```python
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
data={
    'Date':pd.date_range(start='2023-01-01', periods=30, freq='D'),
    'Open':np.random.uniform(15000,20000,30),
    'High':np.random.uniform(20000,25000,30),
    'Low': np.random.uniform(14000, 19000, 30),
    'Close': np.random.uniform(15000, 25000, 30),
    'Volume': np.random.uniform(500000, 1000000, 30),
    'Market Cap': np.random.uniform(300000000, 600000000, 30)
}
df=pd.DataFrame(data)
print(df.head())
```

```
        Date          Open          High           Low
Close  \
0 2023-01-01  16144.243872  23171.761891  16900.713506  18684.604178

1 2023-01-02  18755.360360  22617.447493  15975.580817  16365.262378

2 2023-01-03  15607.716807  21013.870519  17368.341532  24002.368427

3 2023-01-04  17434.177937  23694.117724  14097.496436  20665.980478

4 2023-01-05  17005.642760  23666.866543  15561.966208  21404.929413


          Volume    Market Cap
0  537502.686119  4.162288e+08
1  547545.825420  4.982122e+08
2  501945.592851  3.435210e+08
3  570856.577069  4.260933e+08
4  784086.534267  3.939071e+08
```

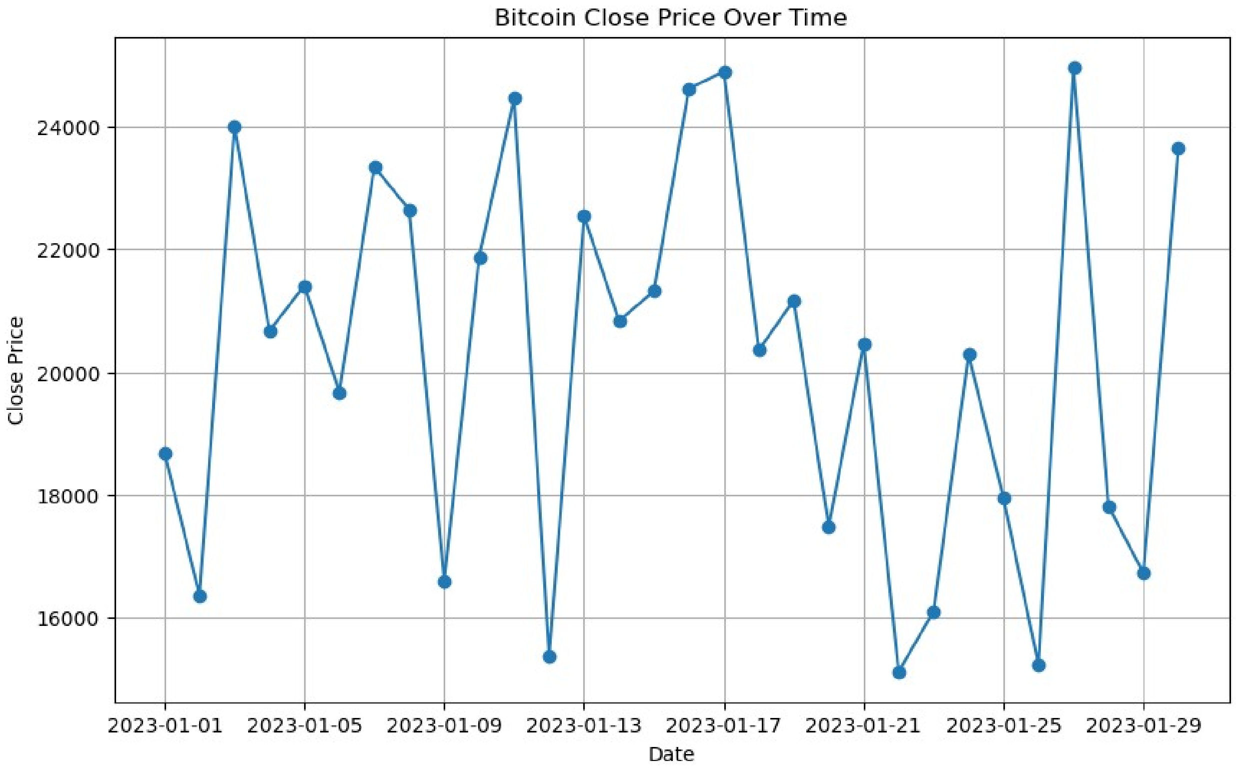
```python
#ADD DATA
#performing EDA

print(df.describe())
```

```
                      Date          Open          High
Low  \
count                   30     30.000000     30.000000     30.000000

mean   2023-01-15 12:00:00  17053.789409  22544.547212  16475.514056

