Contestant Number: _____

Time: _____

Rank: _____

BUSINESS PROFESSIONALS of AMERICA
Giving Purpose to Potential

JAVA PROGRAMMING
(340)

REGIONAL – 2021

PRODUCTION PORTION:

Program 1: StockDriver _____ (355 points)

TOTAL POINTS _____ *(355 points)*

Test Time: 90 minutes

GENERAL GUIDELINES:

Failure to adhere to any of the following rules will result in disqualification:

1. Contestant must hand in this test booklet and all printouts if any. Failure to do so will result in disqualification.
2. No equipment, supplies, or materials other than those specified for this event are allowed in the testing area. No previous BPA tests and/or sample tests (handwritten, photocopied, or keyed) are allowed in the testing area.
3. Electronic devices will be monitored according to ACT standards.
4. All work must be saved on the flashdrive provided.

You will have ninety (90) minutes to complete your work.

Your name and/or school name should *not* appear on work you submit for scoring.

1. Create a folder on the flash drive provided using your contestant number as the name of the folder.
2. Copy your entire solution/project into this folder.
3. Submit your entire solution/project so that the graders may open your project to review the source code.
4. Ensure that the files required to run your program are present and will execute on the flash drive provided.

   *Note that the flash drive letter may *not* be the same when the program is scored as it was when you created the program.

   * It is recommended that you use relative paths rather than absolute paths to ensure that the program will run regardless of the flash drive letter.

   The scorers will *not* compile or alter your source code to correct for this.
   Submissions that do *not* contain source code will *not* be graded.

Assumptions to make when taking this assessment:

- Users will not enter negative numbers as input.
- Users will not enter in numbers as stock names and symbols.
- Users will not enter in letters for number of stocks.
- Users do not need to enter in real stock symbols or names.

Development Standards:

- Your Code must use a consistent variable naming convention.
- All methods (if any) must be documented with comments explaining the purpose of the method, the input parameters (if any), and the output (if any).
- If you create a class, then you must use Javadoc comments.

Stock Market Portfolio Creation Simulation

You are being put in charge of creating a stock trading software that will allow the user to create a simulated portfolio of stocks. The program will allow the user to determine how many stocks they want to create, enter in a symbol and a company name, as well as how many shares for the stock. The program will randomly generate a price for the stock of $1.00 to $200.99.

The program will calculate the value of the stock in the portfolio by finding the product of the price and the amount of shares (price * shares = value). The program will also print out the entire portfolio to the console with the following format:

> Symbol: [SYMBOL_VARIABLE] | Company Name: [NAME_VARIABLE] | Price: [PRICE_VARIABLE] | Total Shares: [SHARES_VARIABLE] | Total Value: [CALCULATED_VALUE]
>
> //*This is a sample of how it will look for a company called XYZ*
> Symbol: XYZ | Company Name: XYZ-Company | Price: $ 174.13 | Total Shares: 10 | Total Value: $ 1,741.30

Input:

The program will require three entries via the console: Stock Symbol (String), Stock Name (String), and number shares (integer). The price will be randomly generated, and the value will be calculated by the price and number of shared entered. The program will construct a stock object that stores all of the information for the stock, and it will be stored into a centralized list in the driver class.

Output:

After all stocks are entered by the user, the program will print out all of the stocks stored in the list. The "Price" and "Total Value" need to be formatted to decimal places and include the symbol "$" in the printed output. All printed decimal places need to account for zeros being in the hundredths place.

> Symbol: [SYMBOL_VARIABLE] | Company Name: [NAME_VARIABLE]| Price: [PRICE_VARIABLE]| Total Shares: [SHARES_VARIABLE]| Total Value: [CALCULATED_VALUE]

<u>Sample Run #1 with error in number entry for number of stocks to create</u>

> How many stocks do you want to create?
> A
> That is an improper value. The program has been stopped.

Sample Run #2 with error in number entry for shares

How many stocks do you want to create?
1

You entered: 1

You will now enter in the company symbol, name and their share quantity. The price will be randomly generated.

