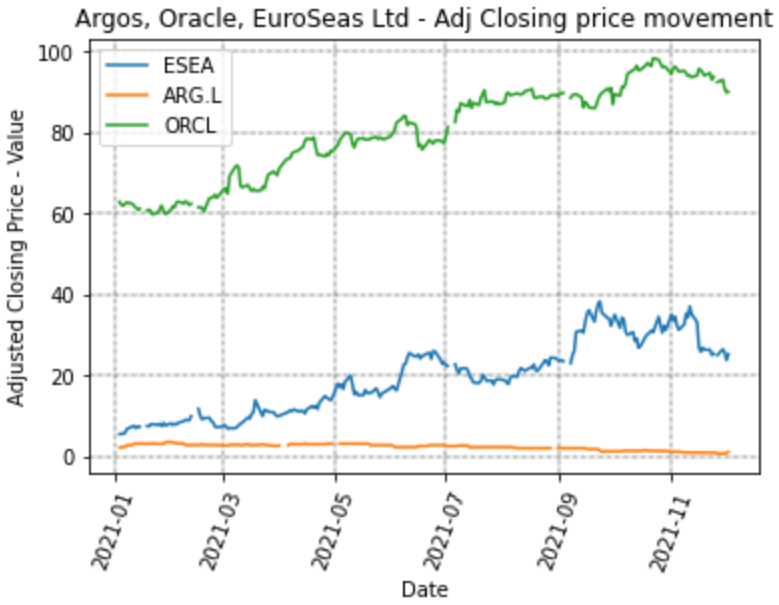
Stock_Data_Analysis_2.0

In [19]:
```
import pandas as pd
import numpy as np
import pandas_datareader as web
from matplotlib import pyplot as plt
%matplotlib inline
```

In [20]:
```
from datetime import datetime
```

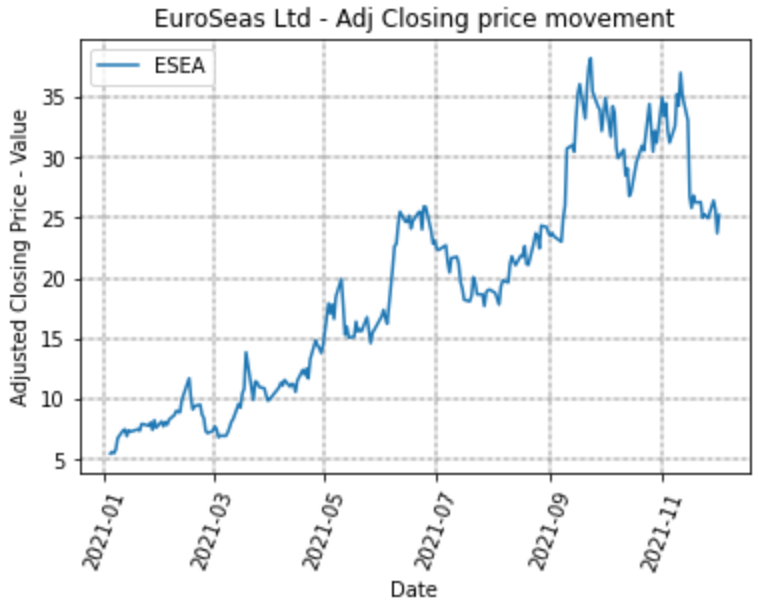
In [21]:
```
#EuroSeasLtd, Argos, Oracle
symbols = ["ESEA", "ARG.L", "ORCL"]
start_date = datetime(2021, 1, 1)
end_date = datetime.today()
stock_data = web.get_data_yahoo(symbols, start_date, end_date)
#print(stock_data.info())
```

In [41]:
```
stock_data = stock_data.reset_index()
x_values = stock_data['Date']
y_values = stock_data['Adj Close']
plt.plot(x_values, y_values)
plt.grid(which="major", color="k", linestyle='-.', linewidth=0.4)
plt.xlabel("Date")
plt.ylabel("Adjusted Closing Price - Value")
plt.title("Argos, Oracle, EuroSeas Ltd - Adj Closing price movement")
plt.legend(['ESEA', 'ARG.L', 'ORCL'])
plt.xticks(rotation=70)
plt.show()
```



In [40]:
```
stock_esea = web.get_data_yahoo('ESEA', start_date, end_date)

stock_esea = stock_esea.reset_index()
x_values = stock_esea['Date']
y_values = stock_esea['Adj Close']
plt.plot(x_values, y_values)
plt.grid(which="major", color="k", linestyle='-.', linewidth=0.4)
plt.xlabel("Date")
plt.ylabel("Adjusted Closing Price - Value")
```

```
plt.title("EuroSeas Ltd - Adj Closing price movement")
plt.legend(['ESEA'])
plt.xticks(rotation=70)
plt.show()
```