min    2023-01-01 00:00:00  15036.057382  20137.746621  14097.496436

25%    2023-01-08 06:00:00  16145.208605  21259.381735  15271.537415
```

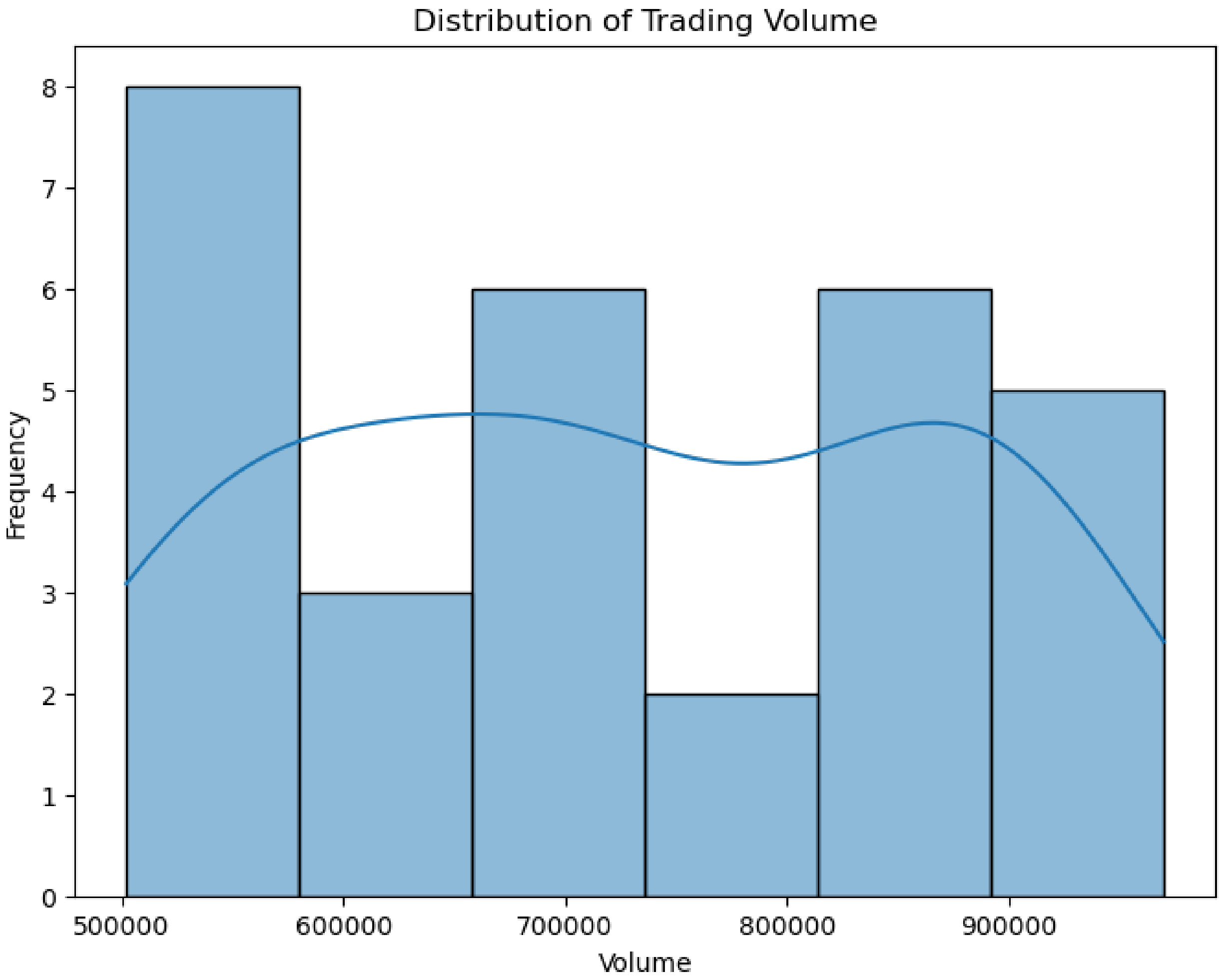
```
50%    2023-01-15 12:00:00  16993.974271  22873.947872  16666.868915

75%    2023-01-22 18:00:00  17736.826400  23573.178068  17330.531382

max    2023-01-30 00:00:00  19796.237447  24950.390256  18960.212715

std                    NaN   1317.654355   1415.291285   1378.717402
```

```
              Close         Volume    Market Cap
count     30.000000      30.000000  3.000000e+01
mean   20217.947782  726100.167418  4.483675e+08
min    15125.677998  501945.592851  3.003021e+08
25%    17567.860838  588814.008733  3.813930e+08
50%    20565.627597  710822.636409  4.525261e+08
75%    22611.556418  879452.881849  5.147255e+08
max    24958.580987  970019.423516  5.990765e+08
std     3150.148161  144289.080042  8.434337e+07
```

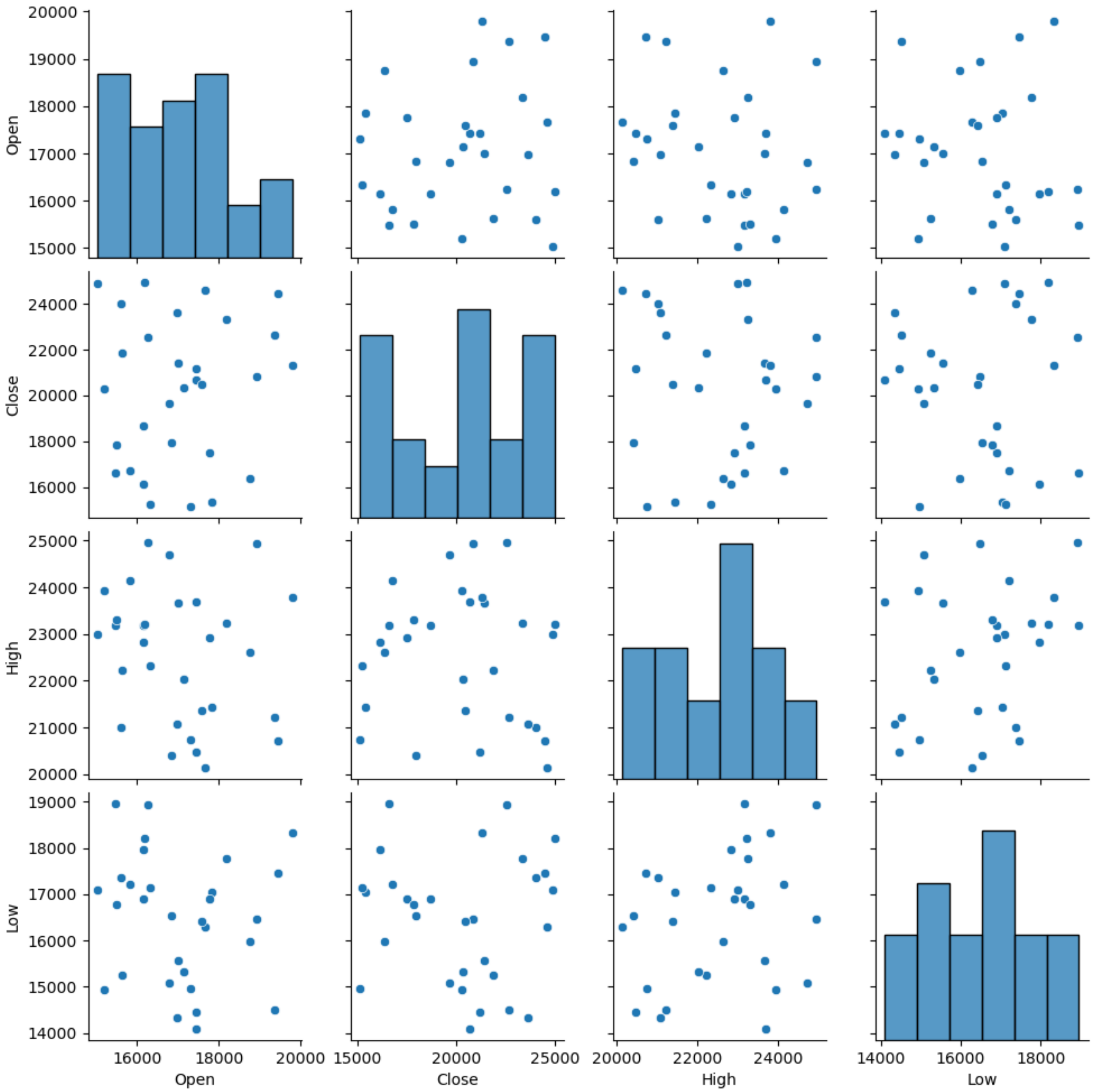
```python
plt.figure(figsize=(10,6))
plt.plot(df['Date'], df['Close'], marker='o')
plt.title('Bitcoin Close Price Over Time')
plt.xlabel('Date')
plt.ylabel('Close Price')
plt.grid(True)
plt.show()
```



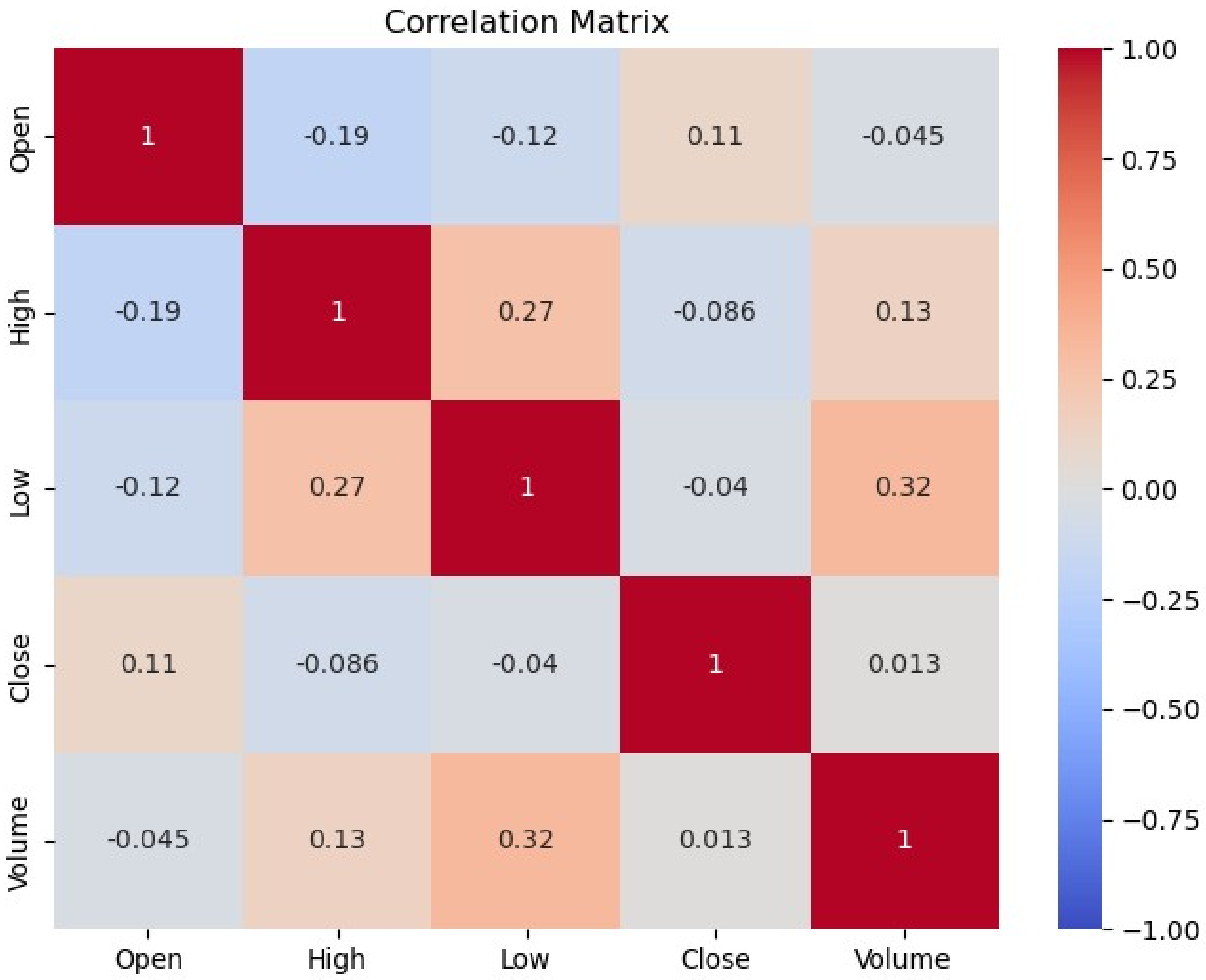
```
plt.figure(figsize=(8, 6))
sns.histplot(df['Volume'], kde=True)
plt.title('Distribution of Trading Volume')
plt.xlabel('Volume')
plt.ylabel('Frequency')
plt.show()
```



```
#creating a pair plot
