Chapter 5 Lab
Methods

Lab Objectives

- Be able to write methods
- Be able to call methods
- Be able to write `javadoc` comments
- Be able to create HTML documentation using the `javadoc` utility

Introduction

Methods are commonly used to break a problem down into small manageable pieces. A large task can be broken down into smaller tasks (methods) that contain the details of how to complete that small task. The larger problem is then solved by implementing the smaller tasks (calling the methods) in the correct order.

This also allows for efficiencies, since the method can be called as many times as needed without rewriting the code each time.

Finally, we will use documentation comments for each method, and generate HTML documents similar to the Java APIs that we have seen.

Task #1 `void` Methods

1. Copy the file *Geometry.java* (Code Listing 5.1) from the Student CD or as directed by your instructor. This program will compile, but, when you run it, it doesn't appear to do anything except wait. That is because it is waiting for user input, but the user doesn't have the menu to choose from yet. We will need to create this.
2. Below 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
4.  Find the circumference of a circle
5.  Find the perimeter of a rectangle
6.  Find the perimeter of a triangle
Enter the number of your choice:
```

3. Add a line in the `main` method that calls the `printMenu` method as indicated by the comments.
4. 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.

Task #2 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 = πr^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 = ½*bh*.
4. Write a `static` method called **`circleCircumference`** that takes in the radius of the circle and returns the circumference using the formula C = *2πr*.
5. Write a `static` method called **`rectanglePerimeter`** that takes in the length and the width of the rectangle and returns the perimeter of the rectangle using the formula P = 2*l* +2*w*.
6. Write a `static` method called **`trianglePerimeter`** that takes in the lengths of the three sides of the triangle and returns the perimeter of the triangle which is calculated by adding up the three sides.

Task #3 Calling Methods

1. Add lines in the `main` method in the `GeometryDemo` 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 6 menu items.**

3. Compile, debug, and run. Test out the program using your sample data.

Task #4 Java Documentation

1. Write `javadoc` comments for each of the 7 `static` methods you just wrote. They should include:
    a. A one line summary of what the method does.
    b. A description of what the program requires to operate and what the result of that operation is.

   c. `@param` listing and describing each of the parameters in the parameter list (if any).
   d. `@return` describing the information that is returned to the calling statement (if any).
2. Generate the documentation. Check the method summary and the method details to ensure your comments were put into the Java Documentation correctly.

Code Listing 5.1 (`Geometry.java`)

```
import java.util.Scanner;

/**
   This program demonstrates static methods
*/

public class Geometry
{
   public static void main(String[] args)
   {
      int choice;        // The user's choice
      double value = 0; // The method's return value
      char letter;       // The user's Y or N decision
      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
      double side1;      // The first side of the triangle
      double side2;      // The second side of the triangle
      double side3;      // The third side of the triangle

      // Create a scanner object to read from the keyboard
      Scanner keyboard = new Scanner(System.in);

      // The do loop allows the menu to be displayed first
      do
      {
         // TASK #1 Call the printMenu method

         choice = keyboard.nextInt();
```

```
switch (choice)
{
   case 1:
      System.out.print("Enter the radius of " +
                       "the circle: ");
      radius = keyboard.nextDouble();

      // TASK #3 Call the circleArea method and
      // store the result in the value variable

      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();

      // TASK #3 Call the rectangleArea method and
      // store the result in the value variable

      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();

      // TASK #3 Call the triangleArea method and
      // store the result in the value variable

      System.out.println("The area of the " +
                         "triangle is " + value);
      break;
   case 4:
      System.out.print("Enter the radius of " +
                       "the circle: ");
      radius = keyboard.nextDouble();
      // TASK #3 Call the circumference method and
      // store the result in the value variable

      System.out.println("The circumference " +
                         "of the circle is " +
                         value);
      break;
   case 5:
      System.out.print("Enter the length of " +
                       "the rectangle: ");
      length = keyboard.nextDouble();
      System.out.print("Enter the width of " +
                       "the rectangle: ");
      width = keyboard.nextDouble();

      // TASK #3 Call the perimeter method and
      // store the result in the value variable

      System.out.println("The perimeter of " +
                         "the rectangle is " +
                         value);
      break;
   case 6:
      System.out.print("Enter the length of " +
                       "side 1 of the " +
                       "triangle: ");
      side1 = keyboard.nextDouble();
      System.out.print("Enter the length of " +
                       "side 2 of the " +
                       "triangle: ");
      side2 = keyboard.nextDouble();
      System.out.print("Enter the length of " +
                       "side 3 of the " +
                       "triangle: ");
      side3 = keyboard.nextDouble();

      // TASK #3 Call the perimeter method and
      // store the result in the value variable

      System.out.println("The perimeter of " +
                         "the triangle is " +
                         value);
      break;
   default:
      System.out.println("You did not enter " +
                         "a valid choice.");
}
```

```
         keyboard.nextLine(); // Consume the new line

         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');
   }

   // TASK #1 Create the printMenu method here
   // TASK #2 Create the value-returning methods here
   // TASK #4 Write javadoc comments for each method
}
```