ITCS 209 Object Oriented Programming	Name:	Lab	Challenge Bonus	
	ID:			
	Section:			

Lab07: Polymorphism

Task 1

Using what you have learned in the classes about Inheritance and Polymorphism, write a program that consists of shapes such as triangles and rectangles.

You are provided with 4 java files:

Try implementing Rectangle.java and Triangle.java from scratch by yourself first. If things get too challenging, find the skeleton files in hint.zip (pw: `polymorphism`)

1. Shape.java – **DO NOT** modify this one.
2. Rectangle.java – subclass of Shape, **Implement this class**
3. Triangle.java – subclass of Shape, **Implement this class**
4. ShapeTester.java – main class (run your program using this class), **DO NOT** modify

Rectangle.java

- **Override**
  - `double getArea()` from Shape.java to compute and return area of a rectangle using **area = length *width**
  - `String toString()` to return the following string : Rectangle[length=4, width=5,Shape[color=red]]

- **Overload** the method `double getArea(double length, double width)` then compute and return area of a rectangle. **Note:** you can call `getArea()` inside this method

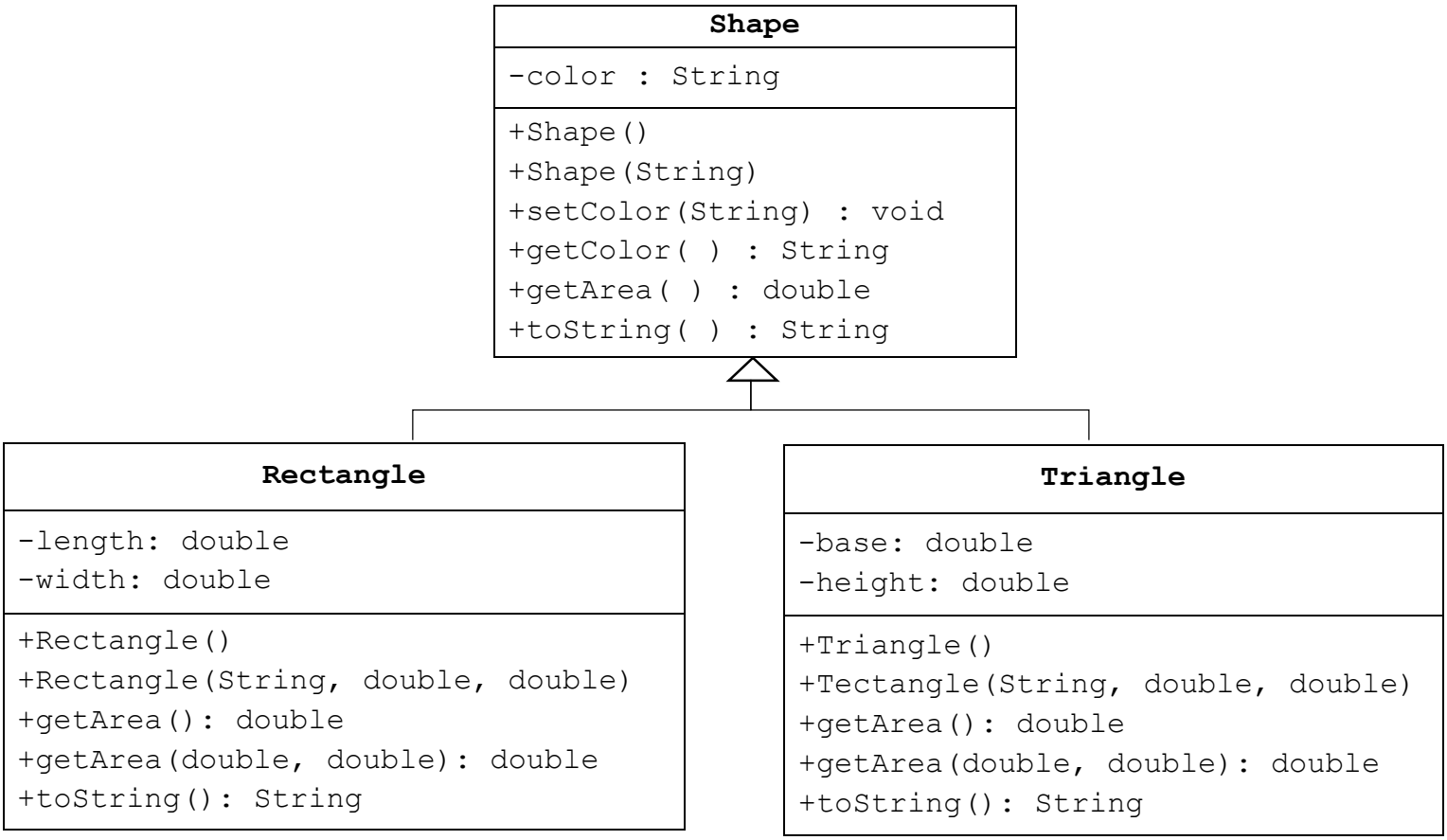
Triangle.java

- **Override** the method
  - `double getArea()` from Shape.java to compute and return area of a triangle using **area = 0.5*base*height**
  - `String toString()` to return the following string : Triangle[base=4,height=5,Shape[color=blue]]
- **Overload** the method `double getArea(double base, double height)` then compute and return area of a triangle. **Note:** you can call `getArea()` inside this method

Output

```
Rectangle[length=4.0,width=5.0,Shape[color=red]]
Area is 20.0
Triangle[base=4.0,height=5.0,Shape[color=blue]]
Area is 10.0
--Test superclass method--
Shape[color=blue]
Shape unknown! Cannot compute area!
Area is 0.0
--Test overload method--
Area is 50.0
Rectangle[length=5.0,width=10.0,Shape[color=green]]
--Test overload method--
Area is 25.0
Triangle[base=5.0,height=10.0,Shape[color=yellow]]
```

You can refer to a class diagram below for more details about variable, constructors and methods.



Challenge Bonus (Optional):

Create a new class of your choice that extends `Shape.java`. **Note**: It could be any shapes such as circles, hexagons and so on.

Modify ShapeTester.java to have 2 objects of your class and compute an area and print the output.