Homework 3 – FAC 407

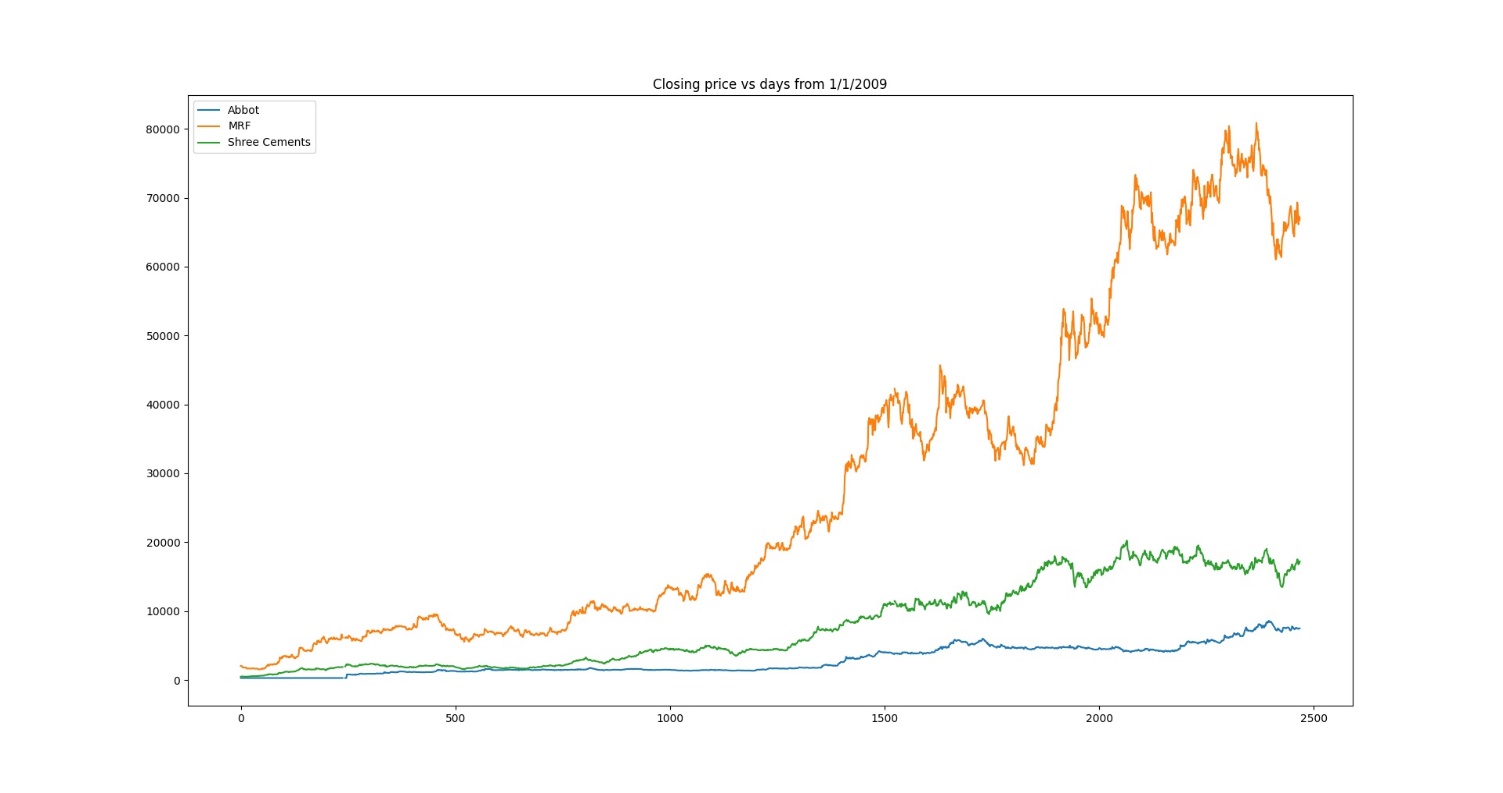
Pranav Sateesh

Financial mini-Project

The Companies in focus:

1. Abbot India
2. MRF
3. Shree Cements

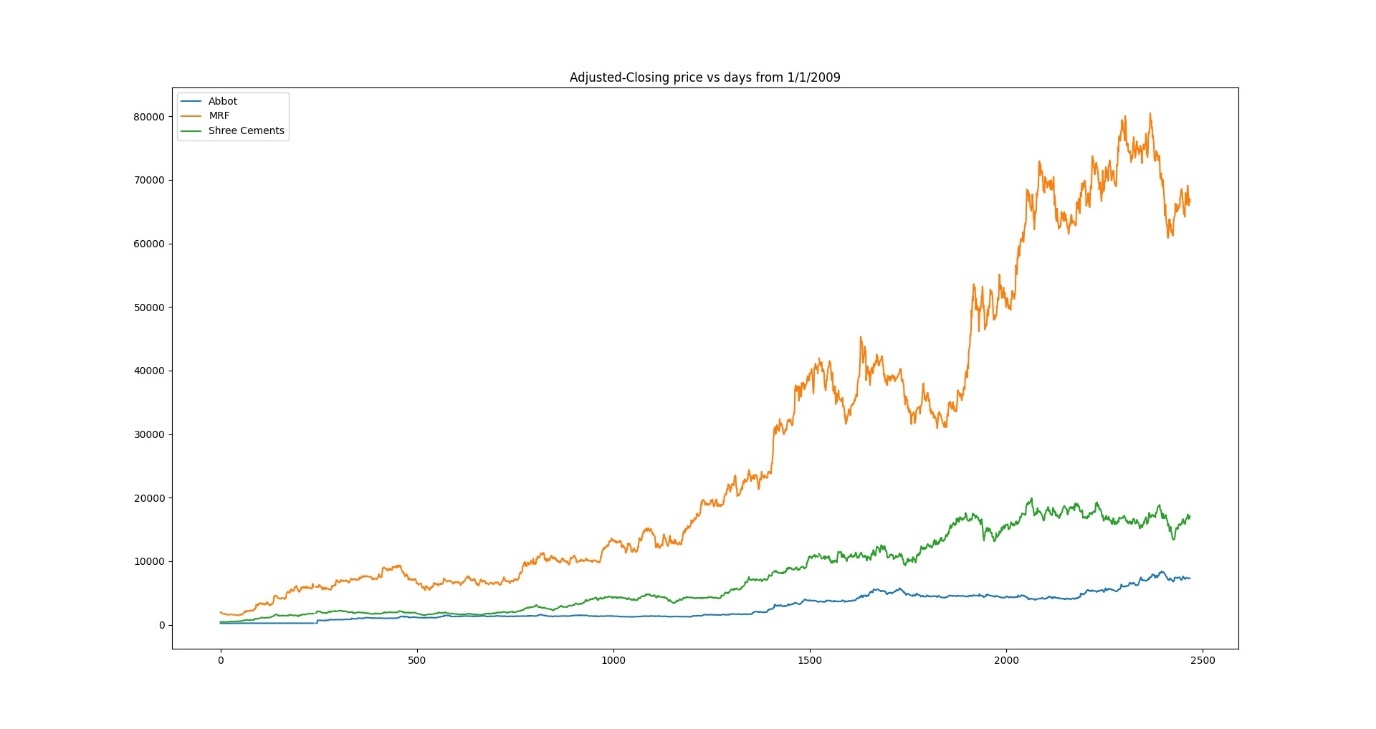
The project has been worked out in python, using the pandas, statistics and numpy modules.