sns.pairplot(df[['Open', 'Close', 'High', 'Low']])
plt.show()
```



```
#correlation
correlation_matrix = df[['Open', 'High', 'Low', 'Close',
'Volume']].corr()
plt.figure(figsize=(8, 6))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', vmin=-1,
vmax=1)
plt.title('Correlation Matrix')
plt.show()
```



```
#Apply Filtering Conditions
avg_volume = df['Volume'].mean()
df_filtered = df[(df['Close'] > df['Open']) & (df['Volume'] >
avg_volume)]
print(df_filtered)
```

```
         Date          Open          High           Low         Close
\
4  2023-01-05  17005.642760  23666.866543  15561.966208  21404.929413

5  2023-01-06  16798.332183  24704.766766  15079.494912  19672.714263

6  2023-01-07  18191.610288  23235.830564  17777.650381  23336.592394

9  2023-01-10  15622.710262  22224.329570  15256.304438  21872.542150

10 2023-01-11  19462.332526  20706.147364  17449.254864  24453.685320

12 2023-01-13  16250.347302  24950.390256  18927.476239  22531.650917
```

```
14 2023-01-15  19796.237447  23786.024648  18330.797687  21313.114311

15 2023-01-16  17652.524327  20137.746621  16293.608392  24610.047470

16 2023-01-17  15036.057382  22983.248793  17097.104930  24881.993283

20 2023-01-21  17594.590039  21371.972430  16414.703965  20465.274717

24 2023-01-25  16836.009752  20395.500052  16545.093685  17941.743689

28 2023-01-29  15822.154312  24135.820387  17217.100932  16729.277366
```

```
           Volume    Market Cap
