Epoch 62/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0015
Epoch 63/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0013
Epoch 64/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0014
```

```
Epoch 65/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0013
Epoch 66/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0012
Epoch 67/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0014
Epoch 68/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0014
Epoch 69/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0015
Epoch 70/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0014
Epoch 71/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0013
Epoch 72/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0013
Epoch 73/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0013
Epoch 74/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0013
Epoch 75/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0011
Epoch 76/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0011
Epoch 77/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0013
Epoch 78/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0011
Epoch 79/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0013
Epoch 80/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0011
Epoch 81/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0012
Epoch 82/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0011
Epoch 83/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0011
Epoch 84/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0013
Epoch 85/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0011
Epoch 86/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0011
Epoch 87/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0010
Epoch 88/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0011
Epoch 89/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0010
Epoch 90/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0010
Epoch 91/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0010
Epoch 92/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0010
Epoch 93/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0011
Epoch 94/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0010
Epoch 95/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0010
Epoch 96/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0010
```

```
Epoch 97/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0011
Epoch 98/100
1199/1199 [==============================] - 9s 7ms/step - loss: 0.0012
Epoch 99/100
1199/1199 [==============================] - 9s 8ms/step - loss: 0.0011
Epoch 100/100
1199/1199 [==============================] - 9s 7ms/step - loss: 8.4624e-04
```

Out[ ]:
```
<keras.callbacks.History at 0x7a0a49375c88>
```

In [ ]:
```python
dataset_test =pd.read_csv("../input/testset.csv")
```

In [ ]:
```python
real_stock_price = dataset_test.iloc[:,1:2].values
```

In [ ]:
```python
dataset_total = pd.concat((dataset_train['Open'],dataset_test['Open']),axis = 0)
dataset_total
```

Out[ ]:
```
0       357.385559
1       360.122742
2       362.313507
3       365.348755
4       365.393463
5       363.769043
6       369.014923
7       368.602600
8       366.118744
9       357.340851
10      358.865936
11      356.536072
12      352.884827
13      350.053253
14      365.617004
15      368.225037
16      372.959259
17      373.451050
18      370.962250
19      374.434662
20      372.830109
21      376.650238
22      381.364594
23      378.105774
24      377.082428
25      382.363098
26      387.544403
27      386.684998
28      388.349152
29      387.544403
           ...
95     1061.859985
96     1074.060059
97     1083.560059
98     1065.130005
99     1079.000000
100    1079.020020
101    1064.890015
102    1063.030029
103    1067.560059
104    1099.349976
105    1122.329956
106    1140.989990
107    1142.170044
108    1131.319946
109    1118.180054
110    1118.599976
111    1131.069946
112    1141.119995
113    1143.849976
114    1148.859985
115    1143.650024
116    1158.500000
