```
!pip install yfinance --upgrade --no-cache-dir
```

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

sns.set(style='whitegrid')
```

```
acoes = ['CSMG3.SA', 'KLBN11.SA', 'PETR4.SA', 'VALE3.SA', 'TAEE11.SA', 'BBAS3.SA']

ydata = yf.download(acoes, '2018-01-01', '2023-12-31', '1d');
dados = ydata["Adj Close"]
```

```
[**********100%%**********]  6 of 6 completed
```

```
dados = dados *100 / dados.iloc[0]
display(dados)
```

| Ticker | BBAS3.SA | CSMG3.SA | KLBN11.SA | PETR4.SA | TAEE11.SA | VALE3.SA |
|---|---|---|---|---|---|---|
| Date | | | | | | |
| **2018-01-02** | 100.000000 | 100.000000 | 100.000000 | 100.000000 | 100.000000 | 100.000000 |
| **2018-01-03** | 101.275424 | 97.628277 | 98.736247 | 100.906351 | 100.139765 | 99.400775 |
| **2018-01-04** | 102.247215 | 96.123144 | 98.901095 | 101.087604 | 98.555430 | 99.808248 |
| **2018-01-05** | 102.247215 | 96.898488 | 97.967017 | 101.691857 | 99.021428 | 101.366232 |
| **2018-01-08** | 102.490169 | 97.491436 | 99.725279 | 102.900307 | 98.555430 | 103.619363 |
| **...** | ... | ... | ... | ... | ... | ... |
| **2023-12-21** | 235.461052 | 240.959337 | 153.854229 | 603.164896 | 311.920020 | 281.420937 |
| **2023-12-22** | 236.111589 | 242.038597 | 156.027106 | 608.966190 | 314.174863 | 279.300310 |
| **2023-12-26** | 238.280153 | 239.917524 | 155.813387 | 618.745453 | 314.842972 | 280.287519 |
| **2023-12-27** | 237.933198 | 238.149933 | 156.454572 | 619.242683 | 315.344037 | 282.993127 |
| **2023-12-28** | 240.231849 | 241.331565 | 158.306960 | 617.253699 | 320.521864 | 282.261850 |

1487 rows × 6 columns

```
nova_ordem = [ 'BBAS3.SA', 'PETR4.SA', 'CSMG3.SA', 'TAEE11.SA', 'KLBN11.SA', 'VALE3.SA']
dados = dados[nova_ordem]
```

```
dados.plot(figsize = (15,5));
```


```
dados_chg = dados.pct_change()
dados_chg = dados_chg.fillna(0)
```

```
ret_acc = (dados.iloc[-1] / dados.iloc[0])-1
print("Retorno acumulado:\n", ret_acc)
```

```
Retorno acumulado:
 Ticker
BBAS3.SA     1.402318
PETR4.SA     5.172537
CSMG3.SA     1.413316
TAEE11.SA    2.205219
KLBN11.SA    0.583070
VALE3.SA     1.822618
dtype: float64
```

```
ret_aa = ((dados.iloc[-1]/dados.iloc[0])**(1/5))-1
print("Retorno anualizado:\n", ret_aa)
```

