Assignment #1
CSCI 201 Fall 2022
4.0% of Course Grade

Title
Stock Symbol Searching

Topics Covered

Java Classes	Basic Java Topics
File I/O	JSON Parsing
Sorting	

Introduction
Much of your computer programming experience is likely using C++. In this course, Java is the language we will primarily use. To help transition your knowledge from C++ to Java, you will be creating a program that looks up public companies or stock symbols based on certain search criteria. This assignment requires that you parse a file containing a set of company's information and their respective stock symbols (a.k.a., the ticker). To ensure accurate parsing, you will then provide a command line interface to allow a user to query the parsed data.

Data Persistence
Many software projects need to store persistent data, and there are a few ways to do this. Databases, which we will cover later in the class, are primarily used for most web applications, but storing data in files is also very common for many types of applications (i.e. mobile, standalone, web). This assignment will require you to parse a file that contains public companies and various information about them and their stock. The data in the file is going to be stored as a ***JavaScript Object Notation (JSON)*** file, formatted as follows:

```
{
  "data": [
          {
            "name": "Tesla Inc",
            "ticker": "TSLA",
            "startDate": "2010-06-29",
            "description": "Tesla Motors, Inc. (Tesla) designs, develops, manufactures and sells electric vehicles and advanced electric vehicle powertrain components.",
            "exchangeCode": "NASDAQ"
          }
  ]
}
```

There could be multiple items in the *data* array, but there will be at least one. In this example, there is only one object in the *data* array. **A sample file, stock.json, is provided**

on the course website. I recommend that you create your own test files that are longer, since the files we will use for testing will be different from the sample one provided.

Parsing JSON

JSON is a lightweight data-interchange format. In other words, it is a syntax that allows for easy storage and organization of data. It is commonly used to exchange information between client and server, and it is popular because of its language independence and human readability. JSON is built upon two basic data structures that you should already be familiar with: dictionaries (maps) and ordered lists. An object in JSON is represented by an unordered set of name/value pairs (i.e., dictionary). Objects are contained by braces, { }, inside of which will list the object's attributes (with the syntax name : value), using a comma as the separator.

There are quite a few Java JSON parsing APIs out there. Unfortunately, the JDK does not have built-in support for JSON, but some of the more popular ones include GSON, Jackson, and JSON.simple. Here are a couple of blogs discussing the merits of choosing one of these APIs over another if you are interested:

http://blog.takipi.com/the-ultimate-json-library-json-simple-vs-gson-vs-jackson-vs-json/
http://javarevisited.blogspot.com/2016/09/top-5-json-library-in-java-JEE.html

In short, it seems GSON is known for its ease and flexibility of converting Java objects into JSON objects (and vice versa), and it is simple and straightforward to use. You may want to start there. No matter which API you choose (and you are not limited to choosing from the ones mentioned in these instructions), you will need to download a JAR file and add it to your Eclipse project. Below are links to the JAR file download for the APIs mentioned above:

GSON:
https://github.com/google/gson

Jackson:
https://github.com/FasterXML/jackson-core

JSON.simple:
https://github.com/fangyidong/json-simple

You will need to add the JAR file to your Java Build Path in Eclipse to use the library. First, move the JAR file to the top directory of your Eclipse project. In other words, if your project is named Assignment 1, put the JAR file into the Assignment 1 directory in your Eclipse workspace directory. Next perform the following steps in Eclipse:

1. Right click on your Eclipse project.
2. Click "Refresh", which should make your JAR file show in the Package Explorer.
3. Right click on your Eclipse project again.
4. Choose "Properties".
5. Select "Java Build Path".
6. Click the "Libraries" tab.
7. Click "Add JARs".

8. Find the JAR in your project directory and add it.
9. Click “Okay”.

Assignment

When your program first runs, you will need to prompt the user to enter the name of the JSON file that contains the data on the different public companies. Your program should validate that the file is formatted properly. You can assume the following data types for each:

data: Array (of JSON Objects)
name: String
ticker: String
description: String
startDate: date formatted as YYYY-MM-DD
exchangeCode: String (valid strings are “NASDAQ” and “NYSE”)

Important Note: Keys in each JSON Objects can appear in any order.

If there is any problem parsing the data in the file, your program should print out as descriptive of an error message as possible. The program should then prompt the user to enter another file. Here are some examples of errors in file parsing you need to catch:

- File not found
- Data cannot be converted to the proper type as shown above
- Missing data parameters

Once a properly formatted file is parsed, your program should display a menu as follows:

1) Display all public companies
2) Search for a stock (by ticker)
3) Search for all stocks on an exchange
4) Add a new company/stocks
5) Remove a company
6) Sort companies
7) Exit

What would you like to do?

Users can enter menu commands by entering the number associated with the command, i.e., entering “1” to display all the public companies. ***All data entered should be case-insensitive, with the exception of the file names.***

Sample Execution

Here is a sample execution of the program with the user input bolded (though the input will not be bolded when you run your program). Assume stock.json contains the data from the sample file provided.

What is the name of the company file? **badfile.json**
The file badfile.json could not be found.

What is the name of the company file? **badformat.json**
The file badformat.json is not formatted properly.

