James McGee 12/4/17

CMPT 220 Professor Arias

Project 2 Writeup

Abstract:

This java application uses stock data from free sources and rearranges them in eclipse to be viewable on another platform. The application shows recent stock prices and how they were traded throughout the day over a month time period. The closing transactions can then be viewed to judge how the company did overall and if the trend seems continuous.

Introduction:

I am a business major with an emphasis in finance and a minor in information systems, so I am aiming to do a project that incorporates business practices. For the project I am interested in making a java application that can track the stock price changes of publicly traded companies. The user can then analyze the data and, one could assume on which stocks to invest in based on recent market changes and shareholders' reactions to them, essentially showing how stock price changed after a period. This can be useful as it can gauge what is happening in the market and show price fluctuations and then give suggested values at which to sell based off how the ratio of days a stock was rising versus falling. The main stock price data is available on financial websites such as yahoo finance, CBS finance, and Bloomberg to name a few and from there can be implemented directly into the program. These sites generally have all the historical data of a company since their creation, and by focusing on more recent time frames can see how many shares were sold throughout as small a time period as a day, and at the highest and lowest prices before closing. Using the java application, I hope to be able to implement this data in live time to

make more accurate predictions. This seems like an application with potential to help shareholders make decisions on stocks they are debating on purchasing.

System Description:

The ReadFiles class is a place where data is selected and checked to see if it exists. This class then analyzes the file to determine the number of rows of information there are and organize them into a table structure. The StockMarket class reads the ReadFiles class and converts the raw data from the financial websites into an array. The Formulas class loops through the ReadFiles class such that all the data inputted is identified. Formulas is then read by the StockMarket to create and print the array with the provided data.

Requirements:

The program is meant to address the difficulty of analyzing vast amounts of financial data. By having a place where data is constructed into a table and viewable all at once allows for fast comparisons to be made. This may aid an analyst looking for specific highs and lows of their stock interests and be able to see how they perform in under certain scenarios such as seasonal periods or during economic cycles such as recessions.

Survey:

The companies that track information such as Yahoo Finance and CBS Finance are similar in that they show how a company has been performing through their listed information. Bloomberg is the most successful and accurate of these types of sites as they can create a wider array of graphs and fields based around information. Even with the more advanced technical capabilities it is still up to the user to process the information and decide their move.

User Manual:

Admittedly my program is not user friendly and requires the user to have some knowledge of the stock market before looking at the data. The column descriptors are not allowed as this reads data not text, so the user would need to know the categories of date, open, high, low, close, adjusted close, and volume from memory. The user should download the data they want from one of these sites the most accessible being yahoo finance and add them to the pre-existing stock folder. Any words in the file will have to be removed however or the program will not run as it can not process anything other than numerical data.

Conclusion:

The program is able to transform raw data into an array that is then printed into a table based around the downloaded information. That table may be then viewed by an interest investor who can then make a decision based on any patterns they perceive.

References:

"Java BufferedReader Class - Javatpoint." Www.javatpoint.com, www.javatpoint.com/java-bufferedreader-class.

Liang, Y. Daniel. *Intro to Java Programming, Brief Version, Global Edition*. 10th ed., Pearson, 2014.

StringTokenizer (Java Platform SE 7), 9 Oct. 2017, docs.oracle.com/javase/7/docs/api/java/util/StringTokenizer.html.