```
Retorno anualizado:
 Ticker
BBAS3.SA     0.191588
PETR4.SA     0.439106
CSMG3.SA     0.192677
TAEE11.SA    0.262326
KLBN11.SA    0.096226
VALE3.SA     0.230638
dtype: float64
```

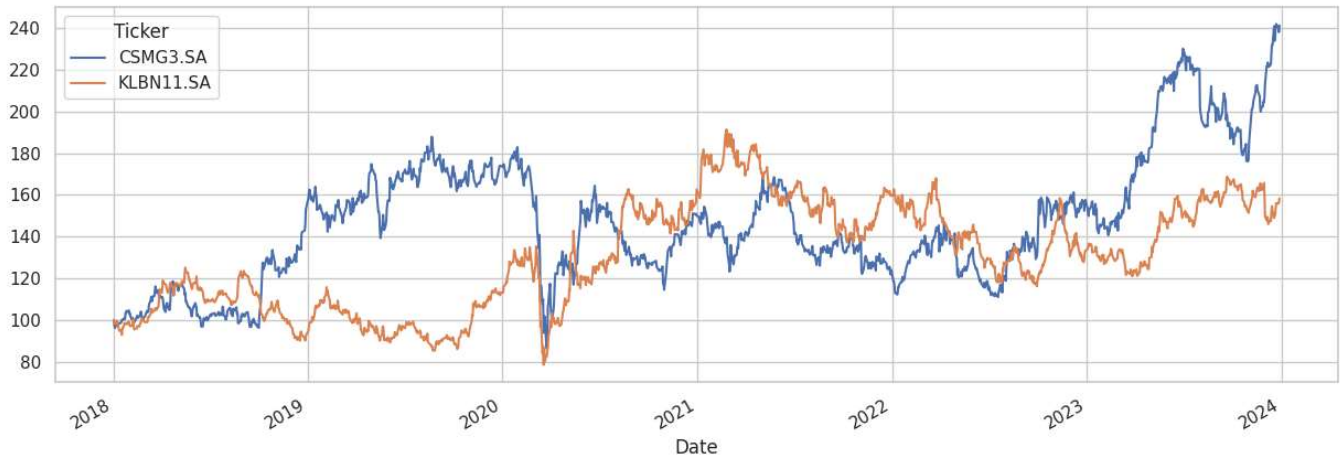
```
vol_aa = dados_chg.std()*np.sqrt(252)
print("Vol anualizada:\n", vol_aa)
```

```
Vol anualizada:
 Ticker
BBAS3.SA     0.390788
PETR4.SA     0.463584
CSMG3.SA     0.365302
TAEE11.SA    0.205062
KLBN11.SA    0.303409
VALE3.SA     0.390549
dtype: float64
```

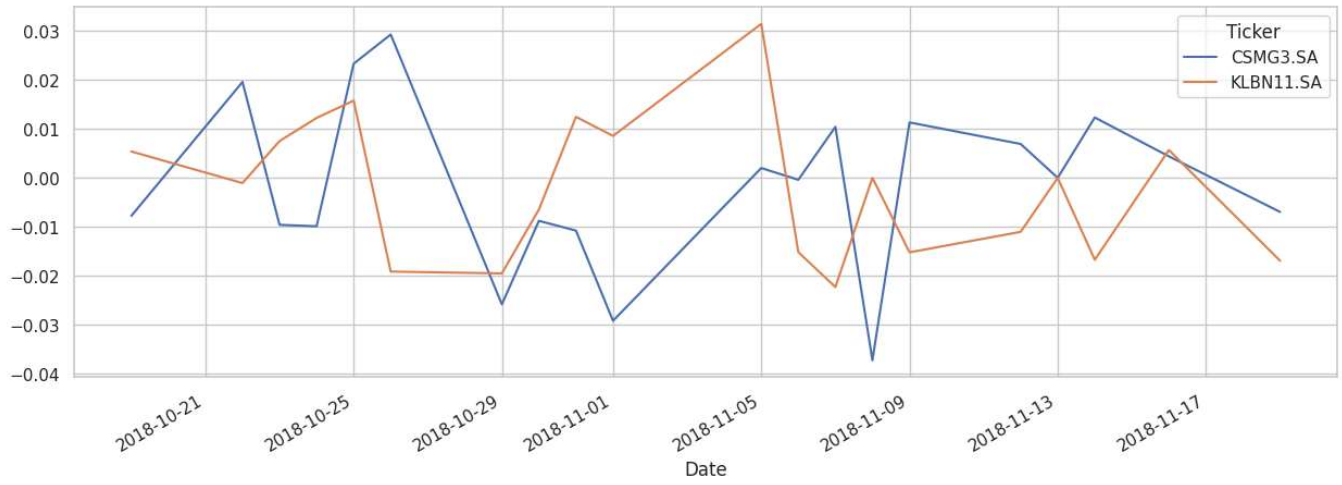
```
#Calculo da correlação dos dados
dados_chg.corr()
```

| Ticker | BBAS3.SA | PETR4.SA | CSMG3.SA | TAEE11.SA | KLBN11.SA | VALE3.SA |
|---|---|---|---|---|---|---|
| **Ticker** | | | | | | |
| **BBAS3.SA** | 1.000000 | 0.640989 | 0.484127 | 0.439334 | 0.176719 | 0.354287 |
| **PETR4.SA** | 0.640989 | 1.000000 | 0.362899 | 0.337913 | 0.239288 | 0.421012 |
| **CSMG3.SA** | 0.484127 | 0.362899 | 1.000000 | 0.438240 | 0.170870 | 0.213889 |
| **TAEE11.SA** | 0.439334 | 0.337913 | 0.438240 | 1.000000 | 0.195351 | 0.192620 |
| **KLBN11.SA** | 0.176719 | 0.239288 | 0.170870 | 0.195351 | 1.000000 | 0.297905 |
| **VALE3.SA** | 0.354287 | 0.421012 | 0.213889 | 0.192620 | 0.297905 | 1.000000 |

