Homework #7 Solution

(Java Programming for Beginners - OnLine)

Note: *Complete your midterm and then try this homework as you can use part of the code from your midterm to complete this homework.*

Description: Convert your Midterm –A Simple Calculator Program-- into a class based program, using your knowledge of Object Oriented Programming techniques.

Write a OOPCalculator class, by giving it proper states and behaviors. For example, you can have menuChoice, firstFloat, secondFloat as member variables. You can give it behaviors like, askCalcChoice, askTwoValues, displayResults to get the choice from users, to get the numbers and display the calculation results respectively. All of the code written for calculation needs to be properly placed within these methods of the OOPCalculator class. For example: askCalcChoice first displays the menu, prompts the user to input the choice, and then waits until user presses enter. Once a valid input is entered, it returns the choice (between 1 to 5 or A for addition, S for subtraction, M for multiplication, D for division or X for exit). Similarly, other methods will implement rest of the behaviors. The users of your OOPCaluclator class do not need to know internal details of how you have implemented your behaviors. These are the only public interface exposed to the users, rest of the details stays with OOPCalculator class.

So, a driver program (written by user of your OOPCalculator or even from within main of your own class) can use your class by writing a code like this:

```
public class TestCalculator {
        public static void main(String[] args) {
           OOPCalculator calc = new OOPCalculator();
           while (calc. askCalcChoice () != 5){ //it will set choice
                    calc. askTwoValues ();              //it will set two values
                    calc.displayResults();       //do calc, display result
                                                  //and wait on press enter key
          }
          calc.displayBye();               //thanks the user for using and waits for press enter key
       } //end main
} //end class
```

All the error checking and exception handling for a wrong choice, or wrong value input are handled by your OOPCalculator class, the user should not need to handle it.

Sample Run of the program: When the driver program is run, the behavior is exactly same as Midterm I, except, provide one more functionality. Allow user to select 'A' or 'a' for addition,' S' or 's' for subtraction,' M' or 'm' for multiplication, 'D' or 'd' for division, and 'X' or 'x' for exit, along with integer choice between 1 to 5.

```
Welcome to <John Doe’s> Handy Calculator


        1.  Addition
        2.  Subtraction
        3.  Multiplication
        4.  Division
        5.  Exit
        What would you like to do?  M

        Please enter two numbers to multiply separate by a space: 24.0  4.0
        Result of multiplying 24.00 and 4.00 is 96.00.

        Press enter key to continue ….

Welcome to <John Doe’>s Handy Calculator

        1.  Addition
        2.  Subtraction
        3.  Multiplication
        4.  Division
        5.  Exit
        What would you like to do?
```

Continue until user selects 5, or ‘x’, or ‘X’ to exit

Note:

1) Replace the <*John Doe*’s> with your name
2) Make sure to properly word the output for the choices and result. Meaning, if user selects ‘M’ from menu, the prompt should be for ‘Multiplication’ not Addition.
3) *3, 24.0 4.0* are shown in the example as underline to emphasize that it is entered by the user and not part of the program. Underline and italics are not a requirement
4) Your program should allow input of integer or decimal numbers. The output should always be in decimals with two decimal digits as precision.
5) Make sure your program will continue displaying menu after result is shown and user has pressed enter key. Your program will exit only when user selects 5 or ‘x’ or ‘X’.
6) If user selects other than 1-5, show a message that they must select between 1 and 5 (or their character equivalents). Give them a chance to re-enter. Continue until a valid number is entered.
7) If they enter invalid values instead of numbers (e.g. strings), provide an error message and give them chances to re-enter. Continue until valid numbers are entered.

8) Make sure to catch divide by zero issue. In case of division choice, the second number should not be zero – so let the user know that the second number cannot be zero.