117    1175.310059
118    1174.849976
119    1159.140015
120    1143.599976
121    1128.000000
122    1121.339966
123    1102.089966
124    1120.000000
Name: Open, Length: 1384, dtype: float64
```

In [ ]:
```
inputs = dataset_total[len(dataset_total) - len(dataset_test)-60:].values
inputs
```

Out[ ]:
```
array([ 955.48999 ,  966.700012,  980.      ,  980.      ,  973.719971,
        987.450012,  992.      ,  992.099976,  990.289978,  991.77002 ,
        986.      ,  989.440002,  989.52002 ,  970.      ,  968.369995,
        980.      , 1009.190002, 1014.      , 1015.219971, 1017.210022,
       1021.76001 , 1022.109985, 1028.98999 , 1027.27002 , 1030.52002 ,
       1033.98999 , 1026.459961, 1023.419983, 1022.590027, 1019.210022,
       1022.52002 , 1034.01001 , 1020.26001 , 1023.309998, 1035.      ,
       1035.869995, 1040.      , 1055.089966, 1042.680054, 1022.369995,
       1015.799988, 1012.659973,  995.940002, 1001.5     , 1020.429993,
       1037.48999 , 1035.5     , 1039.630005, 1046.119995, 1045.      ,
       1054.609985, 1066.079956, 1075.199951, 1071.780029, 1064.949951,
       1061.109985, 1058.069946, 1057.390015, 1051.599976, 1046.719971,
       1048.339966, 1064.310059, 1088.      , 1094.      , 1102.22998 ,
       1109.400024, 1097.099976, 1106.300049, 1102.410034, 1132.51001 ,
       1126.219971, 1131.410034, 1131.829956, 1137.48999 , 1159.849976,
       1177.329956, 1172.530029, 1175.079956, 1176.47998 , 1167.829956,
       1170.569946, 1162.609985, 1122.      , 1090.599976, 1027.180054,
       1081.540039, 1055.410034, 1017.25    , 1048.      , 1045.      ,
       1048.949951, 1079.069946, 1088.410034, 1090.569946, 1106.469971,
       1116.189941, 1112.640015, 1127.800049, 1141.23999 , 1123.030029,
       1107.869995, 1053.079956, 1075.140015, 1099.219971, 1089.189941,
       1115.319946, 1136.      , 1163.849976, 1170.      , 1145.209961,
       1149.959961, 1154.140015, 1120.01001 , 1099.      , 1092.73999 ,
       1081.880005, 1047.030029, 1046.      , 1063.      ,  998.      ,
       1011.630005, 1022.820007, 1013.909973,  993.409973, 1041.329956,
       1020.      , 1016.799988, 1026.439941, 1027.98999 , 1025.040039,
       1040.880005, 1037.      , 1051.369995, 1077.430054, 1069.400024,
       1082.      , 1077.859985, 1052.      , 1025.52002 , 1029.51001 ,
       1046.      , 1030.01001 , 1013.659973, 1028.099976, 1019.      ,
       1016.900024, 1049.22998 , 1058.540039, 1058.099976, 1086.030029,
       1093.599976, 1100.      , 1090.      , 1077.310059, 1079.890015,
       1061.859985, 1074.060059, 1083.560059, 1065.130005, 1079.      ,
       1079.02002 , 1064.890015, 1063.030029, 1067.560059, 1099.349976,
       1122.329956, 1140.98999 , 1142.170044, 1131.319946, 1118.180054,
       1118.599976, 1131.069946, 1141.119995, 1143.849976, 1148.859985,
       1143.650024, 1158.5     , 1175.310059, 1174.849976, 1159.140015,
       1143.599976, 1128.      , 1121.339966, 1102.089966, 1120.      ])
```

In [ ]:
```
inputs = inputs.reshape(-1,1)
```

In [ ]:
```
inputs
```

```
Out[ ]: array([[ 955.48999 ],
               [ 966.700012],
               [ 980.      ],
               [ 980.      ],
               [ 973.719971],
               [ 987.450012],
               [ 992.      ],
               [ 992.099976],
               [ 990.289978],
               [ 991.77002 ],
               [ 986.      ],
               [ 989.440002],
               [ 989.52002 ],
               [ 970.      ],
               [ 968.369995],
               [ 980.      ],
               [1009.190002],
               [1014.      ],
               [1015.219971],
               [1017.210022],
               [1021.76001 ],
               [1022.109985],
               [1028.98999 ],
               [1027.27002 ],
               [1030.52002 ],
               [1033.98999 ],
               [1026.459961],
               [1023.419983],
               [1022.590027],
               [1019.210022],
               [1022.52002 ],
               [1034.01001 ],