```
dados[['CSMG3.SA', 'KLBN11.SA']].plot(figsize = (15,5));
```




```
dados_chg[['CSMG3.SA', 'KLBN11.SA']].iloc[200:220].plot(figsize = (15,5));
```




```
port_pesos = [0, 0, 0.5, 0, 0.5, 0]
dados['PORT1'] = dados.dot(port_pesos)
```

```
display(dados)
```

| Ticker | BBAS3.SA | PETR4.SA | CSMG3.SA | TAEE11.SA | KLBN11.SA | VALE3.SA | PORT1 |
|---|---|---|---|---|---|---|---|
| Date | | | | | | | |
| **2018-01-02** | 100.000000 | 100.000000 | 100.000000 | 100.000000 | 100.000000 | 100.000000 | 100.000000 |
| **2018-01-03** | 101.275424 | 100.906351 | 97.628277 | 100.139765 | 98.736247 | 99.400775 | 98.182262 |
| **2018-01-04** | 102.247215 | 101.087604 | 96.123144 | 98.555430 | 98.901095 | 99.808248 | 97.512120 |
| **2018-01-05** | 102.247215 | 101.691857 | 96.898488 | 99.021428 | 97.967017 | 101.366232 | 97.432753 |
| **2018-01-08** | 102.490169 | 102.900307 | 97.491436 | 98.555430 | 99.725279 | 103.619363 | 98.608357 |
| **...** | ... | ... | ... | ... | ... | ... | ... |
| **2023-12-21** | 235.461052 | 603.164896 | 240.959337 | 311.920020 | 153.854229 | 281.420937 | 197.406783 |
| **2023-12-22** | 236.111589 | 608.966190 | 242.038597 | 314.174863 | 156.027106 | 279.300310 | 199.032852 |
| **2023-12-26** | 238.280153 | 618.745453 | 239.917524 | 314.842972 | 155.813387 | 280.287519 | 197.865456 |
| **2023-12-27** | 237.933198 | 619.242683 | 238.149933 | 315.344037 | 156.454572 | 282.993127 | 197.302252 |
| **2023-12-28** | 240.231849 | 617.253699 | 241.331565 | 320.521864 | 158.306960 | 282.261850 | 199.819262 |

1487 rows × 7 columns

```
dados[['CSMG3.SA', 'KLBN11.SA', 'PORT1']].plot(figsize = (15,5));
```


```
dados_chg = dados.pct_change()
dados_chg = dados_chg.fillna(0)
```

```
ret_acc = (dados.iloc[dados.count()[0]-1] / dados.iloc[0])-1
print("Retorno acumulado:\n", ret_acc)
ret_aa = ((dados.iloc[-1]/dados.iloc[0])**(1/5))-1
print("Ret aa:\n", ret_aa)
```

```
Retorno acumulado:
 Ticker
BBAS3.SA     1.402318
PETR4.SA     5.172537
CSMG3.SA     1.413316
TAEE11.SA    2.205219
KLBN11.SA    0.583070
VALE3.SA     1.822618
PORT1        0.998193
dtype: float64
Ret aa:
 Ticker
BBAS3.SA     0.191588
PETR4.SA     0.439106
CSMG3.SA     0.192677
TAEE11.SA    0.262326
KLBN11.SA    0.096226
VALE3.SA     0.230638
PORT1        0.148491
dtype: float64
```

