DATA PREPROCESSING

Team ID	PNT2022TMID36951
Project Name	Crude Oil Price Prediction

Import The Libraries

In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

Importing the dataset

In [2]:
```
dv=pd.read_csv("Crude Oil Prices Daily.csv")
```

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

Handling Missing Data

In [4]:
```
dv.isnull().any()
```

Out[4]:
```
Date            False
Closing Value    True
dtype: bool
```

In [5]:
```
dv.isnull().sum()
```

Out[5]:
```
Date            0
Closing Value   7
dtype: int64
```

In [6]:
```
dv.dropna(axis=0,inplace=True)
```

In [7]:
```
dv.isnull().sum()
```

Out[7]:
```
Date            0
Closing Value   0
dtype: int64
```

In [9]:
```
dv_oil=dv.reset_index()['Closing Value']
```

In [10]:
```
dv_oil
```

Out[10]:
```
0       25.56
1       26.00
2       26.53
3       25.85
4       25.87
```

```
       ...
8211    73.89
8212    74.19
8213    73.05
8214    73.78
8215    73.93
Name: Closing Value, Length: 8216, dtype: float64
```

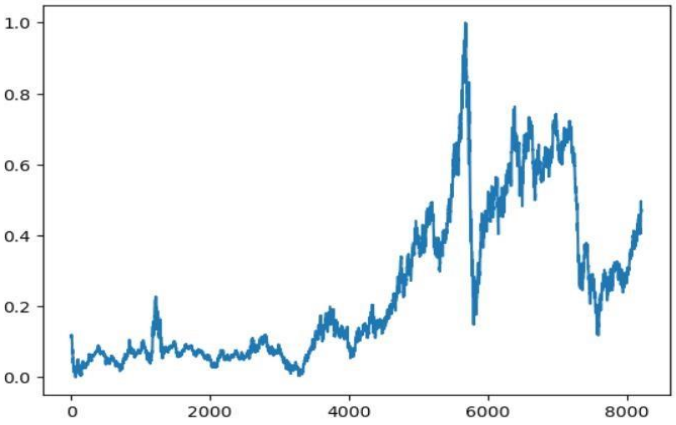
Feature Scaling

In [13]:
```
from sklearn.preprocessing import MinMaxScaler
scaler=MinMaxScaler(feature_range=(0,1))
dv_oil=scaler.fit_transform(np.array(dv_oil).reshape(-1,1))
```

Data Visualization

In [14]:
```
plt.plot(dv_oil)
```

Out[14]: []



Splitting Data Into Train And Test

In [15]:
```
training_size=int(len(dv_oil)*0.65)
test_size=len(dv_oil)-training_size
train_data,test_data=dv_oil[0:training_size,:],dv_oil[training_size:len(dv_oil),:1]
```

In [16]:
```
training_size,test_size
```

Out[16]: (5340, 2876)

In [17]:
```
train_data.shape
```

Out[17]: (5340, 1)

Creating A Dataset With Sliding Windows

In [18]:
```
def create_dataset(dataset, time_step=1):
    dataX, dataY = [], []
    for i in range(len(dataset)-time_step-1):
        a = dataset[i:(i+time_step),0]
        dataX.append(a)
        dataY.append(dataset[i + time_step,0])
    return np.array(dataX), np.array(dataY)
```

In [19]:
```
time_step = 10
X_train, y_train = create_dataset(train_data, time_step)
X_test, y_test = create_dataset(test_data, time_step)
```

In [20]:
```
print(X_train.shape), print(y_train.shape)
```

```
(5329, 10)
(5329,)
```

Out[20]: (None, None)

In [21]:
```
print(X_test.shape), print(y_test.shape)
```

```
(2865, 10)
(2865,)
```

Out[21]: (None, None)

In [22]:
```
X_train
```

Out[22]:
```
array([[0.11335703, 0.11661484, 0.12053902, ..., 0.10980305, 0.1089886 ,
        0.11054346],
       [0.11661484, 0.12053902, 0.11550422, ..., 0.1089886 , 0.11054346,
        0.10165852],
       [0.12053902, 0.11550422, 0.1156523 , ..., 0.11054346, 0.10165852,
        0.09906708],
       ...,
       [0.36731823, 0.35176958, 0.36080261, ..., 0.36391234, 0.37042796,
        0.37042796],
       [0.35176958, 0.36080261, 0.35354657, ..., 0.37042796, 0.37042796,
        0.37879461],
       [0.36080261, 0.35354657, 0.35295424, ..., 0.37042796, 0.37879461,
        0.37916482]])
```