               [1020.26001 ],
               [1023.309998],
               [1035.      ],
               [1035.869995],
               [1040.      ],
               [1055.089966],
               [1042.680054],
               [1022.369995],
               [1015.799988],
               [1012.659973],
               [ 995.940002],
               [1001.5     ],
               [1020.429993],
               [1037.48999 ],
               [1035.5     ],
               [1039.630005],
               [1046.119995],
               [1045.      ],
               [1054.609985],
               [1066.079956],
               [1075.199951],
               [1071.780029],
               [1064.949951],
               [1061.109985],
               [1058.069946],
               [1057.390015],
               [1051.599976],
               [1046.719971],
               [1048.339966],
               [1064.310059],
               [1088.      ],
               [1094.      ],
               [1102.22998 ],
               [1109.400024],
               [1097.099976],
               [1106.300049],
               [1102.410034],
               [1132.51001 ],
               [1126.219971],
               [1131.410034],
               [1131.829956],
               [1137.48999 ],
               [1159.849976],
               [1177.329956],
               [1172.530029],
               [1175.079956],
               [1176.47998 ],
               [1167.829956],
               [1170.569946],
               [1162.609985],
               [1122.      ],
               [1090.599976],
               [1027.180054],
               [1081.540039],
               [1055.410034],
               [1017.25    ],
               [1048.      ],
               [1045.      ],
               [1048.949951],
               [1079.069946],
               [1088.410034],
               [1090.569946],
               [1106.469971],
               [1116.189941],
               [1112.640015],
               [1127.800049],
               [1141.23999 ],
               [1123.030029],
               [1107.869995],
               [1053.079956],
               [1075.140015],
               [1099.219971],
               [1089.189941],
               [1115.319946],
               [1136.      ],
               [1163.849976],
               [1170.      ],
               [1145.209961],
               [1149.959961],
               [1154.140015],
               [1120.01001 ],
               [1099.      ],
               [1092.73999 ],
               [1081.880005],
               [1047.030029],
               [1046.      ],
               [1063.      ],
               [ 998.      ],
               [1011.630005],
               [1022.820007],
               [1013.909973],
               [ 993.409973],
               [1041.329956],
               [1020.      ],
               [1016.799988],
               [1026.439941],
               [1027.98999 ],
               [1025.040039],
               [1040.880005],
               [1037.      ],
               [1051.369995],
               [1077.430054],
               [1069.400024],
               [1082.      ],
               [1077.859985],
               [1052.      ],
               [1025.52002 ],
               [1029.51001 ],
               [1046.      ],
               [1030.01001 ],
               [1013.659973],
               [1028.099976],
               [1019.      ],
               [1016.900024],
               [1049.22998 ],
               [1058.540039],
               [1058.099976],
               [1086.030029],
               [1093.599976],
               [1100.      ],
               [1090.      ],
               [1077.310059],
               [1079.890015],
               [1061.859985],
               [1074.060059],
               [1083.560059],
               [1065.130005],
               [1079.      ],
               [1079.02002 ],
               [1064.890015],
               [1063.030029],
               [1067.560059],
               [1099.349976],
               [1122.329956],
