In [ ]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import pandas_datareader as data
```

In [ ]:
```
start = '2010-01-01'
end = '2019-12-31'

df = data.DataReader('AAPL', 'yahoo', start, end)
df.head()
```

Out[ ]:

Date	High	Low	Open	Close	Volume	Adj Close
2009-12-31	7.619643	7.520000	7.611786	7.526071	352410800.0	6.444380
2010-01-04	7.660714	7.585000	7.622500	7.643214	493729600.0	6.544687
2010-01-05	7.699643	7.616071	7.664286	7.656429	601904800.0	6.556003
2010-01-06	7.686786	7.526786	7.656429	7.534643	552160000.0	6.451721
2010-01-07	7.571429	7.466071	7.562500	7.520714	477131200.0	6.439792

In [ ]:
```
df.tail()
```

Out[ ]:

Date	High	Low	Open	Close	Volume	Adj Close
2019-12-24	71.222504	70.730003	71.172501	71.067497	48478800.0	69.938194
2019-12-26	72.495003	71.175003	71.205002	72.477501	93121200.0	71.325790
2019-12-27	73.492500	72.029999	72.779999	72.449997	146266000.0	71.298737
2019-12-30	73.172501	71.305000	72.364998	72.879997	144114400.0	71.721901
2019-12-31	73.419998	72.379997	72.482498	73.412498	100805600.0	72.245934

In [ ]:
```
df = df.reset_index()
df.head()
```

Out[ ]:

	Date	High	Low	Open	Close	Volume	Adj Close
0	2009-12-31	7.619643	7.520000	7.611786	7.526071	352410800.0	6.444380
1	2010-01-04	7.660714	7.585000	7.622500	7.643214	493729600.0	6.544687
2	2010-01-05	7.699643	7.616071	7.664286	7.656429	601904800.0	6.556003
3	2010-01-06	7.686786	7.526786	7.656429	7.534643	552160000.0	6.451721
4	2010-01-07	7.571429	7.466071	7.562500	7.520714	477131200.0	6.439792

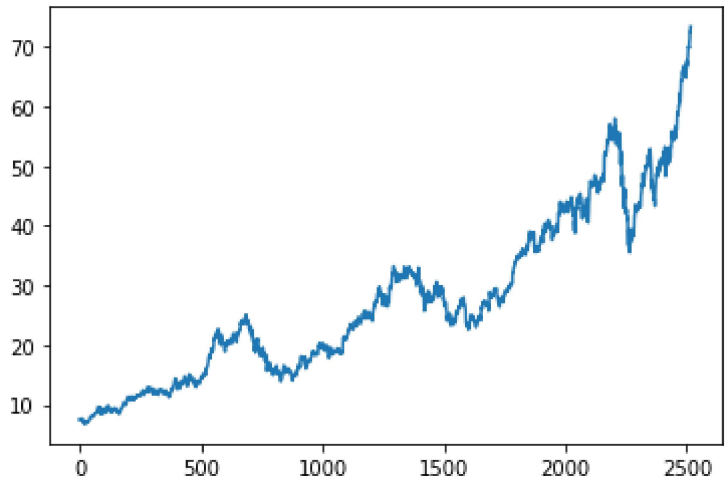
In [ ]:
```
df = df.drop(['Date', 'Adj Close'], axis=1)
df.head()
```

Out[ ]:

	High	Low	Open	Close	Volume
0	7.619643	7.520000	7.611786	7.526071	352410800.0
1	7.660714	7.585000	7.622500	7.643214	493729600.0
2	7.699643	7.616071	7.664286	7.656429	601904800.0
3	7.686786	7.526786	7.656429	7.534643	552160000.0
4	7.571429	7.466071	7.562500	7.520714	477131200.0

In [ ]:
```
plt.plot(df['Close'])
```

Out[ ]:
```
[<matplotlib.lines.Line2D at 0x17dc5ca9fc0>]
```

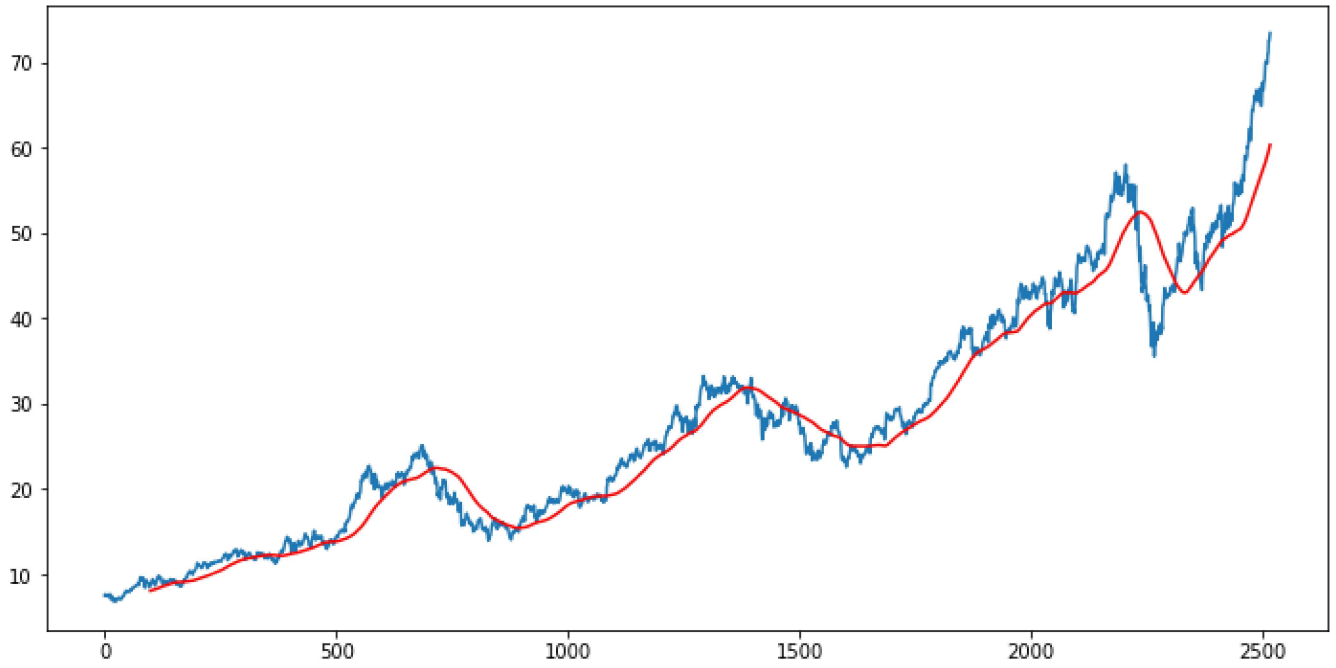


In [ ]:
```
ma100 = df.Close.rolling(window=100).mean()
ma100
```

Out[ ]:
```
0             NaN
1             NaN
2             NaN
3             NaN
4             NaN
           ...
2512    59.401700
2513    59.643125
2514    59.875125
2515    60.106325
2516    60.331875
Name: Close, Length: 2517, dtype: float64
```

In [ ]:
```
plt.figure(figsize=(12, 6))
plt.plot(df['Close'], label='Close')
plt.plot(ma100, 'r')
```

Out[ ]:
```
[<matplotlib.lines.Line2D at 0x17dc5e62fb0>]
```

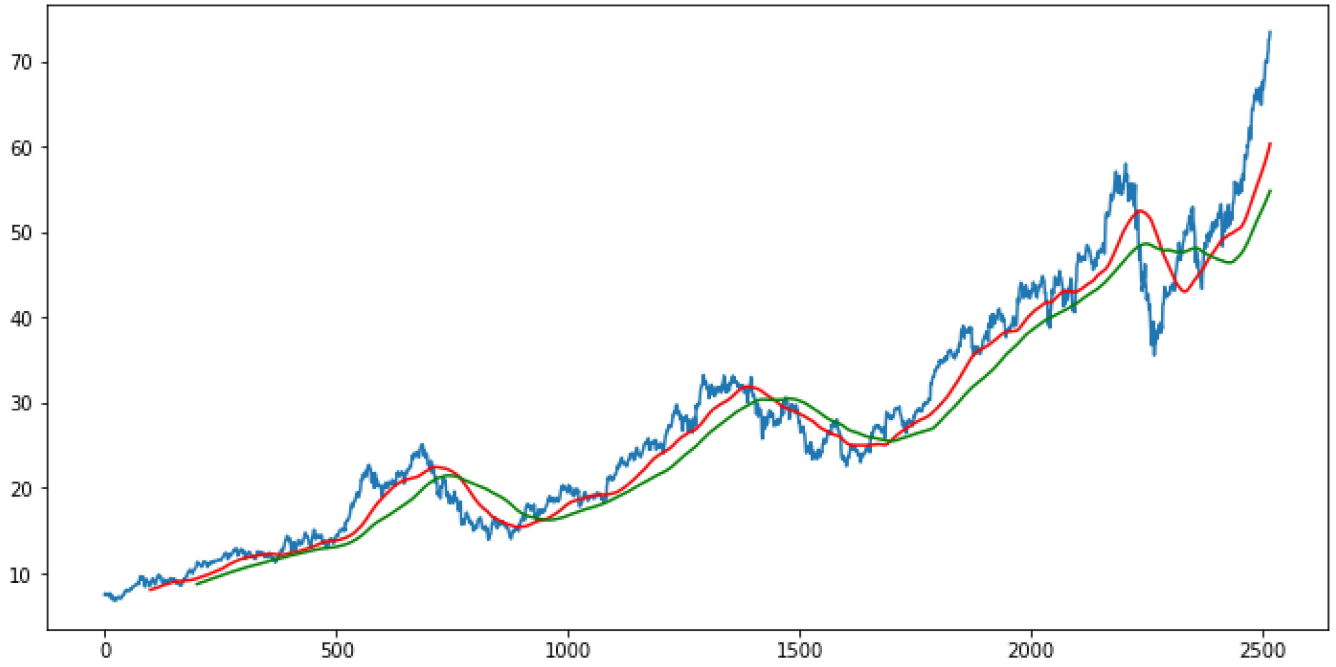


In [ ]:
```
ma200 = df.Close.rolling(window=200).mean()
ma200
```

Out[ ]:
```
0             NaN
1             NaN
2             NaN
3             NaN
4             NaN
           ...
2512    54.261513
2513    54.396763
2514    54.529350
2515    54.661100
2516    54.793137
Name: Close, Length: 2517, dtype: float64
```

In [ ]:
```
plt.figure(figsize=(12, 6))
plt.plot(df['Close'], label='Close')
plt.plot(ma100, 'r')
plt.plot(ma200, 'g')
```

Out[ ]:
```
[<matplotlib.lines.Line2D at 0x17dc5f5edd0>]
```



In [ ]:
```
df.shape
```

Out[ ]:
```
(2517, 5)
```

In [ ]:
```
# Splitting the data into training and testing sets

