In [1]:

```
import pandas as pd
import datetime as dt
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np
import yfinance as yf
import pandas_datareader as pdr
import datetime as dt
import pandas as pd
from sklearn.linear_model import LinearRegression
import scipy.stats
```

https://www.learnpythonwithrune.org/calculate-the-market-sp-500-beta-with-python-for-any-stock/ https://blog.devgenius.io/how-to-calculate-the-daily-returns-and-volatility-of-a-stock-with-python-d4e1de53e53b https://stackoverflow.com/questions/64506283/create-a-pandas-table

Table 1

Annualized Volatility (using trailing 3-months)

In [2]:

```
ticker= "MSFT"
MSFT = yf.download(ticker, start="2022-07-01", end="2022-10-01")

MSFT['daily_returns']=(MSFT['Close'].pct_change())*100

import math
daily_volatility_MSFT = MSFT['daily_returns'].std()
print(daily_volatility_MSFT)

annual_volatility_MSFT = math.sqrt(252) * daily_volatility_MSFT
print(annual_volatility_MSFT)
```

```
[*********************100%***********************]  1 of 1 completed
1.851706500880833
29.39492941447374
```

In [3]:

```
ticker= "AAPL"
AAPL = yf.download(ticker, start="2022-07-01", end="2022-10-01")

AAPL['daily_returns']=(AAPL['Close'].pct_change()) *100

import math
daily_volatility_AAPL = AAPL['daily_returns'].std()
print(daily_volatility_AAPL)

annual_volatility_AAPL = math.sqrt(252) * daily_volatility_AAPL
print(annual_volatility_AAPL)
```

```
[*********************100%***********************]  1 of 1 completed
1.9302554877252651
30.64185592403245
```

In [4]:

```
ticker= "AMZN"
AMZN = yf.download(ticker, start="2022-07-01", end="2022-10-01")

AMZN['daily_returns']=(AMZN['Close'].pct_change())* 100

import math
daily_volatility_AMZN = AMZN['daily_returns'].std()
print(daily_volatility_AMZN)

annual_volatility_AMZN = math.sqrt(252) * daily_volatility_AMZN
print(annual_volatility_AMZN)
```

```
[*********************100%***********************]  1 of 1 completed
2.7597201518278105
43.80919925921881
```

In [5]:

```
ticker= "GOOG"
GOOG = yf.download(ticker, start="2022-07-01", end="2022-10-01")

GOOG['daily_returns']=(GOOG['Close'].pct_change())*100

import math
daily_volatility_GOOG = GOOG['daily_returns'].std()
print(daily_volatility_GOOG)

annual_volatility_GOOG = math.sqrt(252) * daily_volatility_GOOG
print(annual_volatility_GOOG)
```

```
[*********************100%***********************]  1 of 1 completed
2.3390790615485137
37.13172896265428
```

In [6]:

```
ticker= "NFLX"
NFLX = yf.download(ticker, start="2022-07-01", end="2022-10-01")

NFLX['daily_returns']=(NFLX['Close'].pct_change())*100

import math
daily_volatility_NFLX = NFLX['daily_returns'].std()
print(daily_volatility_NFLX)

annual_volatility_NFLX = math.sqrt(252) * daily_volatility_NFLX
print(annual_volatility_NFLX)
```

```
[*********************100%***********************]  1 of 1 completed
3.3093532039715403
52.53435346910104
```

In [7]:

```
ticker= "ACLS"
ACLS = yf.download(ticker, start="2022-07-01", end="2022-10-01")

ACLS['daily_returns']=(ACLS['Close'].pct_change())*100

import math
daily_volatility_ACLS = ACLS['daily_returns'].std()
print(daily_volatility_ACLS)

annual_volatility_ACLS = math.sqrt(252) * daily_volatility_ACLS
print(annual_volatility_ACLS)
```

```
[*********************100%***********************]  1 of 1 completed
4.289293768617285
68.09042767116217
```

In [8]:

```
ticker= "TSLA"
TSLA = yf.download(ticker, start="2022-07-01", end="2022-10-01")

TSLA['daily_returns']=(TSLA['Close'].pct_change())*100

import math
daily_volatility_TSLA = TSLA['daily_returns'].std()
print(daily_volatility_TSLA)

annual_volatility_TSLA = math.sqrt(252) * daily_volatility_TSLA
print(annual_volatility_TSLA)
```

```
[*********************100%***********************]  1 of 1 completed
3.0928442835287693
49.0973809083892
```