Please enter the three character symbol:
EFG

Please enter the company name:
EFG-Company

Please enter the total shares:
two

That is an improper value. The program has been stopped.

Sample Run #3 with only one stock created

How many stocks do you want to create?

1

You entered: 1

You will now enter in the company symbol, name and their share quantity. The price will be randomly generated.

Please enter the three character symbol:

XYZ

Please enter the company name:

XYZ-Company

Please enter the total shares:

10

Listed below is your current portfolio:

Symbol: XYZ | Company Name: XYZ-Company | Price: $ 174.13 | Total Shares: 10 | Total Value: $ 1,741.30

Sample Run #4 with three stocks created

How many stocks do you want to create?
3

You entered: 3

You will now enter in the company symbol, name and their share quantity. The price will be randomly generated.

Please enter the three character symbol:
EFG

Please enter the company name:
EFG-Company

Please enter the total shares:
10

Please enter the three character symbol:
HIJ

Please enter the company name:
HIJ-Incorporated

Please enter the total shares:
2

Please enter the three character symbol:
MNO

Please enter the company name:
MNO-Consolidated

Please enter the total shares:
100

Listed below is your current portfolio:
Symbol: EFG | Company Name: EFG-Company | Price: $ 72.87 | Total Shares: 10 | Total Value: $ 728.70
Symbol: HIJ | Company Name: HIJ-Incorporated | Price: $ 74.00 | Total Shares: 2 | Total Value: $ 148.00
Symbol: MNO | Company Name: MNO-Consolidated | Price: $ 108.94 | Total Shares: 100 | Total Value: $ 10,894.00

Requirements:

1. You must create an application with the main class called StockDriver. This class will host the list to store the stocks that are created and also control the data entry from the user. There are *no* third party files you will need for this program.
2. You will create an object class called Stocks. This class will be instantiated to create the stocks based upon the user entering data. It will also randomly generate the price and calculate the value of the stock based upon number of shares and price. Stocks class will print out the information for the stock, format for two decimal places.
3. Your contestant number must appear as a comment at the top of the main source code file.
4. If incorrect data is entered, then the program should display an appropriate message and exit.
5. The program will perform the required tasks correctly for however many stocks are indicated by the user.
6. The program will display the output like the example above.

Solution and Project for Contestant #: _____

The project is present on the flash drive.	_____	10 points
The projects main class is named **StockDriver.**	_____	10 points
The projects object class is named **Stocks.**	_____	10 points
Javadoc comments are used for new classes.	_____	10 points

Program Execution

The program runs from the USB flash drive.	_____	15 points

If the program does not execute, then the remaining items in this section receive a score of zero.

The program displays an error message if the user enters invalid data in the StockDriver class.	_____	20 points
The program forces the user to create 1 to 10 stocks in the StockDriver class.	_____	10 points
Program gets input from user in the following order: 3 letter symbol, name, and # of shares.	_____	10 points
The StockDriver class creates the correct number of user requested stock entries.	_____	10 points
StockDriver class can correctly print the entire portfolio entry list.	_____	20 points
Stock price is randomly generated.	_____	20 points
Program calculates the value of entry based upon price * shares and is formatted to two decimal places.	_____	20 points
The random generated price is in the correct range (1-200.99) in Stocks class.	_____	20 points
The random generated price is formatted to two decimal places in Stocks class.	_____	30 points
Output matches required format.	_____	30 points

Source Code Review

The source code is properly commented.		
A comment containing the contestant number is present.	_____	10 points
Methods and code sections are commented.	_____	20 points
Code uses try... catch for exception handling.	_____	10 points
Stocks are correctly constructed from data entry from StockDriver class. Must pass the symbol (String), name (String), and shares (int).	_____	10 points
toString() is create to print stocks in the Stocks class and formats the price and value to have a "$" symbol and two decimal places for all values including stocks that have no cents or a zero in the hundredths place (for example: $10.00 or $14.10).	_____	30 points
Random number generation is done in in Stocks class.	_____	20 points
Code uses a consistent variable naming convention.	_____	10 points

Total Points =____ / 355 points