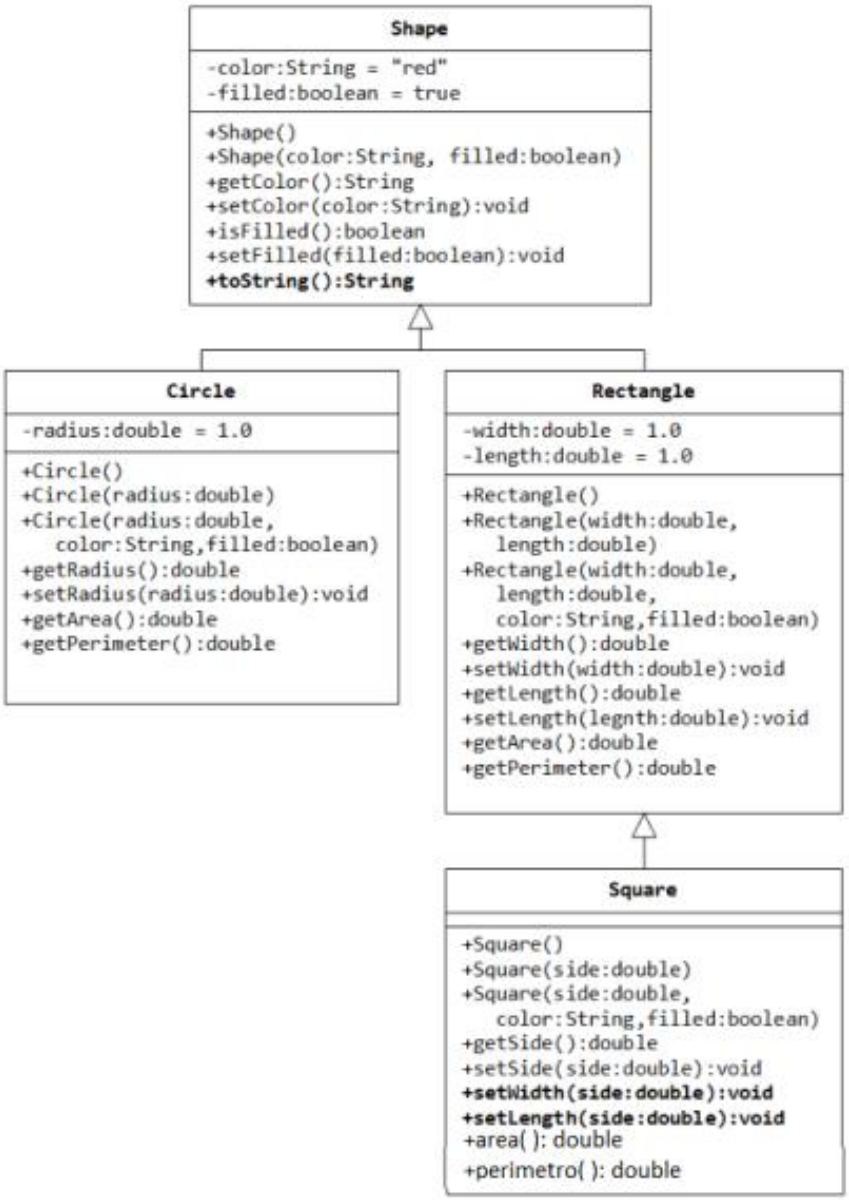
Lab: Inheritance

Objectives

You will learn to apply the concept of Inherence

Exercises

1. Write inheritance hierarchies as show in the diagram below. Use shape as the superclass of the hierarchy. The program has three classes: *Shape.java, Circle.java, Rectangle.java* and *Square.java*.



Shape class:

- Two instance variables `color` (`String`) and `filled` (`boolean`).
- Two constructors: a no-arg (no-argument) constructor that initializes the `color` to "green" and `filled` to `true`, and a constructor that initializes the `color` and `filled` to the given values.
- Getter and setter for all the instance variables. By convention, the getter for a `boolean` variable `xxx` is called `isXXX()` (instead of `getXxx()` for all the other types).
- A `toString()` method that returns "`A Shape with color of xxx and filled/Not filled`".

`Circle` class:

- An instance variable `radius` (`double`).
- Three constructors as shown. The no-arg constructor initializes the radius to `1.0`.
- Getter and setter for the instance variable `radius`.
- Methods `getArea()` and `getPerimeter()`.

`Rectangle` class:

- Two instance variables `width` (`double`) and `length` (`double`).
- Three constructors as shown. The no-arg constructor initializes the `width` and `length` to `1.0`.
- Getter and setter for all the instance variables.
- Methods `getArea()` and `getPerimeter()`.

Square Class

- `Square` has no instance variable, but inherits the instance variables width and length from its superclass Rectangle
- Provide the appropriate constructors (as shown in the class diagram).
- Implement `Area()` and `Perimetro()` using `getArea()` and `getPerimeter` from the `superclass`.

Test Class

- The program must instantiates objects of your classes and outputs the area of each objects.
- Be sure to demo the advantages of use inheritance (i.e. call to methods in the superclass from the subclass)