```
vol_aa = dados_chg.std()*np.sqrt(252)
print("Vol aa:\n", vol_aa)
```

```
Vol aa:
 Ticker
BBAS3.SA     0.390788
PETR4.SA     0.463584
CSMG3.SA     0.365302
TAEE11.SA    0.205062
KLBN11.SA    0.303409
VALE3.SA     0.390549
PORT1        0.260533
dtype: float64
```

```
#Calcular drawdown
ddown = pd.DataFrame()

for ativo in dados.columns:
  list = []
  for ind in range(dados.count()[0]):
    list.append((dados[ativo].iloc[ind]/dados[ativo].iloc[:ind+1].max()-1)*100)
  ddown[ativo]=list

ddown['Data']=dados.index.values
ddown.set_index(keys = 'Data', inplace = True)

#display(ddown)
print(ddown.min())
```

```
BBAS3.SA    -58.286996
PETR4.SA    -63.356055
CSMG3.SA    -54.029209
TAEE11.SA   -23.694037
KLBN11.SA   -42.009959
VALE3.SA    -43.724133
PORT1       -46.258119
dtype: float64
```

```
ddown[['CSMG3.SA', 'KLBN11.SA', 'PORT1']].plot(figsize = (15,5));
```


```
dados=dados.drop(['PORT1'], axis=1)
dados_chg=dados_chg.drop(['PORT1'], axis=1)
```

```
dados
```

| Ticker | BBAS3.SA | PETR4.SA | CSMG3.SA | TAEE11.SA | KLBN11.SA | VALE3.SA |
|---|---|---|---|---|---|---|
| Date | | | | | | |
| **2018-01-02** | 100.000000 | 100.000000 | 100.000000 | 100.000000 | 100.000000 | 100.000000 |
| **2018-01-03** | 101.275424 | 100.906351 | 97.628277 | 100.139765 | 98.736247 | 99.400775 |
| **2018-01-04** | 102.247215 | 101.087604 | 96.123144 | 98.555430 | 98.901095 | 99.808248 |
| **2018-01-05** | 102.247215 | 101.691857 | 96.898488 | 99.021428 | 97.967017 | 101.366232 |
| **2018-01-08** | 102.490169 | 102.900307 | 97.491436 | 98.555430 | 99.725279 | 103.619363 |
| **...** | ... | ... | ... | ... | ... | ... |
| **2023-12-21** | 235.461052 | 603.164896 | 240.959337 | 311.920020 | 153.854229 | 281.420937 |
| **2023-12-22** | 236.111589 | 608.966190 | 242.038597 | 314.174863 | 156.027106 | 279.300310 |
| **2023-12-26** | 238.280153 | 618.745453 | 239.917524 | 314.842972 | 155.813387 | 280.287519 |
| **2023-12-27** | 237.933198 | 619.242683 | 238.149933 | 315.344037 | 156.454572 | 282.993127 |
| **2023-12-28** | 240.231849 | 617.253699 | 241.331565 | 320.521864 | 158.306960 | 282.261850 |

1487 rows × 6 columns

```
def calc_ret_vol(ativos, ativos_chg, port_pesos):
  port = ativos.dot(port_pesos)
  port_chg =port.pct_change()
  port_chg = port_chg.fillna(0)
  ret = ((port.iloc[-1]/port.iloc[0])**(1/5))-1
  vol = port_chg.std()*np.sqrt(252)
  return ret, vol
```