4   784086.534267  3.939071e+08
5   853193.312594  5.990765e+08
6   893487.537225  3.180282e+08
9   898296.347664  4.682099e+08
10  744595.703467  5.868333e+08
12  908228.938806  4.451487e+08
14  881409.058988  5.325931e+08
15  906515.018451  3.470896e+08
16  970019.423516  3.772216e+08
20  814180.628449  4.914450e+08
24  879470.193118  4.909728e+08
28  880156.349030  5.174343e+08
```

```
#For Loop and Conditional Statements
df['Price Change'] = df['Close'] - df['Open']

df['Price Trend'] = df.apply(lambda row: 'Up' if row['Close'] >
row['Open'] else 'Down', axis=1)
print(df[['Date', 'Open', 'Close', 'Price Change', 'Price Trend']])
```

```
         Date          Open         Close  Price Change Price Trend
0  2023-01-01  16144.243872  18684.604178   2540.360306          Up
1  2023-01-02  18755.360360  16365.262378  -2390.097982        Down
2  2023-01-03  15607.716807  24002.368427   8394.651620          Up
3  2023-01-04  17434.177937  20665.980478   3231.802541          Up
4  2023-01-05  17005.642760  21404.929413   4399.286653          Up
5  2023-01-06  16798.332183  19672.714263   2874.382079          Up
6  2023-01-07  18191.610288  23336.592394   5144.982106          Up
7  2023-01-08  19368.954282  22638.191585   3269.237303          Up
8  2023-01-09  15471.880923  16591.329862   1119.448939          Up
9  2023-01-10  15622.710262  21872.542150   6249.831887          Up