Beta against SPY (using trailing 12-months)

In [9]:

```
tickers = ['MSFT', 'AAPL', 'AMZN', 'GOOG', 'NFLX', 'ACLS', 'TSLA', 'SPY',
start = dt.datetime(2021, 10, 1)
end = dt.datetime(2022, 10, 1)

data = pdr.get_data_yahoo(tickers, start, end, interval="d")



returns_data = data['Adj Close'].resample('W').ffill().pct_change()

returns_data[1:]
```

Out[9]:

Symbols	MSFT	AAPL	AMZN	GOOG	NFLX	ACLS	TSLA
Date							
2021-10-10	0.019889	0.001752	0.001633	0.026333	0.031819	-0.013177	0.013248
2021-10-17	0.031745	0.013576	0.036611	0.011560	-0.006907	0.046418	0.073254
2021-							

10-24	0.016272	0.026581	-0.021552	-0.021528	0.058078	-0.001620	0.079060
2021-10-31	0.072649	0.007465	0.011057	0.069580	0.038404	0.114425	0.224607
2021-11-07	0.013389	0.011354	0.043458	0.006546	-0.064594	0.111597	0.097029
2021-11-14	0.001964	-0.008527	0.001751	0.002710	0.057130	0.005732	-0.154383
2021-11-21	0.020841	0.070405	0.042954	0.002052	-0.005582	0.018727	0.100288
2021-11-28	-0.039142	-0.023295	-0.046786	-0.047658	-0.019387	0.004316	-0.048493
2021-12-05	-0.020232	0.032077	-0.032749	-0.001999	-0.095412	-0.013847	-0.061881
2021-12-12	0.060463	0.108811	0.016063	0.043183	0.015827	0.065203	0.002030
2021-12-19	-0.054709	-0.046308	-0.012743	-0.039496	-0.040758	-0.042273	-0.083046
2021-12-26	0.033632	0.030034	0.006182	0.030388	0.046631	0.077045	0.144150
2022-01-02	0.004870	0.007318	-0.025437	-0.016739	-0.018971	0.095182	-0.009578
2022-01-09	-0.066247	-0.030411	-0.024970	-0.053048	-0.101886	-0.078997	-0.028218
2022-01-16	-0.012228	0.005227	-0.002559	0.020306	-0.028407	0.039610	0.022055
2022-01-23	-0.045680	-0.061594	-0.120237	-0.069352	-0.243851	-0.159686	-0.100714
2022-01-30	0.041313	0.048766	0.009359	0.024579	-0.033057	-0.015003	-0.103348
2022-02-06	-0.007526	0.013384	0.094886	0.072973	0.067151	0.064309	0.090943
2022-02-13	-0.035628	-0.021753	-0.027569	-0.062133	-0.045981	0.029575	-0.068579
2022-02-20	-0.022081	-0.007946	-0.004514	-0.027306	-0.000051	0.043243	-0.003512
2022-02-27	0.032577	-0.014644	0.007778	0.031058	-0.001252	0.027979	-0.054972
2022-03-06	-0.025058	-0.010191	-0.052979	-0.017823	-0.074386	-0.112471	0.035092
2022-03-13	-0.033775	-0.051725	-0.000800	-0.012462	-0.059188	0.028395	-0.051223
2022-03-20	0.072696	0.059781	0.108064	0.048484	0.118359	0.270590	0.138354