dataTrain = pd.DataFrame(df['Close'][0:int(df.shape[0]*0.7)])
dataTest = pd.DataFrame(df['Close'][int(df.shape[0]*0.7):int(df.shape[0])])

print(dataTrain.shape)
print(dataTest.shape)
```

```
(1761, 1)
(756, 1)
```

In [ ]:
```
dataTrain.head()
```

Out[ ]:

	Close
0	7.526071
1	7.643214
2	7.656429
3	7.534643
4	7.520714

In [ ]:
```
dataTest.head()
```

Out[ ]:

	Close
1761	29.182501
1762	28.955000
1763	29.037500
1764	29.004999
1765	29.152500

In [ ]:
```
from sklearn.preprocessing import MinMaxScaler
scaler = MinMaxScaler(feature_range=(0, 1))
```

In [ ]:
```
dataTrainArray = scaler.fit_transform(dataTrain)
dataTrainArray
```

Out[ ]:
```
array([[0.02527908],
       [0.02971782],
       [0.03021854],
       ...,
       [0.84388656],
       [0.85089656],
       [0.84616011]])
```

In [ ]:
```
dataTestArray = scaler.fit_transform(dataTest)
dataTestArray
```

Out[ ]:
```
array([[0.00511727],
       [0.        ],
       [0.00185572],
       [0.00112465],
       [0.00444245],
       [0.01175282],
       [0.017826  ],
       [0.01850082],
       [0.02209976],
       [0.01928809],
       [0.01810719],
       [0.0235056 ],
       [0.02344935],
       [0.02226845],
       [0.0235056 ],
       [0.02395548],
       [0.0233369 ],
       [0.03407748],
       [0.03441491],
       [0.03447111],
       [0.03267164],
       [0.03109711],
       [0.0727099 ],
       [0.07147275],
       [0.07456561],
       [0.08136981],
       [0.0883428 ],
       [0.09121067],
       [0.09334757],
       [0.09166055],
       [0.09823986],
       [0.10796831],
       [0.1107237 ],
       [0.10982403],
       [0.11190464],
       [0.1174155 ],
       [0.11972109],
       [0.11645954],
       [0.1171906 ],
       [0.11870884],
       [0.11904632],
       [0.13479162],
       [0.13012432],
       [0.13473542],
       [0.13226113],
       [0.13327338],
       [0.13034922],
       [0.1285497 ],
       [0.13113648],
       [0.13147387],
       [0.13029301],
       [0.13855934],
       [0.13985269],
       [0.13591636],
       [0.14418269],
       [0.13507281],
       [0.14395771],
       [0.14114603],
       [0.1395715 ],
       [0.14092114],
       [0.15734131],
       [0.15914073],
       [0.15807228],
       [0.15655404],
       [0.15677894],
       [0.16279596],
       [0.15857845],
       [0.15655404],
       [0.15475453],
```