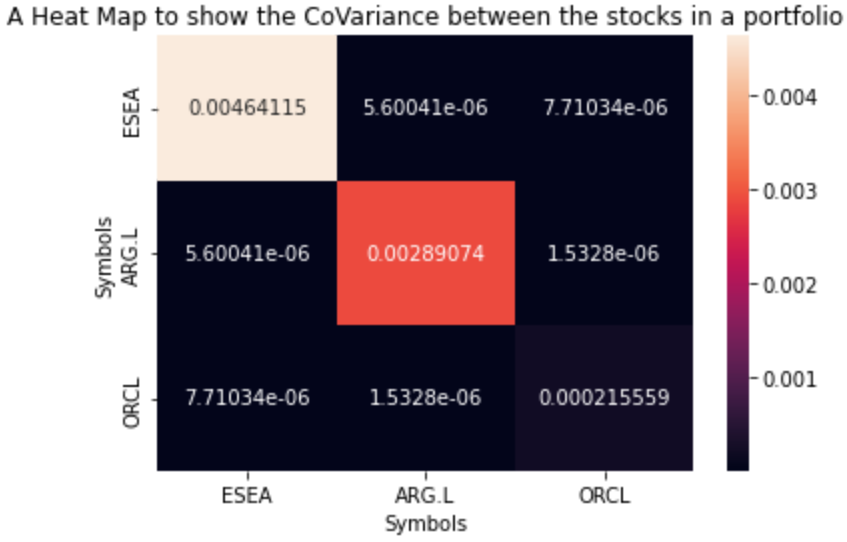
In [24]:

```
daily_returns = stock_data['Adj Close'].pct_change()
daily_returns_cov = daily_returns.cov()
print(daily_returns_cov)
```

```
Symbols       ESEA     ARG.L      ORCL
Symbols
ESEA      0.004641  0.000006  0.000008
ARG.L     0.000006  0.002891  0.000002
ORCL      0.000008  0.000002  0.000216
```

In [25]:

```
import seaborn as sn
sn.heatmap(daily_returns_cov, annot=True, fmt='g')
plt.title("A Heat Map to show the CoVariance between the stocks in a portfolio")
plt.show()
```



Oracle has a CoVariance cloe to Zero with both EuroSeas and Argos. Thereby reducing the risk of the portfolio. As the only way to reduce the risk of a portfolio, is to have stocks that are not correlated at all (Cov=0). Futhermore, ESEA has a negative correlation with ARG.L but all other correlations are positive. This greatly reduces the risk in a portfolio.

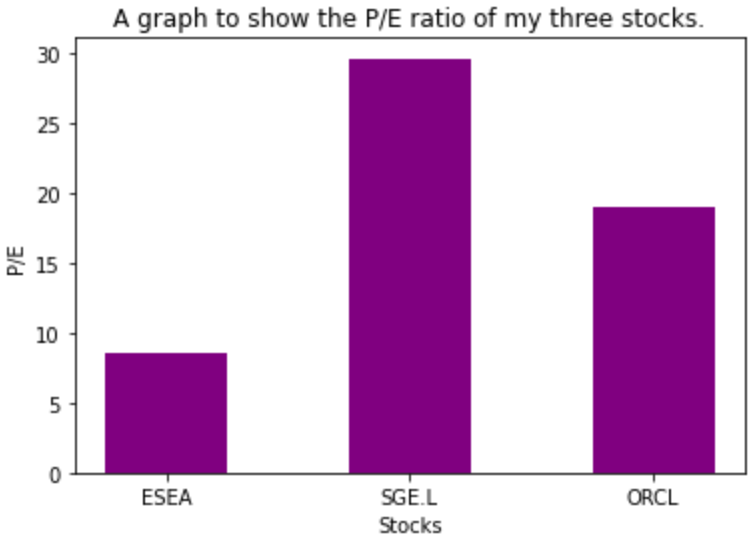
In [42]:
```
#EuroSeas Ltd, SAGE, Oracle
symbol_list = ['ESEA', 'SGE.L', 'ORCL']
pe_list = []
for symbol in symbol_list:
    x = web.get_quote_yahoo(symbol)['trailingPE']
    pe_list.append(x[0])

print(pe_list)
```

```
[8.496291, 29.639845, 19.065678]
```

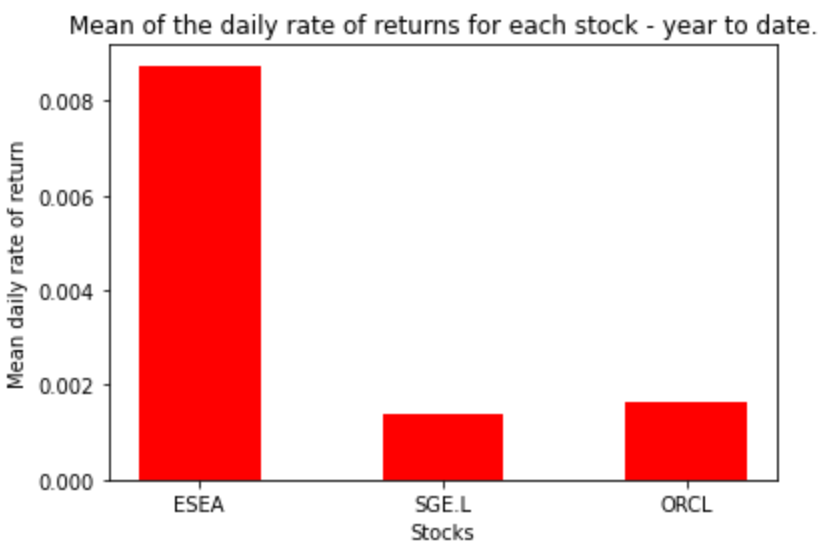
In [44]:
```
df1 = pd.DataFrame({'Ticker':symbol_list, 'PE':pe_list})
print(df1)
```

```
  Ticker         PE
0   ESEA   8.496291
1  SGE.L  29.639845
2   ORCL  19.065678
```

In [45]:
```
plt.bar(df1['Ticker'], df1['PE'], width=0.5, color='purple')
plt.title("A graph to show the P/E ratio of my three stocks.")
plt.ylabel("P/E")
plt.xlabel("Stocks")
plt.show()
```



In [54]:
```
symbol_list = ["ESEA", "SGE.L", "ORCL"]
stock_data_2 = web.get_data_yahoo(symbol_list, start_date, end_date)
stock_data_2 = stock_data_2.reset_index()

daily_returns = stock_data_2['Adj Close'].pct_change()

ESEA_mean_ror = daily_returns["ESEA"].mean()
SGE_mean_ror = daily_returns["SGE.L"].mean()
ORCL_mean_ror = daily_returns["ORCL"].mean()

y_values = [ESEA_mean_ror, SGE_mean_ror, ORCL_mean_ror]
x_values = ["ESEA", "SGE.L", "ORCL"]
plt.bar(x_values, y_values, width=0.5, color='red')
plt.title("Mean of the daily rate of returns for each stock - year to date.")
plt.ylabel("Mean daily rate of return")
plt.xlabel("Stocks")
plt.show()
```

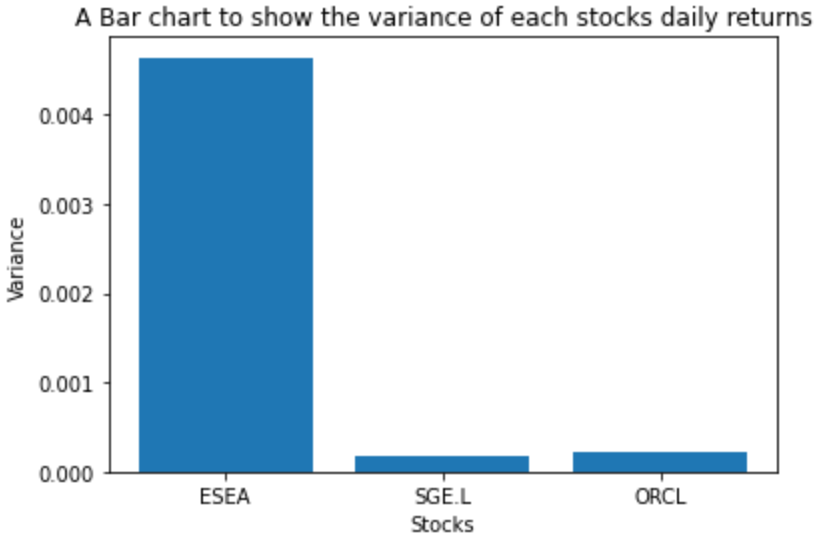


Now we will calculate and plot the variance.

In [56]:

```
stock_daily_var = daily_returns.var()

print(stock_daily_var)

#print(stock_daily_var.keys())

height = []
for key in stock_daily_var.keys():
    height.append(stock_daily_var[key])


x_pos = np.arange(len(stock_daily_var.keys()))

plt.bar(x_pos, height)
plt.xticks(x_pos, stock_daily_var.keys())
plt.title("A Bar chart to show the variance of each stocks daily returns")
plt.xlabel("Stocks")
plt.ylabel("Variance")
plt.show()
```

```
Symbols
ESEA     0.004641
SGE.L    0.000181
ORCL     0.000216
dtype: float64
```



In [ ]: