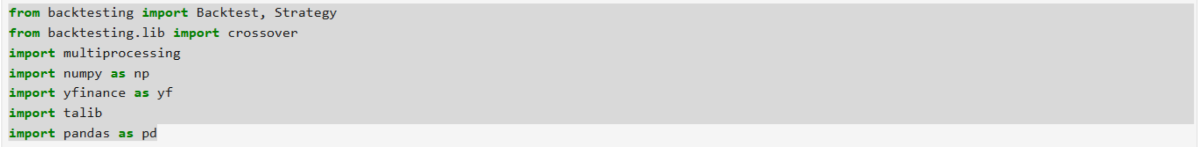
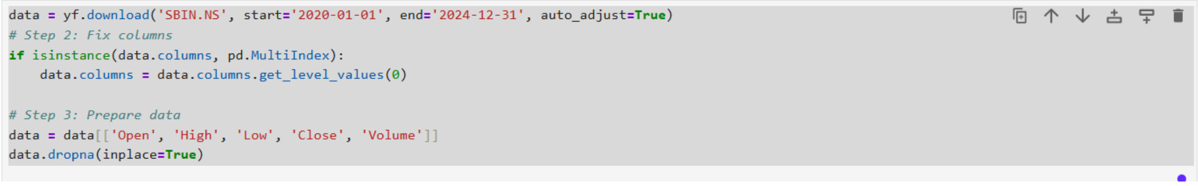
Documentation for the ps

So to start I first installed all the library include bactesting.py and talib and then downloaded data using yfinance.i have used the data of sbi and tata motors



```
from backtesting import Backtest, Strategy
from backtesting.lib import crossover
import multiprocessing
import numpy as np
import yfinance as yf
import talib
import pandas as pd
```



```
data = yf.download('SBIN.NS', start='2020-01-01', end='2024-12-31', auto_adjust=True)
# Step 2: Fix columns
if isinstance(data.columns, pd.MultiIndex):
    data.columns = data.columns.get_level_values(0)

# Step 3: Prepare data
data = data[['Open', 'High', 'Low', 'Close', 'Volume']]
data.dropna(inplace=True)
```

I used the isinstance else-if statement because Sometimes yfinance returns data with MultiIndex columns (e.g., ('Adj Close', '')), especially if group_by='ticker' is used. This line:

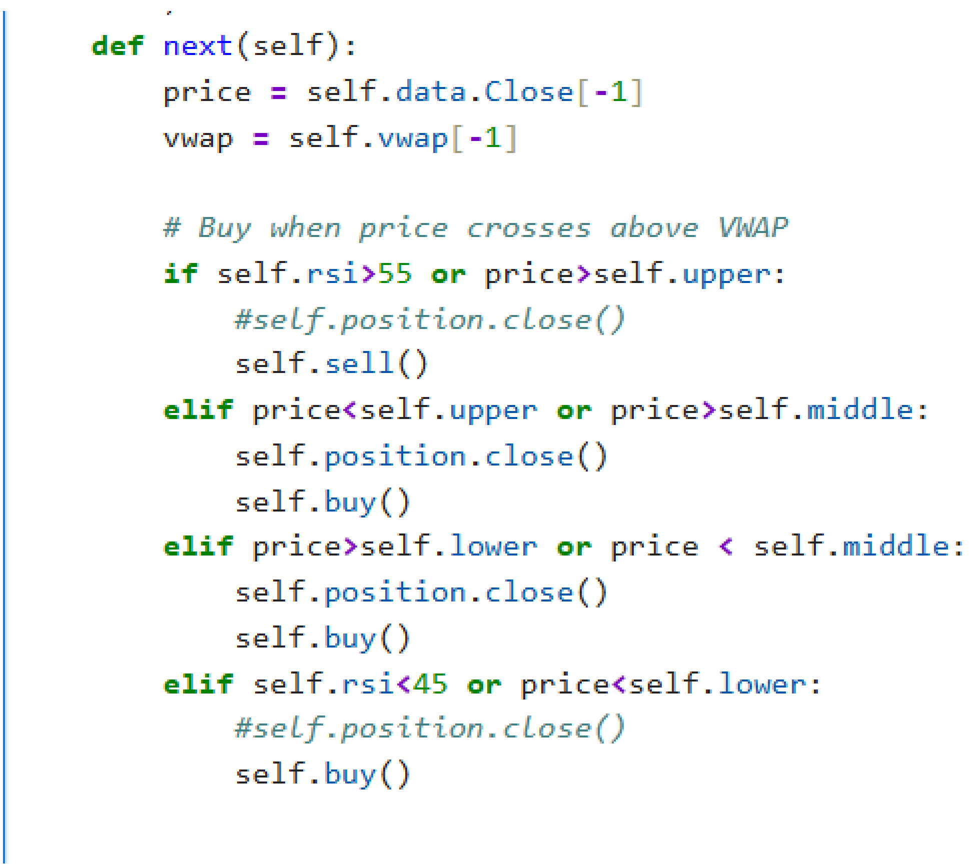
- Checks if columns are a MultiIndex.
- If true, it flattens them by using just the first level (e.g., 'Close' instead of ('Close', '')).

