AWT RECTANGLE2D CLASS

http://www.tutorialspoint.com/awt/awt_rectangle2d_class.htm

Copyright © tutorialspoint.com

Introduction

The Rectangle2D class states a rectangle defined by a location *x, y* and dimension *wxh*.

Class declaration

Following is the declaration for **java.awt.geom.Rectangle2D** class:

```
public abstract class Rectangle2D
   extends RectangularShape
```

Field

Following are the fields for **java.awt.geom.Arc2D** class:

- **static int OUT_BOTTOM** -- The bitmask that indicates that a point lies below this Rectangle2D.
- **static int OUT_LEFT** -- The bitmask that indicates that a point lies to the left of this Rectangle2D.
- **static int OUT_RIGHT** -- The bitmask that indicates that a point lies to the right of this Rectangle2D.
- **static int OUT_TOP** -- The bitmask that indicates that a point lies above this Rectangle2D.

Class constructors

S.N.	Constructor & Description
1	**protected Rectangle2D** This is an abstract class that cannot be instantiated directly.

Class methods

S.N.	Method & Description
1	**void add*doublenewx, doublenewy*** Adds a point, specified by the double precision arguments newx and newy, to this Rectangle2D.
2	**void add*Point2Dpt*** Adds the Point2D object pt to this Rectangle2D.
3	**void add*Rectangle2Dr*** Adds a Rectangle2D object to this Rectangle2D.

4

boolean contains*doublex, doubley*

Tests if the specified coordinates are inside the boundary of the Shape.

5

boolean contains*doublex, doubley, doublew, doubleh*

Tests if the interior of the Shape entirely contains the specified rectangular area.

6

abstract Rectangle2D createIntersection*Rectangle2Dr*

Returns a new Rectangle2D object representing the intersection of this Rectangle2D with the specified Rectangle2D.

7

abstract Rectangle2D createUnion*Rectangle2Dr*

Returns a new Rectangle2D object representing the union of this Rectangle2D with the specified Rectangle2D.

8

boolean equals*Objectobj*

Determines whether or not the specified Object is equal to this Rectangle2D.

9

Rectangle2D getBounds2D

Returns a high precision and more accurate bounding box of the Shape than the getBounds method.

10

PathIterator getPathIterator*AffineTransformat*

Returns an iteration object that defines the boundary of this Rectangle2D.

11

PathIterator getPathIterator*AffineTransformat, doubleflatness*

Returns an iteration object that defines the boundary of the flattened Rectangle2D.

12

int hashCode

Returns the hashcode for this Rectangle2D.

13

static void intersect*Rectangle2Dsrc1, Rectangle2Dsrc2, Rectangle2Ddest*

Intersects the pair of specified source Rectangle2D objects and puts the result into the specified destination Rectangle2D object.

14

boolean intersects*doublex, doubley, doublew, doubleh*

Tests if the interior of the Shape intersects the interior of a specified rectangular area.

15	**boolean intersectsLine*doublex1, doubley1, doublex2, doubley2*** Tests if the specified line segment intersects the interior of this Rectangle2D.
16	**boolean intersectsLine*Line2Dl*** Tests if the specified line segment intersects the interior of this Rectangle2D.
17	**abstract int outcode*doublex, doubley*** Determines where the specified coordinates lie with respect to this Rectangle2D.
18	**int outcode*Point2Dp*** Determines where the specified Point2D lies with respect to this Rectangle2D.
19	**void setFrame*doublex, doubley, doublew, doubleh*** Sets the location and size of the outer bounds of this Rectangle2D to the specified rectangular values.
20	**abstract void setRect*doublex, doubley, doublew, doubleh*** Sets the location and size of this Rectangle2D to the specified double values.
21	**void setRect*Rectangle2Dr*** Sets this Rectangle2D to be the same as the specified Rectangle2D.
22	**static void union*Rectangle2Dsrc1, Rectangle2Dsrc2, Rectangle2Ddest*** Unions the pair of source Rectangle2D objects and puts the result into the specified destination Rectangle2D object.

Methods inherited

This class inherits methods from the following classes:

- java.awt.geom.RectangularShape
- java.lang.Object

Ellipse2D Example

Create the following java program using any editor of your choice in say **D:/ > AWT > com > tutorialspoint > gui >**

AWTGraphicsDemo.java

```
package com.tutorialspoint.gui;

import java.awt.*;
import java.awt.event.*;
import java.awt.geom.*;

public class AWTGraphicsDemo extends Frame {

   public AWTGraphicsDemo(){
      super("Java AWT Examples");
      prepareGUI();
   }

   public static void main(String[] args){
      AWTGraphicsDemo  awtGraphicsDemo = new AWTGraphicsDemo();
      awtGraphicsDemo.setVisible(true);
   }

   private void prepareGUI(){
      setSize(400,400);
      addWindowListener(new WindowAdapter() {
         public void windowClosing(WindowEvent windowEvent){
            System.exit(0);
         }
      });
   }

   @Override
   public void paint(Graphics g) {
      Rectangle2D shape = new Rectangle2D.Float();
      shape.setFrame(100, 150, 200,100);
      Graphics2D g2 = (Graphics2D) g;
      g2.draw (shape);
      Font font = new Font("Serif", Font.PLAIN, 24);
      g2.setFont(font);
      g.drawString("Welcome to TutorialsPoint", 50, 70);
      g2.drawString("Rectangle2D.Rectangle", 100, 120);
   }
}
```

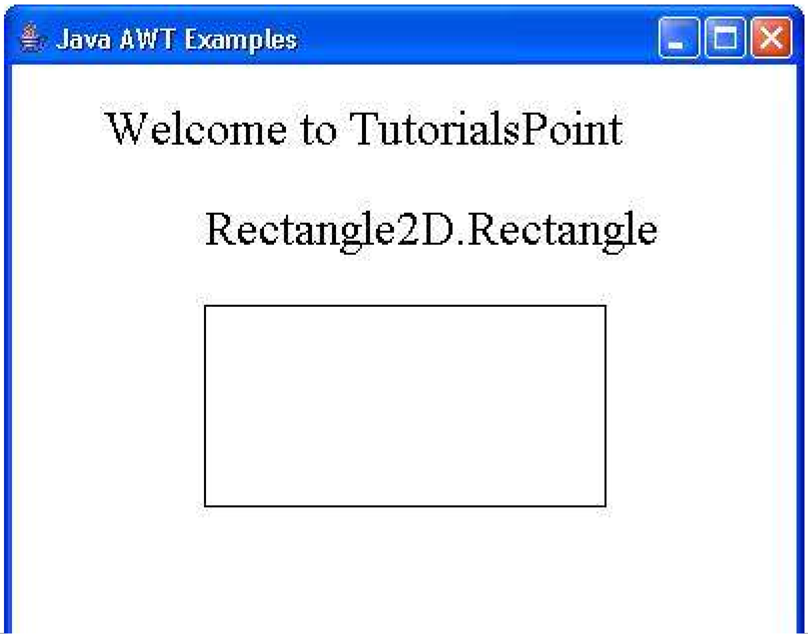
Compile the program using command prompt. Go to **D:/ > AWT** and type the following command.

```
D:\AWT>javac com\tutorialspoint\gui\AWTGraphicsDemo.java
```

If no error comes that means compilation is successful. Run the program using following command.

```
D:\AWT>java com.tutorialspoint.gui.AWTGraphicsDemo
```

Verify the following output



Welcome to TutorialsPoint

Rectangle2D.Rectangle

Loading [MathJax]/jax/output/HTML-CSS/jax.js