| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/RectangularShape.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV CLASS**](http://docs.google.com/java/awt/geom/Rectangle2D.Float.html)   [**NEXT CLASS**](http://docs.google.com/java/awt/geom/RoundRectangle2D.html) | [**FRAMES**](http://docs.google.com/index.html?java/awt/geom/RectangularShape.html)    [**NO FRAMES**](http://docs.google.com/RectangularShape.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#3znysh7) | [METHOD](#2et92p0) | DETAIL: FIELD | [CONSTR](#1t3h5sf) | [METHOD](#2s8eyo1) |

## **java.awt.geom**

Class RectangularShape

[java.lang.Object](http://docs.google.com/java/lang/Object.html)  
 **java.awt.geom.RectangularShape**

**All Implemented Interfaces:** [Shape](http://docs.google.com/java/awt/Shape.html), [Cloneable](http://docs.google.com/java/lang/Cloneable.html) **Direct Known Subclasses:** [Arc2D](http://docs.google.com/java/awt/geom/Arc2D.html), [Ellipse2D](http://docs.google.com/java/awt/geom/Ellipse2D.html), [Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html), [RoundRectangle2D](http://docs.google.com/java/awt/geom/RoundRectangle2D.html)

public abstract class **RectangularShape**extends [Object](http://docs.google.com/java/lang/Object.html)implements [Shape](http://docs.google.com/java/awt/Shape.html), [Cloneable](http://docs.google.com/java/lang/Cloneable.html)

RectangularShape is the base class for a number of [Shape](http://docs.google.com/java/awt/Shape.html) objects whose geometry is defined by a rectangular frame. This class does not directly specify any specific geometry by itself, but merely provides manipulation methods inherited by a whole category of Shape objects. The manipulation methods provided by this class can be used to query and modify the rectangular frame, which provides a reference for the subclasses to define their geometry.

**Since:** 1.2

| **Constructor Summary** | |
| --- | --- |
| protected | [**RectangularShape**](http://docs.google.com/java/awt/geom/RectangularShape.html#RectangularShape())()            This is an abstract class that cannot be instantiated directly. |

| **Method Summary** | |
| --- | --- |
| [Object](http://docs.google.com/java/lang/Object.html) | [**clone**](http://docs.google.com/java/awt/geom/RectangularShape.html#clone())()            Creates a new object of the same class and with the same contents as this object. |
| boolean | [**contains**](http://docs.google.com/java/awt/geom/RectangularShape.html#contains(java.awt.geom.Point2D))([Point2D](http://docs.google.com/java/awt/geom/Point2D.html) p)            Tests if a specified [Point2D](http://docs.google.com/java/awt/geom/Point2D.html) is inside the boundary of the Shape. |
| boolean | [**contains**](http://docs.google.com/java/awt/geom/RectangularShape.html#contains(java.awt.geom.Rectangle2D))([Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html) r)            Tests if the interior of the Shape entirely contains the specified Rectangle2D. |
| [Rectangle](http://docs.google.com/java/awt/Rectangle.html) | [**getBounds**](http://docs.google.com/java/awt/geom/RectangularShape.html#getBounds())()            Returns an integer [Rectangle](http://docs.google.com/java/awt/Rectangle.html) that completely encloses the Shape. |
| double | [**getCenterX**](http://docs.google.com/java/awt/geom/RectangularShape.html#getCenterX())()            Returns the X coordinate of the center of the framing rectangle of the Shape in double precision. |
| double | [**getCenterY**](http://docs.google.com/java/awt/geom/RectangularShape.html#getCenterY())()            Returns the Y coordinate of the center of the framing rectangle of the Shape in double precision. |
| [Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html) | [**getFrame**](http://docs.google.com/java/awt/geom/RectangularShape.html#getFrame())()            Returns the framing [Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html) that defines the overall shape of this object. |
| abstract  double | [**getHeight**](http://docs.google.com/java/awt/geom/RectangularShape.html#getHeight())()            Returns the height of the framing rectangle in double precision. |
| double | [**getMaxX**](http://docs.google.com/java/awt/geom/RectangularShape.html#getMaxX())()            Returns the largest X coordinate of the framing rectangle of the Shape in double precision. |
| double | [**getMaxY**](http://docs.google.com/java/awt/geom/RectangularShape.html#getMaxY())()            Returns the largest Y coordinate of the framing rectangle of the Shape in double precision. |
| double | [**getMinX**](http://docs.google.com/java/awt/geom/RectangularShape.html#getMinX())()            Returns the smallest X coordinate of the framing rectangle of the Shape in double precision. |
| double | [**getMinY**](http://docs.google.com/java/awt/geom/RectangularShape.html#getMinY())()            Returns the smallest Y coordinate of the framing rectangle of the Shape in double precision. |
| [PathIterator](http://docs.google.com/java/awt/geom/PathIterator.html) | [**getPathIterator**](http://docs.google.com/java/awt/geom/RectangularShape.html#getPathIterator(java.awt.geom.AffineTransform,%20double))([AffineTransform](http://docs.google.com/java/awt/geom/AffineTransform.html) at, double flatness)            Returns an iterator object that iterates along the Shape object's boundary and provides access to a flattened view of the outline of the Shape object's geometry. |
| abstract  double | [**getWidth**](http://docs.google.com/java/awt/geom/RectangularShape.html#getWidth())()            Returns the width of the framing rectangle in double precision. |
| abstract  double | [**getX**](http://docs.google.com/java/awt/geom/RectangularShape.html#getX())()            Returns the X coordinate of the upper-left corner of the framing rectangle in double precision. |
| abstract  double | [**getY**](http://docs.google.com/java/awt/geom/RectangularShape.html#getY())()            Returns the Y coordinate of the upper-left corner of the framing rectangle in double precision. |
| boolean | [**intersects**](http://docs.google.com/java/awt/geom/RectangularShape.html#intersects(java.awt.geom.Rectangle2D))([Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html) r)            Tests if the interior of the Shape intersects the interior of a specified Rectangle2D. |
| abstract  boolean | [**isEmpty**](http://docs.google.com/java/awt/geom/RectangularShape.html#isEmpty())()            Determines whether the RectangularShape is empty. |
| abstract  void | [**setFrame**](http://docs.google.com/java/awt/geom/RectangularShape.html#setFrame(double,%20double,%20double,%20double))(double x, double y, double w, double h)            Sets the location and size of the framing rectangle of this Shape to the specified rectangular values. |
| void | [**setFrame**](http://docs.google.com/java/awt/geom/RectangularShape.html#setFrame(java.awt.geom.Point2D,%20java.awt.geom.Dimension2D))([Point2D](http://docs.google.com/java/awt/geom/Point2D.html) loc, [Dimension2D](http://docs.google.com/java/awt/geom/Dimension2D.html) size)            Sets the location and size of the framing rectangle of this Shape to the specified [Point2D](http://docs.google.com/java/awt/geom/Point2D.html) and [Dimension2D](http://docs.google.com/java/awt/geom/Dimension2D.html), respectively. |
| void | [**setFrame**](http://docs.google.com/java/awt/geom/RectangularShape.html#setFrame(java.awt.geom.Rectangle2D))([Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html) r)            Sets the framing rectangle of this Shape to be the specified Rectangle2D. |
| void | [**setFrameFromCenter**](http://docs.google.com/java/awt/geom/RectangularShape.html#setFrameFromCenter(double,%20double,%20double,%20double))(double centerX, double centerY, double cornerX, double cornerY)            Sets the framing rectangle of this Shape based on the specified center point coordinates and corner point coordinates. |
| void | [**setFrameFromCenter**](http://docs.google.com/java/awt/geom/RectangularShape.html#setFrameFromCenter(java.awt.geom.Point2D,%20java.awt.geom.Point2D))([Point2D](http://docs.google.com/java/awt/geom/Point2D.html) center, [Point2D](http://docs.google.com/java/awt/geom/Point2D.html) corner)            Sets the framing rectangle of this Shape based on a specified center Point2D and corner Point2D. |
| void | [**setFrameFromDiagonal**](http://docs.google.com/java/awt/geom/RectangularShape.html#setFrameFromDiagonal(double,%20double,%20double,%20double))(double x1, double y1, double x2, double y2)            Sets the diagonal of the framing rectangle of this Shape based on the two specified coordinates. |
| void | [**setFrameFromDiagonal**](http://docs.google.com/java/awt/geom/RectangularShape.html#setFrameFromDiagonal(java.awt.geom.Point2D,%20java.awt.geom.Point2D))([Point2D](http://docs.google.com/java/awt/geom/Point2D.html) p1, [Point2D](http://docs.google.com/java/awt/geom/Point2D.html) p2)            Sets the diagonal of the framing rectangle of this Shape based on two specified Point2D objects. |

| **Methods inherited from class java.lang.**[**Object**](http://docs.google.com/java/lang/Object.html) |
| --- |
| [equals](http://docs.google.com/java/lang/Object.html#equals(java.lang.Object)), [finalize](http://docs.google.com/java/lang/Object.html#finalize()), [getClass](http://docs.google.com/java/lang/Object.html#getClass()), [hashCode](http://docs.google.com/java/lang/Object.html#hashCode()), [notify](http://docs.google.com/java/lang/Object.html#notify()), [notifyAll](http://docs.google.com/java/lang/Object.html#notifyAll()), [toString](http://docs.google.com/java/lang/Object.html#toString()), [wait](http://docs.google.com/java/lang/Object.html#wait()), [wait](http://docs.google.com/java/lang/Object.html#wait(long)), [wait](http://docs.google.com/java/lang/Object.html#wait(long,%20int)) |

| **Methods inherited from interface java.awt.**[**Shape**](http://docs.google.com/java/awt/Shape.html) |
| --- |
| [contains](http://docs.google.com/java/awt/Shape.html#contains(double,%20double)), [contains](http://docs.google.com/java/awt/Shape.html#contains(double,%20double,%20double,%20double)), [getBounds2D](http://docs.google.com/java/awt/Shape.html#getBounds2D()), [getPathIterator](http://docs.google.com/java/awt/Shape.html#getPathIterator(java.awt.geom.AffineTransform)), [intersects](http://docs.google.com/java/awt/Shape.html#intersects(double,%20double,%20double,%20double)) |

| **Constructor Detail** |
| --- |

### RectangularShape

protected **RectangularShape**()

This is an abstract class that cannot be instantiated directly.

**Since:** 1.2 **See Also:**[Arc2D](http://docs.google.com/java/awt/geom/Arc2D.html), [Ellipse2D](http://docs.google.com/java/awt/geom/Ellipse2D.html), [Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html), [RoundRectangle2D](http://docs.google.com/java/awt/geom/RoundRectangle2D.html)

| **Method Detail** |
| --- |

### getX

public abstract double **getX**()

Returns the X coordinate of the upper-left corner of the framing rectangle in double precision.

**Returns:**the X coordinate of the upper-left corner of the framing rectangle.**Since:** 1.2

### getY

public abstract double **getY**()

Returns the Y coordinate of the upper-left corner of the framing rectangle in double precision.

**Returns:**the Y coordinate of the upper-left corner of the framing rectangle.**Since:** 1.2

### getWidth

public abstract double **getWidth**()

Returns the width of the framing rectangle in double precision.

**Returns:**the width of the framing rectangle.**Since:** 1.2

### getHeight

public abstract double **getHeight**()

Returns the height of the framing rectangle in double precision.

**Returns:**the height of the framing rectangle.**Since:** 1.2

### getMinX

public double **getMinX**()

Returns the smallest X coordinate of the framing rectangle of the Shape in double precision.

**Returns:**the smallest X coordinate of the framing rectangle of the Shape.**Since:** 1.2

### getMinY

public double **getMinY**()

Returns the smallest Y coordinate of the framing rectangle of the Shape in double precision.

**Returns:**the smallest Y coordinate of the framing rectangle of the Shape.**Since:** 1.2

### getMaxX

public double **getMaxX**()

Returns the largest X coordinate of the framing rectangle of the Shape in double precision.

**Returns:**the largest X coordinate of the framing rectangle of the Shape.**Since:** 1.2

### getMaxY

public double **getMaxY**()

Returns the largest Y coordinate of the framing rectangle of the Shape in double precision.

**Returns:**the largest Y coordinate of the framing rectangle of the Shape.**Since:** 1.2

### getCenterX

public double **getCenterX**()

Returns the X coordinate of the center of the framing rectangle of the Shape in double precision.

**Returns:**the X coordinate of the center of the framing rectangle of the Shape.**Since:** 1.2

### getCenterY

public double **getCenterY**()

Returns the Y coordinate of the center of the framing rectangle of the Shape in double precision.