```
[0.15379857],
[0.14513865],
[0.14609461],
[0.1418771 ],
[0.1462633 ],
[0.14272057],
[0.1397964 ],
[0.14969355],
[0.14873759],
[0.15644155],
[0.16144633],
[0.15666644],
[0.15728502],
[0.15649775],
[0.17297421],
[0.17820388],
[0.17567339],
[0.17269302],
[0.1863578 ],
[0.2091323 ],
[0.21464324],
[0.21053813],
[0.21441826],
[0.22650851],
[0.22425912],
[0.22296577],
[0.19361188],
[0.20648931],
[0.20941348],
[0.21464324],
[0.21357479],
[0.21098801],
[0.21396838],
[0.21250634],
[0.21284373],
[0.20772646],
[0.21008825],
[0.22285328],
[0.21430577],
[0.21722993],
[0.2224034 ],
[0.22026659],
[0.18647021],
[0.1664511 ],
[0.17303041],
[0.16498906],
[0.16009669],
[0.14873759],
[0.17162457],
[0.16414551],
[0.16898159],
[0.16763204],
[0.17128718],
[0.16870049],
[0.15694763],
[0.1687567 ],
[0.15666644],
[0.15857845],
[0.15565428],
[0.15897204],
[0.15132428],
[0.15947812],
[0.16442669],
[0.16706967],
[0.16825061],
[0.179666  ],
[0.1868076 ],
[0.18973176],
[0.19265593],
```

[0.19794189],
[0.19411797],
[0.19372438],
[0.20395882],
[0.20761405],
[0.21166287],
[0.19535511],
[0.18939437],
[0.18506437],
[0.19248723],
[0.23235675],
[0.22352814],
[0.22813924],
[0.24174773],
[0.24888941],
[0.25440027],
[0.2221223],
[0.23426867],
[0.24759606],
[0.25743692],
[0.25378169],
[0.23640557],
[0.23438116],
[0.23275043],
[0.24720239],
[0.24832704],
[0.24433451],
[0.24765227],
[0.25670586],
[0.26480349],
[0.26727778],
[0.27093292],
[0.27121411],
[0.26013611],
[0.25918015],
[0.25552492],
[0.24073557],
[0.25687455],
[0.25327561],
[0.24647132],
[0.23876736],
[0.24776476],
[0.24096047],
[0.24129785],
[0.22633981],
[0.21126919],
[0.20283417],
[0.19529891],
[0.20986336],
[0.21599279],
[0.21065062],
[0.21537422],
[0.21363099],
[0.21739862],
[0.21177528],
[0.22251589],
[0.22200981],
[0.22504638],
[0.22538377],
[0.229039],
[0.22594614],
[0.23151328],
[0.24776476],
[0.25108251],
[0.2470899],
[0.22583365],
[0.22735197],
[0.2269021],
[0.23213185],

```
[0.22825173],
[0.23387508],
[0.26559076],
[0.28622844],
[0.29927456],
[0.2871844 ],
[0.29404489],
[0.31873138],
[0.32857224],
[0.33172131],
[0.33976274],
[0.33773833],
[0.33093404],
[0.32699771],
[0.31220828],
[0.29949954],
[0.31085873],
[0.30551648],
[0.30456052],
[0.32233032],
[0.33256486],
[0.33262106],
[0.32767249],
[0.32193673],
[0.30174885],
[0.31507624],
[0.31057755],
[0.30354836],
[0.30264861],
[0.29910587],
[0.30084918],
[0.30113028],
[0.31968734],
[0.31423269],
[0.31743804],
[0.31715685],
[0.32699771],
[0.3407749 ],
[0.33020298],
[0.32913461],
[0.33284596],
[0.33284596],
[0.30787836],
[0.30804706],
[0.31074624],
[0.30034301],
[0.31738175],
[0.31721306],
[0.32171175],
[0.33278976],
[0.32913461],
[0.32902212],
[0.32879714],
[0.33436429],
[0.34454253],
[0.33948155],
[0.35584552],
[0.35674519],
[0.35224658],
[0.34403645],
[0.34426135],
[0.32840355],
[0.31091493],
[0.31316424],
[0.29320142],
[0.28763428],
[0.29022097],
[0.29218917],
[0.2512512 ],
```

```
[0.22870161],
[0.26547827],
[0.24585275],
[0.22116626],
[0.22825173],
[0.26367884],
[0.27284484],
[0.28988358],
[0.32148686],
[0.31833771],
[0.31507624],
[0.31069004],
[0.31873138],
[0.33560143],
[0.35511446],
[0.3518529 ],
[0.35033457],
[0.33278976],
[0.33959404],
[0.34302429],
[0.34218074],
[0.33295845],
[0.34369907],
[0.36079401],
[0.37057866],
[0.3607378 ],
[0.35213409],
[0.35331494],
[0.34977229],
[0.33447678],
[0.33413939],
[0.31181469],
[0.2982062 ],
[0.27621889],
[0.32024971],
[0.29533824],
[0.2848788 ],
[0.29218917],
[0.28600346],
[0.29561942],
[0.31372661],
[0.32041841],
[0.29556322],
[0.3049542 ],
[0.3229489 ],
[0.318394  ],
[0.32795367],
[0.33127143],
[0.33740094],
[0.35100944],
[0.34876004],
[0.32041841],
[0.28060509],
[0.27790591],
[0.26497219],
[0.26896472],
[0.27217007],
[0.26148574],
[0.27801831],
[0.29961203],
[0.34161845],
[0.34341788],
[0.38244393],
[0.389923  ],
[0.39492777],
[0.40229434],
[0.41736488],
[0.40921104],
[0.40673675],
```

