

Assignment 6: Stock Price Simulator

CIS 210 – Fall 2016

Point: 100

Project Due: 11:59 PM on Thursday, 09/29/2016

Learning Objectives:

- ❖ To understand the concept of Class and Object.
- ❖ To identify classes and their fields/methods from a problem statement.
- ❖ To understand overloading methods including class constructors.
- ❖ To generate random numbers with the Random class.
- ❖ To write your own testing cases to test your program.
- ❖ To use debugger for debugging your Java program.

Problem Statement

You are required to write a stock price simulator, which simulates a price change of a stock. When the program runs, it should do the following:

- Create a Stock class which must include fields: name, symbol, currentPrice, nextPrice, priceChange, and priceChangePercentage.
- The Stock class must have two constructor: a no-argument constructor and a constructor with four parameters that correspond to the four fields.
 - For the no-argument constructor, set the default value for each field such as:
 - Name: Microsoft
 - Symbol: MSFT
 - currentPrice: 46.87
 - nextPrice: 46.87
 - For the argument constructor, set the value of four fields based on the arguments.
- The Stock class must have accessors and mutators for all of its fields.
- The setter methods must protect a user from setting the currentPrice/nextPrice into negative number. You can set the values to 0 if a user tries to change them to negative values.
- The Stock class should have a SimulatePrice() method, which increases or decreases the currentPrice by 0 - 10% randomly.
- The main program will ask user to enter a name, symbol, current price of a stock. Then, it simulates the prices for next 30 days. It displays the prices for the next 30 days on the console. If a user enters "NONE", "NA", 0.0 for name, symbol, current price respectively, then the no-argument constructor is used.

Note: your program must be user-friendly and intuitive. This is a part of your grade. In other words, even if your program does everything the problem statement states, your grade may be reduced because of difficulty to it.

Input

This program requires that you read in the following data values:

- A stock's name, symbol, and current price.
 - E.g., Microsoft Corporation, MSFT, 45.87.

You will use *interactive I/O* in this program. All of the input must be validated if it is needed. You can assume that for a numeric input value, the grader will enter a numeric value in the testing cases.

Output

Your program should display the stock's name, symbol, current price, next price priceChange priceChangePercentage for each day.

Test case output: (green texts are user input)

Please enter the name of the stock: Microsoft
Please enter the symbol of the stock: MSFT
Please enter yesterday's price of Microsoft: 45.65

STOCK	SYMBOL	YESTERDAY_PRICE	TODAY_PRICE	PRICE_MOVEMENT	CHANGE_PERCENT
Microsoft	MSFT	45.65	47.48	1.83	4.00%
Microsoft	MSFT	47.48	52.22	4.75	10.00%
Microsoft	MSFT	52.22	49.61	-2.61	-5.00%
Microsoft	MSFT	49.61	47.13	-2.48	-5.00%
Microsoft	MSFT	47.13	51.84	4.71	10.00%
Microsoft	MSFT	51.84	57.03	5.18	10.00%
Microsoft	MSFT	57.03	54.18	-2.85	-5.00%
Microsoft	MSFT	54.18	59.05	4.88	9.00%
Microsoft	MSFT	59.05	62.01	2.95	5.00%
Microsoft	MSFT	62.01	57.05	-4.96	-8.00%
Microsoft	MSFT	57.05	54.19	-2.85	-5.00%
Microsoft	MSFT	54.19	49.32	-4.88	-9.00%
Microsoft	MSFT	49.32	46.85	-2.47	-5.00%
Microsoft	MSFT	46.85	50.60	3.75	8.00%

Microsoft	MSFT	50.60	55.15	4.55	9.00%
Microsoft	MSFT	55.15	58.46	3.31	6.00%
Microsoft	MSFT	58.46	61.38	2.92	5.00%
Microsoft	MSFT	61.38	55.86	-5.52	-9.00%
Microsoft	MSFT	55.86	56.42	0.56	1.00%
Microsoft	MSFT	56.42	56.98	0.56	1.00%

Good bye!

Miscellaneous

Be sure to read and understand the sections in the Course Syllabus handout on general project requirements. Also be sure to study the style, documentation and formatting guidelines discussed in the *Programming Style Guidelines* handout and in the lecture.

What File To Turn In and How to Turn In Your Work using Blackboard

You must turn in your Java program source files in a zip file. The zip file must include a `Stock.java` file which defines the `Stock` class, and a `StockPriceSimulator.java` file which includes the `main()` method. The zip file's name must follow this format: `hw6_StockPriceSimulator.zip`