**Returns:**the Y coordinate of the center of the framing rectangle of the Shape.**Since:** 1.2

### getFrame

public [Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html) **getFrame**()

Returns the framing [Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html) that defines the overall shape of this object.

**Returns:**a Rectangle2D, specified in double coordinates.**Since:** 1.2 **See Also:**[setFrame(double, double, double, double)](http://docs.google.com/java/awt/geom/RectangularShape.html#setFrame(double,%20double,%20double,%20double)), [setFrame(Point2D, Dimension2D)](http://docs.google.com/java/awt/geom/RectangularShape.html#setFrame(java.awt.geom.Point2D,%20java.awt.geom.Dimension2D)), [setFrame(Rectangle2D)](http://docs.google.com/java/awt/geom/RectangularShape.html#setFrame(java.awt.geom.Rectangle2D))

### isEmpty

public abstract boolean **isEmpty**()

Determines whether the RectangularShape is empty. When the RectangularShape is empty, it encloses no area.

**Returns:**true if the RectangularShape is empty; false otherwise.**Since:** 1.2

### setFrame

public abstract void **setFrame**(double x,  
 double y,  
 double w,  
 double h)

Sets the location and size of the framing rectangle of this Shape to the specified rectangular values.

**Parameters:**x - the X coordinate of the upper-left corner of the specified rectangular shapey - the Y coordinate of the upper-left corner of the specified rectangular shapew - the width of the specified rectangular shapeh - the height of the specified rectangular shape**Since:** 1.2 **See Also:**[getFrame()](http://docs.google.com/java/awt/geom/RectangularShape.html#getFrame())

### setFrame

public void **setFrame**([Point2D](http://docs.google.com/java/awt/geom/Point2D.html) loc,  
 [Dimension2D](http://docs.google.com/java/awt/geom/Dimension2D.html) size)

Sets the location and size of the framing rectangle of this Shape to the specified [Point2D](http://docs.google.com/java/awt/geom/Point2D.html) and [Dimension2D](http://docs.google.com/java/awt/geom/Dimension2D.html), respectively. The framing rectangle is used by the subclasses of RectangularShape to define their geometry.

**Parameters:**loc - the specified Point2Dsize - the specified Dimension2D**Since:** 1.2 **See Also:**[getFrame()](http://docs.google.com/java/awt/geom/RectangularShape.html#getFrame())

### setFrame

public void **setFrame**([Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html) r)

Sets the framing rectangle of this Shape to be the specified Rectangle2D. The framing rectangle is used by the subclasses of RectangularShape to define their geometry.

**Parameters:**r - the specified Rectangle2D**Since:** 1.2 **See Also:**[getFrame()](http://docs.google.com/java/awt/geom/RectangularShape.html#getFrame())

### setFrameFromDiagonal

public void **setFrameFromDiagonal**(double x1,  
 double y1,  
 double x2,  
 double y2)

Sets the diagonal of the framing rectangle of this Shape based on the two specified coordinates. The framing rectangle is used by the subclasses of RectangularShape to define their geometry.

**Parameters:**x1 - the X coordinate of the start point of the specified diagonaly1 - the Y coordinate of the start point of the specified diagonalx2 - the X coordinate of the end point of the specified diagonaly2 - the Y coordinate of the end point of the specified diagonal**Since:** 1.2

### setFrameFromDiagonal

public void **setFrameFromDiagonal**([Point2D](http://docs.google.com/java/awt/geom/Point2D.html) p1,  
 [Point2D](http://docs.google.com/java/awt/geom/Point2D.html) p2)

Sets the diagonal of the framing rectangle of this Shape based on two specified Point2D objects. The framing rectangle is used by the subclasses of RectangularShape to define their geometry.

**Parameters:**p1 - the start Point2D of the specified diagonalp2 - the end Point2D of the specified diagonal**Since:** 1.2

### setFrameFromCenter

public void **setFrameFromCenter**(double centerX,  
 double centerY,  
 double cornerX,  
 double cornerY)

Sets the framing rectangle of this Shape based on the specified center point coordinates and corner point coordinates. The framing rectangle is used by the subclasses of RectangularShape to define their geometry.

