CS1110 - Lab 5 - In-Lab Exercises

Task#1- Void Method Exercise

In this coding exercise we will use Java methods to have the computer output count from 1 to 5, and we will make it do this 5 times.

We're going to create our new method first, and it will take care of counting from 1 to 5. Call this new method **counter( ),** and make its return type void.

Your code should look like this at this point:

```
public class FiveCounter {

   public static void counter( ){

      …

   }

   public static void main(String[] args) {

      // TODO code application logic here

           for (int i=0; i< 5 ; i++)



   }

}
```

To call **counter( )** method, inside the for loop you just made add this line of code so we can use our new method:

Counter ( );

Task #2- `boolean` Method

A. Write a `boolean` method called `isOdd()` in a class called `OddTest`, which takes an `int` as input and returns `true` if it is odd. The signature of the method is as follows:

```
public static boolean isOdd(int number)
```

Also write the `main()` method that prompts user for a `number`, and prints "ODD" or "EVEN". You should test for negative input.

B. Write a `boolean` method called `hasEight()`, which takes an `int` as input and returns `true` if the number contains the digit `8` (e.g., `18`, `808`). The signature of the method is as follows:

```
public static boolean hasEight(int number)
```

Write a program called **`MagicSum`**, which prompts user for numbers, and produce the sum of numbers containing the digit `8`. Your program should use the above methods. A sample output of the program is as follows:

```
Enter a positive integer or -1 to exit: 1
Enter a positive integer or -1 to exit: 2
Enter a positive integer or -1 to exit: 3
Enter a positive integer or -1 to exit: 8
Enter a positive integer or -1 to exit: 88
Enter a positive integer or -1 to exit: -1
The magic sum is: 96
```

```
int number;

// Read first input
System.out.print("Enter a positive integer or -1 to exit: ");
number = in.nextInt();

while (number != -1) {  // Read until input is -1
   ......

   // Read next input (Take note that you need to repeat these codes!)
   System.out.print("Enter a positive integer or -1 to exit: ");
   number = in.nextInt();
}
```

Task #3

Below a class called Geometry

public class Geometry {

public static void **printMenu ( ){**

…

}

public static double **circleArea(double r){**

…

}

public static double **rectangleArea(double l, double w){**

…

}

public static double **triangleArea (double h, double b){**

…

}

```
   public static void main(String[] args) {
      // TODO code application logic here
   }
}
```

1. Above the main method, but in the **Geometry** class, create a static method called **printMenu** that has no parameter list and does not return a value. It will simply print out instructions for the user with a menu of options for the user to choose from. The menu should appear to the user as:

```
This is a geometry calculator
Choose what you would like to calculate
1. Find the area of a circle
2. Find the area of a rectangle
3. Find the area of a triangle
Enter the number of your choice:
```

2. Add a line in the main method that calls the **printMenu** method as indicated by the comments.
3. Compile, debug, and run. You should be able to choose any option, but you will always get 0 for the answer. We will fix this in the next task.

Value-Returning Methods

1. Write a static method called **circleArea** that takes in the radius of the circle and returns the area using the formula $A = \pi r^2$.
2. Write a static method called **rectangleArea** that takes in the length and width of the rectangle and returns the area using the formula $A = lw$.
3. Write a static method called **triangleArea** that takes in the base and height of the triangle and returns the area using the formula $A = \frac{1}{2}bh$.

Calling Methods

1. Add lines in the main method in the Geometry class which will call these methods. The comments indicate where to place the method calls.
2. **Below, write some sample data and hand calculated results for you to test all 3 menu items.**

```
public class Geometry {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        // TODO code application logic here
        int choice; //the user's choice
        double value = 0; //the value returned from the method
        char letter; //the Y or N from the user's decision to exit
        double radius; //the radius of the circle
        double length; //the length of the rectangle
        double width; //the width of the rectangle
```

```
        double height; //the height of the triangle
        double base; //the base of the triangle

        //create a scanner object to read from the keyboard
        Scanner keyboard = new Scanner (System.in);
        //do loop was chose to allow the menu to be displayed first
        do {
            //call the printMenu method
        choice = keyboard.nextInt();
        switch (choice){
        case 1:
            System.out.print("Enter the radius of the circle: ");
            radius = keyboard.nextDouble();
            //call the circleArea method and store the result in the value
            System.out.println("The area of the circle is " + value);
        break;

        case 2:
            System.out.print("Enter the length of the rectangle: ");
            length = keyboard.nextDouble();
            System.out.print("Enter the width of the rectangle: ");
            width = keyboard.nextDouble();
            //call the rectangleArea method and store the result in the value
            System.out.println("The area of the rectangle is " + value);
        break;
        case 3:
            System.out.print("Enter the height of the triangle: ");
            height = keyboard.nextDouble();
            System.out.print("Enter the base of the triangle: ");
            base = keyboard.nextDouble();
            //call the triangleArea method and store the result in the value
            System.out.println("The area of the triangle is " + value);
        break;
        default:
            System.out.println("You did not enter a valid choice.");
        }
        //consumes the new line character after the number
        keyboard.nextLine();

        System.out.println("Do you want to exit the program (Y/N)?: ");
        String answer = keyboard.nextLine();
        letter = answer.charAt(0);

        }while (letter != 'Y' && letter != 'y');
    }

}
```