Grade 12 Assignment #4 Java Programming

A Basic Calculator

In this project, you will use classes, methods, and objects to create a simple arithmetic calculator. The calculator will be able to:

- Add two integers
- Subtract two integers
- Multiply two integers
- Divide two integers
- Apply the modulo operator on two integers

The instructions provided are general guidelines. Upon completion of the project, feel free to explore more in the learning environment.

Important: Moving forward, all projects will require that you define the `main` method manually. When you click the "Save" button below, an attempt to run your Java code is made. However, Java will return an error if you attempt to run a Java program without first defining a `main` method. If you encounter such an error, do not worry, you can define the `main` method ahead of time in order to avoid seeing the error.

Tasks:

There are 28 Tasks to complete for this assignment:

1. Open the BlueJ IDE editor to create the code needed for the program called A Basic Calculator.

2. First set up the BlueJ IDE. Create a project (file) with a public class called Calculator.

3. Inside of the public class called Calculator, create a Calculator constructor. You can leave the contents of the constructor empty.

4. Next, create a public method that returns an int and call it add.

5. The add method should accept two int parameters. For example: int a, int b.

6. The add method should add the two integer parameters that a user will specify. Inside of the add method, return the sum of a and b.

7. Next, create another similar method called subtract. The subtract method should accept two int parameters, just like the add method.

8. Inside of the subtract method, return the difference of a and b.

9. Create another method called multiply. The multiply method should accept two int parameters.

10. Inside of the multiply method, return the product of a and b.

11. Create another method called divide. It should accept two int parameters.

12. The divide method is special because we have to make sure that the calculator does not allow division by 0. Create an if statement that checks if the value of b is equal to 0.

13. Inside of the if statement, print out a helpful error message to the user. For example, you can print out: Error! Dividing by zero is not allowed..

14. The divide method must still return an int. On the next line, return 0.

15. Complete the if statement by adding an else block. Inside of the else block, return a divided by b.

16. Create another method called modulo. It should accept two int parameters.

17. The modulo method is another special method because we have to make sure that the calculator does not allow division by 0. Create an if statement that checks if the value of b is equal to 0.

18. Inside of the if statement, print out a helpful error message to the user. For example, you can print out: Error! Dividing by zero is not allowed..

19. The method must still return an int. On the next line, return 0.

20. Complete the if statement by adding an else block. Inside of the else block, return a modulo b.

21. Next, create a main method. This is the main body of programming code that is executed when run. Note: the main method must be defined *exactly the same way* every time it is created. Refer back to the lesson if you need to review the main method.

    In this assignment, the main part of the code is extremely simple. It will focus on creating an object that belongs to the class through the class constructor. Once an object is created, that object can call the methods (specialized functions) that belong to the class. The main part of the code will simply call some of those methods (add, subtract, multiple, divide, modulus) and print out the results to the screen.

22. Inside of main, create a Calculator object called myCalculator.

23. Print out the value of calling the add method on myCalculator. Pass in 5 and 7 as parameters.

    Hint: System.out.println(myCalculator.add(5, 7));

24. On the next line, print out the value of calling the subtract method on myCalculator. Pass in 45 and 11 as parameters.

25. If you completed this project correctly, the output should be 12 and 34. Feel free to explore more with the program. What are some ways in which the program could be improved?

    For example, make sure to include enough printed out statements to the screen (System.out.println()) so that the user knows what is going on and that they are clear regarding the information they are receiving.

26. It would be helpful to describe to other developers what this small Java program does. Write some comments that describes what this program does.

a. Use multi-line comments to (/* comment in the middle of */):
    i. Write one at the top of the code (before the public class Continents designation) that gives a quick intro/description the assignment.
    ii. Write one at the top of the code (before the public static void main(String[] args)) that summarizes the program.

b. Use a single line comment to (//then comment):
    i. Create 3 comments anywhere you deem necessary or important in the code. Remember the comment is to highlight or explain what is going on or what is being done in a particular way or used and why.
    ii. Identify each of the Methods created by indicating its purpose.
    iii. Identify the last line of code (anything that ends with a curly bracket }) in every function by writing “End of BLAH-BLAH function”.
    iv. Identify the end of the program.

27. Once your program is complete, make sure to test it using the BlueJ IDE (do not submit a program that does not work).

28. Lastly, upload (drag and drop) your assignment to the portal. Look for your name under the Assignment 4 webpage.