

Coca Cola Stock - Live and Updated

Objective

- Predict Coca-Cola's stock price (Close price and analyze trends)

Data

- Historical data with Date, Open, High, Low, Close, Volume, Dividends and Stock splits

Deliverables

- Insights from the data (Visualizations and statistics)
- An ML model to predict stock prices
- A live-updating system for predictions.

Data Collection

In [1]: `pip install yfinance`

```
Requirement already satisfied: yfinance in c:\users\accou\anaconda3\lib\site-packages (0.2.52)
Requirement already satisfied: pandas>=1.3.0 in c:\users\accou\anaconda3\lib\site-packages (from yfinance) (2.2.2)
Requirement already satisfied: numpy>=1.16.5 in c:\users\accou\anaconda3\lib\site-packages (from yfinance) (1.26.4)
Requirement already satisfied: requests>=2.31 in c:\users\accou\anaconda3\lib\site-packages (from yfinance) (2.32.2)
Requirement already satisfied: multitasking>=0.0.7 in c:\users\accou\anaconda3\lib\site-packages (from yfinance) (0.0.11)
Requirement already satisfied: lxml>=4.9.1 in c:\users\accou\anaconda3\lib\site-packages (from yfinance) (5.2.1)
Requirement already satisfied: platformdirs>=2.0.0 in c:\users\accou\anaconda3\lib\site-packages (from yfinance) (3.10.0)
Requirement already satisfied: pytz>=2022.5 in c:\users\accou\anaconda3\lib\site-packages (from yfinance) (2024.1)
Requirement already satisfied: frozendict>=2.3.4 in c:\users\accou\anaconda3\lib\site-packages (from yfinance) (2.4.2)
Requirement already satisfied: peewee>=3.16.2 in c:\users\accou\anaconda3\lib\site-packages (from yfinance) (3.17.8)
Requirement already satisfied: beautifulsoup4>=4.11.1 in c:\users\accou\anaconda3\lib\site-packages (from yfinance) (4.12.3)
Requirement already satisfied: html5lib>=1.1 in c:\users\accou\anaconda3\lib\site-packages (from yfinance) (1.1)
Requirement already satisfied: soupsieve>1.2 in c:\users\accou\anaconda3\lib\site-packages (from beautifulsoup4>=4.11.1->yfinance) (2.5)
Requirement already satisfied: six>=1.9 in c:\users\accou\anaconda3\lib\site-packages (from html5lib>=1.1->yfinance) (1.16.0)
Requirement already satisfied: webencodings in c:\users\accou\anaconda3\lib\site-packages (from html5lib>=1.1->yfinance) (0.5.1)
Requirement already satisfied: python-dateutil>=2.8.2 in c:\users\accou\anaconda3\lib\site-packages (from pandas>=1.3.0->yfinance) (2.9.0.post0)
Requirement already satisfied: tzdata>=2022.7 in c:\users\accou\anaconda3\lib\site-packages (from pandas>=1.3.0->yfinance) (2023.3)
Requirement already satisfied: charset-normalizer<4,>=2 in c:\users\accou\anaconda3\lib\site-packages (from requests>=2.31->yfinance) (2.0.4)
Requirement already satisfied: idna<4,>=2.5 in c:\users\accou\anaconda3\lib\site-packages (from requests>=2.31->yfinance) (3.7)
Requirement already satisfied: urllib3<3,>=1.21.1 in c:\users\accou\anaconda3\lib\site-packages (from requests>=2.31->yfinance) (2.2.2)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\accou\anaconda3\lib\site-packages (from requests>=2.31->yfinance) (2024.6.2)
Note: you may need to restart the kernel to use updated packages.
```

In [2]: `pip install pandas-datareader`

```
Requirement already satisfied: pandas-datareader in c:\users\accou\anaconda3\lib\site-packages (0.10.0)
Requirement already satisfied: lxml in c:\users\accou\anaconda3\lib\site-packages (from pandas-datareader) (5.2.1)
Requirement already satisfied: pandas>=0.23 in c:\users\accou\anaconda3\lib\site-packages (from pandas-datareader) (2.2.2)
Requirement already satisfied: requests>=2.19.0 in c:\users\accou\anaconda3\lib\site-packages (from pandas-datareader) (2.32.2)
Requirement already satisfied: numpy>=1.26.0 in c:\users\accou\anaconda3\lib\site-packages (from pandas>=0.23->pandas-datareader) (1.26.4)
Requirement already satisfied: python-dateutil>=2.8.2 in c:\users\accou\anaconda3\lib\site-packages (from pandas>=0.23->pandas-datareader) (2.9.0.post0)
Requirement already satisfied: pytz>=2020.1 in c:\users\accou\anaconda3\lib\site-packages (from pandas>=0.23->pandas-datareader) (2024.1)
Requirement already satisfied: tzdata>=2022.7 in c:\users\accou\anaconda3\lib\site-packages (from pandas>=0.23->pandas-datareader) (2023.3)
Requirement already satisfied: charset-normalizer<4,>=2 in c:\users\accou\anaconda3\lib\site-packages (from requests>=2.19.0->pandas-datareader) (2.0.4)
Requirement already satisfied: idna<4,>=2.5 in c:\users\accou\anaconda3\lib\site-packages (from requests>=2.19.0->pandas-datareader) (3.7)
Requirement already satisfied: urllib3<3,>=1.21.1 in c:\users\accou\anaconda3\lib\site-packages (from requests>=2.19.0->pandas-datareader) (2.2.2)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\accou\anaconda3\lib\site-packages (from requests>=2.19.0->pandas-datareader) (2024.6.2)
Requirement already satisfied: six>=1.5 in c:\users\accou\anaconda3\lib\site-packages (from python-dateutil>=2.8.2->pandas>=0.23->pandas-datareader) (1.16.0)
Note: you may need to restart the kernel to use updated packages.
```

In [3]: `pip install ipywidgets`

Requirement already satisfied: ipywidgets in c:\users\accou\anaconda3\lib\site-packages (7.8.1)
Requirement already satisfied: comm>=0.1.3 in c:\users\accou\anaconda3\lib\site-packages (from ipywidgets) (0.2.1)
Requirement already satisfied: ipython-genutils<=0.2.0 in c:\users\accou\anaconda3\lib\site-packages (from ipywidgets) (0.2.0)
Requirement already satisfied: traitlets>=4.3.1 in c:\users\accou\anaconda3\lib\site-packages (from ipywidgets) (5.14.3)
Requirement already satisfied: widgetsnbextension<=3.6.6 in c:\users\accou\anaconda3\lib\site-packages (from ipywidgets) (3.6.6)
Requirement already satisfied: ipython>=4.0.0 in c:\users\accou\anaconda3\lib\site-packages (from ipywidgets) (8.25.0)
Requirement already satisfied: jupyterlab-widgets<3,>=1.0.0 in c:\users\accou\anaconda3\lib\site-packages (from ipywidgets) (1.0.0)
Requirement already satisfied: decorator in c:\users\accou\anaconda3\lib\site-packages (from ipython>=4.0.0->ipywidgets) (5.1.1)
Requirement already satisfied: jedi>=0.16 in c:\users\accou\anaconda3\lib\site-packages (from ipython>=4.0.0->ipywidgets) (0.18.1)
Requirement already satisfied: matplotlib-inline in c:\users\accou\anaconda3\lib\site-packages (from ipython>=4.0.0->ipywidgets) (0.1.6)
Requirement already satisfied: prompt-toolkit<3.1.0,>=3.0.41 in c:\users\accou\anaconda3\lib\site-packages (from ipython>=4.0.0->ipywidgets) (3.0.43)
Requirement already satisfied: pygments>=2.4.0 in c:\users\accou\anaconda3\lib\site-packages (from ipython>=4.0.0->ipywidgets) (2.15.1)
Requirement already satisfied: stack-data in c:\users\accou\anaconda3\lib\site-packages (from ipython>=4.0.0->ipywidgets) (0.2.0)
Requirement already satisfied: colorama in c:\users\accou\anaconda3\lib\site-packages (from ipython>=4.0.0->ipywidgets) (0.4.6)
Requirement already satisfied: notebook>=4.4.1 in c:\users\accou\anaconda3\lib\site-packages (from widgetsnbextension<=3.6.6->ipywidgets) (7.0.8)
Requirement already satisfied: parso<0.9.0,>=0.8.0 in c:\users\accou\anaconda3\lib\site-packages (from jedi>=0.16->ipython>=4.0.0->ipywidgets) (0.8.3)
Requirement already satisfied: jupyter-server<3,>=2.4.0 in c:\users\accou\anaconda3\lib\site-packages (from notebook>=4.4.1->widgetsnbextension<=3.6.6->ipywidgets) (2.14.1)
Requirement already satisfied: jupyterlab-server<3,>=2.22.1 in c:\users\accou\anaconda3\lib\site-packages (from notebook>=4.4.1->widgetsnbextension<=3.6.6->ipywidgets) (2.25.1)
Requirement already satisfied: jupyterlab<4.1,>=4.0.2 in c:\users\accou\anaconda3\lib\site-packages (from notebook>=4.4.1->widgetsnbextension<=3.6.6->ipywidgets) (4.0.11)
Requirement already satisfied: notebook-shim<0.3,>=0.2 in c:\users\accou\anaconda3\lib\site-packages (from notebook>=4.4.1->widgetsnbextension<=3.6.6->ipywidgets) (0.2.3)
Requirement already satisfied: tornado>=6.2.0 in c:\users\accou\anaconda3\lib\site-packages (from notebook>=4.4.1->widgetsnbextension<=3.6.6->ipywidgets) (6.4.1)
Requirement already satisfied: wcwidth in c:\users\accou\anaconda3\lib\site-packages (from prompt-toolkit<3.1.0,>=3.0.41->ipython>=4.0.0->ipywidgets) (0.2.5)
Requirement already satisfied: executing in c:\users\accou\anaconda3\lib\site-packages (from stack-data->ipython>=4.0.0->ipywidgets) (0.8.3)
Requirement already satisfied: asttokens in c:\users\accou\anaconda3\lib\site-packages (from stack-data->ipython>=4.0.0->ipywidgets) (2.0.5)
Requirement already satisfied: pure-eval in c:\users\accou\anaconda3\lib\site-packages (from stack-data->ipython>=4.0.0->ipywidgets) (0.2.2)
Requirement already satisfied: anyio>=3.1.0 in c:\users\accou\anaconda3\lib\site-packages (from jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension<=3.6.6->ipywidgets) (4.2.0)
Requirement already satisfied: argon2-cffi>=21.1 in c:\users\accou\anaconda3\lib\site-packages (from jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension<=3.6.6->ipywidgets) (21.3.0)
Requirement already satisfied: Jinja2>=3.0.3 in c:\users\accou\anaconda3\lib\site-packages (from jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension<=3.6.6->ipywidgets) (3.1.4)
Requirement already satisfied: jupyter-client>=7.4.4 in c:\users\accou\anaconda3\lib\site-packages (from jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension<=3.6.6->ipywidgets) (8.6.0)
Requirement already satisfied: jupyter-core!=5.0.*,>=4.12 in c:\users\accou\anaconda3\lib\site-packages (from jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension<=3.6.6->ipywidgets) (5.7.2)
Requirement already satisfied: jupyter-events>=0.9.0 in c:\users\accou\anaconda3\lib\site-packages (from jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension<=3.6.6->ipywidgets) (0.10.0)
Requirement already satisfied: jupyter-server-terminals>=0.4.4 in c:\users\accou\anaconda3\lib\site-packages (from jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension<=3.6.6->ipywidgets) (0.4.4)
Requirement already satisfied: nbconvert>=6.4.4 in c:\users\accou\anaconda3\lib\site-packages (from jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension<=3.6.6->ipywidgets) (7.10.0)
Requirement already satisfied: nbformat>=5.3.0 in c:\users\accou\anaconda3\lib\site-packages (from jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension<=3.6.6->ipywidgets) (5.9.2)
Requirement already satisfied: overrides>=5.0 in c:\users\accou\anaconda3\lib\site-packages (from jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension<=3.6.6->ipywidgets) (7.4.0)
Requirement already satisfied: packaging>=22.0 in c:\users\accou\anaconda3\lib\site-packages (from jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension<=3.6.6->ipywidgets) (23.2)
Requirement already satisfied: prometheus-client>=0.9 in c:\users\accou\anaconda3\lib\site-packages (from jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension<=3.6.6->ipywidgets) (0.14.1)
Requirement already satisfied: winpty>=2.0.1 in c:\users\accou\anaconda3\lib\site-packages (from jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension<=3.6.6->ipywidgets) (2.0.10)
Requirement already satisfied: pyzmq>=24 in c:\users\accou\anaconda3\lib\site-packages (from jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension<=3.6.6->ipywidgets) (25.1.2)
Requirement already satisfied: send2trash>=1.8.2 in c:\users\accou\anaconda3\lib\site-packages (from jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension<=3.6.6->ipywidgets) (1.8.2)
Requirement already satisfied: terminado>=0.8.3 in c:\users\accou\anaconda3\lib\site-packages (from jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension<=3.6.6->ipywidgets) (0.17.1)
Requirement already satisfied: websocket-client>=1.7 in c:\users\accou\anaconda3\lib\site-packages (from jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension<=3.6.6->ipywidgets) (1.8.0)
Requirement already satisfied: async-lru>=1.0.0 in c:\users\accou\anaconda3\lib\site-packages (from jupyterlab<4.1,>=4.0.2->notebook>=4.4.1->widgetsnbextension<=3.6.6->ipywidgets) (2.0.4)
Requirement already satisfied: ipykernel in c:\users\accou\anaconda3\lib\site-packages (from jupyterlab<4.1,>=4.0.2->notebook>=4.4.1->widgetsnbextension<=3.6.6->ipywidgets) (6.28.0)
Requirement already satisfied: jupyter-lsp>=2.0.0 in c:\users\accou\anaconda3\lib\site-packages (from jupyterlab<4.1,>=4.0.2->notebook>=4.4.1->widgetsnbextension<=3.6.6->ipywidgets) (2.2.0)
Requirement already satisfied: babel>=2.10 in c:\users\accou\anaconda3\lib\site-packages (from jupyterlab-server<3,>=2.22.1->notebook>=4.4.1->widgetsnbextension<=3.6.6->ipywidgets) (2.11.0)
Requirement already satisfied: json5>=0.9.0 in c:\users\accou\anaconda3\lib\site-packages (from jupyterlab-server<3,>=2.22.1->notebook>=4.4.1->widgetsnbextension<=3.6.6->ipywidgets) (0.9.6)
Requirement already satisfied: jsonschema>=4.18.0 in c:\users\accou\anaconda3\lib\site-packages (from jupyterlab-server<3,>=2.22.1->notebook>=4.4.1->widgetsnbextension<=3.6.6->ipywidgets) (4.19.2)
Requirement already satisfied: requests>=2.31 in c:\users\accou\anaconda3\lib\site-packages (from jupyterlab-server<3,>=2.22.1->notebook>=4.4.1->widgetsnbextension<=3.6.6->ipywidgets) (2.32.2)
Requirement already satisfied: six in c:\users\accou\anaconda3\lib\site-packages (from asttokens->stack-data->ipython>=4.0.0->ipywidgets) (1.16.0)

