```
import pandas as pd
import numpy as np
from datetime import datetime
import matplotlib.pyplot as plt


assets = ['CIPLA',  'PFC',  'RECLTD',   'IOC',  'HINDPETRO',    'AKZONOBEL',    'BPCL', 'GILL
]
weights = np.array([0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1])



df = pd.read_excel (r'/content/Portfolio Master (1).xlsx')
df
```

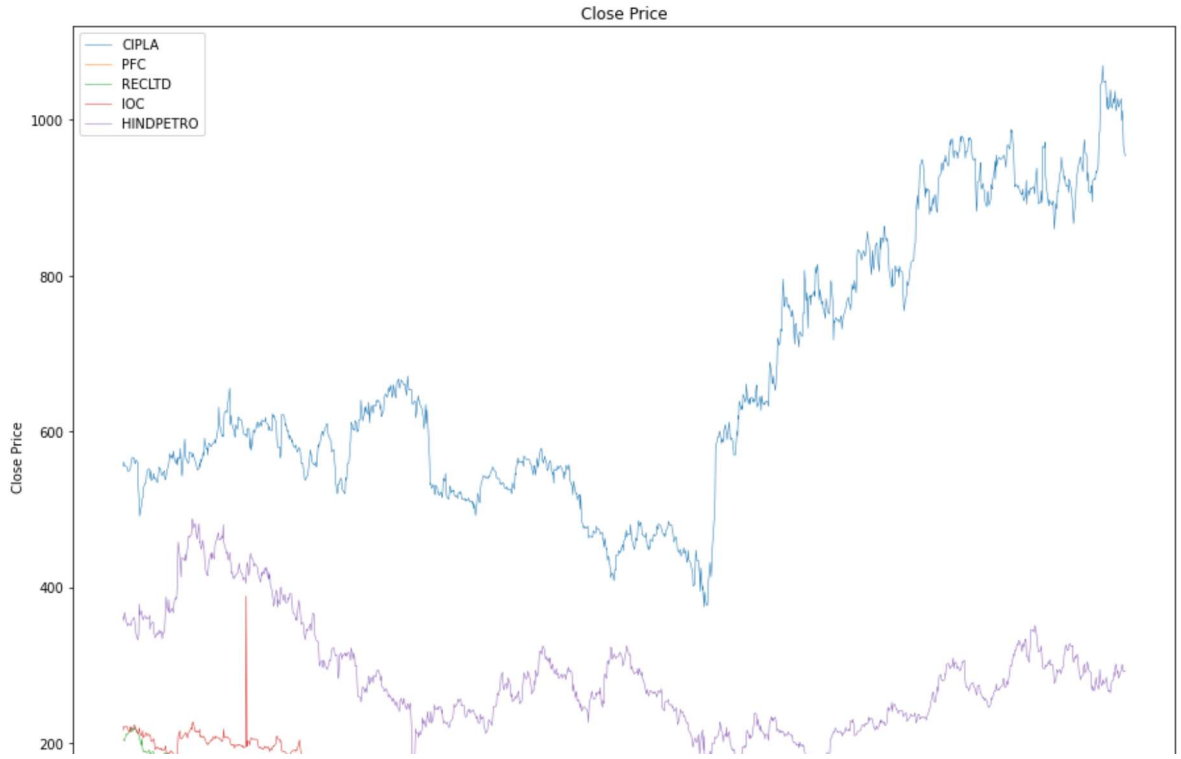
	Date	CIPLA	PFC	RECLTD	IOC	HINDPETRO	AKZONO
0	2017-04-26	555.099976	157.699997	204.699997	216.625000	361.600006	199
1	2017-04-27	560.900024	157.850006	203.850006	221.550003	357.166656	197
2	2017-04-28	556.799988	159.500000	202.649994	220.000000	367.799988	196
3	2017-05-02	554.500000	163.500000	212.050003	221.524994	354.333344	197
4	2017-05-03	554.299988	160.850006	210.899994	220.675003	354.100006	199
...	...	...	...	...	...	...	
1234	2022-04-25	953.650024	116.000000	125.949997	127.699997	292.399994	187
1235	NaT	NaN	NaN	NaN	NaN	NaN	N

```
stock = df
stock["Date"] = stock["Date"].astype("datetime64")
stock
```

	Date	CIPLA	PFC	RECLTD	IOC	HINDPETRO	AKZONO
0	2017-04-26	555.099976	157.699997	204.699997	216.625000	361.600006	199
1	2017-04-27	560.900024	157.850006	203.850006	221.550003	357.166656	197
2	2017-04-28	556.799988	159.500000	202.649994	220.000000	367.799988	196
3	2017-05-02	554.500000	163.500000	212.050003	221.524994	354.333344	197
4	2017-	554.299988	160.850006	210.899994	220.675003	354.100006	199