```
port_pesos
```

```
[0, 0, 0.5, 0, 0.5, 0]
```

```
dados
```

| Ticker | BBAS3.SA | PETR4.SA | CSMG3.SA | TAEE11.SA | KLBN11.SA | VALE3.SA |
|---|---|---|---|---|---|---|
| Date | | | | | | |
| **2018-01-02** | 100.000000 | 100.000000 | 100.000000 | 100.000000 | 100.000000 | 100.000000 |
| **2018-01-03** | 101.275424 | 100.906351 | 97.628277 | 100.139765 | 98.736247 | 99.400775 |
| **2018-01-04** | 102.247215 | 101.087604 | 96.123144 | 98.555430 | 98.901095 | 99.808248 |
| **2018-01-05** | 102.247215 | 101.691857 | 96.898488 | 99.021428 | 97.967017 | 101.366232 |
| **2018-01-08** | 102.490169 | 102.900307 | 97.491436 | 98.555430 | 99.725279 | 103.619363 |
| **...** | ... | ... | ... | ... | ... | ... |
| **2023-12-21** | 235.461052 | 603.164896 | 240.959337 | 311.920020 | 153.854229 | 281.420937 |
| **2023-12-22** | 236.111589 | 608.966190 | 242.038597 | 314.174863 | 156.027106 | 279.300310 |
| **2023-12-26** | 238.280153 | 618.745453 | 239.917524 | 314.842972 | 155.813387 | 280.287519 |
| **2023-12-27** | 237.933198 | 619.242683 | 238.149933 | 315.344037 | 156.454572 | 282.993127 |
| **2023-12-28** | 240.231849 | 617.253699 | 241.331565 | 320.521864 | 158.306960 | 282.261850 |

1487 rows × 6 columns

```
#Portfolio com 2 ativos: KLBN11 e TAEE11
points = []
min_vol_ret = [100, 0] #[vol, ret]
port_pesos = [0, 0, 0, 0, 0, 0]
for w in range(0, 101, 5):
  ret, vol = calc_ret_vol(dados, dados_chg, [0, 0,  w/100, 0, (1-w/100), 0])
  print(f"Aloc:{round(w/100, 2):.2f} {round(1-(w/100),2):.2f} Ret:{round(ret, 3):.3f} Vol:{round(vol, 3):.3f}")
  points.append([ret, vol])
  if vol < min_vol_ret[0]:
    min_vol_ret[0] = vol
    min_vol_ret[1] = ret
    port_pesos[2] = w/100
    port_pesos[4] = 1-w/100
```

```
Aloc:0.00 1.00 Ret:0.096 Vol:0.303
Aloc:0.05 0.95 Ret:0.102 Vol:0.291
Aloc:0.10 0.90 Ret:0.107 Vol:0.280
Aloc:0.15 0.85 Ret:0.113 Vol:0.271
Aloc:0.20 0.80 Ret:0.118 Vol:0.264
Aloc:0.25 0.75 Ret:0.124 Vol:0.259
Aloc:0.30 0.70 Ret:0.129 Vol:0.256
Aloc:0.35 0.65 Ret:0.134 Vol:0.254
Aloc:0.40 0.60 Ret:0.139 Vol:0.254
Aloc:0.45 0.55 Ret:0.144 Vol:0.257
Aloc:0.50 0.50 Ret:0.148 Vol:0.261
Aloc:0.55 0.45 Ret:0.153 Vol:0.266
Aloc:0.60 0.40 Ret:0.158 Vol:0.273
Aloc:0.65 0.35 Ret:0.162 Vol:0.281
Aloc:0.70 0.30 Ret:0.167 Vol:0.290
Aloc:0.75 0.25 Ret:0.171 Vol:0.301
Aloc:0.80 0.20 Ret:0.176 Vol:0.312
Aloc:0.85 0.15 Ret:0.180 Vol:0.324
Aloc:0.90 0.10 Ret:0.184 Vol:0.337
Aloc:0.95 0.05 Ret:0.189 Vol:0.351
Aloc:1.00 0.00 Ret:0.193 Vol:0.365
```

```
print(min_vol_ret)
print(port_pesos)
```

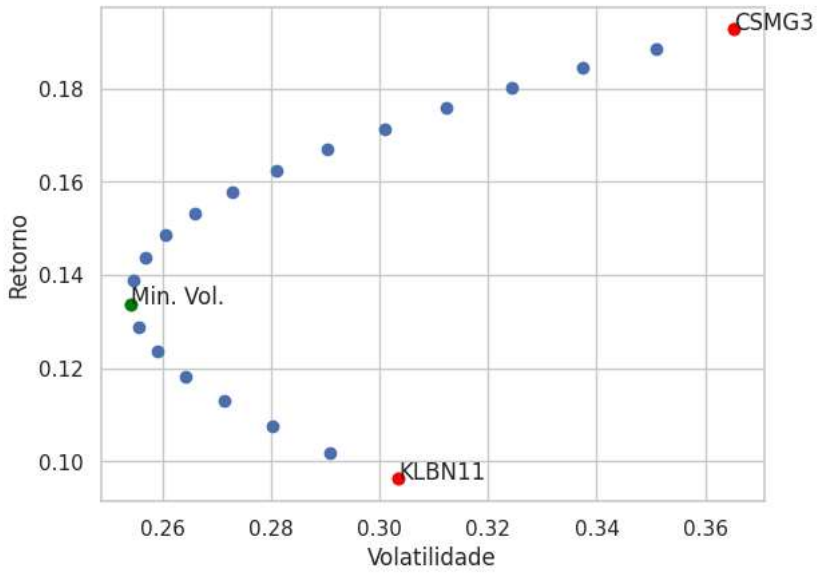
```
[0.25409394048001854, 0.13380393119163303]
[0, 0, 0.35, 0, 0.65, 0]
```