Solution:

```java
/**
 * The OPPCalculator class has two number fields and one string field. It also has methods to
 * fill in the two number, get operation, perform an operation on the numbers, and say goodbye
 * @author Todd Law
  *
 */

import java.util.Scanner;
import static java.lang.System.out;
import java.util.InputMismatchException;

//Change name of your .java file to OOPCalculator.java (otherwise this will not compile as
//class name and file name not matchine.  I named file so that it is easy for you to spot it is
//solutio of your homework#9

public class OOPCalculator {
        private Scanner readInput = new Scanner(System.in);
        private float num1 = 0.0f; //First Number
        private float num2 = 0.0f; //Second Number
        private String ops = ""; // Possible values will be "Adding", "Subtracting",
                                //"Multiplying", "Dividing", and "Exit"

        //constructor - does nothing
        public OOPCalculator(){
        }

        // start of method to ask for operation
        public int askCalcChoice(){
                char userInput;
                int choice_number = 0;
                boolean loop = true;

            do {  // Loop until we have valid operation input
                out.println("Welcome to Todd Law's Handy Calculator\n");
                out.println("1. (A)ddition\n2. (S)ubtraction\n3. (M)ultiplication\n4.
(D)ivision\n5. E(x)it\n");
                        out.print("What would you like to do? ");
                try {
                        userInput = readInput.next().charAt(0); // Reads in user input
                        switch (userInput){
                                case '1':
                                case 'A':
                                case 'a':
                                        ops = "Adding";
                                        choice_number = 1;
                                        loop = false;
                                        break;
                                case '2':
                                case 'S':
                                case 's':
                                        ops = "Subtracting";
                                        choice_number = 2;
                                        loop = false;
                                        break;
                                case '3':
                                case 'M':
                                case 'm':
                                        ops = "Multiplying";
                                        choice_number = 3;
                                        loop = false;
                                        break;
```

```
                              case '4':
                              case 'D':
                              case 'd':
                                      ops = "Dividing";
                                      choice_number = 4;
                                      loop = false;
                                      break;
                              case '5':
                              case 'X':
                              case 'x':
                                      ops = "Exit";
                                      choice_number = 5;
                                      loop = false;
                                      break;
                              default:
                                      loop = true;
                                      System.out.println("You have not entered a number between 1
and 5 or (A, S, M, D, X) . Try again.");
                                      break;
                       } // end of switch
                } catch (final InputMismatchException e) { // catch exception for invalid entry
                    System.out.println("You have entered an invalid input. Try again.");
                    readInput.next();    // discard invalid input
                    continue;            // keep looping until you found right one
                } // end of try

            } while (loop);
            return choice_number;
        } // end of method askCalcChoice()


        //Method to prompt for number and read into array
        public void askTwoValues(){
               float inputFloat1, inputFloat2; // local temporary float variables
            int successful_numbers_read;
           // number of successful reads. Used to determine number of reads needed

               do {  // Loop until we have correct input
                       successful_numbers_read = 0;
                       out.printf("Please enter two numbers, you are %s,", ops);
                              System.out.print(" separated by space: ");
                       try {
                           inputFloat1 = readInput.nextFloat(); // reads in first number
                           successful_numbers_read += 1;
                           inputFloat2 = readInput.nextFloat(); // reads in second number
                           if (inputFloat2 != 0){
                               break;    // Got two valid numbers, breaking out of loop
                           } else if (ops == "Dividing"){ //catch for dividing by zero
                               System.out.println("You can't divide by zero. Try again.");
                               continue; // continue looping due to divide by zero issue
                           }
                       } catch (final InputMismatchException e) {
                           System.out.println("You have entered an invalid input. Try again.");
                           readInput.next();    // discard non-float input
                           if (successful_numbers_read == 0){
                           //if statement for the case when the first input is invalid
                              readInput.next();  // discards second number
                           }
                           continue;            // keep looping until you you get valid inputs
                       }
                   } while (true); //end of do-while loop
                   num1 = inputFloat1;
                   num2 = inputFloat2;
        } // end of askTwoValues() method

        // Method to perform operation given operator and values
        public void displayResults(){
               float results = 0.0f;
```

```
                       System.out.printf("Results of %s ", ops);
                       // switch to print the proper operation and results
                       switch (ops){
                       case "Adding":
                              results = num1 + num2;
                              break;
                       case "Subtracting":
                              results = num1 - num2;
                              break;
                       case "Multiplying":
                              results = num1 * num2;
                              break;
                       case "Dividing":
                              results = num1/num2;
                              break;
                       default: // this should never happen since the number has already been
                               //verified but put in just as a precaution
                              break;
                       } // end of switch
                       out.printf(" %.2f and %.2f is %.2f\n", num1, num2, results);
                      //Print results with two digits after the decimal point
                       readInput.nextLine(); //reads in line before prompting to hit enter
                       out.println("Press Enter to continue ..");
                       while (!readInput.nextLine().equals("")); // loops until enter is hit

               } //end of method displayResults()

        //Method to display goodbye
        public void displayBye() {
               readInput.nextLine(); //reads in line before prompting to hit enter
               out.println("Thanks for using Todd Law's Calculator, press enter to end");
               while (!readInput.nextLine().equals("")); // loops until only enter is hit
        } //end of displayBye Method


        //this is the main to test the OOPCalculator.  This code can be used from any other file
        public static void main(String args[]){
               OOPCalculator calc = new OOPCalculator();
               while (calc. askCalcChoice () != 5){ //it will set choice
               calc. askTwoValues ();                  //it will set two values
               calc.displayResults();         //do calc, display result
                                              //and wait on press enter key
             }
             calc.displayBye();    //thanks the user for using and waits for press enter key
        }

} // end of class OOPCalculator
```