Assignment 2:

Installation

1. To install all the relevant python packages, use the command on the command prompt as 'pip install –r requirements.txt' after extracting the zip folder

2. To run the file, on a command prompt type 'python a1.py'

3. NOTE: Sometimes yahoo finance bugs out and returns an empty dataframe. This is a known bug. Please rerun the program if it does not give a successful run initially.

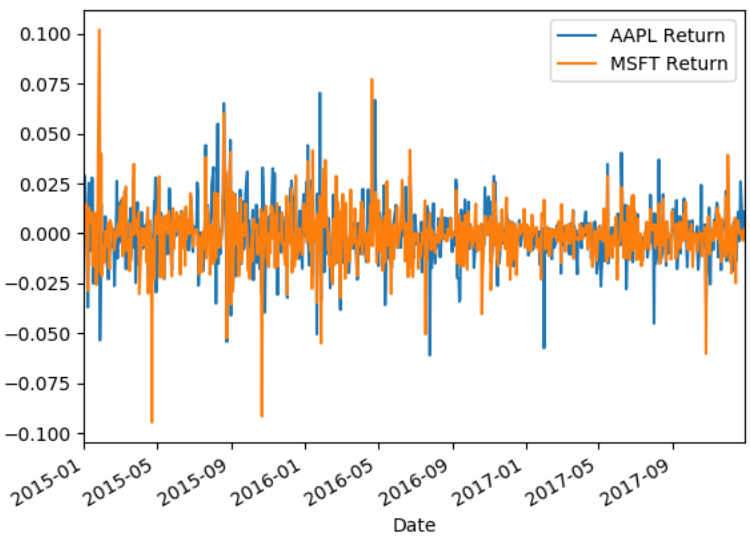
Explanation

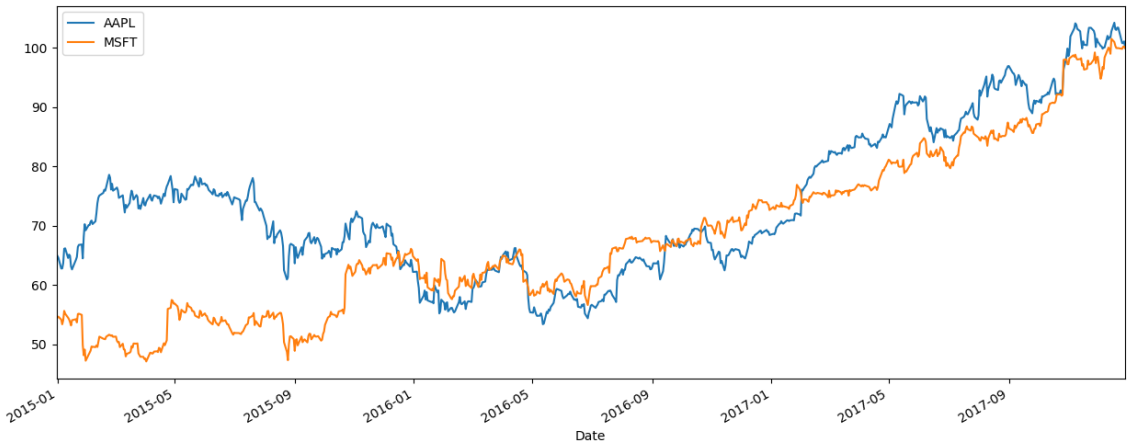
1. **Write a Python program to download the historical data of Apple Inc. (AAPL) and Microsoft Corp (MSFT) for the last 3 years**

   Yahoo Finance was used to fetch data for the past 3 years into a DataFrame for MSFT and AAPL

	AAPL	MSFT
2017-12-29T00:00:00.000000000	169.23000	85.54000
2017-12-28T00:00:00.000000000	171.08000	85.72000
2017-12-27T00:00:00.000000000	170.60001	85.71000
2017-12-26T00:00:00.000000000	170.57001	85.40000
2017-12-22T00:00:00.000000000	175.01000	85.51000
2017-12-21T00:00:00.000000000	175.01000	85.50000
2017-12-20T00:00:00.000000000	174.35001	85.52000
	174.53999	85.83000