Requirement already satisfied: idna>=2.8 in c:\users\accou\anaconda3\lib\site-packages (from anyio>=3.1.0->jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension~=3.6.6->ipywidgets) (3.7)

Requirement already satisfied: sniffio=1.1 in c:\users\accou\anaconda3\lib\site-packages (from anyio>=3.1.0->jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension~=3.6.6->ipywidgets) (1.3.0)

Requirement already satisfied: argon2-cffi-bindings in c:\users\accou\anaconda3\lib\site-packages (from argon2-cffi>=21.1->jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension~=3.6.6->ipywidgets) (21.2.0)

Requirement already satisfied: pytz>=2015.7 in c:\users\accou\anaconda3\lib\site-packages (from babel>=2.10->jupyterlab-server<3,>=2.2.1->notebook>=4.4.1->widgetsnbextension~=3.6.6->ipywidgets) (2024.1)

Requirement already satisfied: MarkupSafe>=2.0 in c:\users\accou\anaconda3\lib\site-packages (from jinja2>=3.0.3->jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension~=3.6.6->ipywidgets) (2.1.3)

Requirement already satisfied: attrs>=22.2.0 in c:\users\accou\anaconda3\lib\site-packages (from jsonschema>=4.18.0->jupyterlab-server<3,>=2.2.1->notebook>=4.4.1->widgetsnbextension~=3.6.6->ipywidgets) (23.1.0)

Requirement already satisfied: jsonschema-specifications>=2023.03.6 in c:\users\accou\anaconda3\lib\site-packages (from jsonschema>=4.18.0->jupyterlab-server<3,>=2.2.1->notebook>=4.4.1->widgetsnbextension~=3.6.6->ipywidgets) (2023.7.1)

Requirement already satisfied: referencing>=0.28.4 in c:\users\accou\anaconda3\lib\site-packages (from jsonschema>=4.18.0->jupyterlab-server<3,>=2.2.1->notebook>=4.4.1->widgetsnbextension~=3.6.6->ipywidgets) (0.30.2)

Requirement already satisfied: rpds-py>=0.7.1 in c:\users\accou\anaconda3\lib\site-packages (from jsonschema>=4.18.0->jupyterlab-server<3,>=2.2.1->notebook>=4.4.1->widgetsnbextension~=3.6.6->ipywidgets) (0.10.6)

Requirement already satisfied: python-dateutil>=2.8.2 in c:\users\accou\anaconda3\lib\site-packages (from jupyter-client>=7.4.4->jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension~=3.6.6->ipywidgets) (2.9.0.post0)

Requirement already satisfied: platformdirs>=2.5 in c:\users\accou\anaconda3\lib\site-packages (from jupyter-core!=5.0.*,>=4.12->jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension~=3.6.6->ipywidgets) (3.10.0)

Requirement already satisfied: pywin32>=300 in c:\users\accou\anaconda3\lib\site-packages (from jupyter-core!=5.0.*,>=4.12->jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension~=3.6.6->ipywidgets) (305.1)

Requirement already satisfied: python-json-logger>=2.0.4 in c:\users\accou\anaconda3\lib\site-packages (from jupyter-events>=0.9.0->jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension~=3.6.6->ipywidgets) (2.0.7)

Requirement already satisfied: pyyaml=5.3 in c:\users\accou\anaconda3\lib\site-packages (from jupyter-events>=0.9.0->jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension~=3.6.6->ipywidgets) (6.0.1)

Requirement already satisfied: rfc3339-validator in c:\users\accou\anaconda3\lib\site-packages (from jupyter-events>=0.9.0->jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension~=3.6.6->ipywidgets) (0.1.4)

Requirement already satisfied: rfc3986-validator>=0.1.1 in c:\users\accou\anaconda3\lib\site-packages (from jupyter-events>=0.9.0->jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension~=3.6.6->ipywidgets) (0.1.1)

Requirement already satisfied: beautifulsoup4 in c:\users\accou\anaconda3\lib\site-packages (from nbconvert>=6.4.4->jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension~=3.6.6->ipywidgets) (4.12.3)

Requirement already satisfied: bleach!=5.0.0 in c:\users\accou\anaconda3\lib\site-packages (from nbconvert>=6.4.4->jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension~=3.6.6->ipywidgets) (4.1.0)

Requirement already satisfied: defusedxml in c:\users\accou\anaconda3\lib\site-packages (from nbconvert>=6.4.4->jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension~=3.6.6->ipywidgets) (0.7.1)

Requirement already satisfied: jupyterlab-pygments in c:\users\accou\anaconda3\lib\site-packages (from nbconvert>=6.4.4->jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension~=3.6.6->ipywidgets) (0.1.2)

Requirement already satisfied: mistune<4,>=2.0.3 in c:\users\accou\anaconda3\lib\site-packages (from nbconvert>=6.4.4->jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension~=3.6.6->ipywidgets) (2.0.4)

Requirement already satisfied: nbclient>=0.5.0 in c:\users\accou\anaconda3\lib\site-packages (from nbconvert>=6.4.4->jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension~=3.6.6->ipywidgets) (0.8.0)

Requirement already satisfied: pandocfilters>=1.4.1 in c:\users\accou\anaconda3\lib\site-packages (from nbconvert>=6.4.4->jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension~=3.6.6->ipywidgets) (1.5.0)

Requirement already satisfied: tinycss2 in c:\users\accou\anaconda3\lib\site-packages (from nbconvert>=6.4.4->jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension~=3.6.6->ipywidgets) (1.2.1)

Requirement already satisfied: fastjsonschema in c:\users\accou\anaconda3\lib\site-packages (from nbformat>=5.3.0->jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension~=3.6.6->ipywidgets) (2.16.2)

Requirement already satisfied: charset-normalizer<4,>=2 in c:\users\accou\anaconda3\lib\site-packages (from requests>=2.31->jupyterlab-server<3,>=2.2.1->notebook>=4.4.1->widgetsnbextension~=3.6.6->ipywidgets) (2.0.4)

Requirement already satisfied: urllib3<3,>=1.21.1 in c:\users\accou\anaconda3\lib\site-packages (from requests>=2.31->jupyterlab-server<3,>=2.2.1->notebook>=4.4.1->widgetsnbextension~=3.6.6->ipywidgets) (2.2.2)

Requirement already satisfied: certifi>=2017.4.17 in c:\users\accou\anaconda3\lib\site-packages (from requests>=2.31->jupyterlab-server<3,>=2.2.1->notebook>=4.4.1->widgetsnbextension~=3.6.6->ipywidgets) (2024.6.2)

Requirement already satisfied: debugpy>=1.6.5 in c:\users\accou\anaconda3\lib\site-packages (from ipykernel->jupyterlab<4.1,>=4.0.2->notebook>=4.4.1->widgetsnbextension~=3.6.6->ipywidgets) (1.6.7)

Requirement already satisfied: nest-asyncio in c:\users\accou\anaconda3\lib\site-packages (from ipykernel->jupyterlab<4.1,>=4.0.2->notebook>=4.4.1->widgetsnbextension~=3.6.6->ipywidgets) (1.6.0)

Requirement already satisfied: psutil in c:\users\accou\anaconda3\lib\site-packages (from ipykernel->jupyterlab<4.1,>=4.0.2->notebook>=4.4.1->widgetsnbextension~=3.6.6->ipywidgets) (5.9.0)

Requirement already satisfied: webencodings in c:\users\accou\anaconda3\lib\site-packages (from bleach!=5.0.0->nbconvert>=6.4.4->jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension~=3.6.6->ipywidgets) (0.5.1)

Requirement already satisfied: fqdn in c:\users\accou\anaconda3\lib\site-packages (from jsonschema[format-nongpl]>=4.18.0->jupyter-events>=0.9.0->jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension~=3.6.6->ipywidgets) (1.5.1)

Requirement already satisfied: isoduration in c:\users\accou\anaconda3\lib\site-packages (from jsonschema[format-nongpl]>=4.18.0->jupyter-events>=0.9.0->jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension~=3.6.6->ipywidgets) (20.11.0)

Requirement already satisfied: jsonpointer>1.13 in c:\users\accou\anaconda3\lib\site-packages (from jsonschema[format-nongpl]>=4.18.0->jupyter-events>=0.9.0->jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension~=3.6.6->ipywidgets) (2.1)

Requirement already satisfied: uri-template in c:\users\accou\anaconda3\lib\site-packages (from jsonschema[format-nongpl]>=4.18.0->jupyter-events>=0.9.0->jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension~=3.6.6->ipywidgets) (1.3.0)

Requirement already satisfied: webcolors>=1.11 in c:\users\accou\anaconda3\lib\site-packages (from jsonschema[format-nongpl]>=4.18.0->jupyter-events>=0.9.0->jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension~=3.6.6->ipywidgets) (24.11.1)

Requirement already satisfied: cffi>=1.0.1 in c:\users\accou\anaconda3\lib\site-packages (from argon2-cffi-bindings->argon2-cffi>=21.1->jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension~=3.6.6->ipywidgets) (1.16.0)

Requirement already satisfied: soupsieve>1.2 in c:\users\accou\anaconda3\lib\site-packages (from beautifulsoup4->nbconvert>=6.4.4->jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension~=3.6.6->ipywidgets) (2.5)

Requirement already satisfied: pyparser in c:\users\accou\anaconda3\lib\site-packages (from cffi>=1.0.1->argon2-cffi-bindings->argon2-cffi>=21.1->jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension~=3.6.6->ipywidgets) (2.21)

Requirement already satisfied: arrow>=0.15.0 in c:\users\accou\anaconda3\lib\site-packages (from isoduration->jsonschema[format-nongpl]>=4.18.0->jupyter-events>=0.9.0->jupyter-server<3,>=2.4.0->notebook>=4.4.1->widgetsnbextension~=3.6.6->ipywidgets) (1.2.3)

Note: you may need to restart the kernel to use updated packages.

```
In [4]: !pip install --upgrade backtesting
```

Requirement already satisfied: backtesting in c:\users\accou\anaconda3\lib\site-packages (0.6.1)
 Requirement already satisfied: numpy>=1.17.0 in c:\users\accou\anaconda3\lib\site-packages (from backtesting) (1.26.4)
 Requirement already satisfied: pandas!=0.25.0,>=0.25.0 in c:\users\accou\anaconda3\lib\site-packages (from backtesting) (2.2.2)
 Requirement already satisfied: bokeh!=3.0.*,>=1.4.0 in c:\users\accou\anaconda3\lib\site-packages (from backtesting) (3.4.1)
 Requirement already satisfied: joblib in c:\users\accou\anaconda3\lib\site-packages (from backtesting) (1.4.2)
 Requirement already satisfied: Jinja2>=2.9 in c:\users\accou\anaconda3\lib\site-packages (from bokeh!=3.0.*,>=1.4.0->backtesting) (3.1.4)
 Requirement already satisfied: contourpy>=1.2 in c:\users\accou\anaconda3\lib\site-packages (from bokeh!=3.0.*,>=1.4.0->backtesting) (1.2.0)
 Requirement already satisfied: packaging>=16.8 in c:\users\accou\anaconda3\lib\site-packages (from bokeh!=3.0.*,>=1.4.0->backtesting) (23.2)
 Requirement already satisfied: pillow>=7.1.0 in c:\users\accou\anaconda3\lib\site-packages (from bokeh!=3.0.*,>=1.4.0->backtesting) (10.3.0)
 Requirement already satisfied: PyYAML>=3.10 in c:\users\accou\anaconda3\lib\site-packages (from bokeh!=3.0.*,>=1.4.0->backtesting) (6.0.1)
 Requirement already satisfied: tornado>=6.2 in c:\users\accou\anaconda3\lib\site-packages (from bokeh!=3.0.*,>=1.4.0->backtesting) (6.4.1)
 Requirement already satisfied: xyzservices>=2021.09.1 in c:\users\accou\anaconda3\lib\site-packages (from bokeh!=3.0.*,>=1.4.0->backtesting) (2022.9.0)
 Requirement already satisfied: python-dateutil>=2.8.2 in c:\users\accou\anaconda3\lib\site-packages (from pandas!=0.25.0,>=0.25.0->backtesting) (2.9.0.post0)
 Requirement already satisfied: pytz>=2020.1 in c:\users\accou\anaconda3\lib\site-packages (from pandas!=0.25.0,>=0.25.0->backtesting) (2024.1)
 Requirement already satisfied: tzdata>=2022.7 in c:\users\accou\anaconda3\lib\site-packages (from pandas!=0.25.0,>=0.25.0->backtesting) (2023.3)
 Requirement already satisfied: MarkupSafe>=2.0 in c:\users\accou\anaconda3\lib\site-packages (from Jinja2>=2.9->bokeh!=3.0.*,>=1.4.0->backtesting) (2.1.3)
 Requirement already satisfied: six>=1.5 in c:\users\accou\anaconda3\lib\site-packages (from python-dateutil>=2.8.2->pandas!=0.25.0,>=0.25.0->backtesting) (1.16.0)

In [5]: `pip install sambo`

Requirement already satisfied: sambo in c:\users\accou\anaconda3\lib\site-packages (1.25.0)
 Requirement already satisfied: numpy>=1.10.0 in c:\users\accou\anaconda3\lib\site-packages (from sambo) (1.26.4)
 Requirement already satisfied: scipy>=1.11.0 in c:\users\accou\anaconda3\lib\site-packages (from sambo) (1.13.1)
 Note: you may need to restart the kernel to use updated packages.