               [1140.98999 ],
               [1142.170044],
               [1131.319946],
               [1118.180054],
               [1118.599976],
               [1131.069946],
               [1141.119995],
               [1143.849976],
               [1148.859985],
               [1143.650024],
               [1158.5     ],
               [1175.310059],
               [1174.849976],
               [1159.140015],
               [1143.599976],
               [1128.      ],
               [1121.339966],
               [1102.089966],
               [1120.      ]])
```

In [ ]:

```
inputs = sc.transform(inputs)
inputs.shape
```

Out[ ]:

```
(185, 1)
```

In [ ]:
```
x_test = []
for i in range(60,185):
    x_test.append(inputs[i-60:i,0])
```

In [ ]:
```
x_test = np.array(x_test)
x_test.shape
```

Out[ ]:
```
(125, 60)
```

In [ ]:
```
x_test = np.reshape(x_test, (x_test.shape[0],x_test.shape[1],1))
x_test.shape
```

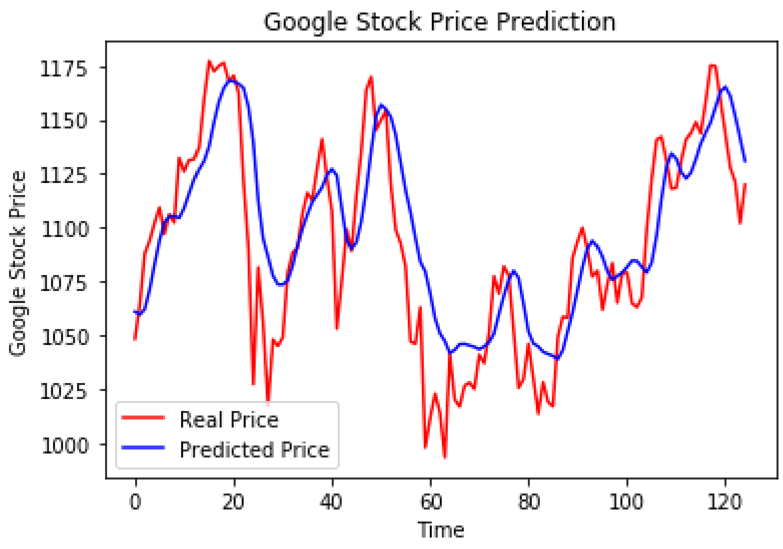
Out[ ]:
```
(125, 60, 1)
```

In [ ]:
```
predicted_price = regressor.predict(x_test)
```

In [ ]:
```
predicted_price = sc.inverse_transform(predicted_price)
predicted_price
```

Out[ ]:
```
array([[1060.9669],
       [1059.5607],
       [1062.1382],
       [1071.1857],
       [1082.9843],
       [1093.968 ],
       [1102.4915],
       [1105.2302],
       [1105.4479],
       [1104.4875],
       [1109.0391],
       [1115.7694],
       [1122.3834],
       [1127.2605],
       [1131.0854],
       [1137.9117],
       [1148.776 ],
       [1158.4279],
       [1164.7532],
       [1168.0515],
       [1167.8665],
       [1166.7338],
       [1164.827 ],
       [1156.1555],
       [1139.8356],
       [1112.7017],
       [1094.9691],
       [1086.4697],
       [1077.6523],
       [1073.6385],
       [1073.5703],
       [1075.4775],
       [1082.2805],
       [1091.726 ],
       [1099.5087],
       [1106.0602],
       [1111.9424],
       [1115.2893],
       [1118.7892],
       [1124.5618],
       [1127.3087],
       [1124.2736],
       [1109.1582],
       [1094.2224],
       [1090.1523],
       [1093.2799],
       [1103.134 ],
       [1117.746 ],
       [1135.7585],
       [1151.6923],
       [1156.9574],
       [1154.9141],
       [1151.2883],
       [1142.9575],
       [1130.1636],
       [1117.4648],
       [1107.2164],
       [1095.2411],
       [1084.1405],
       [1079.6997],
       [1069.5355],
       [1057.868 ],
       [1050.834 ],
       [1047.2107],
       [1041.6935],
       [1043.1472],
       [1045.8274],
       [1045.9597],
       [1045.2379],
       [1044.5918],
       [1043.4572],
       [1044.7418],
       [1046.7887],
       [1050.7759],
       [1059.7622],
       [1068.3588],
       [1075.7533],
       [1079.9879],
       [1076.6641],
       [1064.772 ],
       [1051.8538],
       [1046.1907],
       [1044.7688],
       [1042.2682],
       [1041.2609],
       [1040.3691],
       [1038.9095],
       [1042.9598],
       [1051.852 ],
       [1060.6005],
       [1071.0009],
       [1081.5089],
       [1090.2897],
       [1093.925 ],
       [1091.2341],
       [1086.4099],
       [1079.5878],
       [1075.7708],
       [1077.5172],
       [1079.16  ],
       [1081.7609],
       [1084.7081],
       [1084.5275],
       [1081.7229],
       [1079.1881],
       [1083.7085],
       [1096.3591],
       [1113.5286],
       [1128.1466],
       [1134.6011],
       [1132.142 ],
       [1125.9497],
       [1122.8896],
       [1125.6426],
       [1131.9459],
       [1139.2117],
       [1144.0309],
       [1148.7195],
       [1155.9124],
       [1163.1183],
       [1165.3479],
       [1161.0596],
       [1151.7821],
       [1141.6093],
       [1131.0759]], dtype=float32)
```

In [ ]:
```python
plt.plot(real_stock_price,color = 'red', label = 'Real Price')
plt.plot(predicted_price, color = 'blue', label = 'Predicted Price')
plt.title('Google Stock Price Prediction')
plt.xlabel('Time')
plt.ylabel('Google Stock Price')
plt.legend()
plt.show()
```


Google Stock Price Prediction
Google Stock Price
Real Price
Predicted Price
Time