Website: KEA Securities
Kai Hirota, Eric Lin, Alyssa Lee

URL of the site: http://omis120stockmarket.herokuapp.com/

Purpose & Objective

The purpose of this website is to allow for people, specifically pre-university students, to be able to learn how to and experiment with trading stocks. By giving users a set sum of fake money in a user friendly website, we believe users of the site will be able to quickly get the hang of and develop strategies for trading various American stocks. To create a user friendly site, we streamlined functionalities to quoting, buying, and selling stock, as well as seeing a history of transactions, the current holdings and portfolio value, and leaderboard of users and their portfolio holdings. Unlike sites like Yahoo Finance, the purpose of this site is not to have too much additional (though important) information such as market cap and related news of the specific question.

The objective of this website is to give people interested in investing in stocks to have a user friendly and risk free environment to do so before moving to a platform where transactions are actually made. In other words, the site would be used like a sandbox.

Audience

The intended audience for this website are for people who are interested in investing and want to test out hypothetical investment strategies, but are not ready for the potential financial responsibility. This could appeal to a range people, from high school students, or professionals wanting a sandbox environment.

Problem addressed

We decided to add the cryptocurrency capability because the recent buzz around cryptocurrencies lured a lot of young people to invest without adequate understanding of financial markets or economics; furthermore, the investment decisions were driven mostly by emotions, rather than analysis or research.

Opportunity

There is no website on the Internet that allows users to virtually trade both stocks and cryptocurrencies for free. Ours is the only one.

Languages and tools used

- Language: Python, HTML/CSS, JavaScript, sqlite3
- Frameworks & libraries: Flask, Bootstrap, SQLAlchemy, werkzeug, urllib
- Host: Heroku

Challenges

- Challenge 1: Deployment and Integration
    - Initially, we designed the website with sqlite, which is a great database for local testing and storing data for small websites like the one we created. However, when deploying to Heroku server, we realized that sqlite was not compatible because Heroku uses Postgres SQL.
    - Approach 1: Tried to switch the database to Postgres SQL, but it worked too differently from MySQL or sqlite in terms of syntax, functionality, and integration.

Furthermore, we would have also had to rewrite every queries embedded in the Python code. Failed.
    - Approach 2: Created a pseudo-SQL data structure that works for us by modifying the solution that someone had shared on Stack Overflow. Success.
- Challenge 2: Adding the ability to trade cryptocurrencies post-deployment
    - After finishing the first version of the website, we decided to implement an additional feature that allows users to virtually trade cryptocurrencies. However, we kept getting an error that says “TypeError: 'NoneType' object is not subscriptable.”
    - Approach 1: We didn’t know the cause, so we thought it was a simple error that we had overlooked, so we checked all of the code in all of the files, and investigated the cause. No luck.
- Challenge 3: While trying to fix this, the “Quotes” page stopped working, and showed “404” error every time.
    - Approach 1: At this point, with 2 bugs simultaneously happening, we had to recover the first version of the website and start over.
    - Approach 2: We weren’t sure if the cause was in “quote.html” or “application.py” (back-end Python application) so again, we had to check everything and try to locate the error. After many trials, we realized that we accidentally deleted a section of the routing code. Success.
        - Error: @app.route("/quote") somehow changed to @app.route(“/”), which is the same as index.html.
- Back to Challenge 2:
    - Now that we got rid of the second bug, we copied and pasted the lookup( ) function to Jupyter Notebook. The reason is because this would allow us to run only the code that mattered for querying stock and cryptocurrency prices, run the lines individually and interactively like IPython, and see the output or the value. By taking this approach, we realized that the stock and cryptocurrency data that were returned came in slightly different formats. The price data that we wanted to parse was in the last column for stock price data, but it was the second column for cryptocurrency price data. It was a careless mistake because we just copy and pasted the code block for querying the stock data, and modified what we thought was necessary. By not checking and changing the index, we were getting the market cap data. Success.

Notes

Small problem: Sometimes, when accessing the website, it will display “TypeError: 'NoneType' object is not subscriptable” error. This is because AlphaVantage, which provides stock data for free to developers, has an API call limit of around 1-2 calls per second. This limit does not apply to individual API keys, but to all API keys, meaning that it will not return data if there is a surge in call volume.

Next features & improvements

- Improve user interface to be more aesthetically pleasing, as well as intuitive navigation.
- More advanced features: % changes, interactive visualizations, time-series data instead of single-point data, changes over time in terms of percentage and absolute values.