In [6]: `# import of libraries
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
import os
import yfinance as yf
import warnings
warnings.filterwarnings('ignore')`

In [7]: `# fetching of data
ticker = 'KO'
data = yf.download(ticker, start = '2014-01-01', end = '2024-12-31')`

[*****100%*****] 1 of 1 completed

In [8]: `data.reset_index(inplace= True)`

In [9]: `print(data.info())`

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2767 entries, 0 to 2766
Data columns (total 6 columns):
#   Column          Non-Null Count  Dtype  
---  -
0   (Date, )        2767 non-null   datetime64[ns]
1   (Close, KO)     2767 non-null   float64
2   (High, KO)      2767 non-null   float64
3   (Low, KO)       2767 non-null   float64
4   (Open, KO)      2767 non-null   float64
5   (Volume, KO)    2767 non-null   int64  
dtypes: datetime64[ns](1), float64(4), int64(1)
memory usage: 129.8 KB
None
```

In [10]: `print(data.head())`

Price Ticker	Date	Close KO	High KO	Low KO	Open KO	Volume KO
0	2014-01-02	28.724295	29.126971	28.618326	29.049261	12698600
1	2014-01-03	28.583017	28.773760	28.505307	28.738437	10403400
2	2014-01-06	28.448792	28.674856	28.371081	28.597146	11036300
3	2014-01-07	28.533560	28.766690	28.462916	28.597141	11872300
4	2014-01-08	28.215652	28.568878	28.137945	28.568878	15339600

In [11]: `print(data.tail())`

Price Ticker	Date	Close KO	High KO	Low KO	Open KO	Volume KO
2762	2024-12-23	62.380001	62.759998	61.529999	62.500000	13010800
2763	2024-12-24	62.840000	62.840000	62.009998	62.200001	5019100
2764	2024-12-26	62.570000	62.740002	62.400002	62.619999	7943800
2765	2024-12-27	62.450001	62.950001	62.209999	62.450001	8542800
2766	2024-12-30	62.029999	62.340000	61.680000	62.340000	8972200

Data Cleaning

```
In [12]: # checking for missing values
print(data.isnull().sum())
```

Price Ticker
Date 0
Close KO 0
High KO 0
Low KO 0
Open KO 0
Volume KO 0
dtype: int64

Feature Engineering

- adding moving averages: 20-day and 50-day for trend detection
- adding Daily returns: Hepls capture volatility

```
In [13]: # adding moving Averages
data['ma_20'] = data['Close'].rolling(window=20).mean()
data['ma_50'] = data['Close'].rolling(window=20).mean()

#add Daily returns
data['daily_return'] = data ['Close'].pct_change()

# Adding Volatility(Standard deviation of returns over a rolling window)
data['volatility']= data['daily_return'].rolling(window=20).std()
```

```
In [14]: data.dropna(inplace = True)
```

```
In [15]: data
```

Out[15]:

Price	Date	Close	High	Low	Open	Volume	ma_20	ma_50	daily_return	volatility
Ticker		KO	KO	KO	KO	KO				
20	2014-01-31	26.717985	26.958179	26.647341	26.809825	16185000	27.844065	27.844065	-0.009169	0.009834
21	2014-02-03	26.279987	26.859277	26.209341	26.859277	23164900	27.728913	27.728913	-0.016393	0.010243
22	2014-02-04	26.477781	26.527233	26.287039	26.449523	17759700	27.630363	27.630363	0.007526	0.010568
23	2014-02-05	26.569635	26.647345	26.407152	26.484862	13665200	27.532167	27.532167	0.003469	0.010584
24	2014-02-06	26.866337	27.509207	26.788627	27.382046	22661500	27.464701	27.464701	0.011167	0.010908
...
2762	2024-12-23	62.380001	62.759998	61.529999	62.500000	13010800	63.105224	63.105224	-0.002718	0.009255
2763	2024-12-24	62.840000	62.840000	62.009998	62.200001	5019100	63.052455	63.052455	0.007374	0.009263
2764	2024-12-26	62.570000	62.740002	62.400002	62.619999	7943800	62.977750	62.977750	-0.004297	0.009258
2765	2024-12-27	62.450001	62.950001	62.209999	62.450001	8542800	62.903000	62.903000	-0.001918	0.009258
2766	2024-12-30	62.029999	62.340000	61.680000	62.340000	8972200	62.800500	62.800500	-0.006725	0.009305

2747 rows × 10 columns

```
In [16]: data.info
```

```
Out[16]: <bound method DataFrame.info of Price
Ticker      KO      KO      Date      Close      High      Low      Open      Volume \
20      2014-01-31      26.717985      26.958179      26.647341      26.809825      16185000
21      2014-02-03      26.279987      26.859277      26.209341      26.859277      23164900
22      2014-02-04      26.477781      26.527233      26.287039      26.449523      17759700
23      2014-02-05      26.569635      26.647345      26.407152      26.484862      13665200
24      2014-02-06      26.866337      27.509207      26.788627      27.382046      22661500
...
2762      2024-12-23      62.380001      62.759998      61.529999      62.500000      13010800
2763      2024-12-24      62.840000      62.840000      62.009998      62.200001      5019100
2764      2024-12-26      62.570000      62.740002      62.400002      62.619999      7943800
2765      2024-12-27      62.450001      62.950001      62.209999      62.450001      8542800
2766      2024-12-30      62.029999      62.340000      61.680000      62.340000      8972200

Price      ma_20      ma_50      daily_return      volatility
Ticker
20      27.844065      27.844065      -0.009169      0.009834
21      27.728913      27.728913      -0.016393      0.010243
22      27.630363      27.630363      0.007526      0.010568
23      27.532167      27.532167      0.003469      0.010584
24      27.464701      27.464701      0.011167      0.010908
...
2762      63.105224      63.105224      -0.002718      0.009255
2763      63.052455      63.052455      0.007374      0.009263
2764      62.977750      62.977750      -0.004297      0.009258
2765      62.903000      62.903000      -0.001918      0.009258
2766      62.800500      62.800500      -0.006725      0.009305

[2747 rows x 10 columns]>
```

Exploratory Data Analysis(EDA)

summary statistics

```
In [17]: print(data.describe())
```

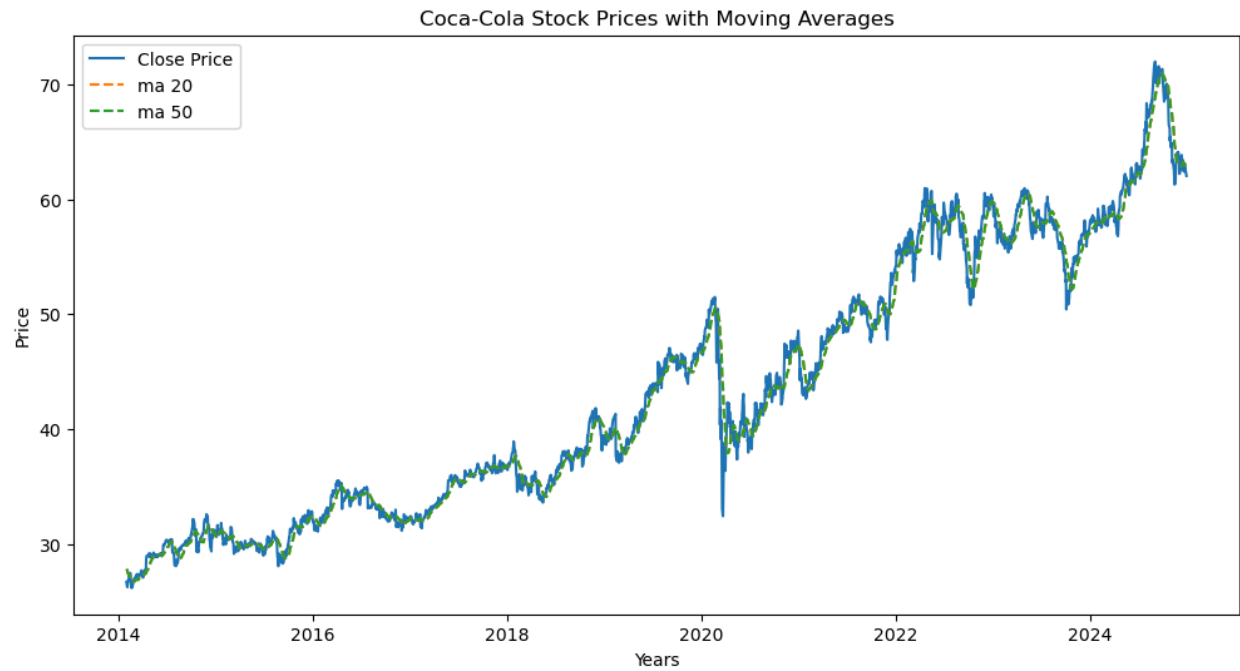
```
Price      Date      Close      High      Low \
Ticker      KO      KO      KO      KO
count      2747      2747.000000      2747.000000      2747.000000
mean      2019-07-15 18:20:18.784128      43.441487      43.730623      43.141359
min      2014-01-31 00:00:00      26.209337      26.506044      26.060979
25%      2016-10-20 12:00:00      33.312366      33.442178      33.140587
50%      2019-07-17 00:00:00      40.687668      41.057746      40.302547
75%      2022-04-05 12:00:00      54.825649      55.188294      54.262469
max      2024-12-30 00:00:00      71.967033      72.479605      71.217888
std      NaN      11.401624      11.483611      11.310104

Price      Open      Volume      ma_20      ma_50      daily_return \
Ticker      KO      KO
count      2747.000000      2.747000e+03      2747.000000      2747.000000      2747.000000
mean      43.439501      1.422592e+07      43.320436      43.320436      0.000365
min      26.181077      2.996300e+06      26.762136      26.762136      -0.096725
25%      33.342908      1.034565e+07      33.360800      33.360800      -0.004554
50%      40.743142      1.288670e+07      40.290018      40.290018      0.000649
75%      54.734846      1.620040e+07      54.439863      54.439863      0.005700
max      72.252895      6.784570e+07      70.896330      70.896330      0.064796
std      11.405039      6.218640e+06      11.348851      11.348851      0.011056

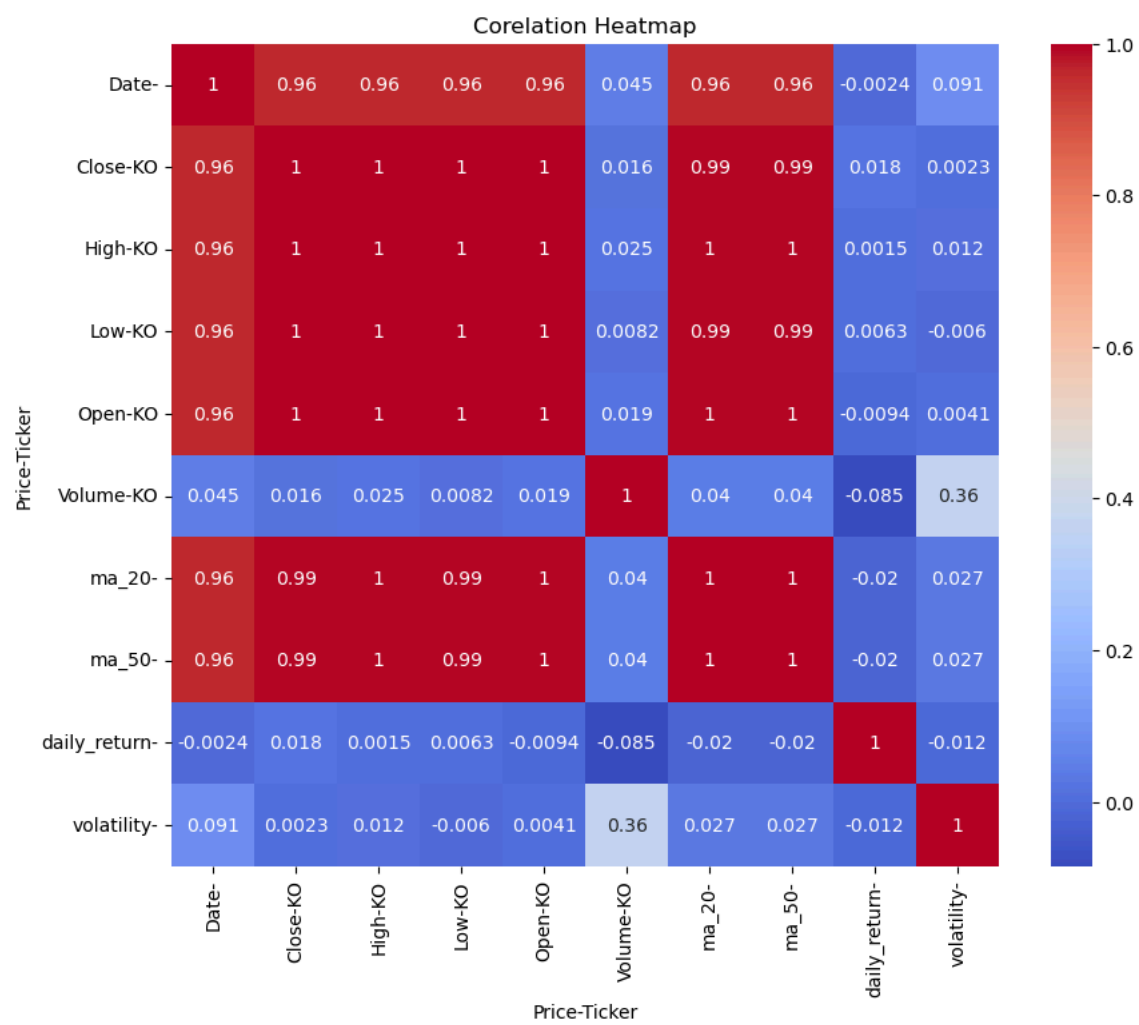
Price      volatility
Ticker
count      2747.000000
mean      0.009684
min      0.003597
25%      0.006694
50%      0.008585
75%      0.010786
max      0.053882
std      0.005452
```

Data Visualization