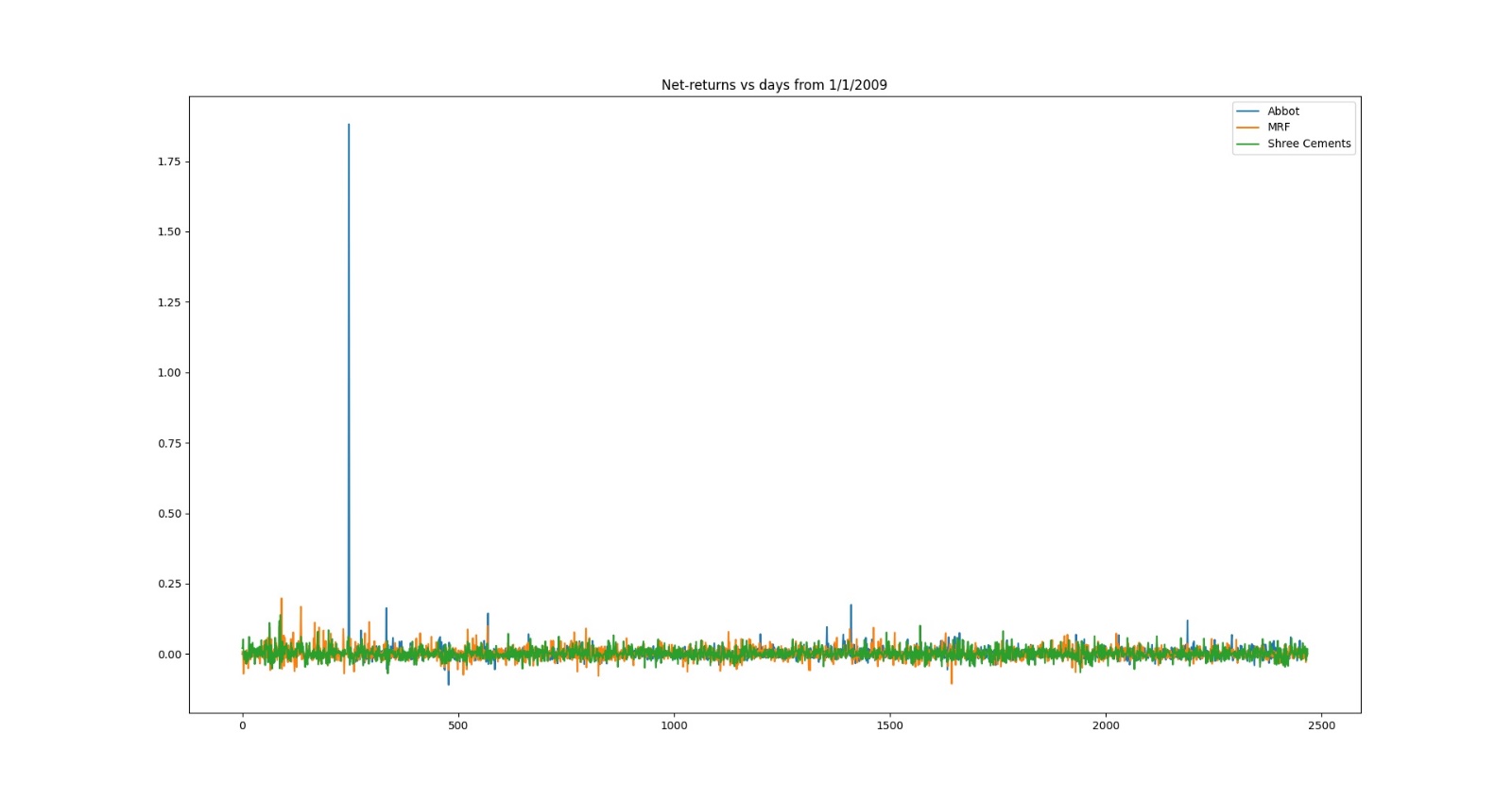
The above plot is a plot of the closing price vs time from 1/1/2009. Here the growth of MRF is most obvious. The other two stocks have had a more horizontal slope.

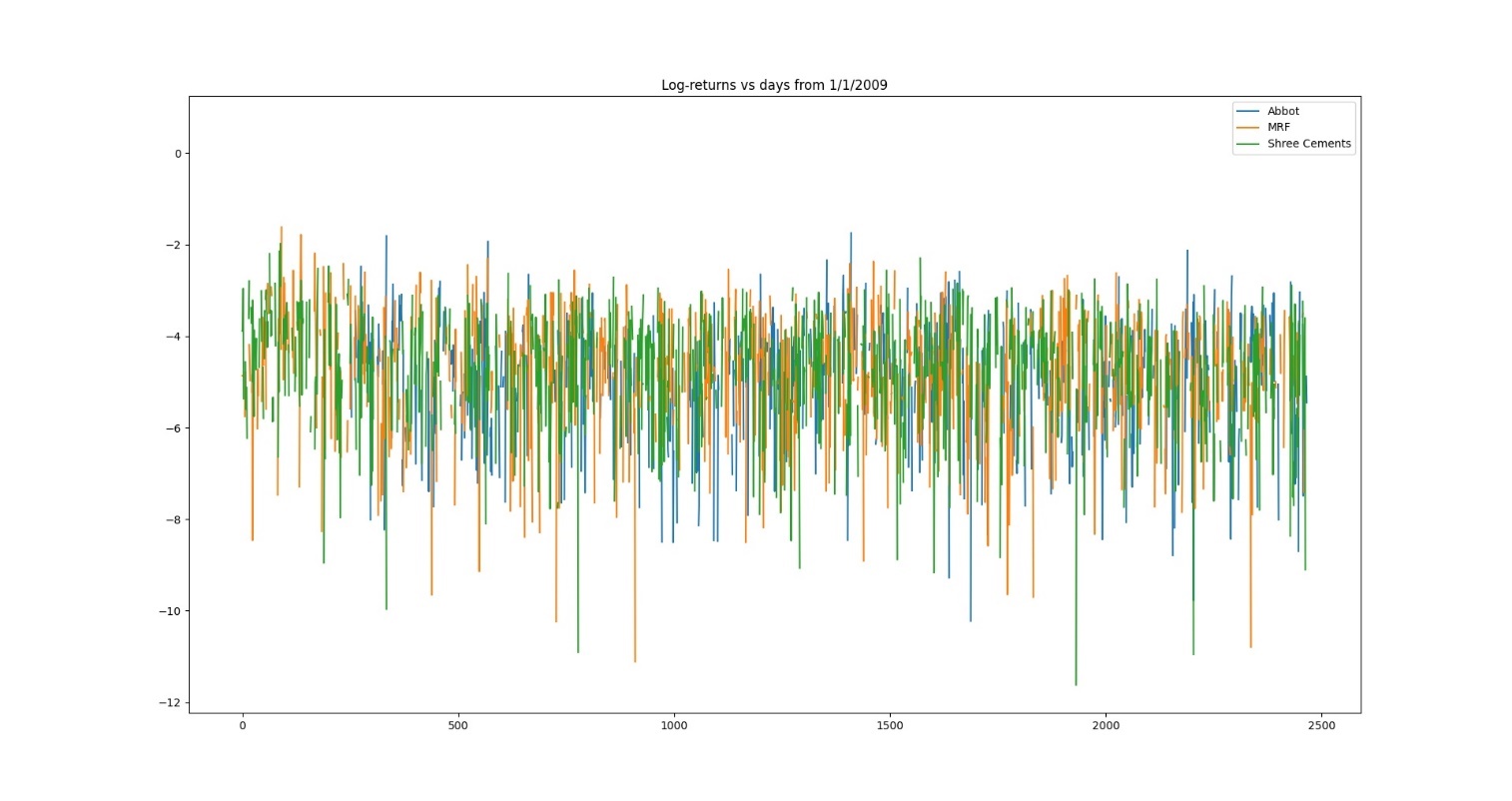
This also shows that the MRF price looks more volatile when compared to the other two companies.