```
[0.39712087],
[0.40690545],
[0.40021373],
[0.39638981],
[0.40381267],
[0.40116969],
[0.40791769],
[0.40673675],
[0.40915484],
[0.40533091],
[0.40308161],
[0.39953887],
[0.41848961],
[0.42743072],
[0.43575325],
[0.43952088],
[0.4365968 ],
[0.42669966],
[0.42405667],
[0.42996121],
[0.42107631],
[0.42163868],
[0.41061688],
[0.41005459],
[0.39290336],
[0.39745826],
[0.39161002],
[0.38857336],
[0.37310915],
[0.38581789],
[0.38429965],
[0.39183491],
[0.38964181],
[0.4012821 ],
[0.38295001],
[0.39127254],
[0.40572459],
[0.42040153],
[0.41910819],
[0.40521851],
[0.42293202],
[0.42461904],
[0.42225725],
[0.42529382],
[0.41938929],
[0.4277119 ],
[0.42523762],
[0.42619357],
[0.43401003],
[0.44424456],
[0.44081431],
[0.42265084],
[0.4166339 ],
[0.41877071],
[0.48180849],
[0.51493   ],
[0.51830405],
[0.52437727],
[0.51335547],
[0.51414274],
[0.52330882],
[0.51571727],
[0.52325254],
[0.52820111],
[0.53095658],
[0.5482765 ],
[0.57223194],
[0.56031047],
[0.55794858],
```

[0.55800487],
[0.56047916],
[0.56424679],
[0.57425635],
[0.58415341],
[0.60259798],
[0.61412587],
[0.62874661],
[0.63285163],
[0.62447281],
[0.60327284],
[0.5931508],
[0.57644945],
[0.60749036],
[0.59185745],
[0.62188611],
[0.60743407],
[0.57391896],
[0.57594337],
[0.57667434],
[0.58600912],
[0.57268182],
[0.59028284],
[0.59815558],
[0.58820223],
[0.61367599],
[0.61811848],
[0.62666591],
[0.6380251],
[0.65371429],
[0.63077101],
[0.60996456],
[0.60704048],
[0.62447281],
[0.56537145],
[0.55463083],
[0.5977057],
[0.57099479],
[0.59793059],
[0.59253223],
[0.56345953],
[0.58196031],
[0.58949557],
[0.60119215],
[0.55822977],
[0.58471578],
[0.56503406],
[0.54220328],
[0.54816401],
[0.57942982],
[0.59832427],
[0.51543609],
[0.48231457],
[0.49457351],
[0.52932576],
[0.52111572],
[0.49850984],
[0.44058933],
[0.42968002],
[0.39914528],
[0.42506892],
[0.43699039],
[0.39385932],
[0.34392396],
[0.34279931],
[0.31755044],
[0.33065286],
[0.32851604],
[0.36619246],

[0.35837601],
[0.35292135],
[0.38801108],
[0.34229323],
[0.33121522],
[0.29618179],
[0.30242371],
[0.29696905],
[0.29961203],
[0.31001518],
[0.27925546],
[0.27059554],
[0.2825733],
[0.25344431],
[0.23061353],
[0.19631107],
[0.17438004],
[0.23252544],
[0.22678961],
[0.22723948],
[0.23573079],
[0.23674296],
[0.14828771],
[0.18242139],
[0.18056568],
[0.19642356],
[0.21081932],
[0.21357479],
[0.20508348],
[0.19220605],
[0.20946977],
[0.2199854],
[0.22515887],
[0.23055732],
[0.21076311],
[0.21424956],
[0.20738907],
[0.2358432],
[0.22763316],
[0.21852328],
[0.27796211],
[0.28465391],
[0.28510379],
[0.3117022],
[0.32817857],
[0.32851604],
[0.30995898],
[0.30697861],
[0.30146766],
[0.30967779],
[0.30568518],
[0.30917171],
[0.30703481],
[0.30990269],
[0.3160884],
[0.31063375],
[0.32137437],
[0.32845975],
[0.32902212],
[0.33205869],
[0.32238653],
[0.33262106],
[0.33756964],
[0.33577012],
[0.33009057],
[0.31873138],
[0.32103698],
[0.35472078],
[0.36602376],

[0.37052246],
[0.38188156],
[0.39532136],
[0.40600577],
[0.39762695],
[0.40679304],
[0.4457628],
[0.42304451],
[0.41005459],
[0.39908899],
[0.40853626],
[0.4099421],
[0.4168588],
[0.42411296],
[0.43974586],
[0.44722493],
[0.44913685],
[0.45650342],
[0.47393583],
[0.47056179],
[0.47685991],
[0.46746893],
[0.46701905],
[0.46904346],
[0.46915595],
[0.49097457],
[0.49507959],
[0.49884722],
[0.51543609],
[0.51363666],
[0.50306474],
[0.49755388],
[0.4992971],
[0.4771411],
[0.53253111],
[0.52482707],
[0.53944781],
[0.52105943],
[0.48945624],
[0.48968114],
[0.47742228],
[0.45751558],
[0.39307206],
[0.40960471],
[0.42231345],
[0.41758986],
[0.41151663],
[0.37828262],
[0.39802063],
[0.3765394],
[0.35899458],
[0.35511446],
[0.35095315],
[0.34617335],
[0.35134682],
[0.33318343],
[0.32323008],
[0.35888209],
[0.37518976],
[0.39026038],
[0.41798345],
[0.43164823],
[0.44418827],
[0.44070182],
[0.44047684],
[0.43254799],
[0.4390148],
[0.46465726],
[0.4613957],

[0.47033689],
[0.46651297],
[0.46538832],
[0.44846207],
[0.47224881],
[0.47191142],
[0.46167689],
[0.48208967],
[0.48872518],
[0.49817245],
[0.4971602],
[0.47348595],
[0.48034645],
[0.49153685],
[0.48321432],
[0.49193053],
[0.50267115],
[0.49867853],
[0.49221171],
[0.50520164],
[0.48793792],
[0.51397405],
[0.52308384],
[0.52212788],
[0.51284939],
[0.51689821],
[0.52780743],
[0.52274646],
[0.54670189],
[0.52077825],
[0.49597935],
[0.43592194],
[0.45650342],
[0.46797501],
[0.49266151],
[0.47894061],
[0.47607265],
[0.52381491],
[0.48883767],
[0.48315812],
[0.50992523],
[0.53157515],
[0.53163136],
[0.54445258],
[0.54344042],
[0.4882191],
[0.50986903],
[0.49676661],
[0.50447057],
[0.5240398],
[0.52252156],
[0.50542653],
[0.52505205],
[0.54805152],
[0.54793903],
[0.5530563],
[0.56728336],
[0.60602823],
[0.60321656],
[0.57881124],
[0.58527806],
[0.58977676],
[0.60141713],
[0.59123888],
[0.57307541],
[0.57864255],
[0.57279422],
[0.59163247],
[0.58522186],

```
[0.57920492],
[0.60816513],
[0.61165158],
[0.57999219],
[0.59045162],
[0.62526007],
[0.62554126],
[0.61058313],
[0.62537256],
[0.64257999],
[0.67699494],
[0.67508294],
[0.67199017],
[0.66664792],
[0.67176519],
[0.6781196 ],
[0.70117527],
[0.6980825 ],
[0.7161896 ],
[0.71843899],
[0.73530904],
[0.74919872],
[0.71680817],
[0.71663948],
[0.7475679 ],
[0.78726881],
[0.79671599],
[0.79463538],
[0.79525387],
[0.80756901],
[0.81156171],
[0.8231458 ],
[0.82179608],
[0.83591074],
[0.82562009],
[0.8431649 ],
[0.85070017],
[0.84614527],
[0.82871286],
[0.82207735],
[0.82078392],
[0.84659506],
[0.83489858],
[0.85486139],
[0.85154364],
[0.83416751],
[0.80768159],
[0.82055893],
[0.84215257],
[0.87100038],
[0.84968801],
[0.85846042],
[0.87133776],
[0.87521789],
[0.89596806],
[0.92245398],
[0.92554692],
[0.9217792 ],
[0.92335373],
[0.92009227],
[0.94573472],
[0.94725297],
[0.97896873],
[0.97835007],
[0.98802224],
[1.        ]])
```

In [ ]:
```
xTrain = []
yTrain = []
```