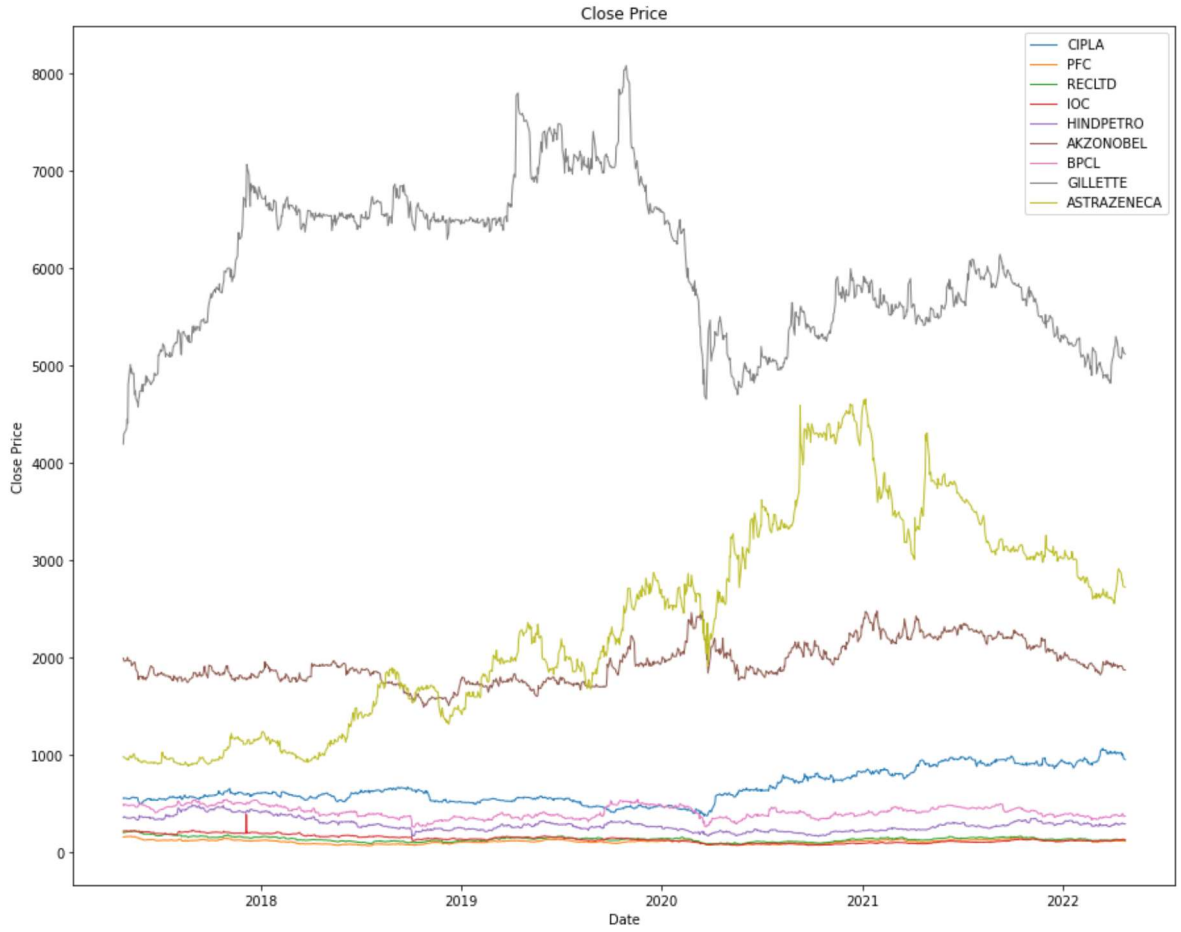
```
plt.figure(figsize=(15, 12))
plt.title('Close Price')

plt.plot(stock.Date, stock.CIPLA, label='CIPLA',
         linewidth=0.5)
plt.plot(stock.Date, stock.PFC, label='PFC',
         linewidth=0.5)
plt.plot(stock.Date, stock.RECLTD, label='RECLTD',
         linewidth=0.5)
plt.plot(stock.Date, stock.IOC, label='IOC',
         linewidth=0.5)
plt.plot(stock.Date, stock.HINDPETRO, label='HINDPETRO',
         linewidth=0.5)
plt.xlabel('Date')
plt.ylabel('Close Price')
plt.legend()
```

```
<matplotlib.legend.Legend at 0x7fddbcb583d0>
```




```
plt.figure(figsize=(15, 12))
plt.title('Close Price')
plt.plot(stock.Date, stock.CIPLA, label='CIPLA',
         linewidth=1)
plt.plot(stock.Date, stock.PFC, label='PFC',
         linewidth=1)
plt.plot(stock.Date, stock.RECLTD, label='RECLTD',
         linewidth=1)
plt.plot(stock.Date, stock.IOC, label='IOC',
         linewidth=1)
plt.plot(stock.Date, stock.HINDPETRO, label='HINDPETRO',
         linewidth=1)
plt.plot(stock.Date, stock.AKZONOBEL, label='AKZONOBEL',
         linewidth=1)
plt.plot(stock.Date, stock.BPCL, label='BPCL',
         linewidth=1)
plt.plot(stock.Date, stock.GILLETTE, label='GILLETTE',
         linewidth=1)
plt.plot(stock.Date, stock.ASTRAZENECA, label='ASTRAZENECA',
         linewidth=1)
#plt.plot(stock.Date, stock.NESTLE, label='NESTLE',
