JavaFX Shapes Documentation

This document contains an overview of the most important methods relating to Shapes in JavaFX. Note that there are many more methods and classes that you are welcome to use in addition to the ones listed here. For a complete list of these, please see the Javadocs. For a more practical overview of JavaFX beyond Shapes, please see our JavaFX Guide.

Table of Contents

javafx.scene.shape.Shape

Inheritance Summary

javafx.scene.Node
    javafx.scene.shape.Shape

Method Summary

Return Type	Method and Description
Paint	**getFill**() Returns the color of the Shape's interior. Note that Color is a subclass of Paint, so you may have to cast the returned object in order to access the actual color it represents.
Paint	**getStroke**() Returns the color of the stroke property (AKA the border). Note that Color is a subclass of Paint, so you may have to cast the returned object in order to access the actual color it represents.
StrokeType	**getStrokeType**() Returns the direction (inside, centered, or outside) that the strokeWidth is applied to the boundary of the shape.
double	**getStrokeWidth**() Returns the width of the Shape's stroke (AKA border).
void	**setFill**(**Paint** value) Sets the color of the Shape's interior.
void	**setStroke**(**Paint** value) Sets the color of the the Shape's stroke (AKA border).
void	**setStrokeType**(**StrokeType** value) Sets the direction (inside, centered, or outside) that the strokeWidth is applied to the boundary of the shape.
void	**setStrokeWidth**(double value) Sets the width of the Shape's stroke (AKA border).
static **Shape**	**subtract**(**Shape** shape1, **Shape** shape2) Returns a new Shape which is created by subtracting the specified second shape from the first shape.
static **Shape**	**union**(**Shape** shape1, **Shape** shape2) Returns a new Shape which is created as a union of the specified input shapes.
static **Shape**	**intersect**(**Shape** shape1, **Shape** shape2) Returns a new Shape which is created as an intersection of the specified input shapes.

javafx.scene.shape.Ellipse

Inheritance Summary

javafx.scene.Node
    javafx.scene.shape.Shape
        javafx.scene.shape.Ellipse

Constructor Summary

Constructors and Descriptions
Ellipse() Creates an empty instance of Ellipse.
Ellipse(double radiusX, double radiusY) Creates an instance of Ellipse of the given size.
Ellipse(double centerX, double centerY, double radiusX, double radiusY) Creates an instance of Ellipse of the given position and size.

Method Summary

Return Type	Method and Description
double	**getCenterX**() Returns the x-coordinate of the Ellipse's center.
double	**getCenterY**() Returns the y-coordinate of the Ellipse's center.
double	**getRadiusX**() Returns the radius along the x-axis of the Ellipse.
double	**getRadiusY**() Returns the radius along the y-axis of the Ellipse.
void	**setCenterX**(double value) Sets the x-coordinate of the Ellipse's center.
void	**setCenterY**(double value) Sets the y-coordinate of the Ellipse's center.
void	**setRadiusX**(double value) Sets the radius along the x-axis of the Ellipse.
void	**setRadiusY**(double value) Sets the radius along the y-axis of the Ellipse.

javafx.scene.shape.Circle

Inheritance Summary

javafx.scene.Node
    javafx.scene.shape.Shape
        javafx.scene.shape.Circle

Constructor Summary

Constructors and Descriptions
Circle() Creates an empty instance of Circle.
Circle(double radius) Creates a new instance of Circle with a specified radius.
Circle(double centerX, double centerY, double radius) Creates a new instance of Circle with a specified position and radius.
Circle(double centerX, double centerY, double radius, **Paint** fill) Creates a new instance of Circle with a specified position, radius and fill color.
Circle(double radius, **Paint** fill) Creates a new instance of Circle with a specified radius and fill color.

Method Summary

Return Type	Method and Description
double	**getCenterX**() Returns the x-coordinate of the Circle's center.
double	**getCenterY**() Returns the y-coordinate of the Circle's center.
double	**getRadius**() Returns the radius of the Circle.
void	**setCenterX**(double value) Sets the x-coordinate of the Circle's center.
void	**setCenterY**(double value) Sets the y-coordinate of the Circle's center.
void	**setRadius**(double value) Sets the radius of the Circle.

javafx.scene.shape.Rectangle

Inheritance Summary

javafx.scene.Node
    javafx.scene.shape.Shape
        javafx.scene.shape.Rectangle

Constructor Summary

Constructors and Descriptions
Rectangle() Creates an empty instance of Rectangle.
Rectangle(double width, double height) Creates a new instance of Rectangle with the given size.
Rectangle(double x, double y, double width, double height) Creates a new instance of Rectangle with the given position and size.
Rectangle(double width, double height, **Paint** fill) Creates a new instance of Rectangle with the given size and fill color.

Method Summary

Return Type	Method and Description
double	**getArcHeight**() Returns the vertical diameter of the arc at the four corners of the Rectangle.
double	**getArcWidth**() Returns the horizontal diameter of the arc at the four corners of the Rectangle.
double	**getHeight**() Returns the height of the Rectangle.
double	**getWidth**() Returns the width of the Rectangle.
double	**getX**() Returns the x-coordinate of the upper-left corner of the Rectangle.
double	**getY**() Returns the y-coordinate of the upper-left corner of the Rectangle.
void	**setArcHeight**(double value) Sets the vertical diameter of the arc at the four corners of the Rectangle.

void	**setArcWidth**(double value) Sets the horizontal diameter of the arc at the four corners of the Rectangle.
void	**setHeight**(double value) Sets the height of the Rectangle.
void	**setWidth**(double value) Sets the width of the Rectangle.
void	**setX**(double value) Sets the x-coordinate of the upper-left corner of the Rectangle.
void	**setY**(double value) Sets the y-coordinate of the upper-left corner of the Rectangle.

javafx.scene.shape.Polygon

Constructor Summary

Constructors and Descriptions
Polygon() Creates an empty instance of Polygon.
Polygon(double... points) Creates a new instance of Polygon. This constructor takes in any number of points (as defined by the "double... points"). This property is called "varargs", a variable number of arguments. Behind the scenes, somewhere in Java, the variable number of arguments are converted into an array. Enter your doubles in x, y pairs. Example: Polygon(0.0, 0.0, 10.0, 0, 20, 20.0) is the geometric equivalent of making a triangle with vertices (0, 0), (10, 0), (20, 20).

Method Summary

Return Type	Method and Description
ObservableList<Text>	**getPoints**() Returns the coordinates of the Polygon vertices. Note: the returned ObservableList is like an ArrayList in that they are both descendants of the List interface. Use it like such!

Polygon does not specify its own location or size methods but you can use the methods it inherits from Node. (See Graphics II lecture for more info on this.) Hint: think about the Layout of the Node.