```
for i in range(100, dataTrainArray.shape[0]):
    xTrain.append(dataTrainArray[i-100:i])
    yTrain.append(dataTrainArray[i, 0])

xTrain, yTrain = np.array(xTrain), np.array(yTrain)
```

In [ ]:
```
xTrain.shape
```

Out[ ]:
```
(1661, 100, 1)
```

In [ ]:
```
# ML Model
```

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

In [ ]:
```
model = Sequential()
model.add(LSTM(units=50, activation='relu', return_sequences=True,
               input_shape=(xTrain.shape[1], 1)))
model.add(Dropout(0.2))

model.add(LSTM(units=60, activation='relu', return_sequences=True))
model.add(Dropout(0.3))

model.add(LSTM(units=80, activation='relu', return_sequences=True))
model.add(Dropout(0.3))

model.add(LSTM(units=120, activation='relu'))
model.add(Dropout(0.5))

model.add(Dense(units=1))
```

In [ ]:
```
model.summary()
```

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 lstm (LSTM)                 (None, 100, 50)           10400

 dropout (Dropout)           (None, 100, 50)           0

 lstm_1 (LSTM)               (None, 100, 60)           26640

 dropout_1 (Dropout)         (None, 100, 60)           0

 lstm_2 (LSTM)               (None, 100, 80)           45120

 dropout_2 (Dropout)         (None, 100, 80)           0

 lstm_3 (LSTM)               (None, 120)               96480

 dropout_3 (Dropout)         (None, 120)               0

 dense (Dense)               (None, 1)                 121

=================================================================
Total params: 178,761
Trainable params: 178,761
Non-trainable params: 0
_________________________________________________________________
```

In [ ]:
```
model.compile(optimizer='adam', loss='mean_squared_error')
model.fit(xTrain, yTrain, epochs=50)
```

```
Epoch 1/50
52/52 [==============================] - 18s 248ms/step - loss: 0.0591
Epoch 2/50
52/52 [==============================] - 14s 263ms/step - loss: 0.0113
Epoch 3/50
52/52 [==============================] - 13s 246ms/step - loss: 0.0097
Epoch 4/50
52/52 [==============================] - 13s 248ms/step - loss: 0.0095
Epoch 5/50
52/52 [==============================] - 12s 230ms/step - loss: 0.0091
Epoch 6/50
52/52 [==============================] - 12s 229ms/step - loss: 0.0083
Epoch 7/50
52/52 [==============================] - 12s 229ms/step - loss: 0.0077
Epoch 8/50
52/52 [==============================] - 12s 227ms/step - loss: 0.0074
Epoch 9/50
52/52 [==============================] - 12s 228ms/step - loss: 0.0081
Epoch 10/50
52/52 [==============================] - 12s 229ms/step - loss: 0.0072
Epoch 11/50
52/52 [==============================] - 12s 228ms/step - loss: 0.0072
Epoch 12/50
52/52 [==============================] - 12s 240ms/step - loss: 0.0070
Epoch 13/50
52/52 [==============================] - 12s 232ms/step - loss: 0.0062
Epoch 14/50
52/52 [==============================] - 11s 215ms/step - loss: 0.0064
Epoch 15/50
52/52 [==============================] - 12s 225ms/step - loss: 0.0061
Epoch 16/50
52/52 [==============================] - 12s 224ms/step - loss: 0.0064
Epoch 17/50
52/52 [==============================] - 12s 226ms/step - loss: 0.0063
Epoch 18/50
52/52 [==============================] - 11s 216ms/step - loss: 0.0054
Epoch 19/50
52/52 [==============================] - 12s 230ms/step - loss: 0.0055
Epoch 20/50
52/52 [==============================] - 12s 223ms/step - loss: 0.0048
Epoch 21/50
52/52 [==============================] - 11s 219ms/step - loss: 0.0047
Epoch 22/50
52/52 [==============================] - 12s 221ms/step - loss: 0.0054
Epoch 23/50
52/52 [==============================] - 12s 221ms/step - loss: 0.0048
Epoch 24/50
52/52 [==============================] - 11s 219ms/step - loss: 0.0046
Epoch 25/50
52/52 [==============================] - 12s 229ms/step - loss: 0.0044
Epoch 26/50
52/52 [==============================] - 12s 225ms/step - loss: 0.0048
Epoch 27/50
52/52 [==============================] - 12s 225ms/step - loss: 0.0047
Epoch 28/50
52/52 [==============================] - 12s 225ms/step - loss: 0.0050
Epoch 29/50
52/52 [==============================] - 12s 227ms/step - loss: 0.0044
Epoch 30/50
52/52 [==============================] - 12s 225ms/step - loss: 0.0042
Epoch 31/50
52/52 [==============================] - 12s 226ms/step - loss: 0.0042
Epoch 32/50
52/52 [==============================] - 12s 226ms/step - loss: 0.0039
Epoch 33/50
52/52 [==============================] - 12s 225ms/step - loss: 0.0041
Epoch 34/50
52/52 [==============================] - 12s 225ms/step - loss: 0.0037
Epoch 35/50
```

```
52/52 [==============================] - 12s 225ms/step - loss: 0.0035
Epoch 36/50
52/52 [==============================] - 12s 227ms/step - loss: 0.0037
Epoch 37/50
52/52 [==============================] - 12s 226ms/step - loss: 0.0035
Epoch 38/50
52/52 [==============================] - 12s 226ms/step - loss: 0.0034
Epoch 39/50
52/52 [==============================] - 12s 225ms/step - loss: 0.0034
Epoch 40/50
52/52 [==============================] - 12s 226ms/step - loss: 0.0034
Epoch 41/50
52/52 [==============================] - 12s 226ms/step - loss: 0.0031
Epoch 42/50
52/52 [==============================] - 12s 225ms/step - loss: 0.0033
Epoch 43/50
52/52 [==============================] - 12s 225ms/step - loss: 0.0032
Epoch 44/50
52/52 [==============================] - 12s 227ms/step - loss: 0.0035
Epoch 45/50
52/52 [==============================] - 12s 228ms/step - loss: 0.0032
Epoch 46/50
52/52 [==============================] - 12s 227ms/step - loss: 0.0034
Epoch 47/50
52/52 [==============================] - 12s 224ms/step - loss: 0.0029
Epoch 48/50
52/52 [==============================] - 12s 229ms/step - loss: 0.0029
Epoch 49/50
52/52 [==============================] - 12s 225ms/step - loss: 0.0027
Epoch 50/50
52/52 [==============================] - 12s 232ms/step - loss: 0.0029
```

Out[ ]:
```
<keras.callbacks.History at 0x17defb23af0>
```

In [ ]:
```
model.save('keras_model.h5')
```

In [ ]:
```
past_100 = dataTrain.tail(100)
```

In [ ]:
```
finalDF = past_100.append(dataTest, ignore_index=True)
```

```
C:\Users\gurup\AppData\Local\Temp\ipykernel_22420\1581833897.py:1: FutureWarning: The frame.a
ppend method is deprecated and will be removed from pandas in a future version. Use pandas.co
ncat instead.
  finalDF = past_100.append(dataTest, ignore_index=True)
```

In [ ]:
```
finalDF.head()
```

Out[ ]:

	Close
0	27.092501
1	27.202499
2	27.000000
3	26.982500
4	27.045000

In [ ]:
```
inputData = scaler.fit_transform(finalDF)
inputData
```

Out[ ]:
```
array([[0.0275037 ],
       [0.02981315],
       [0.02556164],
       [0.02519422],
       [0.02650642],
       [0.03332987],
       [0.03280496],
       [0.03196517],
       [0.03123034],
       [0.0327    ],
       [0.02823853],
       [0.0300231 ],
       [0.02571909],
       [0.02330465],
       [0.01999793],
       [0.01936806],
       [0.01506405],
       [0.01558892],
       [0.01889569],
       [0.02414448],
       [0.02398698],
       [0.0274512 ],
       [0.01254461],
       [0.        ],
       [0.01212474],
       [0.02529918],
       [0.04534957],
       [0.065295  ],
       [0.06188327],
       [0.05484991],
       [0.05479741],
       [0.05469245],
       [0.06030866],
       [0.05028345],
       [0.05117573],
       [0.05227797],
       [0.05679194],
       [0.04750159],
       [0.05206806],
       [0.04928616],
       [0.0518056 ],
       [0.05206806],
       [0.05647702],
       [0.05736931],
       [0.06781443],
       [0.06912663],
       [0.07458534],
       [0.07269581],
       [0.0761075 ],
       [0.07568762],
       [0.07526771],
       [0.07343064],
       [0.07311569],
       [0.07070125],
       [0.07621249],
       [0.07936176],
       [0.06539995],
       [0.05957383],
       [0.05558474],
       [0.05463995],
       [0.04387991],
       [0.04440478],
       [0.03516694],
       [0.0299706 ],
       [0.03821125],
       [0.04162293],
       [0.04067814],
       [0.02445939],
       [0.02781862],
```

```
[0.0135419 ],
[0.02089021],
[0.03600672],
[0.03579677],
[0.03637414],
[0.04513965],
[0.04550706],
[0.04251525],
[0.04545457],
[0.04429982],
[0.04372245],
[0.03878858],
[0.03338233],
[0.03553435],
[0.0313878 ],
[0.03579677],
[0.04146547],
[0.04718668],
[0.05679194],
[0.05338025],
[0.06330047],
[0.06330047],
[0.06660719],
[0.06739452],
[0.0709112 ],
[0.07253832],
[0.07311569],
[0.06907413],
[0.07028133],
[0.07416547],
[0.07154107],
[0.07138362],
[0.06660719],
[0.06833931],
[0.06765694],
[0.07075375],
[0.07757719],
[0.08324586],
[0.08387573],
[0.08723495],
[0.08461056],
[0.08350831],
[0.08854715],
[0.08849465],
[0.08739241],
[0.08854715],
[0.08896706],
[0.08838969],
[0.09841487],
[0.09872982],
[0.09878228],
[0.09710267],
[0.09563301],
[0.13447409],
[0.13331935],
[0.1362062 ],
[0.14255719],
[0.14906573],
[0.15174258],
[0.15373715],
[0.15216249],
[0.15830357],
[0.16738404],
[0.1699559 ],
[0.16911615],
[0.17105818],
[0.17620198],
[0.178354  ],
[0.17530969],
```

```
[0.17599206],
[0.17740917],
[0.17772417],
[0.19242073],
[0.1880643 ],
[0.19236827],
[0.19005879],
[0.19100361],
[0.18827421],
[0.18659456],
[0.18900904],
[0.18932396],
[0.18822176],
[0.19593749],
[0.19714469],
[0.19347055],
[0.20118628],
[0.19268318],
[0.20097628],
[0.19835189],
[0.19688223],
[0.19814197],
[0.21346844],
[0.21514801],
[0.21415073],
[0.21273361],
[0.21294353],
[0.21855977],
[0.21462318],
[0.21273361],
[0.21105396],
[0.21016167],
[0.20207857],
[0.20297085],
[0.19903426],
[0.20312831],
[0.19982154],
[0.19709215],
[0.20633007],
[0.20543779],
[0.21262861],
[0.21730003],
[0.21283853],
[0.2134159 ],
[0.21268107],
[0.22806007],
[0.23294141],
[0.23057947],
[0.22779762],
[0.24055223],
[0.26180977],
[0.26695365],
[0.26312197],
[0.26674365],
[0.2780286 ],
[0.27592904],
[0.27472184],
[0.24732313],
[0.25934283],
[0.26207223],
[0.26695365],
[0.26595637],
[0.26354189],
[0.26632374],
[0.26495908],
[0.265274  ],
[0.26049758],
[0.26270206],
[0.27461684],
```

[0.26663865],
[0.26936805],
[0.27419693],
[0.27220244],
[0.24065715],
[0.22197146],
[0.22811253],
[0.2206068],
[0.21604029],
[0.20543779],
[0.22680034],
[0.21981943],
[0.2243334],
[0.22307374],
[0.22648542],
[0.22407102],
[0.21310098],
[0.22412348],
[0.21283853],
[0.21462318],
[0.21189378],
[0.21499055],
[0.20785219],
[0.21546292],
[0.22008189],
[0.22254883],
[0.22365111],
[0.23430615],
[0.24097206],
[0.24370145],
[0.24643085],
[0.25136473],
[0.24779551],
[0.24742813],
[0.25698089],
[0.26039266],
[0.26417179],
[0.24895025],
[0.24338654],
[0.23934495],
[0.24627339],
[0.28348731],
[0.27524675],
[0.27955072],
[0.29225279],
[0.29891878],
[0.30406257],
[0.27393455],
[0.28527188],
[0.29771158],
[0.30689697],
[0.3034852],
[0.28726645],
[0.28537688],
[0.28385477],
[0.29734412],
[0.29839387],
[0.29466727],
[0.29776404],
[0.3062146],
[0.31377287],
[0.31608235],
[0.31949404],
[0.31975649],
[0.30941636],
[0.30852408],
[0.30511232],
[0.29130804],
[0.30637205],

```
[0.30301283],
[0.29666176],
[0.28947093],
[0.29786904],
[0.29151796],
[0.29183288],
[0.27787115],
[0.26380434],
[0.25593115],
[0.24889779],
[0.26249214],
[0.26821331],
[0.26322697],
[0.26763594],
[0.26600882],
[0.26952551],
[0.26427671],
[0.27430193],
[0.27382955],
[0.27666387],
[0.27697878],
[0.28039055],
[0.27750369],
[0.28270003],
[0.29786904],
[0.3009658 ],
[0.29723913],
[0.2773987 ],
[0.27881589],
[0.27839598],
[0.2832774 ],
[0.27965572],
[0.28490451],
[0.3145077 ],
[0.33377076],
[0.34594792],
[0.33466304],
[0.34106658],
[0.36410878],
[0.37329416],
[0.37623347],
[0.38373929],
[0.38184972],
[0.37549865],
[0.37182451],
[0.35802016],
[0.34615791],
[0.3567605 ],
[0.35177408],
[0.3508818 ],
[0.367468  ],
[0.37702084],
[0.3770733 ],
[0.37245434],
[0.36710063],
[0.3482574 ],
[0.36069709],
[0.35649804],
[0.34993705],
[0.34909723],
[0.34579046],
[0.34741765],
[0.34768003],
[0.36500106],
[0.35990973],
[0.36290158],
[0.36263912],
[0.37182451],
[0.38468403],
```

