In [28]:
```
import numpy as np
import pandas as pd


df = pd.read_csv("AAPL.csv")
df
```

Out[28]:

	Unnamed: 0	symbol	date	close	high	low	open	volume	adjClose	adjHigh	adjLow	adjOpen	adjVolume	divCash	splitFactor
0	0	AAPL	2015-05-27 00:00:00+00:00	132.045	132.260	130.0500	130.34	45833246	121.682558	121.880685	119.844118	120.111360	45833246	0.0	1.0
1	1	AAPL	2015-05-28 00:00:00+00:00	131.780	131.950	131.1000	131.86	30733309	121.438354	121.595013	120.811718	121.512076	30733309	0.0	1.0
2	2	AAPL	2015-05-29 00:00:00+00:00	130.280	131.450	129.9000	131.23	50884452	120.056069	121.134251	119.705890	120.931516	50884452	0.0	1.0
3	3	AAPL	2015-06-01 00:00:00+00:00	130.535	131.390	130.0500	131.20	32112797	120.291057	121.078960	119.844118	120.903870	32112797	0.0	1.0
4	4	AAPL	2015-06-02 00:00:00+00:00	129.960	130.655	129.3200	129.86	33667627	119.761181	120.401640	119.171406	119.669029	33667627	0.0	1.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
1253	1253	AAPL	2020-05-18 00:00:00+00:00	314.960	316.500	310.3241	313.17	33843125	314.960000	316.500000	310.324100	313.170000	33843125	0.0	1.0
1254	1254	AAPL	2020-05-19 00:00:00+00:00	313.140	318.520	313.0100	315.03	25432385	313.140000	318.520000	313.010000	315.030000	25432385	0.0	1.0
1255	1255	AAPL	2020-05-20 00:00:00+00:00	319.230	319.520	316.2000	316.68	27876215	319.230000	319.520000	316.200000	316.680000	27876215	0.0	1.0
1256	1256	AAPL	2020-05-21 00:00:00+00:00	316.850	320.890	315.8700	318.66	25672211	316.850000	320.890000	315.870000	318.660000	25672211	0.0	1.0
1257	1257	AAPL	2020-05-22 00:00:00+00:00	318.890	319.230	315.3500	315.77	20450754	318.890000	319.230000	315.350000	315.770000	20450754	0.0	1.0

1258 rows × 15 columns

In [3]:
```
df.head()
```

Out[3]:

	Unnamed: 0	symbol	date	close	high	low	open	volume	adjClose	adjHigh	adjLow	adjOpen	adjVolume	divCash	splitFactor
0	0	AAPL	2015-05-27 00:00:00+00:00	132.045	132.260	130.05	130.34	45833246	121.682558	121.880685	119.844118	120.111360	45833246	0.0	1.0
1	1	AAPL	2015-05-28 00:00:00+00:00	131.780	131.950	131.10	131.86	30733309	121.438354	121.595013	120.811718	121.512076	30733309	0.0	1.0
2	2	AAPL	2015-05-29 00:00:00+00:00	130.280	131.450	129.90	131.23	50884452	120.056069	121.134251	119.705890	120.931516	50884452	0.0	1.0
3	3	AAPL	2015-06-01 00:00:00+00:00	130.535	131.390	130.05	131.20	32112797	120.291057	121.078960	119.844118	120.903870	32112797	0.0	1.0
4	4	AAPL	2015-06-02 00:00:00+00:00	129.960	130.655	129.32	129.86	33667627	119.761181	120.401640	119.171406	119.669029	33667627	0.0	1.0

In [4]:
```
df.tail()
```

Out[4]:

	Unnamed: 0	symbol	date	close	high	low	open	volume	adjClose	adjHigh	adjLow	adjOpen	adjVolume	divCash	splitFactor
1253	1253	AAPL	2020-05-18 00:00:00+00:00	314.96	316.50	310.3241	313.17	33843125	314.96	316.50	310.3241	313.17	33843125	0.0	1.0
1254	1254	AAPL	2020-05-19 00:00:00+00:00	313.14	318.52	313.0100	315.03	25432385	313.14	318.52	313.0100	315.03	25432385	0.0	1.0
1255	1255	AAPL	2020-05-20 00:00:00+00:00	319.23	319.52	316.2000	316.68	27876215	319.23	319.52	316.2000	316.68	27876215	0.0	1.0
1256	1256	AAPL	2020-05-21 00:00:00+00:00	316.85	320.89	315.8700	318.66	25672211	316.85	320.89	315.8700	318.66	25672211	0.0	1.0
1257	1257	AAPL	2020-05-22 00:00:00+00:00	318.89	319.23	315.3500	315.77	20450754	318.89	319.23	315.3500	315.77	20450754	0.0	1.0

