**DATA 602: Assignment #1**

**Due**

Midnight AoE week 10, 3/4/2018

**Deliverables**



Link to private Github repo containing Python code for assignment, Dockerfile and README file containing link to Docker image in Docker Hub. Starting the app using Docker should not require any additional setup steps beyond installation of Docker, pulling and running of image.

**Objective**

For this assignment, you will build a console-based (shell or terminal based) equities trading system using scraping libraries (i.e. BeautifulSoup) or pandas (Easier, if you know pandas). You can use any data structure you are comfortable with (lists, DataFrames, etc.). The trading system will take orders to buy and sell a list of 5 predefined equities, maintain a blotter and P/L. The purpose of this assignment is to get you familiar with Python and Python data structures.

**Requirements**

A blotter is a list of trades already executed by the user (the trader). A P/L shows the position held in a particular equity and unrealized and realized profit / loss (UPL, RPL). Profit / loss that is realized is profit / loss that has already been incurred by closing a portion. An unrealized profit / loss is potential profit / loss that can occur if a position is theoretically closed now.

**The Main Menu**

When your program starts, the use should be presented with a main menu with at least the following four options to choose from:

*Trade*

*Show Blotter*

*Show P/L*

*Quit*

After executing a menu option the user should always be brought back to the main menu.

**Trade**

If the user wishes to make a trade they will select Trade from the main menu. The user will then be given the list of 5 equities they can trade and be allowed to pick one and state a quantity. The user is then asked to confirm the trade at the market ask price scraped from Yahoo.

If the trade is a buy then the dollar amount of the trade will be deducted from the cash position, if funds are available. Otherwise, the trade is not allowed.

If the trade is a sell then the quantity sold will be deducted from the current equity position held, if any.

Selling short is allowed.

Make sure to pull the market ask price just before executing the trade. Prices change fast and there can be a delay between when you display the market price to the user and when they choose to confirm the trade. The delay is the user’s problem.

**Show Blotter**

Displays the trade blotter, a list of historic trades made by the user. The trade blotter will display, at minimum, the following trade data, with the most recent trade at the top:

Side (Buy/Sell)

Ticker

Quantity

Executed Price

Execution Timestamp

Money In/Out

**Show P/L**

Displays the profit / loss statement. The P/L will display, at minimum, the following trade data, with the most recent trade at the top:

Ticker

Position

Current Market Price

VWAP

UPL (Unrealized P/L)

RPL (Realized P/L)

Portfolio Size

The initial size of the portfolio is $100MM of cash with no equity positions.

Quotes

To obtain market pricing info, scrape delayed quotes from Yahoo using a package like BeautifulSoup.

For the P/L, you will need to display a weighted-average price per share. Here is more information on WAP

<https://www.fool.com/knowledge-center/how-to-calculate-weighted-average-price-per-share.aspx>

**Calculating P/L**

Here is an example of how to calculate unrealized and realized P/L.

As a trader, you start your day off in the morning holding no inventory of equities. Your P/L will look something like this:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Ticker | Position | Market | WAP | UPL | RPL |
| AAPL | 0 | $149.04 | $0 | $0 | $0 |
| AMZN | 0 | $1001.81 | $0 | $0 | $0 |
| INTC | 0 | $34.68 | $0 | $0 | $0 |
| MSFT | 0 | $72.78 | $0 | $0 | $0 |
| SNAP | 0 | $15.27 | $0 | $0 | $0 |
| Cash | $10,000,000 | $10,000,000 |  |  |  |

Notice that your position is zero for all of the equities and your unrealized and realized P/L is also $0 - so far you’ve made no money nor have bought any equities.

You buy 1000 shares of AAPL at $150. A minute later you look at your P/L:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Ticker | Position | Market | WAP | UPL | RPL |
| AAPL | 1000 | $151 | $150 | $1000 | $0 |
| AMZN | 0 | $1001.81 | $0 | $0 | $0 |
| INTC | 0 | $34.68 | $0 | $0 | $0 |
| MSFT | 0 | $72.78 | $0 | $0 | $0 |
| SNAP | 0 | $15.27 | $0 | $0 | $0 |
| Cash | $9,850,000 | $9,850,000 |  |  |  |

Notice that your cash has dropped by $150,000, the cost of purchasing 1000 shares at $150 per share. Also notice the market price for AAPL is currently $151. Since you bought AAPL at $150, you’ve already made $1 per share if you were to sell your shares now. Since you haven’t sold your shares the $1 per share of potential profit is called “unrealized” and you’ll see the UPL column contains the $1000 you’ve potentially made.

If you were to sell the 1000 shares at $151, unrealized profits would go to $0 and realized profits would become $1000, your cash position would go to $10,001,000 and you would have 0 quantity of AAPL.

**Grading**

In addition to the rubric mentioned in the syllabus, I will look for the most cleanly and elegantly written code that make best use of Python data structures and/or pandas / NumPy. Executing trades at the right market price (in this case, the delayed quote from Yahoo) is critical and the correct calculation of WAP, UPL and RPL are important.

Please remember to push work to a private Github repository of your own and add me as a collaborator (Github id **jamiels**)