[0.37481628],
[0.37381907],
[0.37728322],
[0.37728322],
[0.35397864],
[0.3541361],
[0.3566555],
[0.3469452],
[0.36284904],
[0.36269158],
[0.36689063],
[0.37723076],
[0.37381907],
[0.37371408],
[0.37350408],
[0.37870041],
[0.38820072],
[0.38347683],
[0.39875084],
[0.39959059],
[0.39539162],
[0.38772834],
[0.38793826],
[0.37313671],
[0.35681296],
[0.35891244],
[0.34027929],
[0.33508296],
[0.33749736],
[0.33933447],
[0.30112326],
[0.28007563],
[0.3144027],
[0.29608438],
[0.27304219],
[0.27965572],
[0.31272313],
[0.32127861],
[0.33718244],
[0.36668071],
[0.36374132],
[0.36069709],
[0.35660304],
[0.36410878],
[0.37985515],
[0.39806847],
[0.39502416],
[0.39360697],
[0.37723076],
[0.38358183],
[0.3867836],
[0.38599623],
[0.37738822],
[0.38741343],
[0.40336972],
[0.41250265],
[0.40331726],
[0.39528662],
[0.39638882],
[0.39308213],
[0.37880541],
[0.3784905],
[0.35765278],
[0.34495071],
[0.32442791],
[0.36552597],
[0.34227378],
[0.33251102],
[0.33933447],

[0.33356076],
[0.34253623],
[0.35943735],
[0.36568343],
[0.34248377],
[0.35124925],
[0.36804537],
[0.36379386],
[0.37271679],
[0.37581356],
[0.38153481],
[0.39423688],
[0.39213731],
[0.36568343],
[0.32852197],
[0.32600257],
[0.31393033],
[0.31765693],
[0.32064878],
[0.3106761],
[0.32610749],
[0.34626291],
[0.3854714],
[0.38715097],
[0.42357761],
[0.43055851],
[0.43522994],
[0.44210584],
[0.45617257],
[0.44856183],
[0.44625235],
[0.43727696],
[0.44640981],
[0.44016381],
[0.43659459],
[0.44352304],
[0.4410561],
[0.44735463],
[0.44625235],
[0.44850937],
[0.44494015],
[0.44284067],
[0.4395339],
[0.45722239],
[0.46556795],
[0.47333614],
[0.47685282],
[0.47412351],
[0.46488558],
[0.46241864],
[0.46792989],
[0.45963679],
[0.4601617],
[0.44987403],
[0.4493492],
[0.43334037],
[0.43759188],
[0.43213317],
[0.42929877],
[0.41486459],
[0.42672684],
[0.42530972],
[0.43234308],
[0.43029606],
[0.44116102],
[0.42404998],
[0.43181817],
[0.44530761],
[0.45900696],

[0.45779976],
[0.44483524],
[0.4613689],
[0.46294356],
[0.46073907],
[0.46357338],
[0.45806214],
[0.46583041],
[0.46352093],
[0.46441321],
[0.47170903],
[0.48126187],
[0.4780601],
[0.46110645],
[0.45549028],
[0.45748477],
[0.51632377],
[0.54723916],
[0.55038846],
[0.55605717],
[0.5457695],
[0.54650433],
[0.55505989],
[0.54797398],
[0.55500735],
[0.55962631],
[0.56219825],
[0.57836454],
[0.60072437],
[0.58959695],
[0.58739239],
[0.58744493],
[0.58975441],
[0.59327109],
[0.60261394],
[0.61185178],
[0.62906782],
[0.63982786],
[0.65347475],
[0.65730635],
[0.64948562],
[0.62969773],
[0.62024988],
[0.60466096],
[0.63363432],
[0.61904268],
[0.64707122],
[0.63358178],
[0.60229902],
[0.60418859],
[0.60487088],
[0.61358389],
[0.60114428],
[0.61757295],
[0.62492131],
[0.61563092],
[0.63940795],
[0.64355454],
[0.65153264],
[0.66213523],
[0.67677941],
[0.65536432],
[0.63594372],
[0.63321441],
[0.64948562],
[0.59432083],
[0.58429562],
[0.62450139],
[0.59956963],

[0.62471131],
[0.61967251],
[0.59253626],
[0.60980476],
[0.61683812],
[0.62775562],
[0.58765484],
[0.61237669],
[0.59400592],
[0.57269583],
[0.57825954],
[0.60744282],
[0.62507876],
[0.54771153],
[0.51679614],
[0.52823855],
[0.56067605],
[0.55301286],
[0.53191269],
[0.47785011],
[0.46766744],
[0.43916653],
[0.46336347],
[0.47449088],
[0.43423265],
[0.38762335],
[0.3865736],
[0.36300649],
[0.37523619],
[0.3732417],
[0.4084086],
[0.40111278],
[0.39602145],
[0.42877394],
[0.38610123],
[0.3757611],
[0.34306115],
[0.34888731],
[0.34379597],
[0.34626291],
[0.35597313],
[0.32726223],
[0.31917912],
[0.33035908],
[0.30317029],
[0.2818602],
[0.24984253],
[0.22937227],
[0.28364477],
[0.27829098],
[0.27871089],
[0.28663662],
[0.28758137],
[0.20501788],
[0.23687801],
[0.2351459],
[0.24994753],
[0.26338443],
[0.26595637],
[0.25803064],
[0.24601094],
[0.26212477],
[0.27193999],
[0.27676887],
[0.28180774],
[0.26333197],
[0.26658619],
[0.26018266],
[0.28674154],

```
[0.27907835],
[0.27057525],
[0.32605503],
[0.3323011 ],
[0.33272102],
[0.35754778],
[0.37292671],
[0.3732417 ],
[0.35592067],
[0.35313882],
[0.34799494],
[0.35565822],
[0.35193154],
[0.35518584],
[0.35319128],
[0.35586813],
[0.36164184],
[0.3565505 ],
[0.36657571],
[0.37318917],
[0.37371408],
[0.37654839],
[0.36752046],
[0.3770733 ],
[0.38169226],
[0.38001261],
[0.37471136],
[0.36410878],
[0.3662608 ],
[0.39770102],
[0.40825114],
[0.41245019],
[0.4230527 ],
[0.43559731],
[0.44557006],
[0.43774933],
[0.44630489],
[0.48267899],
[0.4614739 ],
[0.4493492 ],
[0.43911399],
[0.447932  ],
[0.4492442 ],
[0.4557002 ],
[0.46247118],
[0.47706282],
[0.48404372],
[0.4858283 ],
[0.4927042 ],
[0.50897549],
[0.50582618],
[0.51170481],
[0.50293933],
[0.50251942],
[0.50440899],
[0.50451399],
[0.52487933],
[0.52871092],
[0.53222761],
[0.54771153],
[0.54603196],
[0.5361642 ],
[0.53102041],
[0.53264752],
[0.51196726],
[0.5636679 ],
[0.556477  ],
[0.5701239 ],
[0.55296032],
```