```
In [18]: # Line plot graph for stock prices
plt.figure(figsize = (12,6))
plt.plot(data['Date'],data['Close'], label = 'Close Price')
plt.plot(data['Date'],data['ma_20'], label = 'ma 20', linestyle= '--')
plt.plot(data['Date'],data['ma_50'], label = 'ma 50', linestyle= '--')
plt.title('Coca-Cola Stock Prices with Moving Averages')
plt.xlabel('Years')
plt.ylabel('Price')
plt.legend()
plt.show()
```



```
In [19]: # Correlation heat map
plt.figure(figsize=(10,8))
sns.heatmap(data.corr(), annot=True, cmap='coolwarm')
plt.title('Correlation Heatmap')
plt.show()
```



Data Splitting

```
In [20]: from sklearn.model_selection import train_test_split
```



```
In [21]: features = ['Open', 'High', 'Low', 'Volume', 'ma_20', 'ma_50', 'daily_return', 'volatility']
        target = 'Close'

In [22]: x = data[features]
        y = data[target]

In [23]: x_train, x_test, y_train, y_test = train_test_split(x, y, test_size = 0.2, random_state = 42, shuffle=False)
```

Model Training

```
In [24]: # using Random Forest for intial Predictions
        from sklearn.ensemble import RandomForestRegressor
        from sklearn.metrics import mean_absolute_error, mean_squared_error
```

```
In [25]: # Initialize the model
        model = RandomForestRegressor(n_estimators=100, random_state=42)

        # Train the model
        model.fit(x_train, y_train)

        # Predict on test set
        y_pred = model.predict(x_test)

        # Evaluate model
        mse = mean_squared_error(y_test, y_pred)
        mae = mean_absolute_error(y_test, y_pred)

        print(f"mean squared error: {mse}")
        print(f"mean absolute error: {mae}")
```

mean squared error: 1.5175407223094595
mean absolute error: 1.5175407223094595

Live Prediction System

Fetching Live stock data

```
In [26]: live_data = yf.download(ticker, period='1d', interval='1m')

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

Preparing live data for prediction

```
In [27]: live_data['ma_20'] = live_data['Close'].rolling(window=20).mean()
        live_data['ma_50'] = live_data['Close'].rolling(window=50).mean()
        live_data['daily_return'] = live_data['Close'].pct_change()
        live_data['volatility'] = live_data['daily_return'].rolling(window=20).std()
```

```
In [28]: # ensuring there are no missing value
        live_data.fillna(0, inplace=True)
```

```
In [29]: # using the latest data point for prediction
        latest_features = live_data[features].iloc[-1:].dropna()
        live_prediction = model.predict(latest_features)

        print(f"precited closing price : {live_prediction[0]}")
```

precited closing price : 60.69389156341553

Deploy the system

```
In [30]: import streamlit as st
```

```
In [32]: st.title('Coca-Cola Stock Price Prediction')
```

```
Out[32]: DeltaGenerator()
```

```
In [33]: data.columns
```

```
Out[33]: MultiIndex([(Date', ''),
 ('Close', 'K0'),
 ('High', 'K0'),
 ('Low', 'K0'),
 ('Open', 'K0'),
 ('Volume', 'K0'),
 ('ma_20', ''),
 ('ma_50', ''),
 ('daily_return', ''),
 ('volatility', '')],
 names=['Price', 'Ticker'])
```

```
In [34]: data.columns = ['_'.join(filter(None, col)) for col in data.columns]
```



```
In [35]: data.columns

Out[35]: Index(['Date', 'Close_KO', 'High_KO', 'Low_KO', 'Open_KO', 'Volume_KO',
               'ma_20', 'ma_50', 'daily_return', 'volatility'],
              dtype='object')

In [36]: st.line_chart(data[['Close_KO', 'ma_20', 'ma_50']])

Out[36]: DeltaGenerator()

In [37]: st.write(f"Predicted Closing Price: {live_prediction[0]}")
```

Stock Analysis of Coca Cola

```
In [38]: sns.set_style('whitegrid')
plt.style.use('fivethirtyeight')
from pandas_datareader.data import DataReader
```

For time series

```
In [39]: from datetime import datetime
from math import sqrt
from sklearn.metrics import mean_squared_error
from sklearn.preprocessing import MinMaxScaler
```

Data set loading

```
In [40]: ko_data = pd.read_csv(r"C:\Users\accou\OneDrive\Desktop\DATA ANALYST AND BUSNIENSS ANALYST\INTERNSHIP\UNIFIED MENTOR\PROJECT\COCA
```

```
In [41]: ko_data
```

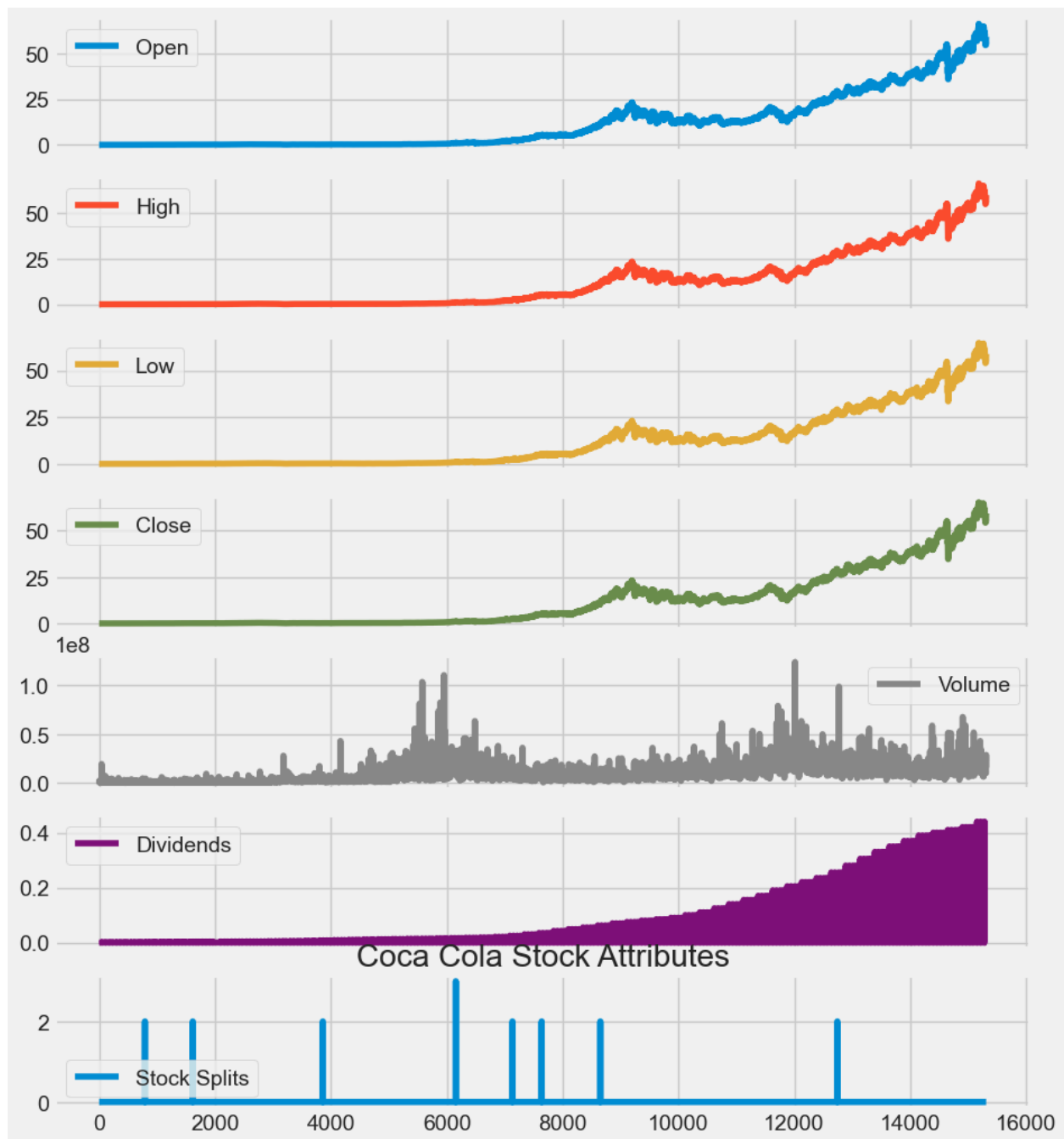
```
Out[41]:
```

	Date	Open	High	Low	Close	Volume	Dividends	Stock Splits
0	1962-01-02	0.050016	0.051378	0.050016	0.050016	806400	0.0	0
1	1962-01-03	0.049273	0.049273	0.048159	0.048902	1574400	0.0	0
2	1962-01-04	0.049026	0.049645	0.049026	0.049273	844800	0.0	0
3	1962-01-05	0.049273	0.049892	0.048035	0.048159	1420800	0.0	0
4	1962-01-08	0.047787	0.047787	0.046735	0.047664	2035200	0.0	0
...
15306	2022-10-20 00:00:00-04:00	55.770000	55.919998	54.959999	55.080002	16905100	0.0	0
15307	2022-10-21 00:00:00-04:00	55.000000	56.110001	54.990002	55.959999	15028000	0.0	0
15308	2022-10-24 00:00:00-04:00	56.639999	57.730000	56.570000	57.570000	17416700	0.0	0
15309	2022-10-25 00:00:00-04:00	59.040001	59.110001	57.750000	58.950001	28829900	0.0	0
15310	2022-10-26 00:00:00-04:00	59.009998	59.779999	58.860001	59.389999	15831400	0.0	0

15311 rows × 8 columns

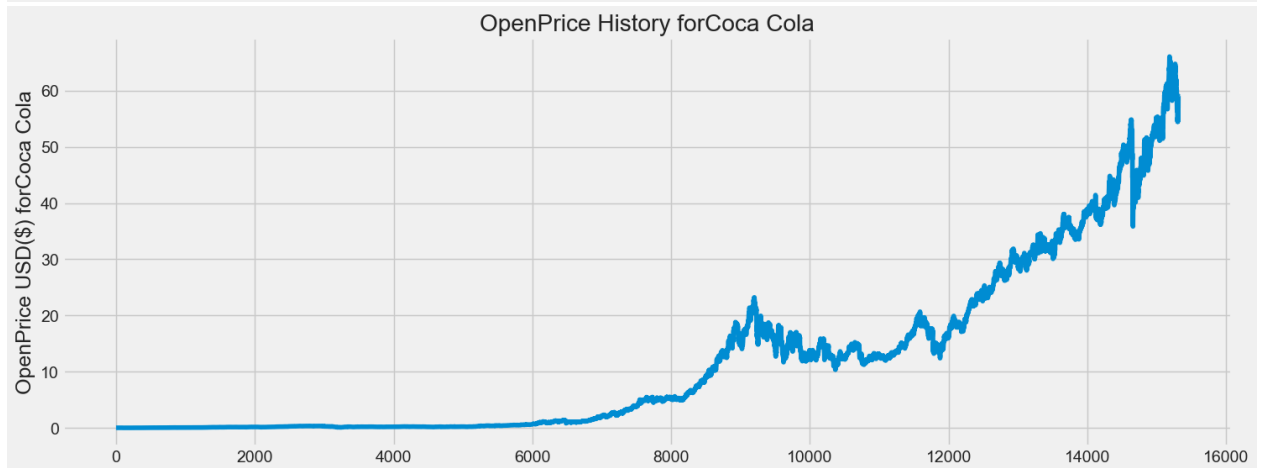
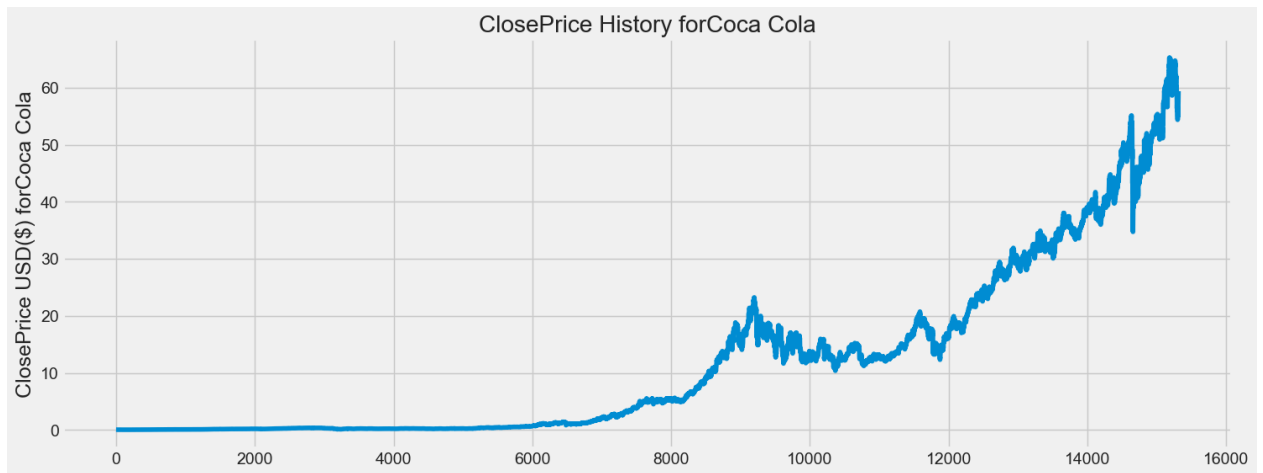
Basic EDA

```
In [42]: ko_data.plot(subplots = True, figsize = (10,12))
plt.title('Coca Cola Stock Attributes')
plt.show()
```