One reason I believe is the reason for this trend is due to the fact that the years 2009 – 2018 did not see great growth in the pharmacy and electrical industries, but the automobile industry did see an increase in revenues, helping MRF a tyre company expand. Abbot has seen exponential growth since due to the pandemic.



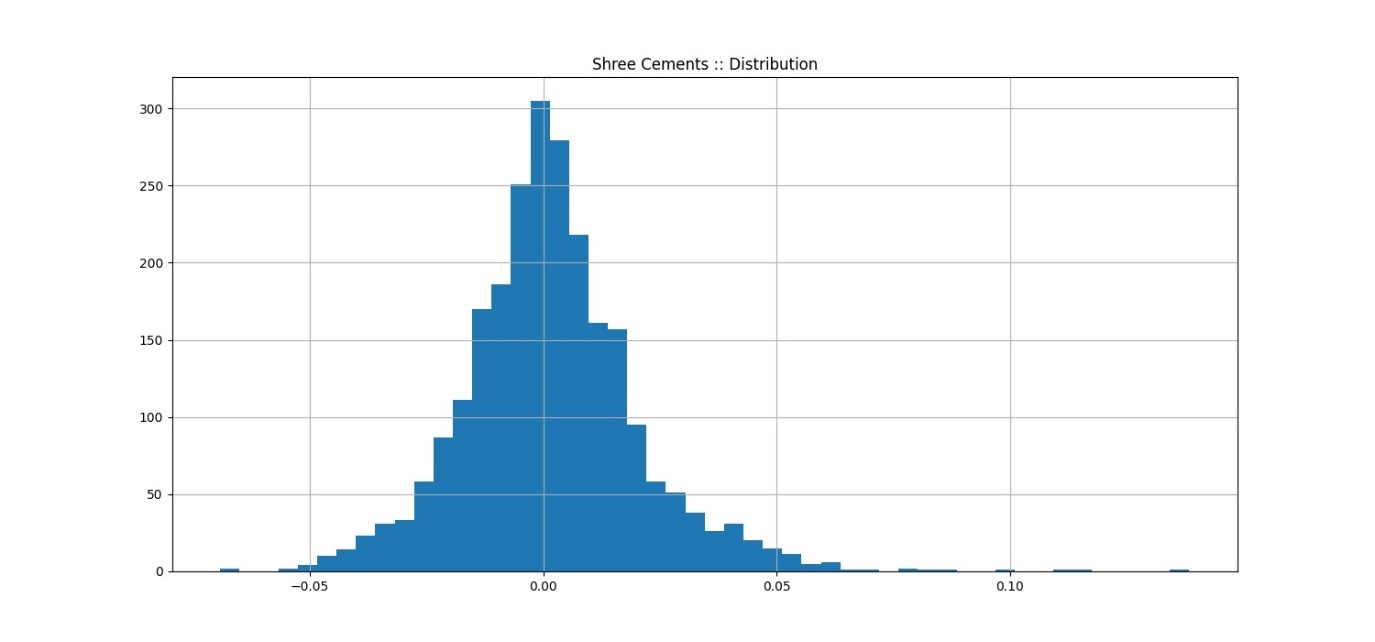
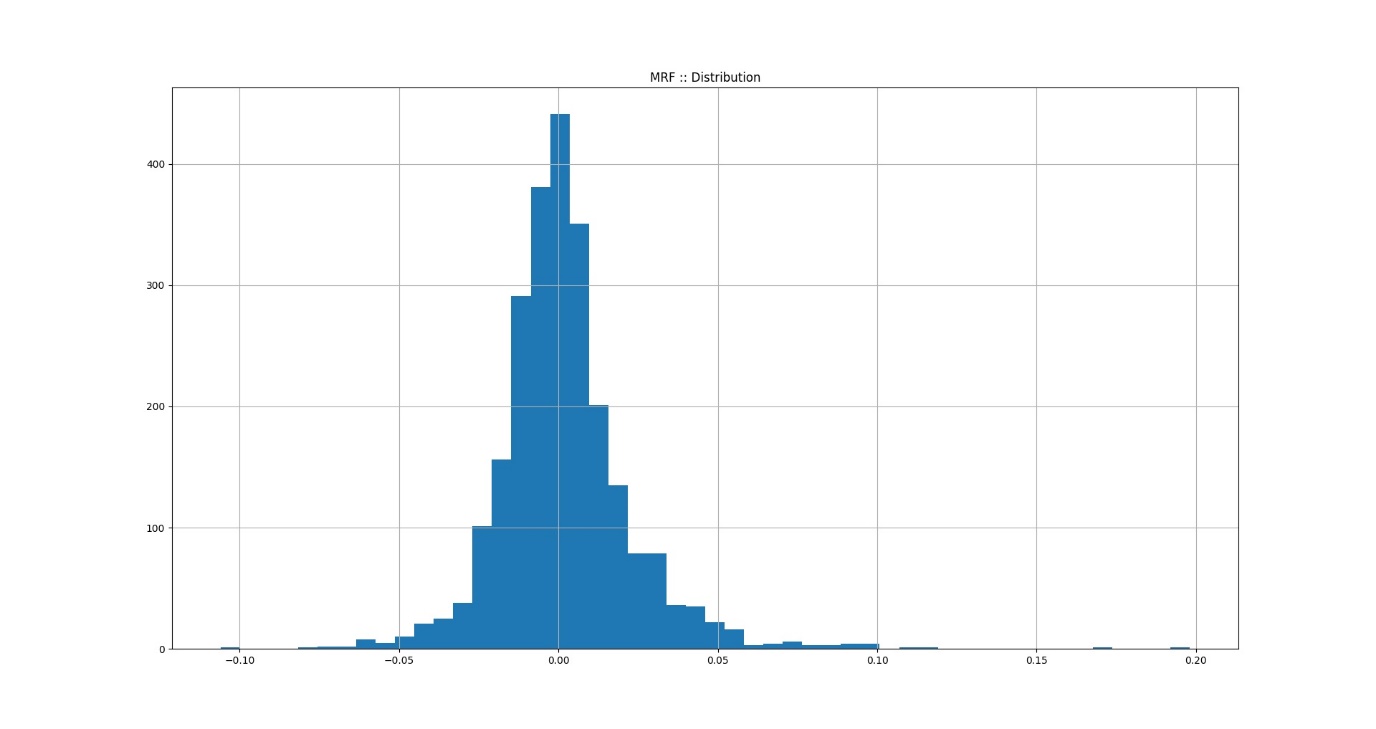
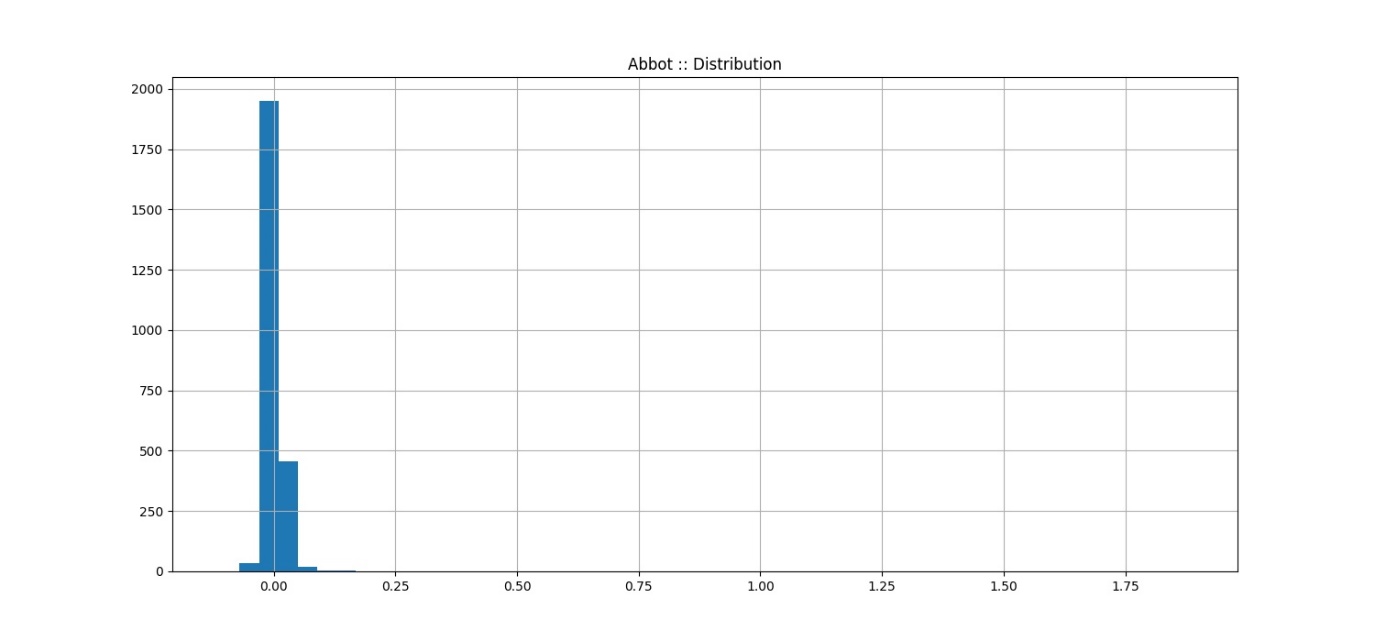
This is the plot for the adjusted price, we see that the adjusted price and the closing price are the same, so the trends remain the same.