[0.52346213],
[0.52367205],
[0.51222972],
[0.49364895],
[0.43349782],
[0.44892929],
[0.46079153],
[0.45638256],
[0.45071386],
[0.41969347],
[0.43811679],
[0.41806636],
[0.40169015],
[0.39806847],
[0.39418434],
[0.38972291],
[0.39455179],
[0.37759821],
[0.36830783],
[0.40158515],
[0.41680662],
[0.43087343],
[0.45674994],
[0.46950455],
[0.48120933],
[0.47795511],
[0.47774511],
[0.47034437],
[0.47638045],
[0.50031493],
[0.49727063],
[0.50561627],
[0.50204705],
[0.5009973],
[0.48519847],
[0.50740084],
[0.50708592],
[0.49753308],
[0.51658622],
[0.52277976],
[0.53159778],
[0.53065295],
[0.50855558],
[0.51495911],
[0.52540416],
[0.51763597],
[0.52577161],
[0.53579683],
[0.53207015],
[0.52603407],
[0.53815877],
[0.52204493],
[0.54634687],
[0.55484989],
[0.5539576],
[0.54529713],
[0.54907627],
[0.55925886],
[0.55453497],
[0.5768948],
[0.55269787],
[0.52955075],
[0.4734936],
[0.4927042],
[0.5034117],
[0.5264539],
[0.51364691],
[0.51096998],
[0.55553226],

```
[0.52288476],
[0.51758351],
[0.54256773],
[0.56277562],
[0.56282808],
[0.57479532],
[0.57385057],
[0.52230739],
[0.54251527],
[0.53028558],
[0.5374764 ],
[0.55574217],
[0.55432506],
[0.53836868],
[0.556687  ],
[0.57815454],
[0.57804954],
[0.58282597],
[0.5961054 ],
[0.63226958],
[0.62964519],
[0.60686545],
[0.61290153],
[0.61710058],
[0.62796562],
[0.61846531],
[0.60151165],
[0.60670799],
[0.6012492 ],
[0.61883269],
[0.61284907],
[0.6072329 ],
[0.63426415],
[0.63751838],
[0.60796773],
[0.61773049],
[0.65022045],
[0.6504829 ],
[0.63652109],
[0.65032544],
[0.66638674],
[0.6985094 ],
[0.69672475],
[0.69383798],
[0.68885157],
[0.69362799],
[0.69955915],
[0.72107915],
[0.71819238],
[0.73509342],
[0.73719298],
[0.75293936],
[0.76590389],
[0.73567079],
[0.73551333],
[0.76438169],
[0.80143824],
[0.81025617],
[0.80831414],
[0.80889143],
[0.8203863 ],
[0.82411306],
[0.83492556],
[0.83366574],
[0.84684026],
[0.83723504],
[0.85361125],
[0.86064461],
[0.8563931 ],
```

```
       [0.84012181],
       [0.83392828],
       [0.832721  ],
       [0.85681294],
       [0.84589552],
       [0.86452867],
       [0.8614319 ],
       [0.84521315],
       [0.82049138],
       [0.832511  ],
       [0.85266635],
       [0.87959268],
       [0.85969987],
       [0.86788797],
       [0.87990759],
       [0.88352927],
       [0.90289733],
       [0.9276191 ],
       [0.93050603],
       [0.92698927],
       [0.92845893],
       [0.9254147 ],
       [0.94934918],
       [0.9507663 ],
       [0.98036957],
       [0.97979212],
       [0.98882005],
       [1.        ]])
```

In [ ]:
```
inputData.shape
```

Out[ ]:
```
(856, 1)
```

In [ ]:
```
xTest = []
yTest = []

for i in range(100, inputData.shape[0]):
    xTest.append(inputData[i-100:i])
    yTest.append(inputData[i, 0])
```

In [ ]:
```
xTest, yTest = np.array(xTest), np.array(yTest)
```

In [ ]:
```
# Prediction
```

In [ ]:
```
yPred = model.predict(xTest)
```

In [ ]:
```
yPred.shape
```

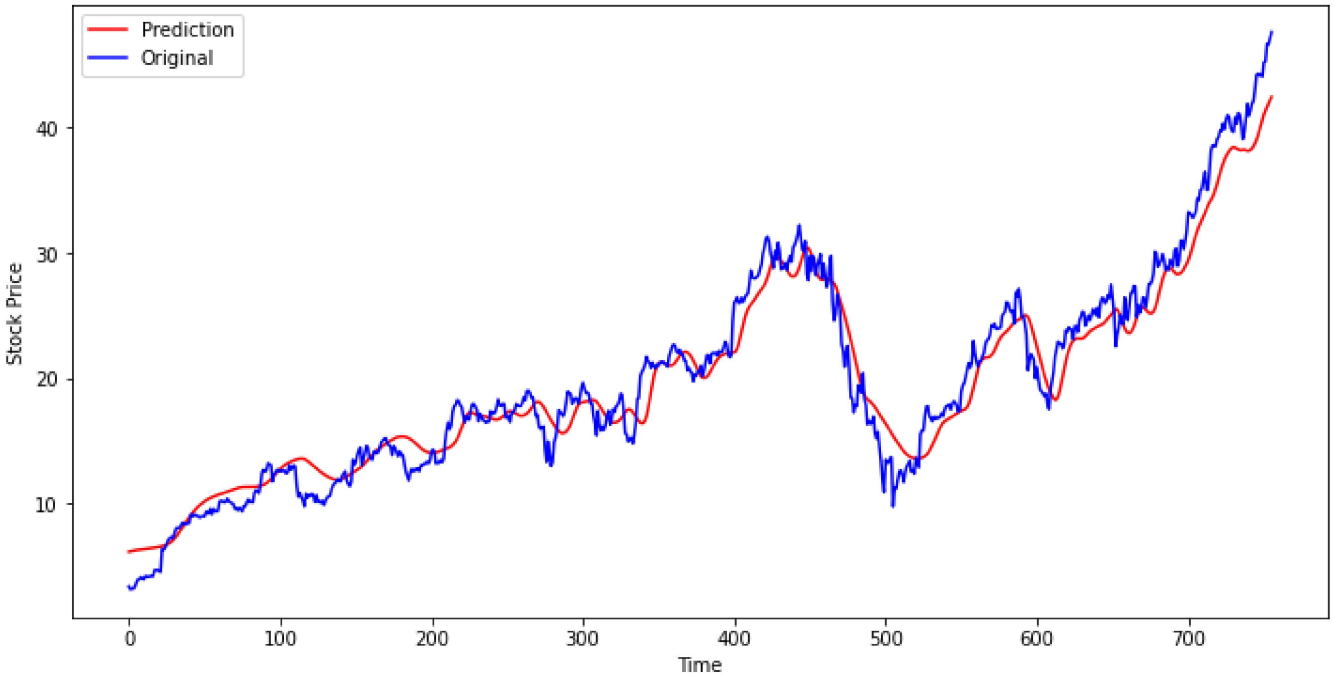
Out[ ]:
```
(756, 1)
```

In [ ]:
```
scaler.scale_
```

Out[ ]:
```
array([0.02099517])
```

In [ ]:
```
scaleFactor = 1/scaler.scale_
yPred = yPred*scaleFactor
yTest = yTest*scaleFactor
```

In [ ]:
```
plt.figure(figsize=(12, 6))
plt.plot(yPred, 'r', label='Prediction')
plt.plot(yTest, 'b', label='Original')
plt.xlabel('Time')
plt.ylabel('Stock Price')
plt.legend()
plt.show()
```


Prediction
Original
Stock Price
Time
10
20
30
40
0
100
200
300
400
500
600
700