2. **Calculate the daily return for AAPL and MSFT and present a comparative graphical analysis.**
   A graphical analysis in the form of line chart depicting daily returns is presented, along with a chart of prices

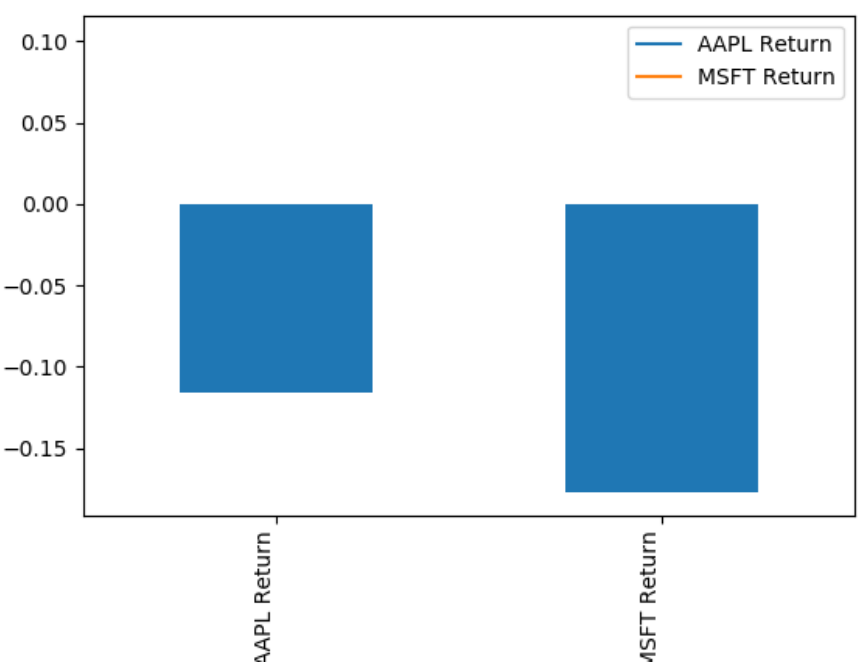




3. **Calculate and graphically represent the expected return for each of the stocks. To do this, we have to calculate the average of the daily returns of the period being analyzed and then annualize**

The expected return is calculated as follows:

```
weights = np.array([0.50, 0.50])  # weight of security in portfolio  weights:
# annual returns of each of the stocks and then calculate the dot product of these returns and the weigh
annual_returns = data2.mean() * 250
print('Annual Returns:\n')
print(annual_returns)
annual_returns.plot.bar()
plt.show()
```



4. **Calculate the Standard Deviation of the portfolio. The used stocks on the portfolio are correlated, so remember to use the appropriate Variance formula**

Portfolio volatility is calculated by the following:

```
pfolio_var = np.dot(weights.T, np.dot(data2.cov() * 250, weights))
print(pfolio_var)


print('Portfolio Volatility:\n')
pfolio_vol = (np.dot(weights.T, np.dot(data2.cov() * 250, weights))) ** 0.5
print (str(round(pfolio_vol, 5) * 100) + ' %')
```

Along with that, a covariance and correlational matrix are also calculated.

The portfolio volatility is:

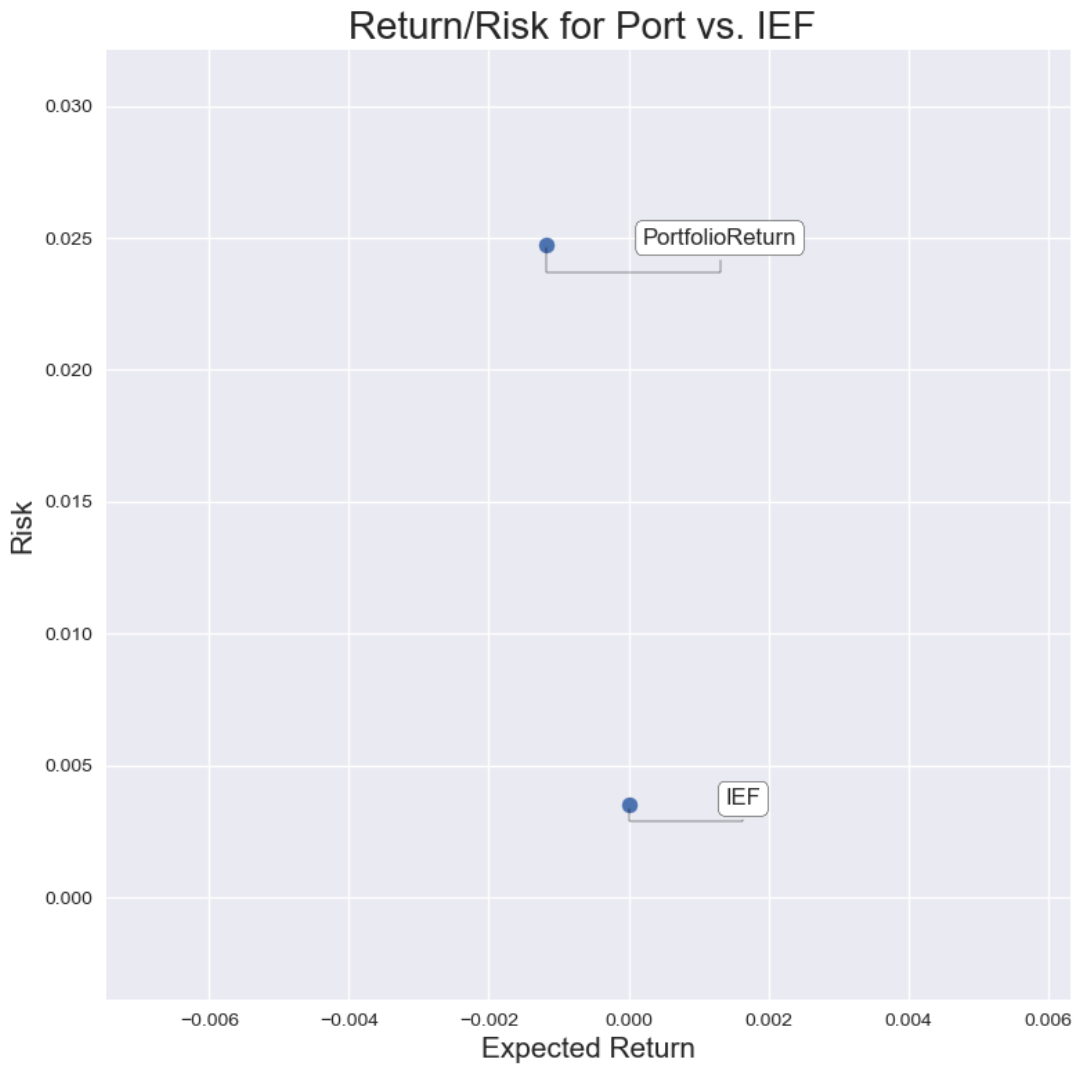
```
Portfolio Volatility:


19.574 %
```

5. **How does the return profile of the current portfolio compare to one that is consisting solely of Treasury Bonds ? How does the Risk Profile of the two compare?**

We use CAPM to calculate Portfolio expected returns

```
Assume a risk-free rate of 1.72% and a risk premium of 5%. Estimate the expected return of Portfolio
14.1926831128 %
```

We then take the IEF to calculate w.r.t. US treasury Bonds. Then we compare both the Risk/Return profiles for comparison



6. **Use Python to print a comparative analysis of the two portfolio in terms of all the major KPIs taught in this course**

   We then display some statistics w.r.t. KPIs using ffn library

Total Return	-68.23%	338.28%
Daily Sharpe	-	-1.39
Daily Sortino	-	-1.35
CAGR	-31.83%	63.85%
Max Drawdown	-306.67%	-271.36%
Calmar Ratio	-0.10	0.24
MTD	-25.84%	261.23%
3m	-231.94%	-194.65%
6m	-143.02%	-61.76%
YTD	-437.40%	-232.05%
1Y	-47.41%	672.36%
3Y (ann.)	-31.83%	63.85%
5Y (ann.)	-31.83%	63.85%
10Y (ann.)	-31.83%	63.85%
Since Incep. (ann.)	-31.83%	63.85%