DB 3 - (Ch. 11 Exception)

Total: 35 pts

1. Design a Triangle class: **(10 pts)**

   Design a class named Triangle that extends GeometricObject. The class contains:
   a. Three double data fields named side1, side2, and side3 with default values 1.0 to denote three sides of a triangle.
   b. A no-arg constructor that creates a default triangle.
   c. A constructor that creates a triangle with the specified side1, side2, and side3.
   d. The accessor methods (getter methods) for all three data fields.
   e. A method named **getArea()** that returns the area of this triangle. (see below for the formula)
   f. A method named **getPerimeter()** that returns the perimeter of this triangle.
   g. A method named **getName()** that returns a string description for the triangle.
   h. The getName() method is implemented as follows:
   return **"Triangle: side1 = " + side1 + " side2 = " + side2 + " side3 = " + side3;**

   The formula for computing the area of a triangle is:

   s = (side1 + side2 + side3) / 2;

   area = Math.sqrt(s * (s - side1) * (s - side2) * (s - side3));

2. Create a user-defined exception class and modify Triangle class**: IllegalTriangleException** and **TriangleWithException** classes **(20 pts)**
   Defined the **TriangleWithException** class from the **Triangle** class (you just create from the above) with three sides. In this class, the sum of any sides is greater than the other side. The **TriangleWithException** class must adhere to this rule.
   Create the **IllegalTriangleException** class, and modify the constructor of the TriangleWithException class to throw an IllegalTriangleException object if a TriangleWithException is created with sides that violate the rule, as follows:

   public TriangleWithException(double side1, double side2, double side3

   throws IllegalTriangleException { //implement it }

3. Write a Driver class T**riangleExceptionTest** in the separate file to test your T**riangleWithException** class.
   You will create two TriangleWithException objects: **(5 pts)**
   TriangleWithException t1 = new TriangleWithException(1.5, 2, 3);
   TriangleWithException t2 = new TriangleWithException(1, 2, 3);

   The result is similar to the following figure:

```
run:
Perimeter for t1: 6.5
Area for t1: 1.3331705629813464
Illegal triangle
Side1: 1.0
Side2: 2.0
Side3: 3.0
BUILD SUCCESSFUL (total time: 0 seconds)
```