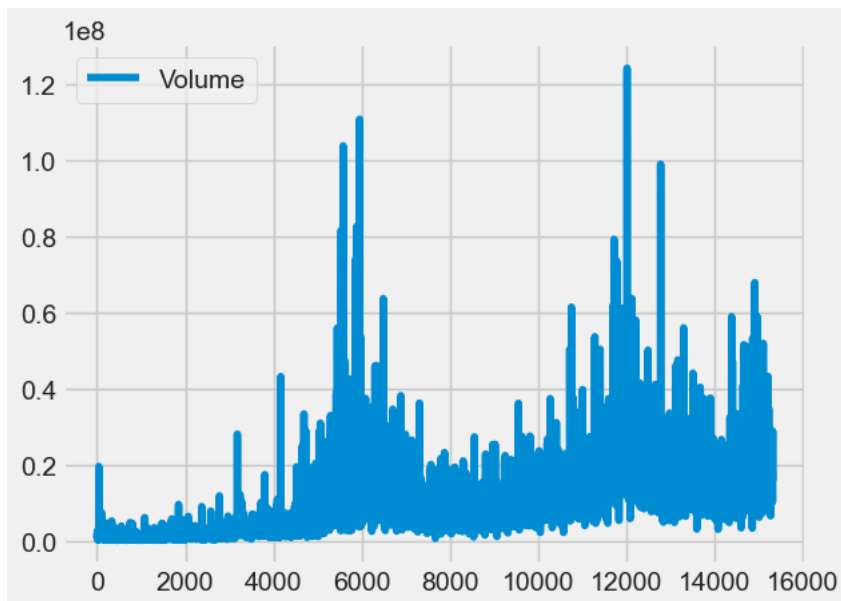
```
In [43]: def plot_close_val(data_frame, column, stock):
plt.figure(figsize = (16,6))
plt.title(column + 'Price History for' + stock)
plt.plot(data_frame[column])
plt.ylabel(column + 'Price USD($)' for' + stock, fontsize = 18)
plt.show()
```

```
In [44]: plot_close_val(ko_data, 'Close', 'Coca Cola')
plot_close_val(ko_data, 'Open', 'Coca Cola')
```



In [45]: `ko_data[['Volume']].plot()`

Out[45]: <Axes: >



Basic Company Info

In [46]: `ko_info = pd.read_csv(r'C:\Users\accou\OneDrive\Desktop\DATA ANALYST AND BUSNIENSS ANALYST\INTERNSHIP\UNIFIED MENTOR\PROJECT\COCA
names=(['Description','Information']))`

In [47]: `ko_info`

Out[47]:

Description		Information
0	Key	Value
1	zip	30313
2	sector	Consumer Defensive
3	fullTimeEmployees	80300
4	longBusinessSummary	The Coca-Cola Company, a beverage company, man...
...
149	dayHigh	60.345
150	regularMarketPrice	59.6
151	preMarketPrice	NaN
152	logo_url	https://logo.clearbit.com/coca-colacompany.com
153	trailingPegRatio	2.6848

154 rows × 2 columns

```
In [48]: ko_info.dropna()
ko_info.drop(ko_info.loc[ko_info['Information'] == 'nan'].index, inplace = True)

In [49]: ko = ko_info.sort_values('Information').style

In [50]: ko
```

Out[50]:

	Description	Information
49	gmtOffsetMilliseconds	-18000000
145	bid	0
138	ask	0
77	heldPercentInsiders	0.00636
68	sharesPercentSharesOut	0.0074
100	shortPercentOfFloat	0.0074
108	trailingAnnualDividendYield	0.027917083
147	dividendYield	0.028099999
29	returnOnAssets	0.07831
75	SandP52WeekChange	0.15025425
18	revenueGrowth	0.161
58	52WeekChange	0.21709049
15	profitMargins	0.23313999
19	operatingMargins	0.31123
14	ebitdaMargins	0.35199
33	returnOnEquity	0.39722002
74	lastDividendValue	0.42
93	earningsQuarterlyGrowth	0.423
27	earningsGrowth	0.425
16	grossMargins	0.60723996
71	heldPercentInstitutions	0.7005
84	beta	0.712113
109	payoutRatio	0.82269996
11	maxAge	1
41	quickRatio	1.173
28	currentRatio	1.516
81	shortRatio	1.6
116	trailingAnnualDividendRate	1.67
122	dividendRate	1.68
139	askSize	1000
76	priceToBook	11.606621
17	operatingCashflow	12855000064
20	ebitda	13306000384
88	lastSplitDate	1344816000
35	totalCash	14871000064
70	lastFiscalYearEnd	1609372800
80	mostRecentQuarter	1633046400
91	lastDividendDate	1638230400
82	sharesShortPreviousMonthDate	1638230400
123	exDividendDate	1638230400
95	dateShortInterest	1640908800
78	nextFiscalYearEnd	1672444800
32	debtToEquity	172.826
136	averageVolume	17746368
140	volume	18219394
129	regularMarketVolume	18219394
23	grossProfits	19581000000
86	priceHint	2
73	trailingEps	2.031
42	recommendationMean	2.1

	Description	Information
60	forwardEps	2.43
153	trailingPegRatio	2.6848
96	pegRatio	2.77
119	averageVolume10days	20867790
113	averageDailyVolume10Day	20867790
57	enterpriseToEbitda	21.583
98	forwardPE	24.526747
101	sharesShortPriorMonth	24026403
30	numberOfAnalystOpinions	25
133	marketCap	257437417472
85	enterpriseValue	287178719232
128	trailingPE	29.34515
89	lastSplitFactor	2:1
143	fiveYearAvgDividendYield	3.21
38	totalCashPerShare	3.443
1	zip	30313
67	sharesShort	31874471
37	totalRevenue	37802000384
83	floatShares	3890760972
6	phone	404 676 2121
36	totalDebt	41707999232
102	impliedSharesOutstanding	4311130112
62	sharesOutstanding	4319419904
144	fiftyTwoWeekLow	48.11
66	bookValue	5.135
107	twoHundredDayAverage	55.77645
115	fiftyDayAverage	57.6512
21	targetLowPrice	58
137	dayLow	59.21
126	regularMarketDayLow	59.21
150	regularMarketPrice	59.6
26	currentPrice	59.6
117	open	59.79
106	regularMarketOpen	59.79
114	regularMarketPreviousClose	59.82
105	previousClose	59.82
94	priceToSalesTrailing12Months	6.8101535
149	dayHigh	60.345
111	regularMarketDayHigh	60.345
141	fiftyTwoWeekHigh	61.45
31	targetMeanPrice	63.72
25	targetMedianPrice	64
55	enterpriseToRevenue	7.597
34	targetHighPrice	70
24	freeCashflow	7007374848
40	revenuePerShare	8.771
148	bidSize	800
3	fullTimeEmployees	80300
72	netIncomeToCommon	8812999680
46	exchangeTimezoneName	America/New_York

	Description	Information
5	city	Atlanta
13	industry	Beverages—Non-Alcoholic
44	shortName	Coca-Cola Company (The)
2	sector	Consumer Defensive
50	quoteType	EQUITY
47	exchangeTimezoneShortName	EST
146	tradeable	False
48	isEsgPopulated	False
7	state	GA
51	symbol	KO
43	exchange	NYSE
12	address1	One Coca-Cola Plaza
45	longName	The Coca-Cola Company
4	longBusinessSummary	<p>The Coca-Cola Company, a beverage company, manufactures, markets, and sells various nonalcoholic beverages worldwide. The company provides sparkling soft drinks; water, enhanced water, and sports drinks; juice, dairy, and plant-based beverages; tea and coffee; and energy drinks. It also offers beverage concentrates and syrups, as well as fountain syrups to fountain retailers, such as restaurants and convenience stores. The company sells its products under the Coca-Cola, Diet Coke/Coca-Cola Light, Coca-Cola Zero Sugar, Fanta, Fresca, Schweppes, Sprite, Thums Up, Aquarius, Ciel, Dasani, glac��au smartwater, glac��au vitaminwater, Ice Dew, I LOHAS, Powerade, Topo Chico, AdeS, Del Valle, fairlife, innocent, Minute Maid, Minute Maid Pulpy, Simply, Ayataka, Costa, dogadan, FUZE TEA, Georgia, Gold Peak, HONEST TEA, and Kochakaden brands. It operates through a network of independent bottling partners, distributors, wholesalers, and retailers, as well as through bottling and distribution operators. The company was founded in 1886 and is headquartered in Atlanta, Georgia.</p>
127	currency	USD
39	financialCurrency	USD
8	country	United States
0	Key	Value
9	companyOfficers	[]
22	recommendationKey	buy
52	messageBoardId	finmb_26642
152	logo_url	https://logo.clearbit.com/coca-colacompany.com
10	website	https://www.coca-colacompany.com
53	market	us_market
54	annualHoldingsTurnover	nan
56	beta3Year	nan
59	morningStarRiskRating	nan
61	revenueQuarterlyGrowth	nan
63	fundInceptionDate	nan
64	annualReportExpenseRatio	nan
65	totalAssets	nan
69	fundFamily	nan
79	yield	nan
87	threeYearAverageReturn	nan
90	legalType	nan
92	morningStarOverallRating	nan
97	ytdReturn	nan
99	lastCapGain	nan
103	category	nan
104	fiveYearAverageReturn	nan
110	volume24Hr	nan
112	navPrice	nan
118	toCurrency	nan
120	expireDate	nan
121	algorithm	nan

	Description	Information
124	circulatingSupply	nan
125	startDate	nan
130	lastMarket	nan
131	maxSupply	nan
132	openInterest	nan
134	volumeAllCurrencies	nan
135	strikePrice	nan
142	fromCurrency	nan
151	preMarketPrice	nan

Basic CAGR

unflod_moreshow hidden cell

Basic Rolling Averages

```
In [51]: # Isolate the adjusted closing prices
adj_close_px = ko_data['Close']

# Calucate the moving average
moving_avg = adj_close_px.rolling(window= 40).mean()

# Inspect the result
moving_avg[-10:]
```

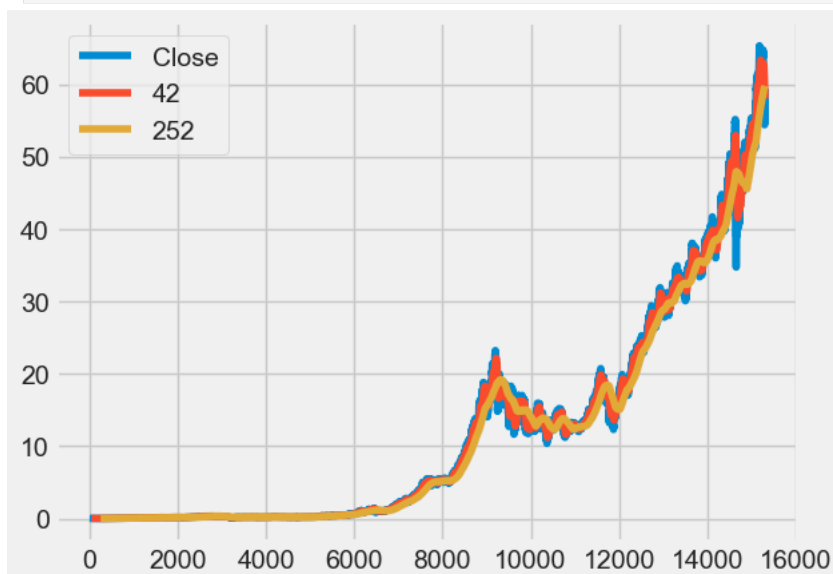
```
Out[51]: 15301    59.573229
15302    59.329031
15303    59.103823
15304    58.921440
15305    58.725320
15306    58.504966
15307    58.298918
15308    58.171838
15309    58.088689
15310    58.030935
Name: Close, dtype: float64
```

```
In [52]: # Short moving window rolling mean
ko_data['42'] = adj_close_px.rolling(window= 40). mean()

# Long moving window rolling mean
ko_data['252'] = adj_close_px.rolling(window= 252). mean()

# Plot the adjusted closing price, the short and long windows of rooling means
ko_data[['Close', '42', '252']].plot()

plt.show()
```



```
In [53]: daily_close_px = ko_data[['Close']]

# Calculate the daily percentage change for 'daily_close_px'
```

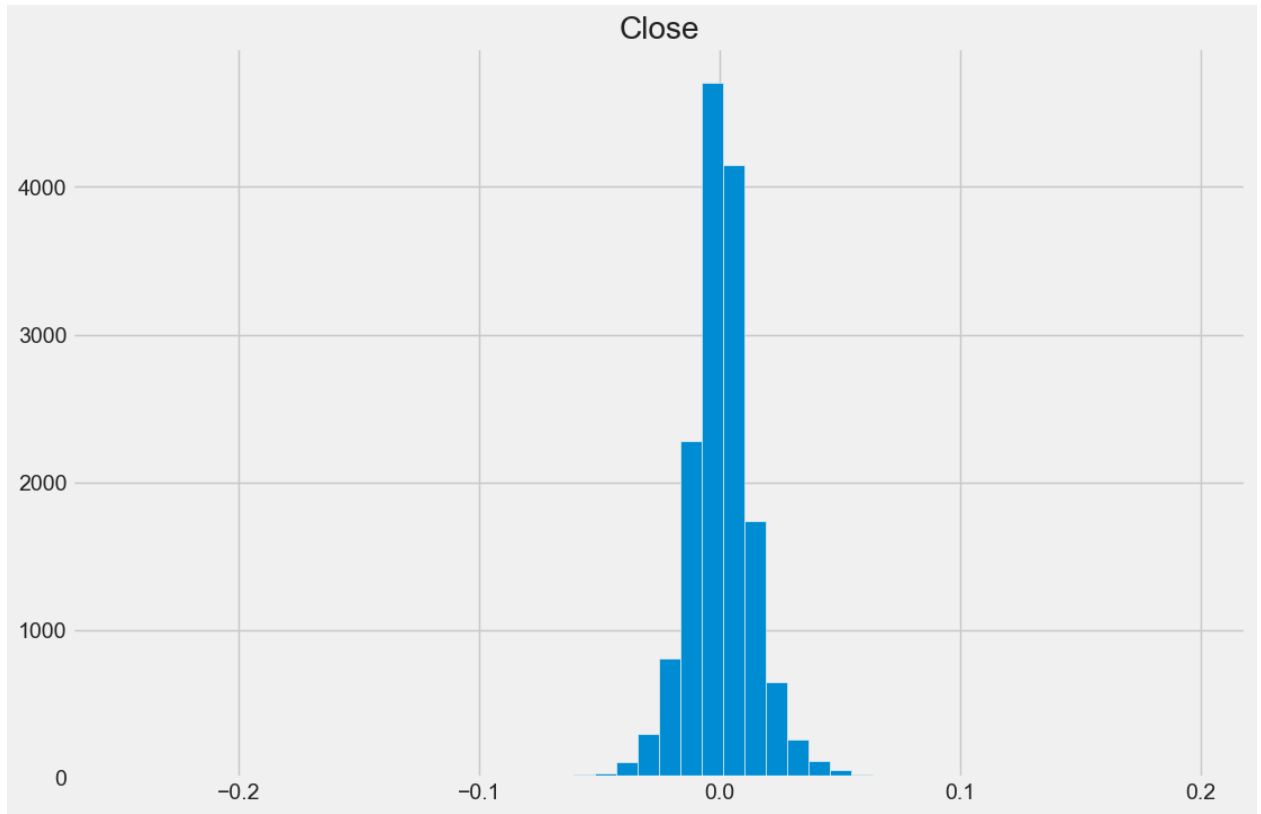
```

daily_pct_change = daily_close_px.pct_change()

#plot the distributions
daily_pct_change.hist(bins =50, sharex =True, figsize= (12,8))

#showing the resulting plot
plt.show()