2022-03-27	0.010818	0.065496	0.021848	0.034503	-0.017735	-0.025084	0.116248
2022-04-03	0.018901	-0.002347	-0.007365	-0.005805	-0.001016	-0.088651	0.073171
2022-04-10	-0.040237	-0.024210	-0.055634	-0.047544	-0.047099	-0.174004	-0.054491
2022-04-17	-0.057716	-0.028220	-0.017830	-0.050425	-0.041447	-0.045178	-0.039483
2022-04-24	-0.020727	-0.021175	-0.048492	-0.060030	-0.368217	-0.028354	0.020355
2022-05-01	0.012736	-0.025589	-0.139027	-0.038854	-0.116741	-0.006931	-0.133615
2022-05-08	-0.010053	-0.000881	-0.076512	0.006032	-0.049328	0.068871	-0.005868
2022-05-15	-0.049539	-0.064662	-0.014964	0.007397	0.036857	0.007732	-0.110969
2022-05-22	-0.030529	-0.064714	-0.048330	-0.061816	-0.006875	-0.040921	-0.137333
2022-05-29	0.081881	0.087579	0.070224	0.031890	0.047438	0.109156	0.144193
2022-06-05	-0.011784	-0.028468	0.062559	0.015647	0.019417	-0.006411	-0.073825
2022-06-12	-0.063069	-0.056748	-0.103801	-0.027378	-0.080611	-0.066946	-0.009751
2022-06-19	-0.021108	-0.040618	-0.031281	-0.031967	-0.040614	-0.100795	-0.066615
2022-06-26	0.080961	0.076771	0.096404	0.098943	0.087402	0.119016	0.133542
2022-07-03	-0.030333	-0.019272	-0.059248	-0.079780	-0.057113	-0.176117	-0.075062
2022-07-10	0.031127	0.058375	0.054582	0.101645	0.039066	0.090094	0.103404
2022-07-17	-0.040873	0.021287	-0.017223	-0.061593	0.011392	0.075187	-0.042656
2022-07-24	0.014179	0.026104	0.078115	-0.039081	0.165671	0.132562	0.134032
2022-07-31	0.078276	0.054643	0.102353	0.076412	0.020232	0.104949	0.091487
2022-08-07	0.007730	0.018889	0.043349	0.013546	0.008359	0.111617	-0.030220
2022-08-14	0.031812	0.040822	0.019531	0.037473	0.099303	-0.016756	0.041156
2022-	-0.017651	-0.003370	-0.037060	-0.036934	-0.032651	-0.034474	-0.011210

08-21							
2022-08-28	-0.063114	-0.046059	-0.054113	-0.057738	-0.074142	-0.026004	-0.028910
2022-09-04	-0.044873	-0.047733	-0.024780	-0.023540	0.012675	-0.123115	-0.062064
2022-09-11	0.032805	0.010012	0.045173	0.028524	0.032993	0.035021	0.109063
2022-09-18	-0.074567	-0.042384	-0.073085	-0.072911	0.028086	-0.066453	0.012246
2022-09-25	-0.027866	-0.001792	-0.078928	-0.043038	-0.057136	-0.063837	-0.092369
2022-10-02	-0.021100	-0.081300	-0.006855	-0.030453	0.039883	0.056156	-0.036611

In [10]:

```
cov = returns_data.cov()
var_SPY= returns_data['SPY'].var()

BETA_SPY_MSFT = cov.loc['MSFT', 'SPY']/var_SPY
BETA_SPY_AAPL = cov.loc['AAPL', 'SPY']/var_SPY
BETA_SPY_AMZN = cov.loc['AMZN', 'SPY']/var_SPY
BETA_SPY_GOOG = cov.loc['GOOG', 'SPY']/var_SPY
BETA_SPY_NFLX = cov.loc['NFLX', 'SPY']/var_SPY
BETA_SPY_ACLS = cov.loc['ACLS', 'SPY']/var_SPY
BETA_SPY_TSLA = cov.loc['TSLA', 'SPY']/var_SPY

print(BETA_SPY_MSFT)
print(BETA_SPY_AAPL)
print(BETA_SPY_AMZN)
print(BETA_SPY_GOOG)
print(BETA_SPY_NFLX)
print(BETA_SPY_ACLS)
print(BETA_SPY_TSLA)
```

```
1.1954446612431402
1.1979781787555914
1.4862992252892226
1.156317186503966
1.5918756445795306
2.008856135835389
2.07523391083857
```

Beta against IWM (using trailing 12-months)