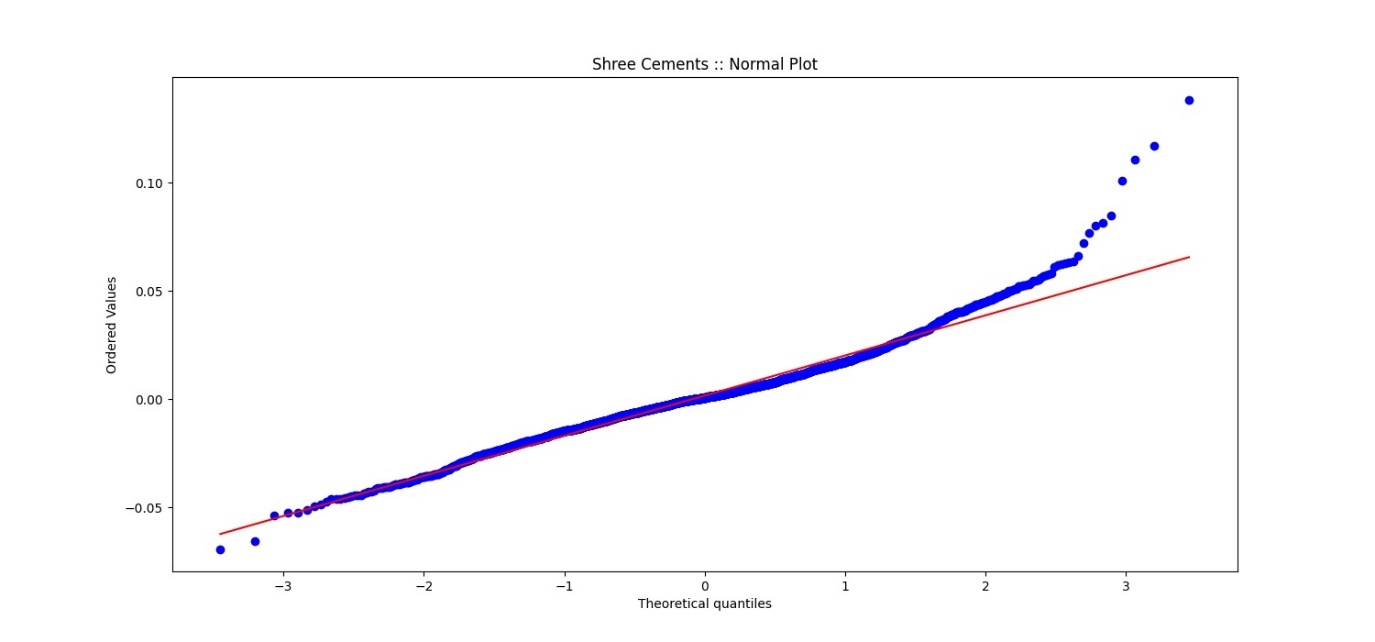
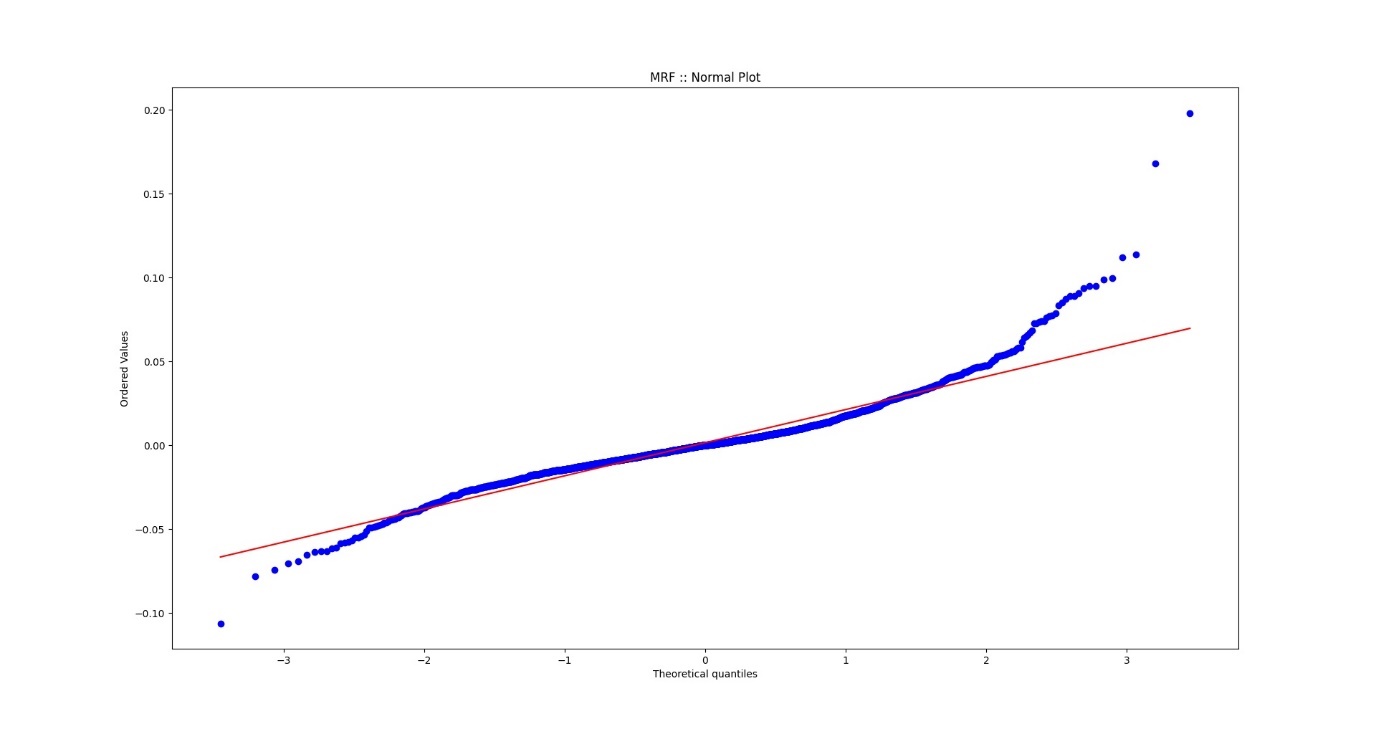
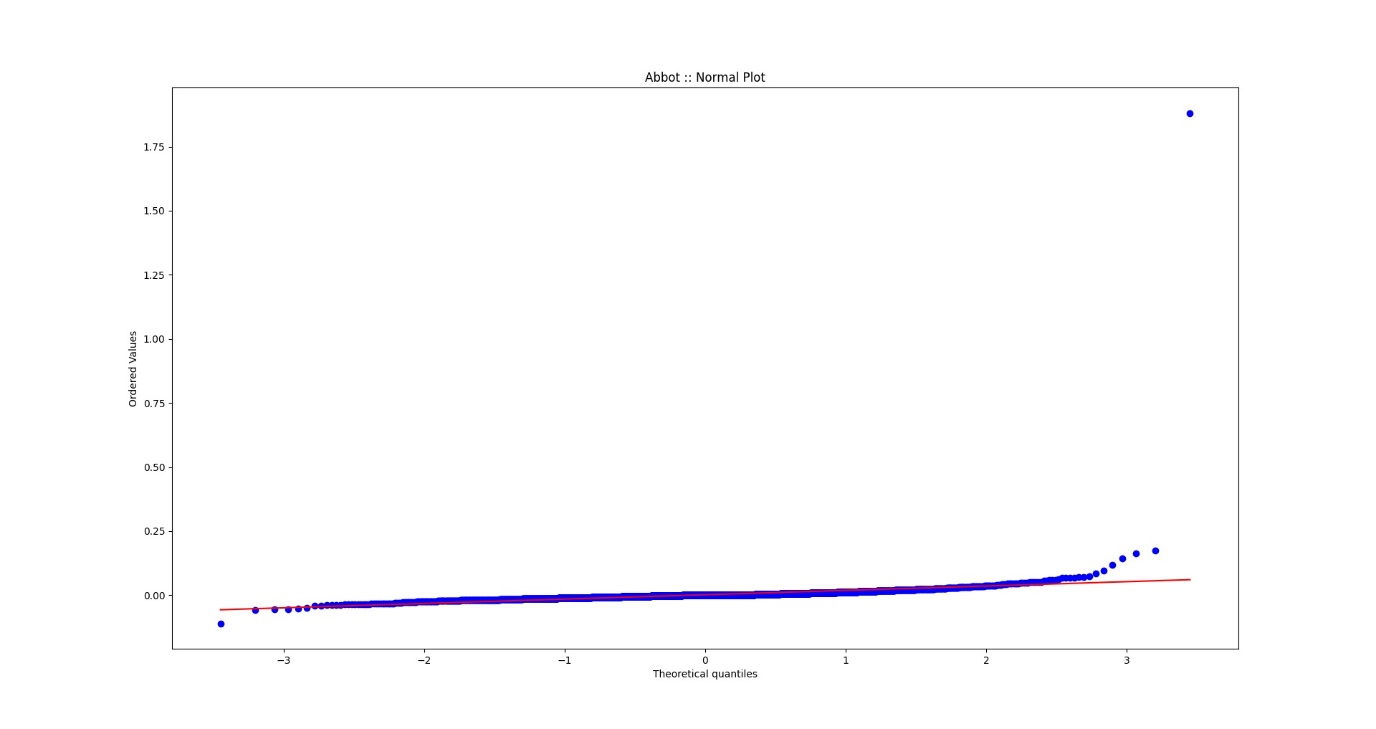
This is a plot of net returns, the graph looks fairly consistent, that the perturbations do not look very large except for the case of Abbot, where we observe a huge spike, this is due to the case of a merger in 2010 with Solvay India at a 2:3 ratio.