10 2023-01-11  19462.332526  24453.685320   4991.352794          Up
11 2023-01-12  17843.240157  15366.433870  -2476.806287        Down
12 2023-01-13  16250.347302  22531.650917   6281.303615          Up
13 2023-01-14  18931.845666  20830.877923   1899.032257          Up
14 2023-01-15  19796.237447  21313.114311   1516.876864          Up
15 2023-01-16  17652.524327  24610.047470   6957.523143          Up
```

```
16 2023-01-17  15036.057382  24881.993283   9845.935901          Up
17 2023-01-18  17132.153136  20353.414153   3221.261017          Up
18 2023-01-19  17436.845492  21167.769653   3730.924162          Up
19 2023-01-20  17764.927091  17484.708241   -280.218849        Down
20 2023-01-21  17594.590039  20465.274717   2870.684678          Up
21 2023-01-22  17313.883867  15125.677998  -2188.205869        Down
22 2023-01-23  16148.102805  16106.673900    -41.428905        Down
23 2023-01-24  15190.447443  20285.045461   5094.598018          Up
24 2023-01-25  16836.009752  17941.743689   1105.733938          Up
25 2023-01-26  16330.529328  15236.534667  -1093.994662        Down
26 2023-01-27  16192.060919  24958.580987   8766.520068          Up
27 2023-01-28  15496.457816  17817.318629   2320.860813          Up
28 2023-01-29  15822.154312  16729.277366    907.123054          Up
29 2023-01-30  16982.305783  23644.095779   6661.789996          Up
```

```
#close the project by analysing the bitcoin data
```