In [11]:

```
cov = returns_data.cov()
var_IWM = returns_data['IWM'].var()

BETA_IWM_MSFT = cov.loc['MSFT', 'IWM']/var_IWM
BETA_IWM_AAPL = cov.loc['AAPL', 'IWM']/var_IWM
BETA_IWM_AMZN = cov.loc['AMZN', 'IWM']/var_IWM
BETA_IWM_GOOG = cov.loc['GOOG', 'IWM']/var_IWM
BETA_IWM_NFLX = cov.loc['NFLX', 'IWM']/var_IWM
BETA_IWM_ACLS = cov.loc['ACLS', 'IWM']/var_IWM
BETA_IWM_TSLA = cov.loc['TSLA', 'IWM']/var_IWM

print(BETA_IWM_MSFT)
print(BETA_IWM_AAPL)
print(BETA_IWM_AMZN)
print(BETA_IWM_GOOG)
print(BETA_IWM_NFLX)
print(BETA_IWM_ACLS)
print(BETA_IWM_TSLA)
```

```
0.8729749080438319
0.780800424499311
1.2694870793985422
0.8573503278664563
1.386672962616075
1.8308145618350335
1.632842310491568
```

Beta against DIA (using trailing 12-months)

In [12]:

```
cov = returns_data.cov()
var_DIA = returns_data['DIA'].var()

BETA_DIA_MSFT = cov.loc['MSFT', 'DIA']/var_DIA
BETA_DIA_AAPL = cov.loc['AAPL', 'DIA']/var_DIA
BETA_DIA_AMZN = cov.loc['AMZN', 'DIA']/var_DIA
BETA_DIA_GOOG = cov.loc['GOOG', 'DIA']/var_DIA
BETA_DIA_NFLX = cov.loc['NFLX', 'DIA']/var_DIA
BETA_DIA_ACLS = cov.loc['ACLS', 'DIA']/var_DIA
BETA_DIA_TSLA = cov.loc['TSLA', 'DIA']/var_DIA

print(BETA_DIA_MSFT)
print(BETA_DIA_AAPL)
print(BETA_DIA_AMZN)
print(BETA_DIA_GOOG)
print(BETA_DIA_NFLX)
print(BETA_DIA_ACLS)
print(BETA_DIA_TSLA)
```

```
1.3285846110806225
1.364533689630722
1.6310609727970924
1.2159602349687109
1.6868964041178118
2.174587057021318
2.1776286050565883
```

Average Weekly Drawdown (52-week Low minus 52-week High) / 52-week High

In [13]:

```
data_week = pdr.get_data_yahoo(tickers, start, end, interval="w")

data_week = data_week['Adj Close']
```

In [14]:

```
data_weekly_returns = data_week

data_weekly_returns = data_weekly_returns[1:]

data_weekly_returns
weekly_drawdowns = (data_weekly_returns[1:].min()-data_weekly_returns[1:]
weekly_drawdowns
```

Out[14]:

```
Symbols
MSFT   -0.315637
AAPL   -0.264858
AMZN   -0.422179
GOOG   -0.358797
NFLX   -0.745752
ACLS   -0.404570
TSLA   -0.467895
SPY    -0.239272
IWM    -0.312754
DIA    -0.197648
dtype: float64
```

In [16]:

```
avg_wkl_drawdown_MSFT = weekly_drawdowns['MSFT']
avg_wkl_drawdown_AAPL = weekly_drawdowns['AAPL']
avg_wkl_drawdown_AMZN = weekly_drawdowns['AMZN']
avg_wkl_drawdown_GOOG = weekly_drawdowns['GOOG']
avg_wkl_drawdown_NFLX = weekly_drawdowns['NFLX']
avg_wkl_drawdown_ACLS = weekly_drawdowns['ACLS']
avg_wkl_drawdown_TSLA = weekly_drawdowns['TSLA']

print(avg_wkl_drawdown_MSFT)
```

```
-0.31563713626840245
```

Maximum Weekly Drawdown (52-week Low minus 52-week High) / 52-week High

In [17]:

```
max_drawdown_MSFT = (data_weekly_returns.MSFT.min() - data_weekly_returns
max_drawdown_AAPL = (data_weekly_returns.AAPL.min() - data_weekly_returns
max_drawdown_AMZN = (data_weekly_returns.AMZN.min() - data_weekly_returns
max_drawdown_GOOG = (data_weekly_returns.GOOG.min() - data_weekly_returns
max_drawdown_NFLX = (data_weekly_returns.NFLX.min() - data_weekly_returns
max_drawdown_ACLS = (data_weekly_returns.ACLS.min() - data_weekly_returns
max_drawdown_TSLA = (data_weekly_returns.TSLA.min() - data_weekly_returns
```

