


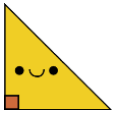

Getting into Shape Peer Assessment

Name _____ Date _____ Period ____

Peer Reviewer _____

Circle in pencil methods that don't exist, need repair, or need improved documentation. Method comments must have a quality method summary first, followed by the documentation of parameters (if any), and finally followed by return documentation (if any). There are no default constructors. Methods that are in a red, bold font should be documented as having been overridden (`@Override`). When reading the documentation, encourage the documentation to be both complete and concise without just repeating the words that make up the method name.

GeometricShape	Constructor with one parameter that is a name	getLabel	setLabel	calculatePerimeter	calculateArea	**compareTo**	**compareAnotherWay**
	isNearlyEqual	**compareToNearly**	isPolygon	**toString**			
AnotherComparable	compareAnotherWay						
Nearable	isNearlyEqual	compareToNearly					
Circle	Constructor with one parameter that is a name	Constructor with two parameters	getRadius	setRadius	**calculatePerimeter**	**calculateArea**	
		equals Should call compareTo	**compareTo** Should NOT call equals	**isPolygon**	**toString**		
Polygon	Constructor with two parameters	**isPolygon**	isRegular	getNumSides			
Triangle	Constructor with four parameters	getSideA	getSideB	getSideC	**calculatePerimeter**	**calculateArea**	
	orderSides	setSideA	setSideB	setSideC	**equals** Should NOT call equals	**compareTo** Should call equals	
		isRegular	**toString**				
Equilateral	Constructor with two parameters: a name and side length	**setSideA**	**setSideB**	**setSideC**			
IsoscelesRightTriangle	Constructor with two parameters: a name and leg length	**setSideA**	**setSideB**	**setSideC**			
Rectangle	Constructor with three parameters	getWidth	getLength	setWidth	setLength	**calculatePerimeter**	**calculateArea**
		equals Should NOT call compareTo	**compareTo** Should call equals	**isRegular**	**toString**		
Square	Constructor with one parameter that is a name	Constructor with two parameters	**setWidth**	**setLength**			

Run the tester. Finish the peer review on the next page.

Getting into Shape Peer Assessment

Compare the output to the printed output. Circle in pencil each output that needs attention.

First illegal	Second illegal	Third illegal	Fourth illegal	Fifth illegal
Isosceles Right setSideC	Equilateral setSideB	Isosceles Right setSideA		
alex.compareTo(testObject)	rex.compareTo(testObject)	alex.compareAnotherWay(round)	rex.compareAnotherWay(round)	
alex	almost	bob	cathy	derek
edward	fred	gail	helena	iris
jerome	kate			

Indicate your general feedback by circling the best wording that reflects your assessment.

Documentation	**High quality**	**Mostly complete and correct**	**Needs improvement**
Checkstyle	**All classes/interface pass CS**	**All but 1 or 2 do not pass**	**More than two do not pass**
DiffNow	**Only 1 difference – the top line**	**Write in how many differences _____**	

Inspected by _____ Final date of inspection _____

Remember these remaining steps:

Step 1: Give an electronic copy to the person whose code you are reviewing.

Step 2: Upload an electronic copy of this peer review to the correct CW: submission in Schoology along with a zip of the final code that you reviewed.

Step 3: Upload your own project with its peer assessment to the correct PROJ: submission in Schoology.