# Chapter 10 Programming Assignment

**Purpose:** To write an Object-Oriented application using abstraction, inheritance, encapsulation, and polymorphism to handle an inheritance structure of various shapes.

**Details:**

Implement the following Shape hierarchy:

*Shape*

Circle

Square

Triangle

Use the following as a guide:

* The Shape class should be abstract and contain *two abstract* methods:
  + getArea – Which calculates the area of the shape and returns that value as a double.
  + getPerimeter – Which calculates the perimeter of the shape and return that value as a double.

And *one* *concrete* method:

* + getName – that returns the shape’s name as a String.
* The Circle class’s constructor should take *a single* parameter of type double representing its diameter. No error checking is required.
* The Square class’s constructor should take *a single* parameter of type double representing the length of its sides. No error checking is required.
* The Triangle class’s constructor should take *a single* parameter of type double representing the length of its sides. No error checking is required.
* No set and get methods are needed. Do not include them in the any of the classes.

Create another class called TestShapes that contains the main method. This test class does not need to ask users for input. The class should do the following:

1. Create an array of size three, that uses Shape references. Fill the array with one of each concrete class in the hierarchy.
2. Iterate through the array of shapes and doing the following:
   1. Display the shape’s name.
   2. Display the shape’s area.
   3. Display the shape’s perimeter.

Upload all source files to Blackboard.

**Note:** Ensure that your program is properly formatted and it follows all Java naming conventions.