Total Return (using trailing 10-years)

In [18]:

```
data_ten_years = pdr.get_data_yahoo(tickers, start = 2012-10-1, end = 202

data_ten_years_returns = data_ten_years['Adj Close'].resample('D').ffill(
```

In [19]:

```
total_ret_MSFT = data_ten_years_returns.MSFT.sum()
total_ret_AAPL = data_ten_years_returns.AAPL.sum()
total_ret_AMZN = data_ten_years_returns.AMZN.sum()
total_ret_GOOG = data_ten_years_returns.GOOG.sum()
total_ret_NFLX = data_ten_years_returns.NFLX.sum()
total_ret_ACLS = data_ten_years_returns.ACLS.sum()
total_ret_TSLA = data_ten_years_returns.TSLA.sum()
total_ret_SPY = data_ten_years_returns.SPY.sum()
total_ret_IWM = data_ten_years_returns.IWM.sum()
total_ret_DIA = data_ten_years_returns.DIA.sum()

print(total_ret_MSFT)
```

```
98.54458587924918
```

Annualized Total Return (using trailing 10-years)

In [20]:

```
annualized_ret_MSFT = data_ten_years_returns.MSFT.mean() * 252
annualized_ret_AAPL = data_ten_years_returns.AAPL.mean() * 252
annualized_ret_AMZN = data_ten_years_returns.AMZN.mean() * 252
annualized_ret_GOOG = data_ten_years_returns.GOOG.mean() * 252
annualized_ret_NFLX = data_ten_years_returns.NFLX.mean() * 252
annualized_ret_ACLS = data_ten_years_returns.ACLS.mean() * 252
annualized_ret_TSLA = data_ten_years_returns.TSLA.mean() * 252
annualized_ret_SPY = data_ten_years_returns.SPY.mean() * 252
annualized_ret_IWM = data_ten_years_returns.IWM.mean() * 252
annualized_ret_DIA = data_ten_years_returns.DIA.mean() * 252
```

Table

In [21]:

```
d = {
    'Tickers' : ['MSFT', 'AAPL', 'AMZN', 'GOOG', 'NFLX', 'ACLS', 'TSLA'],
    'Portfolio Weight in %' : [100/7,100/7,100/7,100/7,100/7,100/7,100/7,
    'Annualized Volatility':[annual_volatility_MSFT,annual_volatility_AAP
    'Beta against SPY':[BETA_SPY_MSFT,BETA_SPY_AAPL,BETA_SPY_AMZN,BETA_SP
    'Beta against IWM':[BETA_IWM_MSFT,BETA_IWM_AAPL,BETA_IWM_AMZN,BETA_IW
    'Beta against DIA':[BETA_DIA_MSFT,BETA_DIA_AAPL,BETA_DIA_AMZN,BETA_DI
    'Average Weekly Drawdown':[avg_wkl_drawdown_MSFT, avg_wkl_drawdown_AA
    'Maximum Weekly Drawdown':[max_drawdown_MSFT,max_drawdown_AAPL,max_dr
    'Total Return in %':[total_ret_MSFT,total_ret_AAPL,total_ret_AMZN,tot
    'Annualized Total Return in %':[annualized_ret_MSFT,annualized_ret_AA

}
df = pd.DataFrame(data=d)
df
```

Out[21]:

	Tickers	Portfolio Weight in %	Annualized Volatility	Beta against SPY	Beta against IWM	Beta against DIA	Average Weekly Drawdown	Maximum Weekly Drawdown
0	MSFT	14.285714	29.394929	1.195445	0.872975	1.328585	-0.315637	-0.315637
1	AAPL	14.285714	30.641856	1.197978	0.780800	1.364534	-0.264858	-0.264858
2	AMZN	14.285714	43.809199	1.486299	1.269487	1.631061	-0.422179	-0.422179
3	GOOG	14.285714	37.131729	1.156317	0.857350	1.215960	-0.358797	-0.358797
4	NFLX	14.285714	52.534353	1.591876	1.386673	1.686896	-0.745752	-0.745752
5	ACLS	14.285714	68.090428	2.008856	1.830815	2.174587	-0.404570	-0.414131
6	TSLA	14.285714	49.097381	2.075234	1.632842	2.177629	-0.467895	-0.467895

Table 2

Portfolio Return

In [22]:

```
ticker = ['MSFT', 'AAPL', 'AMZN', 'GOOG', 'NFLX', 'ACLS', 'TSLA']
start = dt.datetime(2021, 10, 1)
end = dt.datetime(2022, 10, 1)

data_port_cor = pdr.get_data_yahoo(ticker, start, end, interval="d")

data_port_cor = data_port_cor['Adj Close']

portfolio_007 = data_port_cor.pct_change()[1:]

portfolio_007
```

Out[22]:

Symbols	MSFT	AAPL	AMZN	GOOG	NFLX	ACLS	TSLA
Date							
2021-10-04	-0.020719	-0.024606	-0.028472	-0.019767	-0.015983	-0.047898	0.008140
2021-10-05	0.019957	0.014159	0.009788	0.018032	0.052142	0.009446	-0.001203
2021-10-06	0.015064	0.006307	0.012732	0.008643	0.006758	-0.001523	0.002767
2021-10-07	0.005936	0.009084	0.012391	0.013334	-0.011344	0.034438	0.013874
2021-10-08	0.000000	-0.002722	-0.004182	0.006254	0.001282	-0.005900	-0.010232
...	...	...	...	...	...	...	...
2022-09-26	-0.001975	0.002260	0.012041	-0.003630	-0.010335	0.004011	0.002470
2022-09-27	-0.004380	0.006566	-0.006426	-0.007287	0.001294	0.036477	0.025108
2022-09-28	0.019712	-0.012652	0.031466	0.027016	0.092886	0.048768	0.017212
2022-09-29	-0.014809	-0.049119	-0.027201	-0.026305	-0.022390	0.000639	-0.068101
2022-09-30	-0.019368	-0.030039	-0.015679	-0.019778	-0.017813	-0.032897	-0.011036

251 rows × 7 columns

In [23]:

```
w = [1/7,1/7,1/7,1/7,1/7,1/7,1/7]

weighted_returns = (w * portfolio_007)

port_ret_007 = weighted_returns.sum(axis=1)

port_ret_007
```

Out[23]:

```
Date
2021-10-04   -0.021329
2021-10-05    0.017474
2021-10-06    0.007250
2021-10-07    0.011102
2021-10-08   -0.002214
                ...
2022-09-26    0.000692
2022-09-27    0.007336
2022-09-28    0.032058
2022-09-29   -0.029612
2022-09-30   -0.020944
Length: 251, dtype: float64
```

Correlation against ETF

In [24]:

```
tickers = ['MSFT', 'AAPL', 'AMZN', 'GOOG', 'NFLX', 'ACLS', 'TSLA']
tickers1 = ['SPY', 'IWM', 'DIA']


data_week = pdr.get_data_yahoo(tickers, start, end, interval="w")
data_week_adj = data_week['Adj Close']
data_week_adj['portfolio']= data_week_adj.mean(axis=1)

data_week_ETF = pdr.get_data_yahoo(tickers1, start, end, interval="w")
data_week_ETF_adj = data_week_ETF['Adj Close']
#print(data_week_adj['portfolio'])
correlation_port_SPY = scipy.stats.linregress(data_week_ETF_adj['SPY'], d
correlation_port_IWM = scipy.stats.linregress(data_week_ETF_adj['IWM'], d
correlation_port_DIA = scipy.stats.linregress(data_week_ETF_adj['DIA'], d
corr_port_SPY = correlation_port_SPY.rvalue
corr_port_IWM = correlation_port_IWM.rvalue
corr_port_DIA = correlation_port_DIA.rvalue
print(corr_port_SPY)
print(corr_port_IWM)
print(corr_port_DIA)
```

```
/var/folders/z2/909fbvdx5n1f_z6b8_t_fkmm0000gn/T/ipykernel_40081/21539209
23.py:7: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-do
cs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  data_week_adj['portfolio']= data_week_adj.mean(axis=1)
0.9219327912471224
0.9610673617905101
0.883185389796672
```

Covariance of Portfolio against ETF

In [25]:

```
tickers = ['MSFT', 'AAPL', 'AMZN', 'GOOG', 'NFLX', 'ACLS', 'TSLA']
tickers1 = ['SPY', 'IWM', 'DIA']


data_week = pdr.get_data_yahoo(tickers, start, end, interval="w")
data_week_adj = data_week['Adj Close']
data_week_adj['portfolio']= data_week_adj.mean(axis=1)

data_week_ETF = pdr.get_data_yahoo(tickers1, start, end, interval="w")
data_week_ETF_adj = data_week_ETF['Adj Close']

np_a = np.array([data_week_adj['portfolio'], data_week_ETF_adj['SPY']])
np_b = np.array([data_week_adj['portfolio'], data_week_ETF_adj['IWM']])
np_c = np.array([data_week_adj['portfolio'], data_week_ETF_adj['DIA']])

cov_SPY_port = np.cov(np_a)
cov_IWM_port = np.cov(np_b)
cov_DIA_port = np.cov(np_c)
covariance_SPY_port = cov_SPY_port[0,1]
covariance_IWM_port = cov_IWM_port[0,1]
covariance_DIA_port = cov_DIA_port[0,1]
print(covariance_SPY_port)
print(covariance_IWM_port)
print(covariance_DIA_port)
```

```
/var/folders/z2/909fbvdx5n1f_z6b8_t_fkmm0000gn/T/ipykernel_40081/14534145
05.py:7: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-do
cs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  data_week_adj['portfolio']= data_week_adj.mean(axis=1)
1095.9105756199872
749.3867992975183
618.2532386605703
```

Tracking Errors (using trailing 10-years)

In [26]:

```
annualized_ret_port = ((1/7 * annualized_ret_MSFT) + (1/7 * annualized_re
print(annualized_ret_port)

total_ret_SPY = data_ten_years_returns.SPY.sum()
annualized_ret_SPY = data_ten_years_returns.SPY.mean() * 252
annualized_std_SPY = data_ten_years_returns.SPY.std() * math.sqrt(252)
print(annualized_std_SPY)
print(annualized_ret_SPY)


total_ret_IWM = data_ten_years_returns.IWM.sum()
annualized_ret_IWM = data_ten_years_returns.IWM.mean() * 252
annualized_std_IWM = data_ten_years_returns.IWM.std() * math.sqrt(252)
print(annualized_std_IWM)
print(annualized_ret_IWM)


total_ret_DIA = data_ten_years_returns.DIA.sum()
annualized_ret_DIA = data_ten_years_returns.DIA.mean() * 252
annualized_std_DIA = data_ten_years_returns.DIA.std() * math.sqrt(252)
print(annualized_std_DIA)
print(annualized_ret_DIA)

tracking_err_SPY = annualized_ret_port - annualized_ret_SPY
tracking_err_IWM = annualized_ret_port - annualized_ret_IWM
tracking_err_DIA = annualized_ret_port - annualized_ret_DIA

print(tracking_err_SPY)
print(tracking_err_IWM)
print(tracking_err_DIA)
```

```
26.507945490175796
18.151865913627198
2.746791005313539
21.754458532491597
6.841131178999487
17.23180396052415
3.621983257678736
23.761154484862256
19.66681431117631
22.88596223249706
```

Sharpe Ratio (using current risk-free rate)

In [27]:

```
#https://www.investopedia.com/terms/s/sharperatio.asp

risk_free_rate = 0.0334*100
data_week = pdr.get_data_yahoo(tickers, start, end, interval="w")
data_week_adj = data_week['Adj Close']
data_week_adj['portfolio']= data_week_adj.mean(axis=1)

np_data_port = np.array(data_week_adj['portfolio'])
std_port = np.std(np_data_port)

sharpe_SPY = (annualized_ret_SPY-risk_free_rate)/ annualized_std_SPY
sharpe_IWM = (annualized_ret_IWM-risk_free_rate)/  annualized_std_IWM
sharpe_DIA = ( annualized_ret_DIA-risk_free_rate)/ annualized_std_DIA
sharpe_port = (annualized_ret_port-risk_free_rate)/ std_port

print(sharpe_SPY)
print(sharpe_IWM)
print(sharpe_DIA)
print(sharpe_port)
```

```
-0.03268033146064172
0.160938557664873
0.016364117089813893
0.5821716437192509
```

```
/var/folders/z2/909fbvdx5n1f_z6b8_t_fkmm0000gn/T/ipykernel_40081/41341598
17.py:6: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-do
cs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  data_week_adj['portfolio']= data_week_adj.mean(axis=1)
```

