**Task: Interday Swing Trade Visualization (Part 2)**

1. Import the following modules at the start of your algorithm

A white background with black and orange text

Description automatically generated

2. Create the following variables:



3. Create a list of all the tickers from task 21

A close-up of a computer code

Description automatically generated

4. Create the asynchronous function “fetch”

A close-up of a computer screen

Description automatically generated

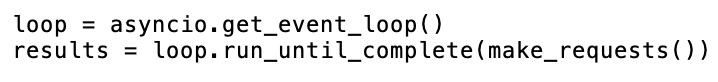
5. Create the asynchronous function “make\_requests” that makes requests to the Polygon.io API



A screenshot of a computer

Description automatically generated

6. Create the event loop and run it using the following code:



7. For each ticker in “tickers” create an object that has the following parameters (ticker, five day average volume, three month average volume, today’s close, and each of the past four trading sessions’ closes)

A white background with black text

Description automatically generated

8. Create a new empty list called “trendup”

9. Create a new empty list called “trenddown”

10. Create an instance method that checks if the stock has greater average volume over the past five days than over the past three months. If so, check to see if the stock has had an increase in closing price in each of the last five trading sessions or a decrease in closing price in each of the last five trading sessions. If so, add the stock to the list “trendup” or “trenddown” accordingly.

A computer code with black text

Description automatically generated

11. For each stock in both “trendup” and “trenddown” obtain the day by day data by making a request to the Polygon.io (<https://polygon.io/docs/stocks/get_v2_aggs_ticker__stocksticker__range__multiplier___timespan___from___to>) and you will get a JSON object as a return value.

A screenshot of a computer

Description automatically generated

Hint: Use an f-string with the two variables from step 2 for the “from” and “to” parameters

12. For each stock in both “trendup” and “trenddown” create a list of the stock’s day by day open price

13. For each stock in both “trendup” and “trenddown” create a list of the stock’s day by day close price

14. For each stock in both “trendup” and “trenddown” create a list of the stock’s day by day high price

15. For each stock in both “trendup” and “trenddown” create a list of the stock’s day by day low price

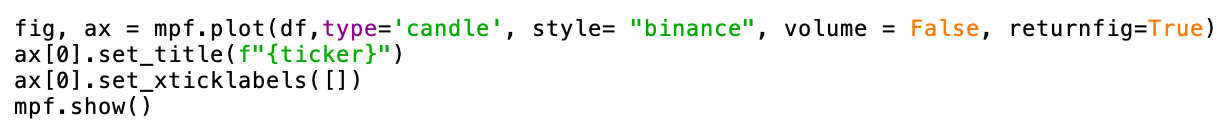
16. For each stock in both “trendup” and “trenddown” create a list of the stock’s day by day volume

17. For each stock in both “trendup” and “trenddown” create an empty dataframe using pandas

18. For each stock in both “trendup” and “trenddown” add 5 columns to the data frame (one for each of the following: open, close, high, low, and volume.

19. For each stock in both “trendup” and “trenddown” manipulate the first column in the dataframe so that it displays the time. It doesn’t matter which time you use since we will be omitting the x-axis labels.

20. For each stock in both “trendup” and “trenddown” display the dataframe using the following code:



21. Save your algorithm as a “.py” file in your documents folder:

A screenshot of a computer

Description automatically generated

22. Run your algorithm from the terminal (change directory to documents before running code)

A screenshot of a computer

Description automatically generated