#         linewidth=1)
plt.xlabel('Date')
plt.ylabel('Close Price')
plt.legend()
```

```
<matplotlib.legend.Legend at 0x7fddbbab5650>
```




```
plt.plot(stock.Date, stock.IOC, label='IOC',
         linewidth=1)
plt.plot(stock.Date, stock.PFC, label='PFC',
         linewidth=1)
plt.plot(stock.Date, stock.RECLTD, label='RECLTD',
         linewidth=1)
```

```
[<matplotlib.lines.Line2D at 0x7fddb9d0d6d0>]
```


```
CIPLA = stock['CIPLA'].pct_change()
PFC = stock['PFC'].pct_change()
RECLTD  =stock['RECLTD'].pct_change()
IOC= stock['IOC'].pct_change()
HINDPETRO   =stock['HINDPETRO'].pct_change()
AKZONOBEL   =stock['AKZONOBEL'].pct_change()
BPCL =  stock['BPCL'].pct_change()
GILLETTE=stock['GILLETTE'].pct_change()
ASTRAZENECA =stock['ASTRAZENECA'].pct_change()
NESTLE= stock['NESTLE'].pct_change()


ERs = {"CIPLA": CIPLA,
        "PFC": PFC,
        "RECLTD": RECLTD,
        "IOC" : IOC,
        "HINDPETRO": HINDPETRO,
        "AKZONOBEL": AKZONOBEL,
        "BPCL": BPCL,
        "GILLETTE": GILLETTE,
        "ASTRAZENECA": ASTRAZENECA,
        "NESTLE": NESTLE}


CIPLA.head(10)
```

```
0         NaN
1    0.010449
2   -0.007310
3   -0.004131
4   -0.000361
5   -0.008389
6   -0.001910
7    0.003737
8    0.001634
9    0.014867
Name: CIPLA, dtype: float64
```

```
Returndf = pd.concat(ERs,
               axis = 1)


Returndf.drop(Returndf.index[0])
Returndf
```

