Close Lab 5: Create and use a class

Problem Description:

Design a class named <u>Triangle</u> that extends <u>GeometricObject</u>. The class contains:

- Three <u>double</u> data fields named <u>side1</u>, <u>side2</u>, and <u>side3</u> with default values <u>1.0</u> to denote three sides of the triangle.
- A no-arg constructor that creates a default triangle.
- A constructor that creates a triangle with the specified <u>side1</u>, <u>side2</u>, and <u>side3</u>.
- The accessor methods for all three data fields.
- A method named <u>getArea()</u> that returns the area of this triangle.
- A method named <u>getPerimeter()</u> that returns the perimeter of this triangle.
- A method named <u>toString()</u> that returns a string description for the triangle.

For the formula to compute the area of a triangle, see Exercise 5.19. The <u>toString()</u> method is implemented as follows:

```
return "Triangle: side1 = " + side1 + " side2 = " + side2 +
   " side3 = " + side3;
```

Draw the UML diagram that involves the classes <u>Triangle</u> and <u>GeometricObject</u>. Implement the class. Write a test program that creates a <u>Triangle</u> object with sides <u>1</u>, 1.5, <u>1</u>, color <u>yellow</u> and <u>filled</u> <u>true</u>, and displays the area, perimeter, color, and whether filled or not.

Design:
Draw the UML class diagram here

Coding:

```java
public class Exercise11_01 {
  public static void main(String[] args) {
    Triangle triangle = new Triangle(1, 1.5, 1);
    triangle.setColor("yellow");
    triangle.setFilled(true);

    System.out.println(triangle);
    System.out.println("The area is " + triangle.getArea());
    System.out.println("The perimeter is "
      + triangle.getPerimeter());
    System.out.println(triangle);
  }
}

class GeometricObject {
  // Copy it from the book
}
```

1

```
class Triangle extends GeometricObject {
  // Implement it
}
```

Submission:

Follow our class coding standard to complete this lab, check out for credit.