What is the name of the company file? **stock.json**
The file has been properly read.

1) Display all public companies
2) Search for a stock (by ticker)
3) Search for all stocks on an exchange
4) Add a new company/stocks
5) Remove a company
6) Sort companies
7) Exit

What would you like to do? **0**

That is not a valid option.

What would you like to do? **hello**

That is not a valid option.

What would you like to do? **1**

Tesla Inc, symbol TSLA, started on 2010-06-29, listed on NASDAQ,
Tesla Motors, Inc. (Tesla) designs, develops, manufactures and sells electric vehicles and advanced electric vehicle powertrain components.
Apple Inc, symbol AAPL, started on 1980-12-12, listed on NYSE,
Apple Inc. (Apple) designs, manufactures and markets mobile communication and media devices, personal computers, and portable digital music players, and a variety of related software, services, peripherals, networking solutions, and third-party digital content and applications.
Microsoft Corporation, symbol MSFT, started on 1986-03-13, listed on NASDAQ,
Microsoft enables digital transformation for the era of an intelligent cloud and an intelligent edge. Its mission is to empower every person and every organization on the planet to achieve more.

1) Display all public companies
2) Search for a stock (by ticker)
3) Search for all stocks on an exchange
4) Add a new company/stocks

5) Remove a company
6) Sort companies
7) Exit

What would you like to do? **2**

What is the ticker of the company you would like to search for? **USC**

USC could not be found.

What is the ticker of the company you would like to search for? **TSLA**

Tesla, symbol TSLA, started on 2010-06-29, listed on NASDAQ

1) Display all public companies
2) Search for a stock (by ticker)
3) Search for all stocks on an exchange
4) Add a new company/stocks
5) Remove a company
6) Sort companies
7) Exit

What would you like to do? **3**

What Stock Exchange would you like to search for? **APPLE**

No exchange named APPLE found.

What Stock Exchange would you like to search for? **NASDAQ**

TSLA and MSFT found on the NASDAQ exchange.

1) Display all public companies
2) Search for a stock (by ticker)
3) Search for all stocks on an exchange
4) Add a new company/stocks
5) Remove a company
6) Sort companies
7) Exit

What would you like to do? **4**

What is the name of the company you would like to add? **Tesla Inc**

There is already an entry for Tesla Inc.

What is the name of the company you would like to add? **AT&T Inc**

What is the stock symbol of AT&T Inc? **T**

What is the start date of AT&T Inc? **1984-07-19**

What is the exchange where AT&T Inc is listed? **NYSE**

What is the description of AT&T Inc? **AT&T Inc. is a diversified, global leader in telecommunications, media and entertainment, and technology. AT&T Communications provides more than 100 million U.S. consumers with entertainment and communications experiences across TV, mobile and broadband.**

There is now a new entry for:
AT&T Inc, symbol T, started on 1984-07-19, listed on NYSE,
AT&T Inc. is a diversified, global leader in telecommunications, media and entertainment, and technology. AT&T Communications provides more than 100 million U.S. consumers with entertainment and communications experiences across TV, mobile and broadband.

1) Display all public companies
2) Search for a stock (by ticker)
3) Search for all stocks on an exchange
4) Add a new company/stocks
5) Remove a company
6) Sort companies
7) Exit

What would you like to do? **5**

1) Tesla Inc
2) Apple Inc
3) AT&T Inc.
4) Microsoft Corporation

Which company would you like to remove? **2**

Apple Inc is now removed.

1) Display all public companies
2) Search for a stock (by ticker)
3) Search for all stocks on an exchange
4) Add a new company/stocks
5) Remove a company
6) Sort companies
7) Exit

What would you like to do? **6**

1) A to Z
2) Z to A

How would you like to sort by? **1**

Your companies are now sorted from in alphabetical order (A-Z).

1) Display all public companies
2) Search for a stock (by ticker)
3) Search for all stocks on an exchange
4) Add a new company/stocks
5) Remove a company
6) Sort companies
7) Exit

What would you like to do? **7**

1) Yes
2) No

Would you like to save your edits? **1**

Your edits have been saved to stock.json
Thank you for using my program!

Grading Criteria

The manner by which you go about implementing the solution is not specifically graded, but the output must match exactly what you see in the execution above.

Note: Any program crash will result in -0.2

File I/O (0.8%)

0.2% - The filename is read from the user, and the file is appropriately parsed
0.1% - If the file could not be found, an appropriate error message is displayed
0.1% - If the file cannot be parsed, an appropriate error message is displayed
0.1% - If the file has missing parameters, an appropriate error message is displayed
0.3% - The file is properly saved upon exiting the program

Reading Inputs (1%)

0.3% - Invalid user input is handled properly
0.7% - Users can properly navigate the menu options by using numerical inputs

Outputs (1.2%)

0.2% - “Display all public companies” displays all public companies from the file
0.1% - “Search for a stock (by ticker)” displays the correct information for the search query
0.1% - “Search for all stocks on an exchange” displays the correct information for the search query
0.3% - “Add a new company/stocks” works properly
0.1% - “Remove a company” works properly
0.4% - Each of the two methods for sorting companies works properly