Chapter 9 – Object Oriented Programming -- helper file

See the file **Chapter9HelperFile.zip** for sample solution files. See the file UML.pdf for a sample UML diagram.

UML.PDF --- the UML diagram for the Rectangle utility class. Note the process of showing public and private methods and attributes.
Recatngle.java --- the utility class that defines an object with the attributes and characteristics outlined, below.
RectangleTest.java --- the test class that shows how to define objects and use the functionality of the methods.

1. Design/Write a **utility class** named **Rectangle** to represent any rectangular object. The class contains:

- 4 double data fields representing the width, length, area and perimeter of the rectangle. All fields must be defined as **private**.
- A no-arg constructor that creates a default rectangle with values of 0 for all of the data fields.
- A constructor that creates a rectangle with the specified length and width.
- Methods to **SET** and **GET** values for the all private data fields. The area of a rectangle is length * width. The perimeter of a rectangle is 2 * (length + width).
- A method to display the values of all the data fields. Don't forget to properly document the output. For example, display Area = 200 square units, instead of displaying 200.

✓ Write a **test class** (a new class file in a new .Java source file) that:
✓ Creates a rectangle object using the no-arg constructor. Display the values of the rectangle.
✓ Creates 2 rectangle objects with width 10 and length 20 and the other with width 7 and length 30. Display the values of each rectangle.
✓ Asks the user for the length and width of a rectangular object and create the object, calculate the area and perimeter and display the data fields of the object.
✓ Defines an array of 10 rectangular objects. Assign the values 1, 1 to the length and the width of the first object, 2, 2 to the length and the width of the second object, etc. This way the 10^{th} object will have length and width values of 10 and 10. I sure hope you use Loops. Use your methods to calculate the area and perimeter for these objects and display the result.