

uaspraktikum-g-211-19-0136

June 12, 2023

```
[5]: #import dataset
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
from google.colab import files
uploaded = files.upload()
```

<IPython.core.display.HTML object>

Saving Stock.csv to Stock.csv

```
[6]: df = pd.read_csv("Stock.csv")
```

```
[7]: df
```

```
[7]:
```

	Stock Name	Syymbol	Shares	Purchase Price	Cost Basis	\
0	Apple	AAPL	100	90.0	9,000.00	
1	Microsoft	MSFT	200	32.0	6,400.00	
2	Salesforce	CRM	150	25.0	3,750.00	
3	Oracle	ORCL	250	50.0	12,500.00	
4	Hewlett Packard Enterprise	HPE	500	18.0	9,000.00	
5	Alphabet	GOOG	100	225.0	22,500.00	
6	Intel	INTC	200	22.0	4,400.00	
7	Cisco	CSCO	225	18.0	4,050.00	
8	Qualcomm	QCOM	185	65.0	12,025.00	
9	Amazon	AMZN	50	800.0	40,000.00	
10	Redhat	RHT	100	95.0	9,500.00	
11	Facebook	FB	1000	17.0	17,000.00	
12	Twitter	TWTR	500	45.0	22,500.00	

	Current Price	Market Value	Gain/Loss	Devidend/share	Annual Yield
0	144.13	14,413.27	14,269.14	2.28	0.02
1	65.57	13,114.14	13,048.57	1.56	0.02
2	82.57	12,385.50	12,302.57	-	-
3	44.56	11,138.75	11,094.20	0.64	0.01
4	17.69	8,842.50	8,824.82	0.26	0.01

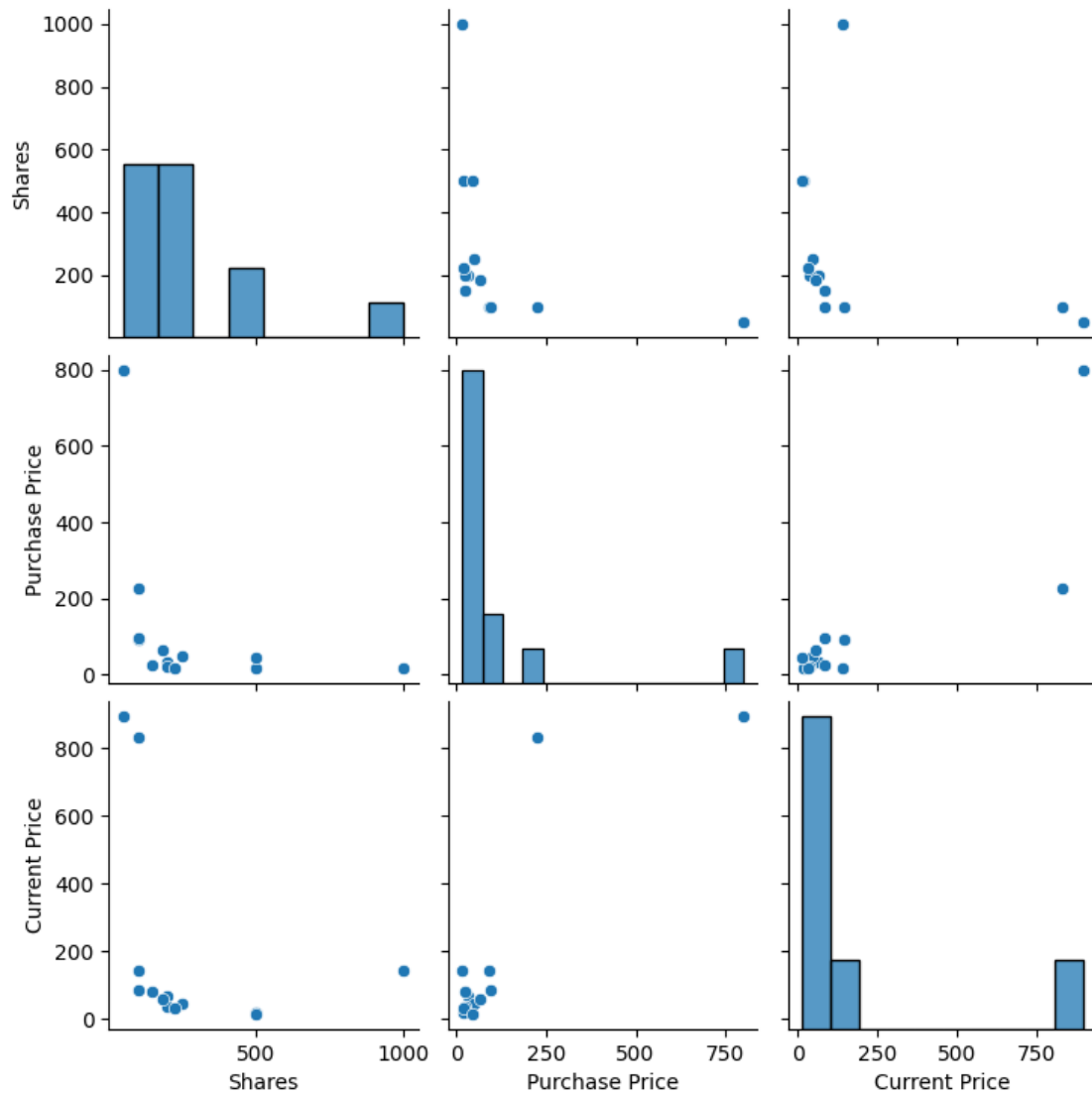
5	833.36	83,336.00	82,502.64	-	-
6	36.07	7,213.00	7,176.94	1.09	0.03
7	33.24	7,478.78	7,445.54	1.16	0.03
8	56.48	10,447.88	10,391.40	2.12	0.04
9	897.64	44,882.00	43,984.36	-	-
10	86.26	8,626.00	8,539.74	-	-
11	141.64	141,640.00	141,498.36	-	-
12	14.61	7,302.55	7,287.94	-	-

```
[8]: df.columns
```

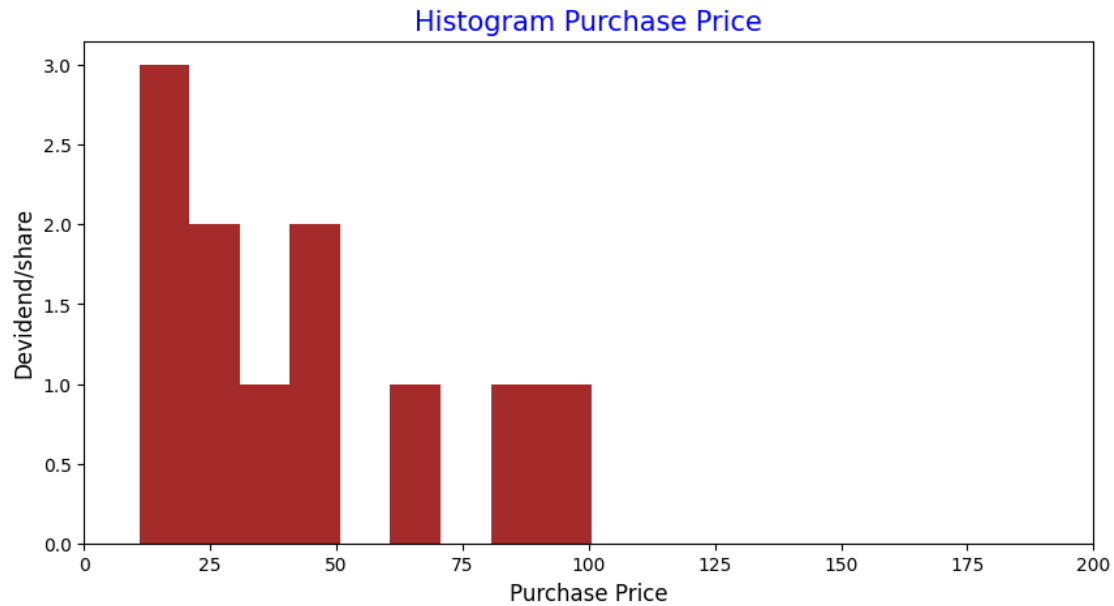
```
[8]: Index(['Stock Name', 'Symbol', 'Shares', 'Purchase Price', 'Cost Basis',
          'Current Price', 'Market Value', 'Gain/Loss', 'Devidend/share',
          'Annual Yield'],
          dtype='object')
```

```
[9]: sns.pairplot(df)
```

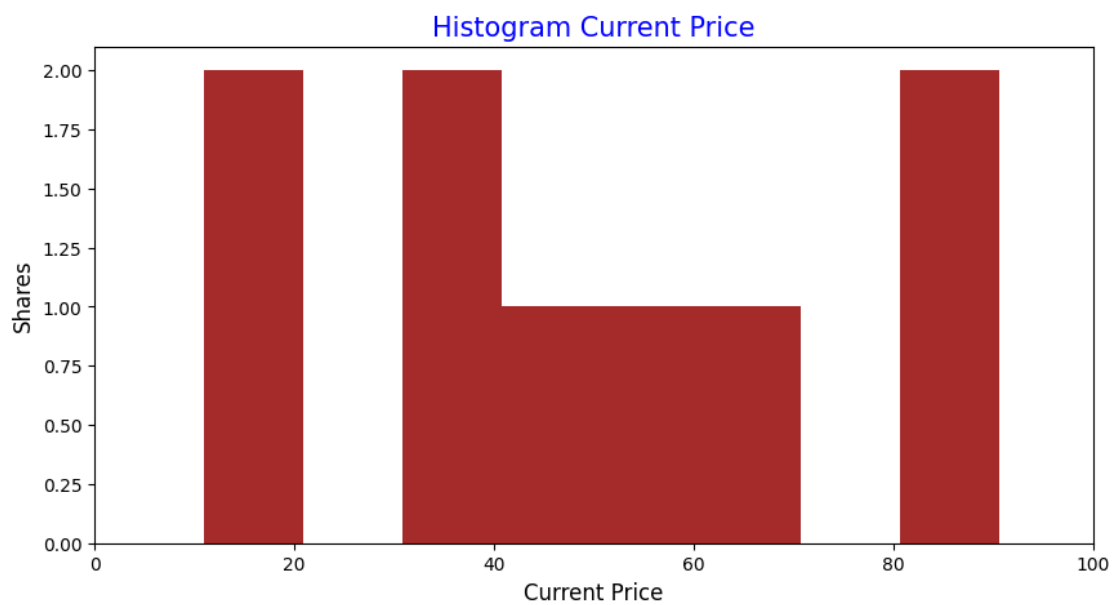
```
[9]: <seaborn.axisgrid.PairGrid at 0x7f5992c2eb60>
```