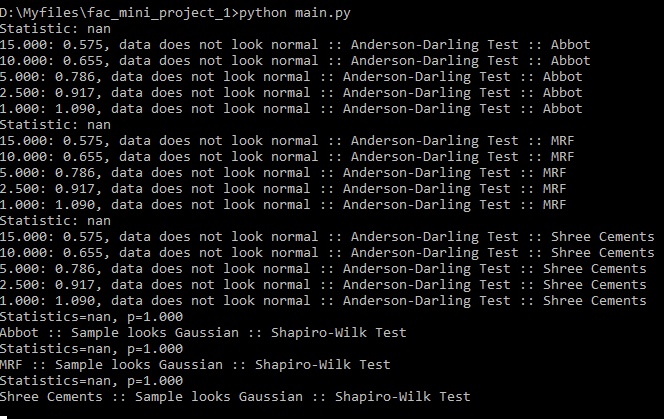


This is a plot of the log-returns, this graph is similar to that of the net-returns. The graph shares the perturbations observed in the net-returns graph.

The Next page covers graphs of distribution and normal plot of the closing prices.







The above are the results obtained for the tests for gaussian and normal distribution.

The results obtained for the correlation between returns:

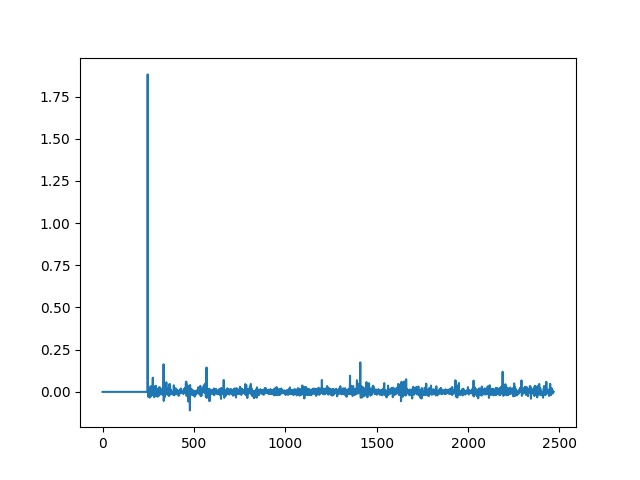
Abbot :: MRF = 0.0404

Abbot :: Shree Cements = 0.0102

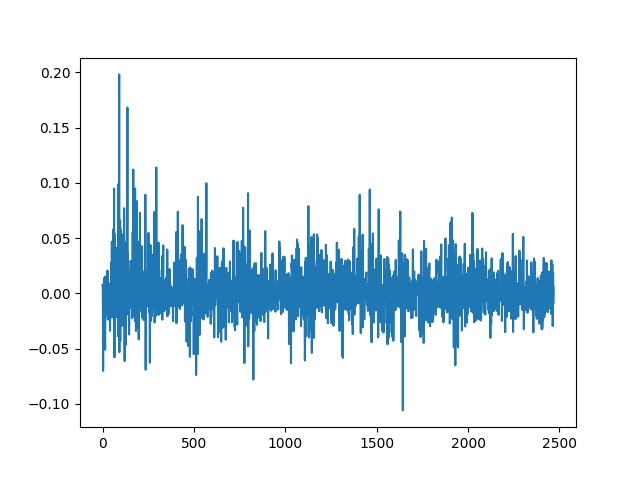
MRF :: Shree Cements = 0.2098

This shows that the correlation between the Pharma industry and the tyre industry is low and the correlation between the Pharma industry and electricals is even lower as observed from the correlations of 0.0404 and 0.0102 respectively.

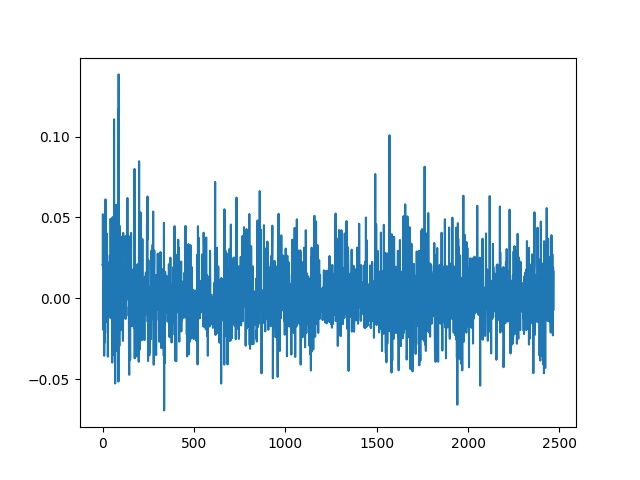
We see a higher correlation between MRF and Shree cements, even though the number is not high enough to commend a specific reason for such behaviour. It makes logical sense that the tyre industry is correlated to the Cement industry because of the industrial and manufacturing nature of both sectors.



ABBOT-NET RETURNS



MRF-NET RETURNS



SHREE-CEMENTS RETURNS

As observed here the net-returns for abbot is very distinct from both MRF and Shree cements. This is observed in the correlation as well, the graphs for MRF and Shree Cements are similar and between (-0.1, 0.2) and (-0.05,-0.1). The peaks do not necessarily correspond but the overall structure looks very similar.

For the probability plots that have been obtained, we observe that both Shree Cements and MRF have a more Gaussian distribution compared to the more linear Abbot. This is also observed in the very less volatile nature of the Abbot stock.