	CIPLA	PFC	RECLTD	IOC	HINDPETRO	AKZONOBEL	BPC
0	NaN	NaN	NaN	NaN	NaN	NaN	Nal
1	0.010449	0.000951	-0.004152	0.022735	-0.012260	-0.011180	-0.02182
2	-0.007310	0.010453	-0.005887	-0.006996	0.029771	-0.005146	0.02850
3	-0.004131	0.025078	0.046385	0.006932	-0.036614	0.005096	-0.00709
4	-0.000361	-0.016208	-0.005423	-0.003837	-0.000659	0.009431	-0.00462
...	...	...	...	...	...	...	.
1234	-0.013806	-0.023569	-0.020987	-0.049144	0.003087	-0.001066	0.00784
1235	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.00000
1236	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.00000
1237	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.00000
1238	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.00000

```
Ret = pd.read_excel (r'/content/Returns.xlsx')
Ret["Date"] = Ret["Date"].astype("datetime64")
```

```
Ret
```

```
            Date      CIPLA       PFC     RECLTD       IOC  HINDPETRO  AKZONOBEL
```

```
import seaborn as sn


covmat = Ret.cov() * 252
covmat
```

	CIPLA	PFC	RECLTD	IOC	HINDPETRO	AKZONOBEL
CIPLA	0.081048	0.020620	0.018956	0.013056	0.003798	0.003279
PFC	0.020620	0.150426	0.113657	0.053570	0.001562	0.003159
RECLTD	0.018956	0.113657	0.145884	0.049960	-0.002657	0.003272
IOC	0.013056	0.053570	0.049960	0.355825	0.005563	0.001239
HINDPETRO	0.003798	0.001562	-0.002657	0.005563	0.179038	0.000880
AKZONOBEL	0.003279	0.003159	0.003272	0.001239	0.000880	0.064046
BPCL	0.004656	0.005841	-0.000275	0.006243	0.123218	0.001978
GILLETTE	0.004331	0.004958	0.005221	-0.002625	0.014963	-0.003393
ASTRAZENECA	0.002003	0.013654	0.012821	0.006361	0.001194	0.020275

```
plt.figure(figsize=(15, 12))
sn.heatmap(covmat, annot=True, fmt='g')
plt.show()
```


```
variance = np.dot(weights.T, np.dot(covmat,weights))
variance
```

```
0.025654590445933385
```

```
volatility = np.sqrt(variance)
volatility
```

```
0.16017050429443427
```

```
annualizedreturns = np.sum(Ret.mean()* weights) * 252

annualizedreturns
```

```
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:1: FutureWarning: DataFrame
  """Entry point for launching an IPython kernel.
0.08567519669609704
```

```
pip install PyPortfolioOpt
```

```
Collecting PyPortfolioOpt
  Downloading pyportfolioopt-1.5.2-py3-none-any.whl (61 kB)
     |████████████████████████████████| 61 kB 3.3 MB/s
Requirement already satisfied: pandas>=0.19 in /usr/local/lib/python3.7/dist-packages (
Collecting cvxpy<2.0.0,>=1.1.10
  Downloading cvxpy-1.2.0-cp37-cp37m-manylinux_2_24_x86_64.whl (2.8 MB)
     |████████████████████████████████| 2.8 MB 10.5 MB/s
Requirement already satisfied: scipy<2.0,>=1.3 in /usr/local/lib/python3.7/dist-packages
Requirement already satisfied: numpy<2.0,>=1.12 in /usr/local/lib/python3.7/dist-package
Requirement already satisfied: ecos>=2 in /usr/local/lib/python3.7/dist-packages (from c
Requirement already satisfied: scs>=1.1.6 in /usr/local/lib/python3.7/dist-packages (fro
```

```
Requirement already satisfied: osqp>=0.4.1 in /usr/local/lib/python3.7/dist-packages (fr
Requirement already satisfied: qdldl in /usr/local/lib/python3.7/dist-packages (from osq
Requirement already satisfied: python-dateutil>=2.7.3 in /usr/local/lib/python3.7/dist-p
Requirement already satisfied: pytz>=2017.3 in /usr/local/lib/python3.7/dist-packages (f
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.7/dist-packages (from
Installing collected packages: cvxpy, PyPortfolioOpt
  Attempting uninstall: cvxpy
    Found existing installation: cvxpy 1.0.31
    Uninstalling cvxpy-1.0.31:
      Successfully uninstalled cvxpy-1.0.31
Successfully installed PyPortfolioOpt-1.5.2 cvxpy-1.2.0
```

```
from pypfopt.efficient_frontier import EfficientFrontier
from pypfopt import risk_models
from pypfopt import expected_returns
```

```
stock = stock.drop(['Date'], axis = 1)
stock
```

	CIPLA	PFC	RECLTD	IOC	HINDPETRO	AKZONOBEL	COCHINSHIP	
0	555.099976	157.699997	204.699997	216.625000	361.600006	1994.70	522.000000	4
1	560.900024	157.850006	203.850006	221.550003	357.166656	1972.40	541.549988	4
2	556.799988	159.500000	202.649994	220.000000	367.799988	1962.25	536.549988	4
3	554.500000	163.500000	212.050003	221.524994	354.333344	1972.25	536.500000	4
4	554.299988	160.850006	210.899994	220.675003	354.100006	1990.85	532.750000	4
...	...	...	...	...	...	...	...	
1234	953.650024	116.000000	125.949997	127.699997	292.399994	1873.75	NaN	3
1235	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1236	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1237	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1238	NaN	NaN	NaN	NaN	NaN	NaN	NaN	

1239 rows × 11 columns

```
mu = expected_returns.mean_historical_return(stock)
S = risk_models.sample_cov(stock)
ef = EfficientFrontier(mu,S)
weights = ef.max_sharpe()
cleanedweights = ef.clean_weights()
print(cleanedweights)
ef.portfolio_performance(verbose=True)
```

```
OrderedDict([('CIPLA', 0.18522), ('PFC', 0.0), ('RECLTD', 0.0), ('IOC', 0.0), ('HINDPETR
Expected annual return: 20.3%
Annual volatility: 17.7%
Sharpe Ratio: 1.03
(0.20256956562418288, 0.17650351617978455, 1.0343678674266157)
```

```
from pypfopt import objective_functions

ef = EfficientFrontier(mu, S)
ef.add_objective(objective_functions.L2_reg, gamma=0.1)
w = ef.max_sharpe()
print(ef.clean_weights())
```

```
OrderedDict([('CIPLA', 0.19148), ('PFC', 0.0), ('RECLTD', 0.0), ('IOC', 0.0), ('HINDPETR
/usr/local/lib/python3.7/dist-packages/pypfopt/efficient_frontier/efficient_frontier.py
  "max_sharpe transforms the optimization problem so additional objectives may not work
```