This ensures column names are clean and standard.

Also used dropna to remove the na values as it hampers the calculation of averages

now lets come to strategy in this I first tried to implement basic Bollinger bands with which I got negative sharpe this indicated its inefficiency

then with Bollinger bands I added rsi signals. As both of them indicate whether a stock is overbought or sold so this increase the efficiency of buying and selling signals



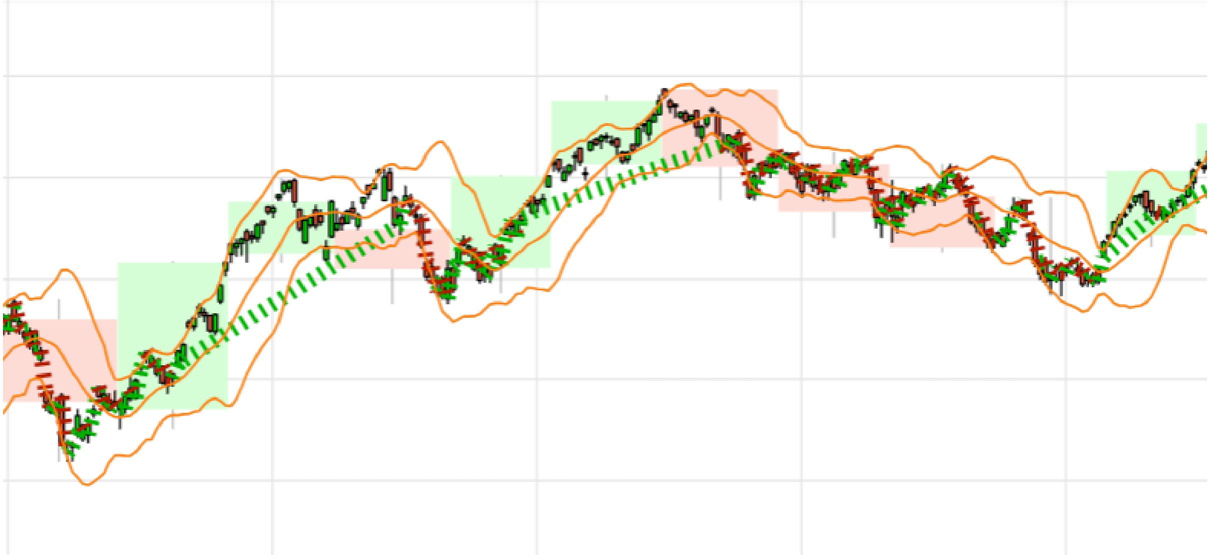
```
    def next(self):
        price = self.data.Close[-1]
        vwap = self.vwap[-1]

        # Buy when price crosses above VWAP
        if self.rsi>55 or price>self.upper:
            #self.position.close()
            self.sell()
        elif price<self.upper or price>self.middle:
            self.position.close()
            self.buy()
        elif price>self.lower or price < self.middle:
            self.position.close()
            self.buy()
        elif self.rsi<45 or price<self.lower:
            #self.position.close()
            self.buy()
```

```
Start                    2020-01-01 00:00:00
End                       2024-12-30 00:00:00
Duration                    1825 days 00:00:00
Exposure Time [%]                  95.95469
Equity Final [$]              24537.30078
Equity Peak [$]                27108.45117
Return [%]                     145.37301
Buy & Hold Return [%]              94.93714
Return (Ann.) [%]                  20.08237
Volatility (Ann.) [%]            21.03637
CAGR [%]                          13.19523
Sharpe Ratio                        0.95465
Sortino Ratio                       1.6317
Calmar Ratio                          0.929
Alpha [%]                         63.75504
Beta                              0.85971
Max. Drawdown [%]                  -21.61722
Avg. Drawdown [%]                   -2.1673
Max. Drawdown Duration     402 days 00:00:00
Avg. Drawdown Duration      22 days 00:00:00
# Trades                            533
```

```
Win Rate [%]                53.47092
Best Trade [%]               17.69107
Worst Trade [%]              -11.27971
Avg. Trade [%]                0.18036
Max. Trade Duration     102 days 00:00:00
Avg. Trade Duration       4 days 00:00:00
Profit Factor               1.43023
Expectancy [%]                  0.2
SQN                       2.20348
Kelly Criterion              0.1447
_strategy                    VWAP
_equity_curve                   ...
_trades               Size  Entry...
dtype: object
```

(this result is for nifty50)



Now I got this result with the help of above strategy it has a decent sharpe and sortino ratio with also a good number of trades in a period of 4 years this strategy basically checks if the price crosses the upper band and rsi>55 then the stock is overpriced hence a down trend would be followed so sell it and vice versa for price being lesser than lower bound in between I instruct to buy as this ensures a continues trading of stock without any risk of returns

I tried to improve(which I have added at the end in code) this but adding to check volatility using atr and adding macd but then my result failed also I was not able to fetch data from y finance at the end