CIS355A Week 4 Lab—Processing Arrays of Objects

**Objectives**

* Create a GUI that uses JList and JTabbedPanes.
* Process multiple objects in an ArrayList.
* Code event handlers for multiple events.

**Problem: Stocks4U Portfolio Management System**

Stocks4U needs to develop an app for you to manage your stock purchases. You should be able to store a list of stock purchases, view the individual stocks, add and remove stocks.

**Functional Requirements**

You can code the GUI by hand or use NetBeans GUI builder interface.

The GUI should have two tabs using JTabbedPane.

* One tab (Show stocks) should have the following.
  + A JList to display all the stock purchases
  + A text field or label to display information about a stock
  + A JButton to remove a stock
* One tab (Add stock) should have textboxes, labels, and a button to input a stock.

Create a Stock class to manage the stock activity. It should have private instance variables of

* company name;
* number of shares;
* purchase price; and
* current price.

Create a default and parameterized constructor.

Create sets and gets for all instance variables.

Create a get method to calculate and return the profit or loss. This would be calculated as follows.

Number of shares \* (current price – purchase price)

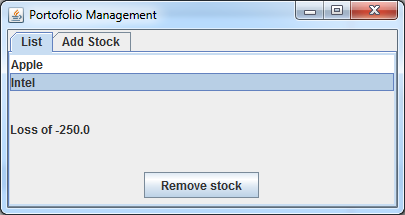
Create toString to display the name of the stock.

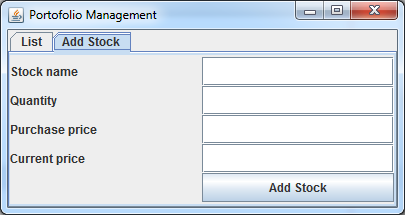
As you add stocks, they are displayed in the JList.

If you select an element in the JList, the gain or loss is displayed in the label or text field.

If you select an element in the JList and click Remove, the stock is removed from the list.

**Sample GUI**





**Grading Rubric**

|  |  |
| --- | --- |
| Stock class   * It has all required functionality. | 10 |
| GUI class   * Use the Stock class to process data. * As you add stocks, they are displayed in the JList. * If you select an element in the JList, the gain or loss is displayed in the label or text field. * If you select an element in the JList and click Remove, the stock is removed from the list. * Use error messages for any invalid or missing user input using JOptionPane. | 25 |
| Code style | 5 |
| Total | 40 |

**Code Style Requirements**

* Include meaningful comments throughout your code.
* Use meaningful names for variables.
* Code must be properly indented.
* Include a comment header at beginning of each file (example below).

**/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
Program Name: ProgramName.java   
Programmer's Name: Student Name   
Program Description: Describe here what this program will do   
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/**

**Deliverables**

Submit the following as a single zip folder.

* All Java files

Follow the assignment specification regarding class and method names.

Note that your Java file name must match class name (do not rename).