**Introduction**

In this assignment, we implemented a program to work with geometric objects, specifically triangles, using Java. The primary goal was to create a system that represents triangles, handles exceptions for invalid triangles, and allows user input to create and display triangle objects. This assignment will help us cover the basic concepts of java programming like class creation, polymorphism and exception handling.

**GeometricObject Class**

In the abstract class Geometric Object, this will be used a foundation to create other shapes, which includes the following fields- color and filled (whether it’s filled or not), then we created getters and setters for the color and filled properties. This class also includes the two abstract methods, getArea() and getPerimeter().

**Triangle Class**

Later in this class we created the Triangle class that extends GeometricObject. The class encompasses the three sides of the triangle fields, methods to get the area and perimeter. We used the formula given in the document. A toString() method was method that provides us with the description of the triangle sides, color, and fill.

**Triangle2 Class**

Triangle2 class extends ‘GeometricObject’ and includes extra validation for creating a triangle, whether the triangle has legal sides or illegal sides. If it doesn’t satisfy the conditions a TriangleException is thrown. (Custom created).

**TestClass1 and TestClass2**

Two test classes were created to satisfy the following conditions, allow user input to create a triangle and the second class to ensure that the input is valid. An exception ‘TriangleException’ was created and used to catch and handle various conditions that didn’t meet the conditions.

**Code Execution Instructions**

To run the code:

* Ensure you have a Java development environment set up.
* Compile all Java files in the GroupAssigment package.
* Run TestClass1 and TestClass2 to test the functionality.

**Conclusion**

This assignment helped us team members to polish our understanding of the classes and objects, exception handling and validation of input. We made sure that the code follows the coding and naming conventions practiced by the industry.

**References:**

Heron's Formula: Wikipedia - Heron's Formula

Java Documentation: Oracle Java Documentation