```



```

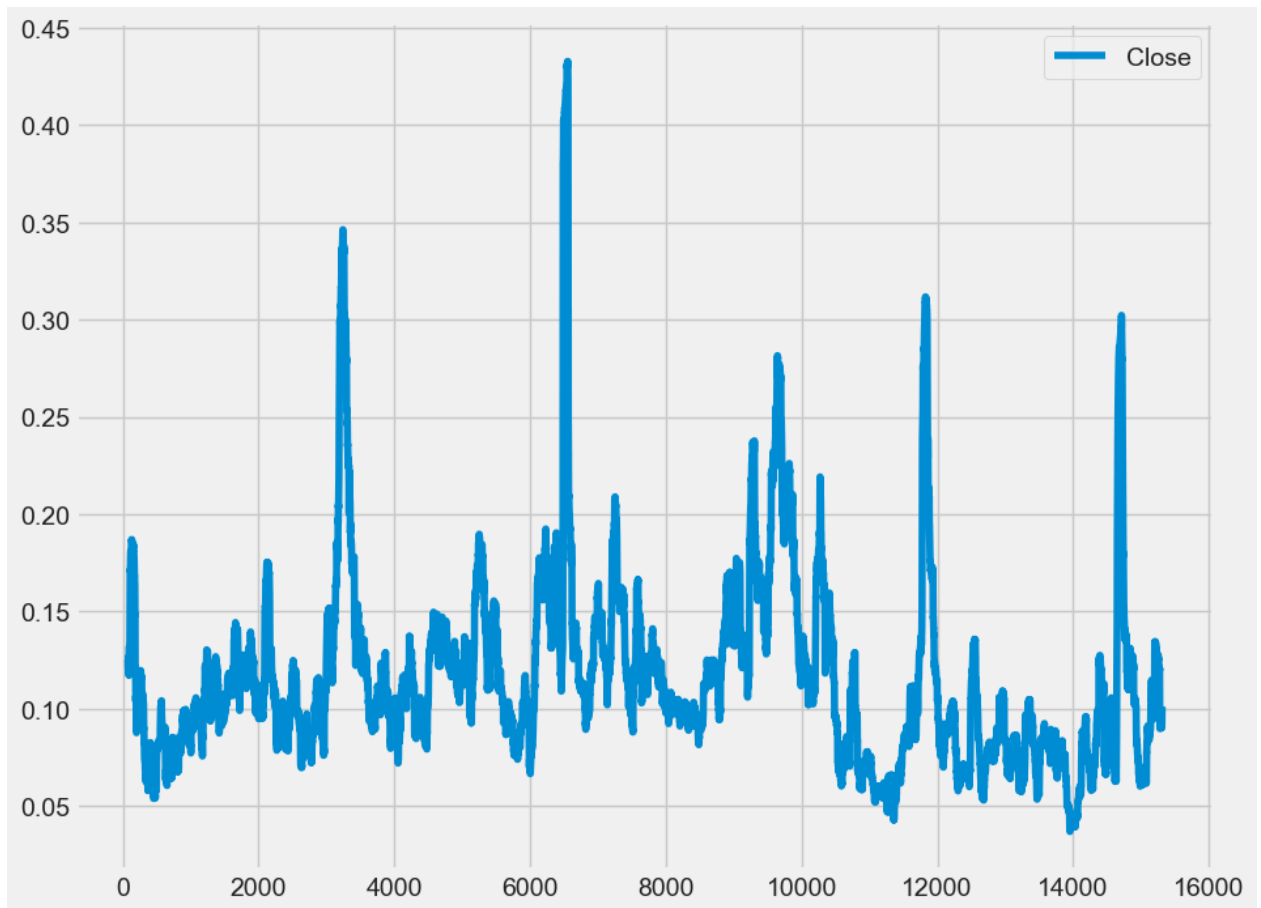
In [54]: # Define the minimum of periods to consider
min_periods = 75

#Calculate the volatility
vol = daily_pct_change.rolling(min_periods).std() * np.sqrt(min_periods)

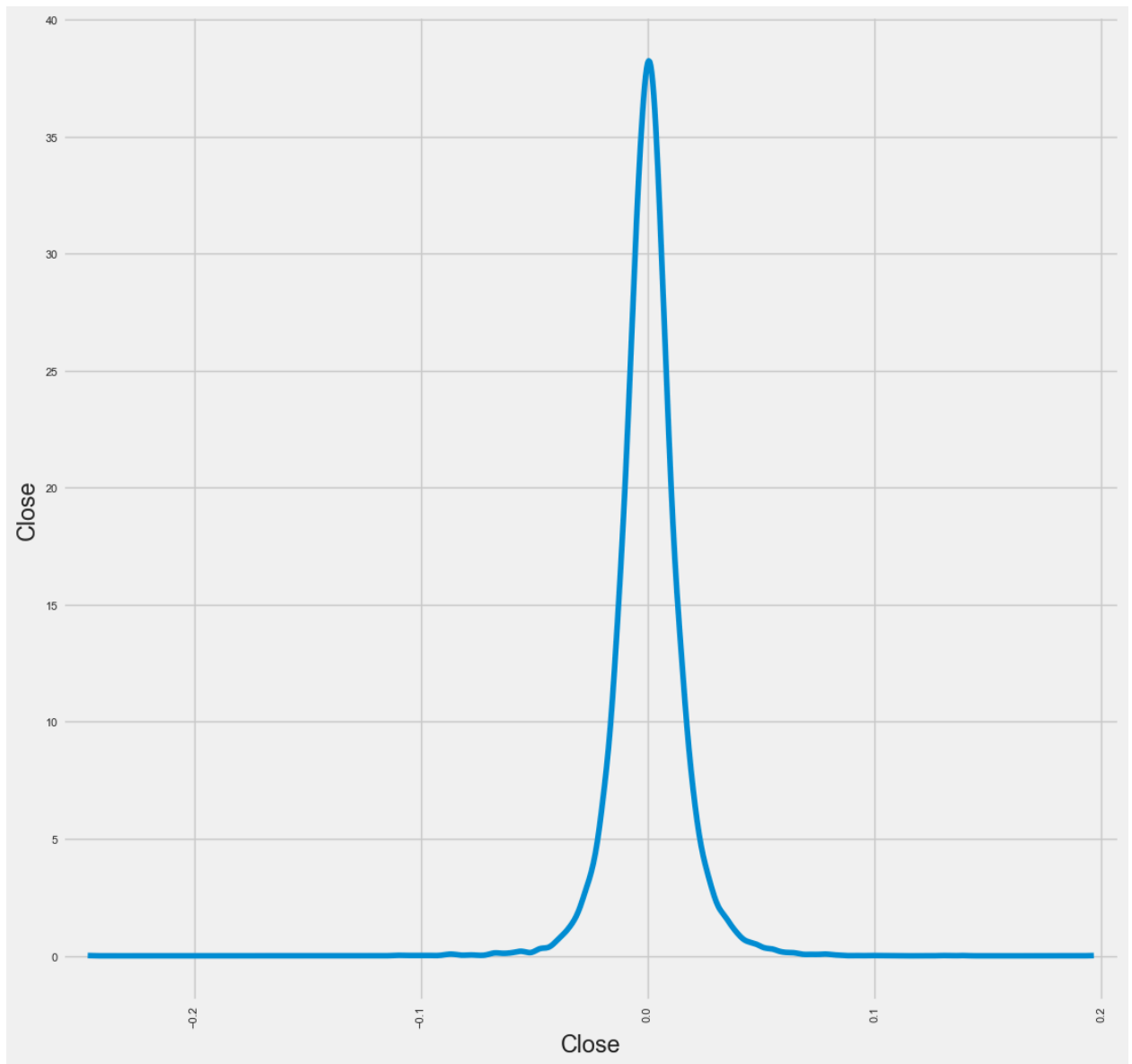
#plot the volatility
vol.plot(figsize = (10,8))

plt.show()

```



```
In [55]: # Plot a scatter matrix with the `daily_pct_change` data
pd.plotting.scatter_matrix(daily_pct_change, diagonal = 'kde', alpha = 0.1, figsize = (12,12))
plt.show()
```



Basic MACD

```
In [56]: import plotly.graph_objects as go
```

```
In [57]: ko_data = ko_data.reset_index()
```

```
In [58]: fig = go.Figure(data=go.Ohlc(x=ko_data['Date'],  
open=ko_data['Open'],  
high=ko_data['High'],  
low=ko_data['Low'],  
close=ko_data['Close']))  
fig.show()
```

60
50
40
30

Basic SMA

```
In [59]: ko_data = ko_data.reset_index()
```

```
In [60]: ko_data['sma5'] = ko_data.Close.rolling(5).mean()  
ko_data['sma20'] = ko_data.Close.rolling(20).mean()  
ko_data['sma50'] = ko_data.Close.rolling(50).mean()  
ko_data['sma200'] = ko_data.Close.rolling(200).mean()  
ko_data['sma500'] = ko_data.Close.rolling(500).mean()
```

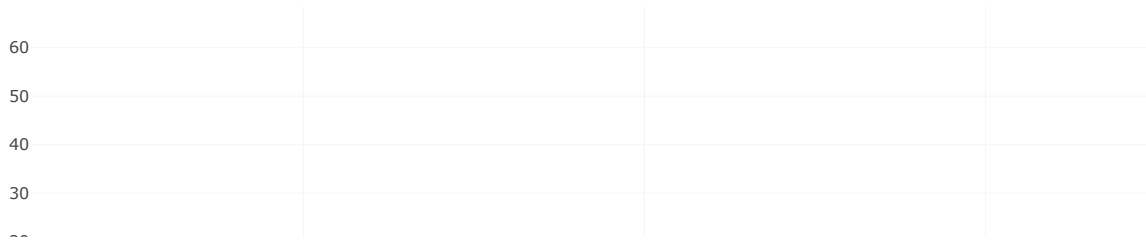
```
In [61]: fig = go.Figure(data=[go.Ohlc(x=ko_data['Date'],  
open = ko_data['Open'],high =ko_data['High'],low= ko_data['Low'],close = ko_data['Close'], name = 'OHLC'),  
go.Scatter(x=ko_data.Date,y=ko_data.sma5, line=dict(color = 'orange', width =1),name = 'sma5'),  
go.Scatter(x=ko_data.Date, y=ko_data.sma20, line =dict(color = 'green', width = 1), name = 'sma20'),  
go.Scatter(x=ko_data.Date, y= ko_data.sma50, line =dict(color= 'blue', width =1), name = 'sma50'),  
go.Scatter(x=ko_data.Date, y=ko_data.sma200, line =dict(color = 'violet', width = 1), name = 'sma200'),  
go.Scatter(x=ko_data.Date, y=ko_data.sma500, line =dict(color = 'purple', width = 1), name = 'sma500')])  
fig.show()
```

60
50
40
30

BASIC EMA

```
In [62]: ko_data['ema5'] = ko_data.Close.ewm(span =5, adjust =False).mean()
ko_data['ema20'] = ko_data.Close.ewm(span = 20, adjust = False).mean()
ko_data['ema50'] = ko_data.Close.ewm(span = 50, adjust = False).mean()
ko_data['ema200'] = ko_data.Close.ewm(span = 50, adjust = False).mean()
ko_data['ema500'] = ko_data.Close.ewm(span = 50, adjust = False).mean()
```

```
In [63]: fig= go.Figure(data = [go.Ohlc(x =ko_data['Date'],
open = ko_data['Open'], high= ko_data['High'], low = ko_data['Low'], close = ['Close'], name = 'OHLC'),
go.Scatter(x = ko_data.Date, y = ko_data.sma5, line = dict(color = 'orange', width = 1), name = 'sma5'),
go.Scatter(x = ko_data.Date, y = ko_data.sma20, line = dict(color = 'orange', width = 1), name = 'sma20'),
go.Scatter(x = ko_data.Date, y = ko_data.sma50, line = dict(color = 'orange', width = 1), name = 'sma50'),
go.Scatter(x = ko_data.Date, y = ko_data.sma200, line = dict(color = 'orange', width = 1), name = 'sma200'),
go.Scatter(x = ko_data.Date, y = ko_data.sma500, line = dict(color = 'orange', width = 1), name = 'sma500'))])
fig.show()
```



```
In [64]: ko_data.head()
```

```
Out[64]:
```

	level_0	index	Date	Open	High	Low	Close	Volume	Dividends	Stock Splits	...	sma5	sma20	sma50	sma200	sma500
0	0	0	1962-01-02	0.050016	0.051378	0.050016	0.050016	806400	0.0	0	...	NaN	NaN	NaN	NaN	NaN
1	1	1	1962-01-03	0.049273	0.049273	0.048159	0.048902	1574400	0.0	0	...	NaN	NaN	NaN	NaN	NaN
2	2	2	1962-01-04	0.049026	0.049645	0.049026	0.049273	844800	0.0	0	...	NaN	NaN	NaN	NaN	NaN
3	3	3	1962-01-05	0.049273	0.049892	0.048035	0.048159	1420800	0.0	0	...	NaN	NaN	NaN	NaN	NaN
4	4	4	1962-01-08	0.047787	0.047787	0.046735	0.047664	2035200	0.0	0	...	0.048803	NaN	NaN	NaN	NaN