```
#                                a) GRAFICO DA FRONTEIRA EFICIENTE                                #
lp = np.array(points).T
plt.scatter(lp[[1][:]],lp[[0][:]]);
plt.ylabel("Retorno");
plt.xlabel("Volatilidade");

plt.scatter(vol_aa['CSMG3.SA'], ret_aa['CSMG3.SA'], color='red');
plt.text(vol_aa['CSMG3.SA'], ret_aa['CSMG3.SA'], 'CSMG3');

plt.scatter(vol_aa['KLBN11.SA'], ret_aa['KLBN11.SA'], color='red');
plt.text(vol_aa['KLBN11.SA'], ret_aa['KLBN11.SA'], 'KLBN11');

plt.scatter(min_vol_ret[0], min_vol_ret[1], color='green');
plt.text(min_vol_ret[0], min_vol_ret[1], 'Min. Vol.');
```




```
port_pesos
```

```
     [0, 0, 0.35, 0, 0.65, 0]
```

```
dados['PORT1'] = dados.dot(port_pesos)
dados_chg = dados.pct_change()
dados_chg = dados_chg.fillna(0)
```

```
#          c) A RENTABILIDADE ANUIZADA DOS ATIVOS QUE COMPÕE O PORTIFÓLIO E O PORTIFOLIO_1          #
ret_aa = (((dados.iloc[-1]/dados.iloc[0])**(1/5))-1)*100
print("RENTABILIDADE ANUIZADA:\n", ret_aa.round(2))

print("\n")

#          d) A VOLATILIDADE ANUIZADA DOS ATIVOS QUE COMPÕE O PORTIFÓLIO E O PORTIFOLIO_1          #
vol_aa = (dados_chg.std()*np.sqrt(252))*100
print("VOLATILIDADE ANUIZADA:\n", vol_aa.round(2))
```

```
     Ret aa:
      Ticker
     BBAS3.SA     0.191588
     PETR4.SA     0.439106
     CSMG3.SA     0.192677
     TAEE11.SA    0.262326
     KLBN11.SA    0.096226
     VALE3.SA     0.230638
     PORT1        0.133804
     dtype: float64
     Vol aa:
      Ticker
     BBAS3.SA     0.390788
     PETR4.SA     0.463584
     CSMG3.SA     0.365302
     TAEE11.SA    0.205062
     KLBN11.SA    0.303409
     VALE3.SA     0.390549
     PORT1        0.254094
     dtype: float64
```

```
#Calcular drawdown
ddown = pd.DataFrame()

for ativo in dados.columns:
  list = []
  for ind in range(dados.count()[0]):
    list.append((dados[ativo].iloc[ind]/dados[ativo].iloc[:ind+1].max()-1)*100)
  ddown[ativo]=list

ddown['Data']=dados.index.values
ddown.set_index(keys = 'Data', inplace = True)

#display(ddown)
print(ddown.min())
```

```
     BBAS3.SA    -58.286996
     PETR4.SA    -63.356055
     CSMG3.SA    -54.029209
```

```
TAEE11.SA   -23.694037
KLBN11.SA   -42.009959
VALE3.SA    -43.724133
PORT1       -44.136837
dtype: float64
```

```
#                              b) CRIANDO O GRÁFICO DE DRAWDOWN                                  #
ddown[['CSMG3.SA', 'KLBN11.SA', 'PORT1']].plot(figsize = (15,5));
```