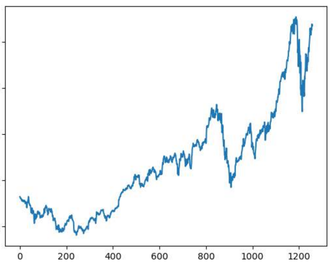
In [ ]:
```
### Data Collection
import pandas_datareader as pdr
key=""
df = pdr.get_data_tiingo('AAPL', api_key=key)
df.to_csv('AAPL.csv')
```

In [5]:
```
df1=df.reset_index()['close']
df1
```

Out[5]:
```
0       132.045
1       131.780
2       130.280
3       130.535
4       129.960
         ...
1253    314.960
1254    313.140
1255    319.230
1256    316.850
1257    318.890
Name: close, Length: 1258, dtype: float64
```

In [6]:
```
import matplotlib.pyplot as plt
plt.plot(df1)
```

Out[6]:
```
[<matplotlib.lines.Line2D at 0x1fd8b3dd790>]
```



In [7]:
```
### LSTM are sensitive to the scale of the data. so we apply MinMax scaler
import numpy as np


df1
```

Out[7]:
```
0       132.045
1       131.780
2       130.280
3       130.535
4       129.960
         ...
1253    314.960
1254    313.140
1255    319.230
1256    316.850
1257    318.890
Name: close, Length: 1258, dtype: float64
```

In [ ]:
```
from sklearn.preprocessing import MinMaxScaler
scaler=MinMaxScaler(feature_range=(0,1))
df1=scaler.fit_transform(np.array(df1).reshape(-1,1))
```

In [8]:
```
print(df1)
```

```
0       132.045
1       131.780
2       130.280
3       130.535
4       129.960
         ...
1253    314.960
1254    313.140
1255    319.230
1256    316.850
1257    318.890
Name: close, Length: 1258, dtype: float64
```

In [ ]:
```
##splitting dataset into train and test split
training_size=int(len(df1)*0.65)
test_size=len(df1)-training_size
train_data,test_data=df1[0:training_size,:],df1[training_size:len(df1),:1]
```

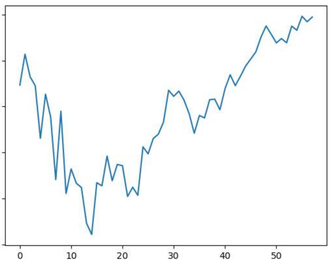
In [11]:
```
training_size,test_size
```

Out[11]:
```
(817, 441)
```

In [12]:
```
day_new=np.arange(1,101)
day_pred=np.arange(101,131)
```

In [62]:
```
df3=df1.tolist()
df3.extend(lst_output)
plt.plot(df3[1200:])
```

Out[62]:
```
[<matplotlib.lines.Line2D at 0x1608329b890>]
```

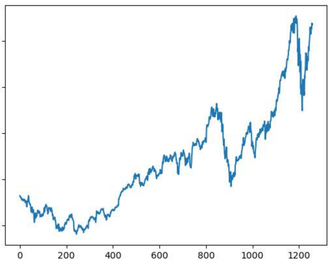


In [27]:
```
df3=df.reset_index()['close']
df3
```

Out[27]:
```
0       132.045
1       131.780
2       130.280
3       130.535
4       129.960
         ...
1253    314.960
1254    313.140
1255    319.230
1256    316.850
1257    318.890
Name: close, Length: 1258, dtype: float64
```

In [24]:
```
import matplotlib.pyplot as plt
plt.plot(df3)
```

Out[24]:
```
[<matplotlib.lines.Line2D at 0x1fd8e304f10>]
```



In [ ]:
```
```