5 rows × 22 columns



6 FINTA Tech Analysis Ratios

- Simple Moving Average 'SMA'
- Simple Moving Median 'SMM'
- Smoothed Simple Moving Average 'SSMA'
- Exponential Moving Average 'EM'
- Double Exponential Moving Average 'DE-A'
- Triple Exponential Moving Average 'T-IMA'
- Triangular Moving Average 'T-IMA'
- Triple Exponential Moving Average Oscillator -TRIX'
- Volume Adjusted Moving Average-'VAMA'
- Kaufman Efficiency Indica-or 'ER'
- Kaufman's Adaptive Moving Avera-e 'KAMA'
- Zero Lag Exponential Moving Avera-e 'ZLEMA'
- Weighted Moving Av-rage 'WMA'
- Hull Moving A-erage 'HMA'
- Elastic Volume Moving
- Volume Weighted Average Price 'VWAP'
- Smoothed Moving Average 'SMMA'
- Fractal Adaptive Moving Average 'FRAMA'
- Moving Average Convergence Divergence 'MAC'
- Percentage Price Oscillator 'P-O'
- Volume-Weighted MACD 'VW_M-CD'
- Elastic-Volume weighted MACD 'EV_-ACD'
- Market Momentum-'MOM'
- Rate-of-Chang- 'ROC'
- Relative Strenght Ind-x 'RSI'
- Inverse Fisher Transform RSI -IFT_RSI'
- True -ange 'TR'
- Average True -ange 'ATR'
- Stop-and-R-verse 'SAR'
- Bollinger B-

nds 'BBANDS' • Bollinger Bands W-dth 'BBWIDTH' • Momentum Breako-t Bands 'MOBO' • Perce-t B 'PERCENT_B' • Kelt-er Channels 'KC' • Don-hian Channel 'DO' • Directional Movem-nt Indicator 'DMI' • Average Dir-ctional Index 'ADX' •-Pivot Points 'PIVOT' • Fibonacci Pi-ot Points 'PIVOT_FIB' • Stochasti- Oscillator %K 'STOCH' • Stochasti- oscillator %D 'STOCHD' • -tochastic RSI 'STOCHRSI'- • Williams %R 'WILLIAMS'

- Awesome Oscillator 'AO'

-• Mass Index 'MI'- • Vortex Indicator 'VORTEX'- • Know Sure Thing 'KS-' • True Strength Index 'T-I' • Typical Price -TP' • Accumulation-Distribution Line -ADL' • Chaikin Oscillator 'CH-IKIN' • Money Flow Inde- 'MFI' • On Balance Volu-e 'OBV' • Weighter O-V 'WOBV' • Volume Zone Oscill-tor 'VZO' • Price Zone Oscil-ator 'PZO' • Elder's Force-Index 'EFI' • Cumulative Forc- Index 'CFI' • Bull power and Bea- Power 'EBBP' • Ease-of-Movement 'EMV' • Commodity Cha-nel Index 'CCI' • Cop-ock Curve 'COPP' • Buy and Se-l Pressure 'BASP' • Norm-lized BASP 'BASPN' • Chande Momen-um Oscillator 'CMO' • Chandel-er Exit 'CHANDELI-R' • Qstick 'QSTICK' • T-iggs Money Index 'TMF' • Wav- Trend Oscillator 'WTO'

- Fisher Transform 'FISH'

•-Ichimoku Cloud 'ICHIMOKU'

- Squeeze Momentum Indicator 'SQZMI'

-• Volume Price Trend 'VPT'- • Finite Volume Element 'FVE'- • Volume Flow Indicator 'VF-' • Moving Standard deviation 'M-D' • Schaff Trend Cycle 'STC' • Adaptive Price Zone 'APZ' • Ultimate Oscillator 'UO' Average 'EVWMA'

```
In [65]: try:
        from finta import TA
        from backtesting import Backtest, Strategy
        from backtesting.lib import crossover

    except:
        !pip install finta backtesting
        from finta import TA
        from backtesting import Backtest, Strategy
        from backtesting.lib import crossover
```



BokehJS 3.4.1 successfully loaded.

```
In [66]: fin_ma = pd.read_csv(r'C:\Users\accou\OneDrive\Desktop\DATA ANALYST AND BUSNIENSS ANALYST\INTERNSHIP\UNIFIED MENTOR\PROJECT\COCA
```

```
In [67]: print(fin_ma.head())
         ohlc = fin_ma
         print(TA.SMA(ohlc, 42))
```

	Date	Open	High	Low	Close	Volume	Dividends	\
0	1962-01-02	0.050016	0.051378	0.050016	0.050016	806400	0.0	
1	1962-01-03	0.049273	0.049273	0.048159	0.048902	1574400	0.0	
2	1962-01-04	0.049026	0.049645	0.049026	0.049273	844800	0.0	
3	1962-01-05	0.049273	0.049892	0.048035	0.048159	1420800	0.0	
4	1962-01-08	0.047787	0.047787	0.046735	0.047664	2035200	0.0	

Stock Splits

0	0
1	0
2	0
3	0
4	0
0	NaN
1	NaN
2	NaN
3	NaN
4	NaN

...	
15306	58.759467
15307	58.572686
15308	58.422110
15309	58.297065
15310	58.219369

Name: 42 period SMA, Length: 15311, dtype: float64

```
In [ ]:
```

```
In [68]: function_dict = {' Simple Moving Average ' : 'SMA',
                        ' Simple Moving Median ' : 'SMM',
                        ' Smoothed Simple Moving Average ' : 'SSMA',
                        ' Exponential Moving Average ' : 'EMA',
                        ' Double Exponential Moving Average ' : 'DEMA',
                        ' Triple Exponential Moving Average ' : 'TEMA',
                        ' Triangular Moving Average ' : 'TRIMA',
                        ' Triple Exponential Moving Average Oscillator': 'TRIX',
                        ' Volume Adjusted Moving Average ' : 'VAMA',
                        ' Kaufman Efficiency Indicator ' : 'ER',
                        ' Kaufmans Adaptive Moving Average ' : 'KAMA',
                        ' Zero Lag Exponential Moving Average ' : 'ZLEMA',
                        ' Weighted Moving Average ' : 'WMA',
```



```
' Hull Moving Average ' : 'HMA',
' Elastic Volume Moving Average ' : 'EVWMA',
' Volume Weighted Average Price ' : 'VWAP',
' Smoothed Moving Average ' : 'SMMA',
' Fractal Adaptive Moving Average ' : 'FRAMA',
' Moving Average Convergence Divergence ' : 'MACD',
' Percentage Price Oscillator ' : 'PPO',
' Volume-Weighted MACD ' : 'VW_MACD',
' Elastic-Volume weighted MACD ' : 'EV_MACD',
' Market Momentum ' : 'MOM',
' Rate-of-Change ' : 'ROC',
' Relative Strength Index ' : 'RSI',
' Inverse Fisher Transform RSI ' : 'IFT_RSI',
' True Range ' : 'TR',
' Average True Range ' : 'ATR',
' Stop-and-Reverse ' : 'SAR',
' Bollinger Bands ' : 'BBANDS',
' Bollinger Bands Width ' : 'BBWIDTH',
' Momentum Breakout Bands ' : 'MOBO',
' Percent B ' : 'PERCENT_B',
' Keltner Channels ' : 'KC',
' Donchian Channel ' : 'DO',
' Directional Movement Indicator ' : 'DMI',
' Average Directional Index ' : 'ADX',
' Pivot Points ' : 'PIVOT',
' Fibonacci Pivot Points ' : 'PIVOT_FIB',
' Stochastic Oscillator Percent K ' : 'STOCH',
' Stochastic oscillator Percent D ' : 'STOCHD',
' Stochastic RSI ' : 'STOCHRSI',
' Williams Percent R ' : 'WILLIAMS',
' Ultimate Oscillator ' : 'UO',
' Awesome Oscillator ' : 'AO',
' Mass Index ' : 'MI',
' Know Sure Thing ' : 'KST',
' True Strength Index ' : 'TSI',
' Typical Price ' : 'TP',
' Accumulation-Distribution Line ' : 'ADL',
' Chaikin Oscillator ' : 'CHAIKIN',
' Money Flow Index ' : 'MFI',
' On Balance Volume ' : 'OBV',
' Weighter OBV ' : 'WOBV',
' Volume Zone Oscillator ' : 'VZO',
' Price Zone Oscillator ' : 'PZO',
' Elders Force Index ' : 'EFI',
' Cumulative Force Index ' : 'CFI',
' Bull power and Bear Power ' : 'EBBP',
' Ease of Movement ' : 'EMV',
' Commodity Channel Index ' : 'CCI',
' Coppock Curve ' : 'COPP',
' Buy and Sell Pressure ' : 'BASP',
' Normalized BASP ' : 'BASPN',
' Chande Momentum Oscillator ' : 'CMO',
' Chandelier Exit ' : 'CHANDELIER',
' Qstick ' : 'QSTICK',
' Wave Trend Oscillator ' : 'WTO',
' Fisher Transform ' : 'FISH',
' Ichimoku Cloud ' : 'ICHIMOKU',
' Adaptive Price Zone ' : 'APZ',
' Volume Price Trend ' : 'VPT',
' Finite Volume Element ' : 'FVE',
' Volume Flow Indicator ' : 'VFI',
' Moving Standard deviation ' : 'MSD',
' Schaff Trend Cycle ' : 'STC'}
```

```
In [69]: for key, value in function_dict.items():
         function_name = 'TA.' + value + "(ohlcv).plot(title='" + key + " for Coca Cola / Coke Stock')"
         result = eval(function_name)
```

```

-----
AttributeError                                Traceback (most recent call last)
~\AppData\Local\Temp\ipykernel_19348\393203403.py in ?()
      1 for key, value in function_dict.items():
      2     function_name = 'TA.' + value + "(ohlcv.plot(title='" + key + " for Coca Cola / Coke Stock'))"
----> 3     result = eval(function_name)

<string> in ?()
----> 1 'Could not get source, probably due dynamically evaluated source code.'

~\anaconda3\Lib\site-packages\finta\finta.py in ?(*args, **kwargs)
     30         raise LookupError(
     31             'Must have a dataframe column named "{0}"'.format(inputs[1])
     32         )
     33
--> 34     return func(*args, **kwargs)

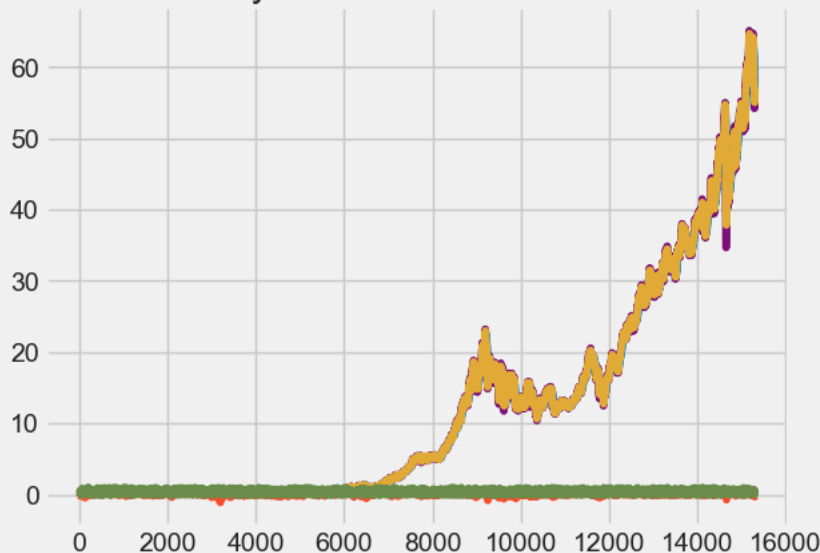
~\anaconda3\Lib\site-packages\finta\finta.py in ?(cls, ohlc, er, ema_fast, ema_slow, period, column)
    288 ) ## first KAMA is SMA
    289 kama = []
    290 # Current KAMA = Prior KAMA + smoothing_constant * (Price - Prior KAMA)
    291 for s, ma, price in zip(
--> 292     sc.iteritems(), sma.shift().iteritems(), ohlc[column].iteritems()
    293 ):
    294     try:
    295         kama.append(kama[-1] + s[1] * (price[1] - kama[-1]))

~\anaconda3\Lib\site-packages\pandas\core\generic.py in ?(self, name)
    6295     and name not in self._accessors
    6296     and self._info_axis._can_hold_identifiers_and_holds_name(name)
    6297 ):
    6298     return self[name]
-> 6299     return object.__getattr__(self, name)

AttributeError: 'Series' object has no attribute 'iteritems'

```

Kaufman Efficiency Indicator for Coca Cola / Coke Stock



BACK TESTING TRADING STRATEGY

```

In [70]: class DemaCross(Strategy):
          def init(self):
              self.ma1 = self.I(TA.DEMA, ohlc, 10)
              self.ma2 = self.I(TA.DEMA, ohlc, 20)

          def next(self):
              if crossover(self.ma1, self.ma2):
                  self.buy()
              elif crossover(self.ma2, self.ma1):
                  self.sell()

In [71]: ohlc.head()
          print(ohlc.Date)

```

```

0          1962-01-02
1          1962-01-03
2          1962-01-04
3          1962-01-05
4          1962-01-08
...
15306      2022-10-20 00:00:00-04:00
15307      2022-10-21 00:00:00-04:00
15308      2022-10-24 00:00:00-04:00
15309      2022-10-25 00:00:00-04:00
15310      2022-10-26 00:00:00-04:00
Name: Date, Length: 15311, dtype: object

```

```
In [72]: bt = Backtest(ohlc, DemaCross, cash = 100000, commission = 0.015, exclusive_orders = True)
```