::PYTHON CODE::

MODULE WITH FUCNTIONS

import pandas as pd

import datetime

import matplotlib.pyplot as plt

from numpy.random import seed

from numpy.random import randn

from scipy.stats import shapiro, anderson

import scipy.stats as stats

import numpy as np

import os

import statistics

import time

def normal\_test\_sw():

    abbot = pd.read\_csv('ABBOTINDIA.NS.csv')

    mrf = pd.read\_csv('MRF.NS.csv')

    shreecem = pd.read\_csv('SHREECEM.NS.csv')

    stat, p = shapiro(abbot["Close"])

    print('Statistics=%.3f, p=%.3f' % (stat, p))

    alpha = 0.05

    if p > alpha:

        print('Abbot :: Sample looks Gaussian :: Shapiro-Wilk Test')

    else:

        print('Abbot :: Sample does not look Gaussian :: Shapiro-Wilk Test')

    stat, p = shapiro(mrf["Close"])

    print('Statistics=%.3f, p=%.3f' % (stat, p))

    alpha = 0.05

    if p > alpha:

        print('MRF :: Sample looks Gaussian :: Shapiro-Wilk Test')

    else:

        print('MRF :: Sample does not look Gaussian :: Shapiro-Wilk Test')

    stat, p = shapiro(shreecem["Close"])

    print('Statistics=%.3f, p=%.3f' % (stat, p))

    alpha = 0.05

    if p > alpha:

        print('Shree Cements :: Sample looks Gaussian :: Shapiro-Wilk Test')

    else:

        print('Shree Cements :: Sample does not look Gaussian :: Shapiro-Wilk Test')

    time.sleep(5)

def normal\_test\_ad():

    abbot = pd.read\_csv('ABBOTINDIA.NS.csv')

    mrf = pd.read\_csv('MRF.NS.csv')

    shreecem = pd.read\_csv('SHREECEM.NS.csv')

    result = anderson(abbot["Close"])

    print('Statistic: %.3f' % result.statistic)

    p = 0

    for i in range(len(result.critical\_values)):

        sl, cv = result.significance\_level[i], result.critical\_values[i]

        if result.statistic < result.critical\_values[i]:

            print('%.3f: %.3f, data looks normal :: Anderson-Darling Test :: Abbot' % (sl, cv))

        else:

            print('%.3f: %.3f, data does not look normal :: Anderson-Darling Test :: Abbot' % (sl, cv))

    result = anderson(mrf["Close"])

    print('Statistic: %.3f' % result.statistic)

    p = 0

    for i in range(len(result.critical\_values)):

        sl, cv = result.significance\_level[i], result.critical\_values[i]

        if result.statistic < result.critical\_values[i]:

            print('%.3f: %.3f, data looks normal :: Anderson-Darling Test  :: MRF' % (sl, cv))

        else:

            print('%.3f: %.3f, data does not look normal :: Anderson-Darling Test :: MRF' % (sl, cv))

    result = anderson(shreecem["Close"])

    print('Statistic: %.3f' % result.statistic)

    p = 0

    for i in range(len(result.critical\_values)):

        sl, cv = result.significance\_level[i], result.critical\_values[i]

        if result.statistic < result.critical\_values[i]:

            print('%.3f: %.3f, data looks normal :: Anderson-Darling Test :: Shree Cements' % (sl, cv))

        else:

            print('%.3f: %.3f, data does not look normal :: Anderson-Darling Test :: Shree Cements' % (sl, cv))

    time.sleep(5)

def distribution\_returns\_normdist():

    abbot = pd.read\_csv('ABBOTINDIA.NS.csv')

    mrf = pd.read\_csv('MRF.NS.csv')

    shreecem = pd.read\_csv('SHREECEM.NS.csv')

    fig = plt.figure(figsize=(15, 7))

    ax1 = fig.add\_subplot(1, 1, 1)

    abbot["Return"] = abbot["Close"].pct\_change().fillna(method='bfill')

    abbot['Return'].hist(bins=50, ax=ax1)

    plt.title("Abbot :: Distribution")

    plt.show()

    fig = plt.figure(figsize=(15, 7))

    ax1 = fig.add\_subplot(1, 1, 1)

    stats.probplot(abbot['Return'], dist='norm', plot=ax1)

    plt.title("Abbot :: Normal Plot")

    plt.show()

    fig = plt.figure(figsize=(15, 7))

    ax1 = fig.add\_subplot(1, 1, 1)

    mrf["Return"] = mrf["Close"].pct\_change().fillna(method='bfill')

    mrf['Return'].hist(bins=50, ax=ax1)

    plt.title("MRF :: Distribution")

    plt.show()

    fig = plt.figure(figsize=(15, 7))

    ax1 = fig.add\_subplot(1, 1, 1)

    stats.probplot(mrf['Return'], dist='norm', plot=ax1)

    plt.title("MRF :: Normal Plot")

    plt.show()

    fig = plt.figure(figsize=(15, 7))

    ax1 = fig.add\_subplot(1, 1, 1)

    shreecem["Return"] = shreecem["Close"].pct\_change().fillna(method='bfill')

    shreecem['Return'].hist(bins=50, ax=ax1)

    plt.title("Shree Cements :: Distribution")

    plt.show()

    fig = plt.figure(figsize=(15, 7))

    ax1 = fig.add\_subplot(1, 1, 1)

    stats.probplot(shreecem['Return'], dist='norm', plot=ax1)

    plt.title("Shree Cements :: Normal Plot")

    plt.show()

def plot\_the\_closing():

    abbot = pd.read\_csv('ABBOTINDIA.NS.csv')

    mrf = pd.read\_csv('MRF.NS.csv')

    shreecem = pd.read\_csv('SHREECEM.NS.csv')

    plt.plot(abbot["Close"])

    plt.plot(mrf["Close"])

    plt.plot(shreecem["Close"])

    plt.legend(["Abbot","MRF","Shree Cements"])

    plt.title("Closing price vs days from 1/1/2009")

    plt.show()

def plot\_the\_net\_returns():

    abbot = pd.read\_csv('ABBOTINDIA.NS.csv')

    mrf = pd.read\_csv('MRF.NS.csv')

    shreecem = pd.read\_csv('SHREECEM.NS.csv')

    abbot["Return"] = abbot["Close"].pct\_change().fillna(method='bfill')

    mrf["Return"] = mrf["Close"].pct\_change().fillna(method='bfill')

    shreecem["Return"] = shreecem["Close"].pct\_change().fillna(method='bfill')

    plt.plot(abbot["Return"])

    plt.show()

    plt.plot(mrf["Return"])

    plt.show()

    plt.plot(shreecem["Return"])

*#plt.legend(["Abbot","MRF","Shree Cements"])*

*#plt.title("Net-returns vs days from 1/1/2009")*

    plt.show()

def plot\_the\_gross\_returns():

    abbot = pd.read\_csv('ABBOTINDIA.NS.csv')

    mrf = pd.read\_csv('MRF.NS.csv')

    shreecem = pd.read\_csv('SHREECEM.NS.csv')

    abbot["Return"] = abbot["Close"].pct\_change().fillna(method='bfill')

    mrf["Return"] = mrf["Close"].pct\_change().fillna(method='bfill')

    shreecem["Return"] = shreecem["Close"].pct\_change().fillna(method='bfill')

    abbot["Gross"] = abbot["Return"] + 1

    mrf["Gross"] = mrf["Return"] + 1

    shreecem["Gross"] = shreecem["Return"] + 1

    plt.plot(abbot["Gross"])

    plt.plot(mrf["Gross"])

    plt.plot(shreecem["Gross"])

    plt.legend(["Abbot","MRF","Shree Cements"])

    plt.title("Gross-returns vs days from 1/1/2009")

    plt.show()

def plot\_the\_logret():

    abbot = pd.read\_csv('ABBOTINDIA.NS.csv')

    mrf = pd.read\_csv('MRF.NS.csv')

    shreecem = pd.read\_csv('SHREECEM.NS.csv')

    abbot["Return"] = abbot["Close"].pct\_change().fillna(method='bfill')

    mrf["Return"] = mrf["Close"].pct\_change().fillna(method='bfill')

    shreecem["Return"] = shreecem["Close"].pct\_change().fillna(method='bfill')

    abbot["Log"] = np.log(abbot["Return"])

    mrf["Log"] = np.log(mrf["Return"])

    shreecem["Log"] = np.log(shreecem["Return"])

    plt.plot(abbot["Log"])

    plt.plot(mrf["Log"])

    plt.plot(shreecem["Log"])

    plt.legend(["Abbot","MRF","Shree Cements"])

    plt.title("Log-returns vs days from 1/1/2009")

    plt.show()

def plot\_the\_adjclosing():

    abbot = pd.read\_csv('ABBOTINDIA.NS.csv')

    mrf = pd.read\_csv('MRF.NS.csv')

    shreecem = pd.read\_csv('SHREECEM.NS.csv')

    plt.plot(abbot["Adj Close"])

    plt.plot(mrf["Adj Close"])

    plt.plot(shreecem["Adj Close"])

    plt.legend(["Abbot","MRF","Shree Cements"])

    plt.title("Adjusted-Closing price vs days from 1/1/2009")

    plt.show()

def get\_mean\_median\_mode\_logret():

    abbot = pd.read\_csv('ABBOTINDIA.NS.csv')

    mrf = pd.read\_csv('MRF.NS.csv')

    shreecem = pd.read\_csv('SHREECEM.NS.csv')

    abbot["Return"] = abbot["Close"].pct\_change().fillna(method='bfill')

    mrf["Return"] = mrf["Close"].pct\_change().fillna(method='bfill')

    shreecem["Return"] = shreecem["Close"].pct\_change().fillna(method='bfill')

    abbot["Log"] = np.log(abbot["Return"])

    mrf["Log"] = np.log(mrf["Return"])

    shreecem["Log"] = np.log(shreecem["Return"])

    abbot\_r = list(abbot["Return"])

    mrf\_r = list(mrf["Return"])

    shreecem\_r = list(shreecem["Return"])

    abbot\_l = []

    mrf\_l = []

    shreecem\_l = []

    count\_abbot = 0

    count\_mrf = 0

    count\_shreecem = 0

    for i in abbot\_r:

        abbot\_l.append(np.log(i))

    for i in mrf\_r:

        mrf\_l.append(np.log(i))

    for i in shreecem\_r:

        shreecem\_l.append(np.log(i))

    abbot\_mean = statistics.mean(abbot\_l)

    abbot\_median = statistics.median(abbot\_l)

    abbot\_mode = statistics.mode(abbot\_l)

    mrf\_mean = statistics.mean(mrf\_l)

    mrf\_median = statistics.median(mrf\_l)

    mrf\_mode = statistics.mode(mrf\_l)

    shreecem\_mean = statistics.mean(shreecem\_l)

    shreecem\_median = statistics.median(shreecem\_l)

    shreecem\_mode = statistics.mode(shreecem\_l)

    print('Abbot :: mean=%.3f, median=%.3f, mode=%.3f' % (abbot\_mean, abbot\_median, abbot\_mode))

    print('MRF :: mean=%.3f, median=%.3f, mode=%.3f' % (mrf\_mean, mrf\_median, mrf\_mode))

    print('Shree Cements :: mean=%.3f, median=%.3f, mode=%.3f' % (shreecem\_mean, shreecem\_median, shreecem\_mode))

def correlation():

    abbot = pd.read\_csv('ABBOTINDIA.NS.csv')

    mrf = pd.read\_csv('MRF.NS.csv')

    shreecem = pd.read\_csv('SHREECEM.NS.csv')

    abbot["Return"] = abbot["Close"].pct\_change().fillna(method='bfill')

    mrf["Return"] = mrf["Close"].pct\_change().fillna(method='bfill')

    shreecem["Return"] = shreecem["Close"].pct\_change().fillna(method='bfill')

    abbot\_mrf = abbot["Return"].corr(mrf["Return"])

    abbot\_shreecem = abbot["Return"].corr(shreecem["Return"])

    mrf\_shreecem = mrf["Return"].corr(shreecem["Return"])

    abbot["Log"] = np.log(abbot["Return"])

    mrf["Log"] = np.log(mrf["Return"])

    shreecem["Log"] = np.log(shreecem["Return"])

    abbot\_mrf\_log = abbot["Log"].corr(mrf["Log"])

    abbot\_shreecem\_log = abbot["Log"].corr(shreecem["Log"])

    mrf\_shreecem\_log = mrf["Log"].corr(shreecem["Log"])

    print("abbot:mrf:")

    print(abbot\_mrf)

    print("abbot:shreecem:")

    print(abbot\_shreecem)

    print("mrf:shreecem:")

    print(mrf\_shreecem)

    print("abbot:mrf\_log:")

    print(abbot\_mrf\_log)

    print("abbot:shreecem\_log:")

    print(abbot\_shreecem\_log)

    print("mrf:shreecem\_log:")

    print(mrf\_shreecem\_log)

    time.sleep(10)

MAIN PYTHON FILE

from fac407\_prnvstsh\_module import plot\_the\_closing, plot\_the\_adjclosing, plot\_the\_net\_returns, plot\_the\_gross\_returns, plot\_the\_logret, distribution\_returns\_normdist, get\_mean\_median\_mode\_logret, normal\_test\_ad, normal\_test\_sw, correlation

import os

def main():

    while True:

        os.system('cls' if os.name == 'nt' else 'clear')

        var = input(" Hello Choose a number for the result/analysis \

        1. Plot the closing price \

        2. Plot the adjusted closing price \

        3. Plot the net returns \

        4. Plot the gross returns \

        5. Plot the log returns \

        6. Plot the Distribution of Returns \

        7. Statiscal information about the log-returns \

        8. Normal distribution test Shapiro-Wilk \

        9. Normal distribution test Anderson-Darling \

        10. Correlation \

        11. EXIT ")

        if var == "1":

            plot\_the\_closing()

        elif var == "2":

            plot\_the\_adjclosing()

        elif var == "3":

            plot\_the\_net\_returns()

        elif var == "4":

            plot\_the\_gross\_returns()

        elif var == "5":

            plot\_the\_logret()

        elif var == "6":

            distribution\_returns\_normdist()

        elif var == "7":

            get\_mean\_median\_mode\_logret()

        elif var == "8":

            normal\_test\_sw()

        elif var == "9":

            normal\_test\_ad()

        elif var == "10":

            correlation()

        elif var == "11":

            exit()

if \_\_name\_\_ == "\_\_main\_\_":

    main()