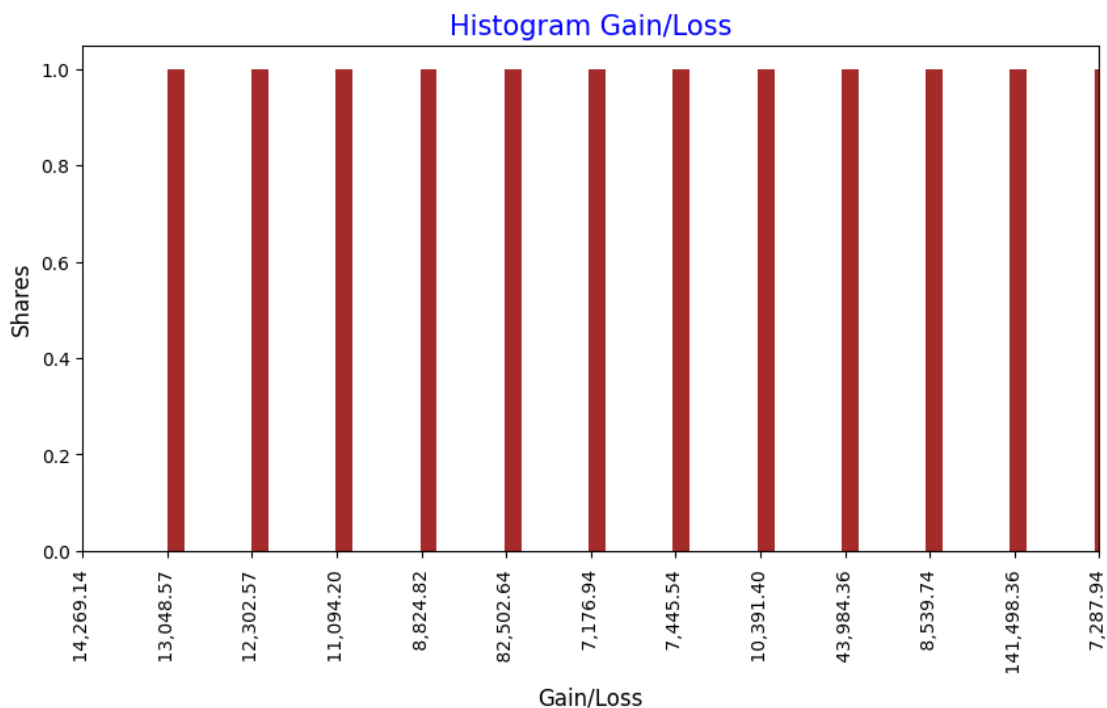
```
[53]: #membuat histogram Purchase Price
import matplotlib.pyplot as plt
plt.figure(figsize=(10,5))
plt.hist(df['Purchase Price'], bins=20, range=(1,200), color='brown')
plt.title('Histogram Purchase Price', fontsize=15, color='blue')
plt.xlabel('Purchase Price', fontsize = 12)
plt.ylabel('Devidend/share', fontsize = 12)
plt.xlim(xmin=0,xmax=200)
plt.show()
```



```
[44]: #membuat histogram Current Price
import matplotlib.pyplot as plt
plt.figure(figsize=(10,5))
plt.hist(df['Current Price'], bins=20, range=(1,200), color='brown')
plt.title('Histogram Current Price', fontsize=15, color='blue')
plt.xlabel('Current Price', fontsize = 12)
plt.ylabel('Shares', fontsize = 12)
plt.xlim(xmin=0,xmax=100)
plt.show()
```



```
[56]: #membuat histogram Gain/Loss
import matplotlib.pyplot as plt
plt.figure(figsize=(10,5))
plt.hist(df['Gain/Loss'], bins=1000, range=(1,200), color='brown')
plt.title('Histogram Gain/Loss', fontsize=15, color='blue')
plt.xlabel('Gain/Loss', fontsize = 12)
plt.ylabel('Shares', fontsize = 12)
plt.xlim(xmin=0,xmax=12)
plt.xticks(rotation=90)
plt.show()
```



```
[10]: print (df.columns.values.tolist())
```

```
['Stock Name', 'Symbol', 'Shares', 'Purchase Price', 'Cost Basis', 'Current
Price', 'Market Value', 'Gain/Loss', 'Devidend/share', 'Annual Yield']
```

```
[19]: #Devidend/share 5 tertinggi
top_devidends = (df.groupby('Stock Name')['Devidend/share']
                 .sum()
                 .reset_index()
                 .sort_values(by='Devidend/share', ascending=False)
                 .head(5))
```

```
print(top_devidends)
```

	Stock Name	Devidend/share
2	Apple	2.28
9	Qualcomm	2.12
7	Microsoft	1.56
3	Cisco	1.16
6	Intel	1.09

```
[12]: #market value 5 tertinggi
top_marketvalues = (df.groupby('Stock Name')['Market Value']
                    .sum()
                    .reset_index()
                    .sort_values(by='Market Value', ascending=False)
                    .head(5))
print(top_marketvalues)
```

	Stock Name	Market Value
0	Alphabet	83,336.00
5	Hewlett Packard Enterprise	8,842.50
10	Redhat	8,626.00
3	Cisco	7,478.78
12	Twitter	7,302.55

```
[13]: #Annual yield 7 tertinggi
top_annualyields = (df.groupby('Stock Name')['Annual Yield']
                    .sum()
                    .reset_index()
                    .sort_values(by='Annual Yield', ascending=False)
                    .head(7))
print(top_annualyields)
```

	Stock Name	Annual Yield
9	Qualcomm	0.04
3	Cisco	0.03
6	Intel	0.03
2	Apple	0.02
7	Microsoft	0.02
5	Hewlett Packard Enterprise	0.01
8	Oracle	0.01