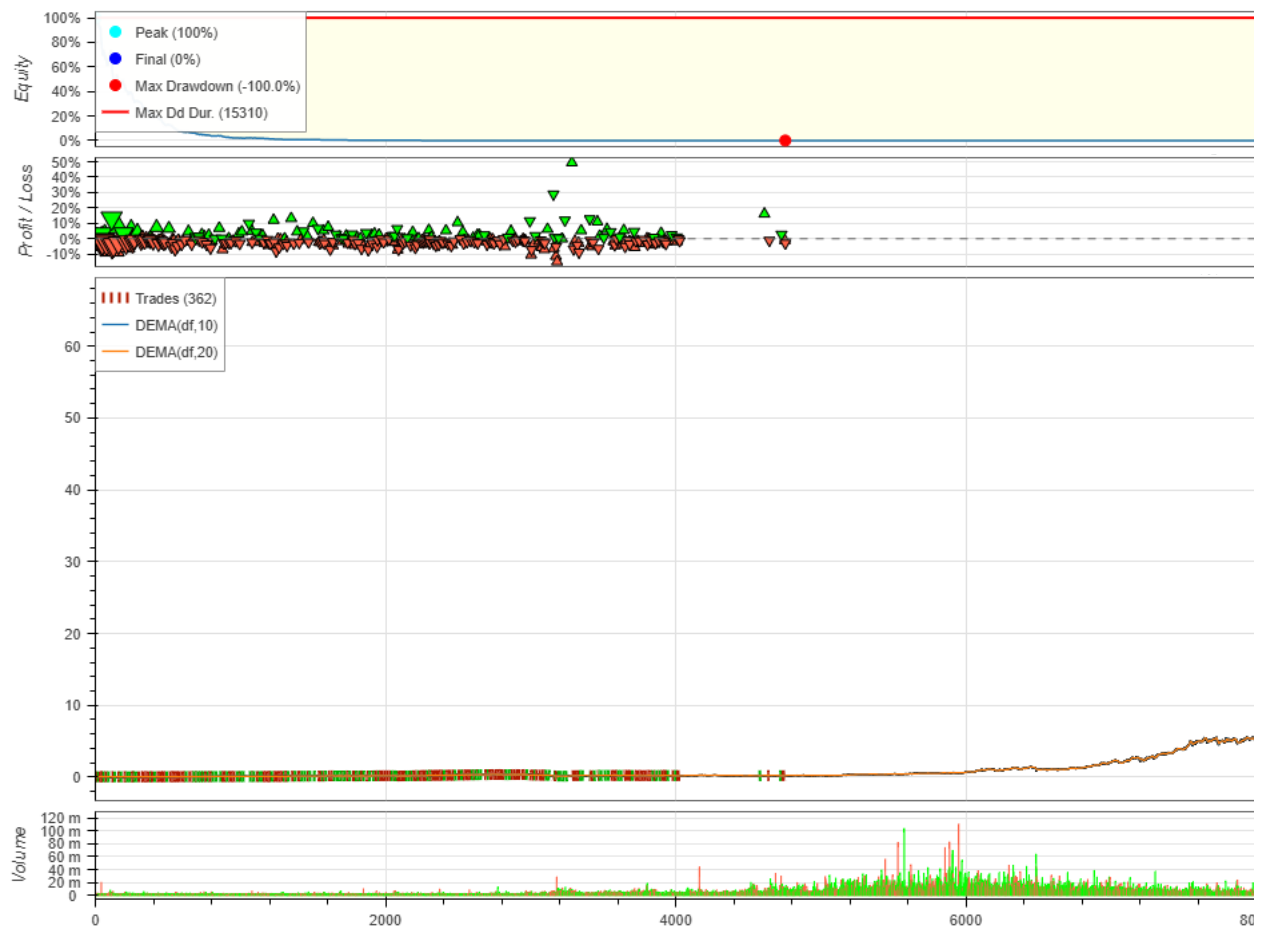
```
In [73]: bt.run()
```

```

Out[73]: Start                0.0
End                15310.0
Duration           15310.0
Exposure Time [%]  26.88263
Equity Final [$]   0.19041
Equity Peak [$]    100000.0
Commissions [$]    77449.51548
Return [%]         -99.99981
Buy & Hold Return [%] 118642.19364
Return (Ann.) [%]   0.0
Volatility (Ann.) [%] NaN
Sharpe Ratio       NaN
Sortino Ratio      NaN
Calmar Ratio       0.0
Max. Drawdown [%]  -99.99981
Avg. Drawdown [%]  -99.99981
Max. Drawdown Duration 15304.0
Avg. Drawdown Duration 15304.0
# Trades           362.0
Win Rate [%]       26.79558
Best Trade [%]     49.37035
Worst Trade [%]    -14.86996
Avg. Trade [%]     -0.78197
Max. Trade Duration 59.0
Avg. Trade Duration 11.35635
Profit Factor       0.64345
Expectancy [%]     -0.67403
SQN                -1.6788
Kelly Criterion     -0.25192
_strategy          DemaCross
_equity_curve      Equ...
_trades            Size  En...
dtype: object

```

```
In [74]: bt.plot()
```



Out[74]: **GridPlot**(id = 'p1301', ...)

In [75]: data = ohlc

Back Testing Trading Strategy Heatmaps

```
In [76]: from backtesting import Strategy
from backtesting.lib import crossover
from backtesting.test import SMA
```

```
In [77]: def BBANDS(data, n_lookback, n_std):
    """Bollinger Bands Indicator"""
    hlc3 = (data["High"] + data["Low"] + data["Close"]) / 3 # Typical price
    mean = hlc3.rolling(n_lookback).mean()
    std = hlc3.rolling(n_lookback).std()

    upper = mean + (n_std * std)
    lower = mean - (n_std * std)

    return upper, lower

# Calculate SMAs
close = data.Close.values
sma10 = SMA(data.Close, 10)
sma20 = SMA(data.Close, 20)
sma50 = SMA(data.Close, 50)
sma100 = SMA(data.Close, 100)
# Calculate Bollinger Bands
upper, lower = BBANDS(data, 20, 2)
```

```
In [78]: data['X_SMA10'] = (close - sma10) / close
data['X_SMA20'] = (close - sma20) / close
data['X_SMA50'] = (close - sma50) / close
data['X_SMA100'] = (close - sma100) / close

data['X_DELTA_SMA10'] = (sma10 - sma20) / close
data['X_DELTA_SMA20'] = (sma20 - sma50) / close
data['X_DELTA_SMA50'] = (sma50 - sma100) / close
```

```
In [79]: data.index = pd.to_datetime(data.index) # Convert index to datetime
```

```
In [80]: data['X_MOM'] = data['Close'].pct_change( periods=2)
data['X_BB_upper'] = (upper - data['Close']) / data['Close']
data['X_BB_lower'] = (lower - data['Close']) / data['Close']
data['X_BB_width'] = (upper - lower) / data['Close']

data['X_Sentiment'] = ~data.index.to_series().between('2017-09-27', '2017-12-14')
```

```
In [81]: class Sma4Cross(Strategy):
# Define default parameters
n1 = 50
n2 = 100
n_enter = 20
n_exit = 10

def init(self):
# Correct function syntax and fixed typo (SAM → SMA)
self.sma1 = self.I(SMA, self.data.Close, self.n1)
self.sma2 = self.I(SMA, self.data.Close, self.n2)
self.sma_enter = self.I(SMA, self.data.Close, self.n_enter)
self.sma_exit = self.I(SMA, self.data.Close, self.n_exit)

def next(self):
if not self.position: # No open position
if self.sma1 > self.sma2:
if crossover(self.data.Close, self.sma_enter):
self.buy()
else:
if crossover(self.sma_enter, self.data.Close):
self.sell()

else: # If there is an open position
if (self.position.is_long and crossover(self.sma_exit, self.data.Close)) or \
(self.position.is_short and crossover(self.data.Close, self.sma_exit)):
self.position.close()
```

```
In [82]: from backtesting import Backtest
from backtesting.test import GOOG

backtest = Backtest(ohlc, Sma4Cross, commission=0.002)

stats, heatmap = backtest.optimize(
n1=range(10, 110, 10),
n2=range(20, 210, 20),
n_enter=range(15, 35, 5),
n_exit=range(1, 25, 5),
constraint=lambda p: p.n_exit < p.n_enter < p.n1 < p.n2,
maximize='Equity Final [$]',
max_tries=200,
random_state=0,
return_heatmap=True)
```

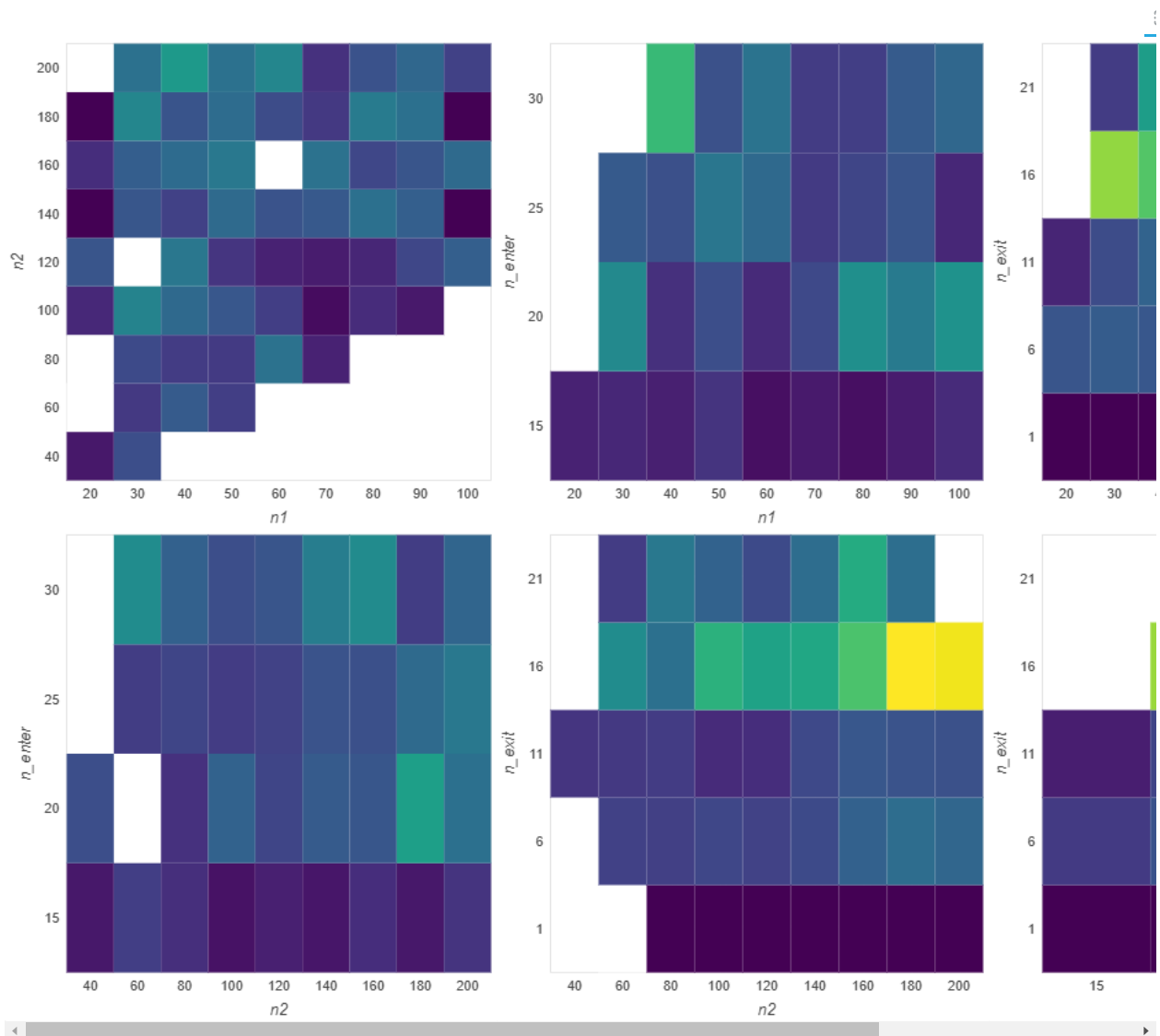
Backtest.optimize: 0%| | 0/184 [00:00<?, ?it/s]

```
In [83]: hm = heatmap.groupby(['n1', 'n2']).mean().unstack()
hm
```

```
Out[83]:
```

	n2	40	60	80	100	120	140	160	180	200
n1										
20	200.126887	NaN	NaN	355.028171	794.915220	0.000000	391.744291	0.000000	NaN	
30	728.450627	503.613367	678.713319	1354.955908	NaN	817.481822	909.376210	1390.996228	1137.443514	
40	NaN	876.022267	534.254173	1036.041979	1201.999181	588.527760	1060.119543	792.734939	1630.512221	
50	NaN	562.806531	518.871240	827.472457	468.905605	1045.180643	1213.577989	1069.720447	1127.904131	
60	NaN	NaN	1137.862532	559.717428	291.716186	774.426709	NaN	708.211503	1388.738763	
70	NaN	NaN	294.413807	104.229593	228.207686	846.436596	1141.466459	498.835292	427.758471	
80	NaN	NaN	NaN	388.572966	325.606870	1116.898855	645.086784	1265.971792	758.640605	
90	NaN	NaN	NaN	211.403791	643.259499	912.496645	795.328651	1121.830289	1007.408018	
100	NaN	NaN	NaN	NaN	917.793196	0.000000	1051.502905	0.000000	585.918056	

```
In [84]: from backtesting.lib import plot_heatmaps
plot_heatmaps(heatmap, agg='mean')
```



Out[84]: GridPlot(id = 'p1595', ...)

```
In [85]: %%capture
!pip install scikit_optimize
```

```
In [86]: stats_skopt, heatmap, optimize_result = backtest.optimize(
    n1 = [10, 100],
    n2 = [20, 200],
    n_enter = [10, 40],
    n_exit = [10, 30],
    constraint = lambda p: p.n_exit < p.n_enter < p.n1 < p.n2,
    maximize = 'Equity Final [$]', # Correct column name
    method = 'skopt',
    max_tries = 200, # Correct spelling: max_tries
    random_state = 0,
    return_heatmap = True,
    return_optimization = True
)
```

Backtest.optimize: 0%| | 0/200 [00:00<?, ?it/s]

```
In [87]: from skopt.plots import plot_objective
# Plot the optimization objective
_ = plot_objective(optimize_result, n_points=10)
```

```
-----  
AttributeError                                Traceback (most recent call last)  
Cell In[87], line 5  
      1 from skopt.plots import plot_objective  
      3 # Plot the optimization objective  
----> 5 _ = plot_objective(optimize_result, n_points=10)  
  
File ~\anaconda3\Lib\site-packages\skopt\plots.py:805, in plot_objective(result, levels, n_points, n_samples, size, wspace, hspace, zscale, dimensions, sample_source, minimum, n_minimum_search, plot_dims, show_points, cmap, ax)  
      802 if plot_dims is None:  
      803     # Get all dimensions.  
      804     plot_dims = []  
--> 805     for row in range(space.n_dims):  
      806         if space.dimensions[row].is_constant:  
      807             continue  
  
AttributeError: 'Space' object has no attribute 'n_dims'
```

In []: