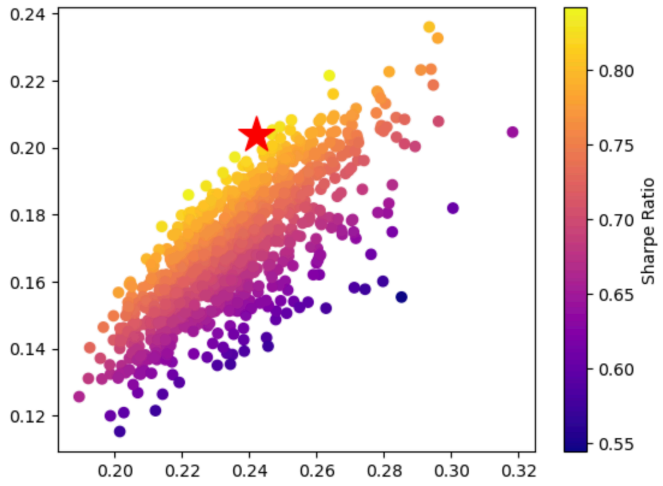
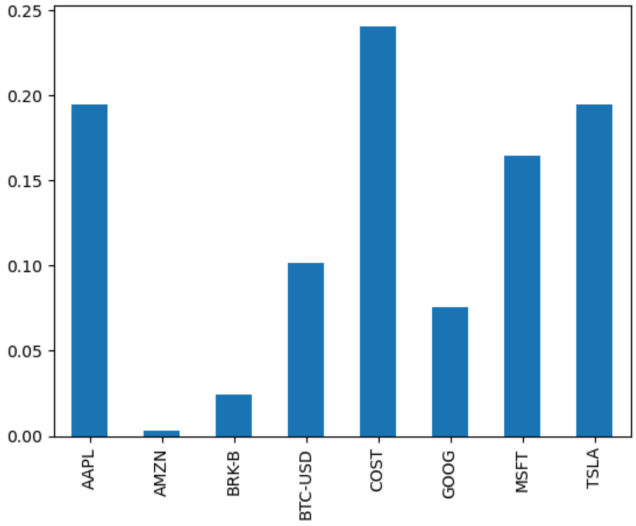
```
# This Python program optimizes an equity portfolio by finding the allocation that results in the maximum
# Sharpe ratio using Monte Carlo Simulation. It displays the best Sharpe ratios on an Efficient Frontier chart,
# and the best allocation numbers on a bar chart.

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import yfinance as yf

# create an equity portfolio
stocks = ['AAPL', 'AMZN', 'MSFT', 'GOOG', 'TSLA', 'BTC-USD', 'BRK-B', 'COST']

# get portfolio price data since 1/1/2018 from Yahoo Finance
data = yf.download(stocks, start='2018-01-01')

# find daily returns
data = data['Close']
x = data.pct_change()

p_weights = []
p_returns = []
p_risk = []
p_sharpe = []

# Monte Carlo Simulation
count = 1000
for k in range(count):
    # generate a random allocation
    wts = np.random.uniform(size = len(x.columns))
    wts = wts/np.sum(wts)
    p_weights.append(wts)

    # calculate annual returns
    mean_ret = (x.mean() * wts).sum()*252
    p_returns.append(mean_ret)

    # calculate volatility
    ret = (x * wts).sum(axis = 1)
    annual_std = np.std(ret) * np.sqrt(252)
    p_risk.append(annual_std)

    #calculate Sharpe ratio
    sharpe = (np.mean(ret) / np.std(ret))*np.sqrt(252)
    p_sharpe.append(sharpe)

# find the index of the portfolio with the maximum Sharpe ratio
max_ind = np.argmax(p_sharpe)
```

```
# create an Efficient Frontier chart showing the returns on the Y-axis and volatility on the X-axis.
plt.scatter(p_risk, p_returns, c=p_sharpe, cmap='plasma')
plt.colorbar(label='Sharpe Ratio')

# add a color bar for the Sharpe ratio, a red star marker to the chart that
# showing the most efficient portfolio with the best Sharpe ratio.
plt.scatter(p_risk[max_ind], p_returns[max_ind], color='r', marker='*', s=500)
plt.show()

# display the best allocation using a bar chart
s = pd.Series(p_weights[max_ind], index=x.columns)
s.plot(kind='bar')

# display the best allocation numbers
p_weights[max_ind]
```

```
[**********100%**********]  8 of 8 completed
```






```
array([0.07697208, 0.05355683, 0.09458436, 0.11446065, 0.33803085,
       0.00110767, 0.10307572, 0.21821184])
```