Annualized Volatility (252 days) Spread (Portfolio Volatility – ETF Volatility)

In [28]:

```
spread_SPY = (std_port - annualized_std_SPY)
spread_DIA =  (std_port - annualized_std_DIA)
spread_IWM = (std_port -  annualized_std_IWM)
spread_port = 0

np_data_port = np.array(data_week_adj['portfolio'])
std_port = np.std(np_data_port)
print(std_port)
```

```
39.79572990220804
```

In [29]:

```
d = {
    'Tickers' : ['SPY','IWM','DIA','Portfolio'],
    'Correlation against ETF':[corr_port_SPY,corr_port_IWM,corr_port_DIA,
    'Covariance of Portfolio against ETF' :[covariance_SPY_port,covarianc
    'Tracking Errors':[tracking_err_SPY,tracking_err_IWM,tracking_err_DIA
    'Sharpe Ratio':[sharpe_SPY,sharpe_IWM,sharpe_DIA,sharpe_port],
    'Annualized Volatility (252 days) Spread ':[spread_SPY,spread_IWM,spr

}
df = pd.DataFrame(data=d)
df
```

Out[29]:

	Tickers	Correlation against ETF	Covariance of Portfolio against ETF	Tracking Errors	Sharpe Ratio	Annualized Volatility (252 days) Spread
0	SPY	0.921933	1095.910576	23.761154	-0.032680	21.643864
1	IWM	0.961067	749.386799	19.666814	0.160939	18.041271
2	DIA	0.883185	618.253239	22.885962	0.016364	22.563926
3	Portfolio	1	1	0	0.582172	0.000000

Correlation Matrix

In [30]:

```
portfolio_007 ['Portfolio Return'] = port_ret_007

portfolio_007
```

Out[30]:

Symbols	MSFT	AAPL	AMZN	GOOG	NFLX	ACLS	TSLA
Date							
2021-10-04	-0.020719	-0.024606	-0.028472	-0.019767	-0.015983	-0.047898	0.008140
2021-10-05	0.019957	0.014159	0.009788	0.018032	0.052142	0.009446	-0.001203
2021-10-06	0.015064	0.006307	0.012732	0.008643	0.006758	-0.001523	0.002767
2021-10-07	0.005936	0.009084	0.012391	0.013334	-0.011344	0.034438	0.013874
2021-10-08	0.000000	-0.002722	-0.004182	0.006254	0.001282	-0.005900	-0.010232
...	...	...	...	...	...	...	...
2022-09-26	-0.001975	0.002260	0.012041	-0.003630	-0.010335	0.004011	0.002470
2022-09-27	-0.004380	0.006566	-0.006426	-0.007287	0.001294	0.036477	0.025108
2022-09-28	0.019712	-0.012652	0.031466	0.027016	0.092886	0.048768	0.017212
2022-09-29	-0.014809	-0.049119	-0.027201	-0.026305	-0.022390	0.000639	-0.068101
2022-09-30	-0.019368	-0.030039	-0.015679	-0.019778	-0.017813	-0.032897	-0.011036

251 rows × 8 columns

In [31]:

```
corr_data = portfolio_007[1:].corr(method='pearson')

corr_data
```

Out[31]:

Symbols	MSFT	AAPL	AMZN	GOOG	NFLX	ACLS	TSLA	Portf Re
Symbols								
MSFT	1.000000	0.781685	0.717617	0.815703	0.491669	0.622195	0.588690	0.839
AAPL	0.781685	1.000000	0.693581	0.738541	0.453638	0.646168	0.645906	0.836
AMZN	0.717617	0.693581	1.000000	0.712540	0.576695	0.580936	0.550923	0.832
GOOG	0.815703	0.738541	0.712540	1.000000	0.523805	0.639043	0.548395	0.837
NFLX	0.491669	0.453638	0.576695	0.523805	1.000000	0.426847	0.464509	0.726
ACLS	0.622195	0.646168	0.580936	0.639043	0.426847	1.000000	0.578879	0.808
TSLA	0.588690	0.645906	0.550923	0.548395	0.464509	0.578879	1.000000	0.795
Portfolio Return	0.839288	0.836562	0.832043	0.837877	0.726203	0.808883	0.795283	1.000