```
dados=dados.drop(['PORT1'], axis=1)
dados_chg=dados_chg.drop(['PORT1'], axis=1)
```

```
dados
```

| Ticker | BBAS3.SA | PETR4.SA | CSMG3.SA | TAEE11.SA | KLBN11.SA | VALE3.SA |
|---|---|---|---|---|---|---|
| Date | | | | | | |
| **2018-01-02** | 100.000000 | 100.000000 | 100.000000 | 100.000000 | 100.000000 | 100.000000 |
| **2018-01-03** | 101.275424 | 100.906351 | 97.628277 | 100.139765 | 98.736247 | 99.400775 |
| **2018-01-04** | 102.247215 | 101.087604 | 96.123144 | 98.555430 | 98.901095 | 99.808248 |
| **2018-01-05** | 102.247215 | 101.691857 | 96.898488 | 99.021428 | 97.967017 | 101.366232 |
| **2018-01-08** | 102.490169 | 102.900307 | 97.491436 | 98.555430 | 99.725279 | 103.619363 |
| **...** | ... | ... | ... | ... | ... | ... |
| **2023-12-21** | 235.461052 | 603.164896 | 240.959337 | 311.920020 | 153.854229 | 281.420937 |
| **2023-12-22** | 236.111589 | 608.966190 | 242.038597 | 314.174863 | 156.027106 | 279.300310 |
| **2023-12-26** | 238.280153 | 618.745453 | 239.917524 | 314.842972 | 155.813387 | 280.287519 |
| **2023-12-27** | 237.933198 | 619.242683 | 238.149933 | 315.344037 | 156.454572 | 282.993127 |
| **2023-12-28** | 240.231849 | 617.253699 | 241.331565 | 320.521864 | 158.306960 | 282.261850 |

1487 rows × 6 columns

```
#Portfolio com 3 ativos: CSMG3, KLBN11 e VALE3
points = []
min_vol_ret = [100, 0]
port_pesos = [0, 0, 0, 0, 0, 0]
for w1 in range(0, 101, 5):
  for w2 in range(0, 101-w1, 5):
    ret, vol = calc_ret_vol(dados, dados_chg, [0, 0, w1/100, 0, w2/100, (1-w1/100-w2/100)])
    #print("Aloc:", round(w1/100, 2), round(w2/100, 2), round(1-w1/100-w2/100, 2), "Ret:", round(ret, 3), "Vol:", round(vol, 3))
    print(f"Aloc:{round(w1/100, 2):.2f} {round(w2/100, 2):.2f} {round(1-w1/100-w2/100, 2):.2f} Ret:{round(ret, 3):.3f} Vol:{round(vol, 3
    points.append([ret, vol])
    if vol < min_vol_ret[0]:
      min_vol_ret[0] = vol
      min_vol_ret[1] = ret
      port_pesos[2] = w1/100
      port_pesos[4] = w2/100
      port_pesos[5] = 1-w1/100-w2/100
```

```
print(min_vol_ret)
print(port_pesos)
```

```
[0.2406876025071771, 0.1507672473324957]
[0, 0, 0.3, 0, 0.55, 0.1499999999999999]
```

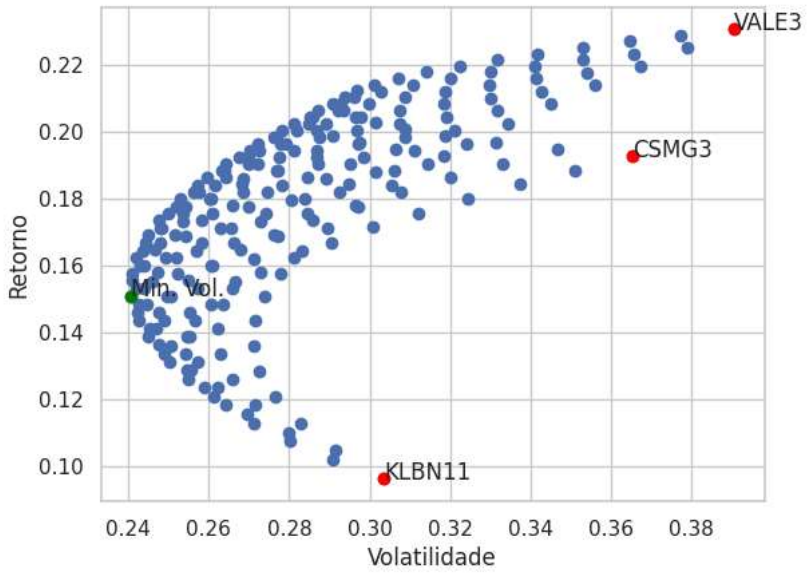
```
#                              a) GRAFICO DA FRONTEIRA EFICIENTE                                  #
lp = np.array(points).T
plt.scatter(lp[[1][:]],lp[[0][:]]);
plt.ylabel("Retorno");
plt.xlabel("Volatilidade");

plt.scatter(vol_aa['CSMG3.SA'], ret_aa['CSMG3.SA'], color='red');
plt.text(vol_aa['CSMG3.SA'], ret_aa['CSMG3.SA'], 'CSMG3');

plt.scatter(vol_aa['KLBN11.SA'], ret_aa['KLBN11.SA'], color='red');
plt.text(vol_aa['KLBN11.SA'], ret_aa['KLBN11.SA'], 'KLBN11');

plt.scatter(vol_aa['VALE3.SA'], ret_aa['VALE3.SA'], color='red');
plt.text(vol_aa['VALE3.SA'], ret_aa['VALE3.SA'], 'VALE3');

plt.scatter(min_vol_ret[0], min_vol_ret[1], color='green');
plt.text(min_vol_ret[0], min_vol_ret[1], 'Min. Vol.');
```




```
dados['PORT1'] = dados.dot(port_pesos)
dados_chg = (dados - dados.shift(1)) / dados.shift(1)
dados_chg = dados_chg.fillna(0)
```

```
#          c) A RENTABILIDADE ANUIZADA DOS ATIVOS QUE COMPÕE O PORTIFÓLIO E O PORTIFOLIO_1          #
ret_aa = (((dados.iloc[-1]/dados.iloc[0])**(1/5))-1)*100
print("RENTABILIDADE ANUIZADA:\n", ret_aa.round(2))

print("\n")

#          d) A VOLATILIDADE ANUIZADA DOS ATIVOS QUE COMPÕE O PORTIFÓLIO E O PORTIFOLIO_1          #
vol_aa = (dados_chg.std()*np.sqrt(252))*100
print("VOLATILIDADE ANUIZADA:\n", vol_aa.round(2))
```

```
RENTABILIDADE ANUIZADA:
 Ticker
BBAS3.SA     19.16
PETR4.SA     43.91
CSMG3.SA     19.27
TAEE11.SA    26.23
KLBN11.SA     9.62
VALE3.SA     23.06
PORT1        15.08
dtype: float64


VOLATILIDADE ANUIZADA:
 Ticker
BBAS3.SA     39.08
PETR4.SA     46.36
CSMG3.SA     36.53
TAEE11.SA    20.51
KLBN11.SA    30.34
VALE3.SA     39.05
PORT1        24.07
dtype: float64
```

```
#Calcular drawdown (valor de queda de um ativo em comparação ao valor máximo de cotação anterior)
ddown = pd.DataFrame()

for ativo in dados.columns:
  list = []
  for ind in range(dados.count()[0]):
    list.append((dados[ativo].iloc[ind]/dados[ativo].iloc[:ind+1].max()-1)*100)
  ddown[ativo]=list

ddown['Data']=dados.index.values
ddown.set_index(keys = 'Data', inplace = True)

#display(ddown)
print(ddown.min())
```

```
BBAS3.SA    -58.286996
PETR4.SA    -63.356055
CSMG3.SA    -54.029209
TAEE11.SA   -23.694037
KLBN11.SA   -42.009959
VALE3.SA    -43.724133
PORT1       -42.788396
dtype: float64
```

```
#                                b) CRIANDO O GRÁFICO DE DRAWDOWN                                #
ddown[['CSMG3.SA', 'KLBN11.SA', 'VALE3.SA', 'PORT1']].plot(figsize = (15,5));
```