**Parameters:**centerX - the X coordinate of the specified center pointcenterY - the Y coordinate of the specified center pointcornerX - the X coordinate of the specified corner pointcornerY - the Y coordinate of the specified corner point**Since:** 1.2

### setFrameFromCenter

public void **setFrameFromCenter**([Point2D](http://docs.google.com/java/awt/geom/Point2D.html) center,  
 [Point2D](http://docs.google.com/java/awt/geom/Point2D.html) corner)

Sets the framing rectangle of this Shape based on a specified center Point2D and corner Point2D. The framing rectangle is used by the subclasses of RectangularShape to define their geometry.

**Parameters:**center - the specified center Point2Dcorner - the specified corner Point2D**Since:** 1.2

### contains

public boolean **contains**([Point2D](http://docs.google.com/java/awt/geom/Point2D.html) p)

Tests if a specified [Point2D](http://docs.google.com/java/awt/geom/Point2D.html) is inside the boundary of the Shape.

**Specified by:**[contains](http://docs.google.com/java/awt/Shape.html#contains(java.awt.geom.Point2D)) in interface [Shape](http://docs.google.com/java/awt/Shape.html) **Parameters:**p - the specified Point2D to be tested **Returns:**true if the specified Point2D is inside the boundary of the Shape; false otherwise.**Since:** 1.2

### intersects

public boolean **intersects**([Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html) r)

Tests if the interior of the Shape intersects the interior of a specified Rectangle2D. The Shape.intersects() method allows a Shape implementation to conservatively return true when:

* there is a high probability that the Rectangle2D and the Shape intersect, but
* the calculations to accurately determine this intersection are prohibitively expensive.

This means that for some Shapes this method might return true even though the Rectangle2D does not intersect the Shape. The [Area](http://docs.google.com/java/awt/geom/Area.html) class performs more accurate computations of geometric intersection than most Shape objects and therefore can be used if a more precise answer is required.

**Specified by:**[intersects](http://docs.google.com/java/awt/Shape.html#intersects(java.awt.geom.Rectangle2D)) in interface [Shape](http://docs.google.com/java/awt/Shape.html) **Parameters:**r - the specified Rectangle2D **Returns:**true if the interior of the Shape and the interior of the specified Rectangle2D intersect, or are both highly likely to intersect and intersection calculations would be too expensive to perform; false otherwise.**Since:** 1.2 **See Also:**[Shape.intersects(double, double, double, double)](http://docs.google.com/java/awt/Shape.html#intersects(double,%20double,%20double,%20double))

### contains

public boolean **contains**([Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html) r)

Tests if the interior of the Shape entirely contains the specified Rectangle2D. The Shape.contains() method allows a Shape implementation to conservatively return false when:

* the intersect method returns true and
* the calculations to determine whether or not the Shape entirely contains the Rectangle2D are prohibitively expensive.

This means that for some Shapes this method might return false even though the Shape contains the Rectangle2D. The [Area](http://docs.google.com/java/awt/geom/Area.html) class performs more accurate geometric computations than most Shape objects and therefore can be used if a more precise answer is required.

**Specified by:**[contains](http://docs.google.com/java/awt/Shape.html#contains(java.awt.geom.Rectangle2D)) in interface [Shape](http://docs.google.com/java/awt/Shape.html) **Parameters:**r - The specified Rectangle2D **Returns:**true if the interior of the Shape entirely contains the Rectangle2D; false otherwise or, if the Shape contains the Rectangle2D and the intersects method returns true and the containment calculations would be too expensive to perform.**Since:** 1.2 **See Also:**[Shape.contains(double, double, double, double)](http://docs.google.com/java/awt/Shape.html#contains(double,%20double,%20double,%20double))

### getBounds

public [Rectangle](http://docs.google.com/java/awt/Rectangle.html) **getBounds**()

Returns an integer [Rectangle](http://docs.google.com/java/awt/Rectangle.html) that completely encloses the Shape. Note that there is no guarantee that the returned Rectangle is the smallest bounding box that encloses the Shape, only that the Shape lies entirely within the indicated Rectangle. The returned Rectangle might also fail to completely enclose the Shape if the Shape overflows the limited range of the integer data type. The getBounds2D method generally returns a tighter bounding box due to its greater flexibility in representation.

**Specified by:**[getBounds](http://docs.google.com/java/awt/Shape.html#getBounds()) in interface [Shape](http://docs.google.com/java/awt/Shape.html) **Returns:**an integer Rectangle that completely encloses the Shape.**Since:** 1.2 **See Also:**[Shape.getBounds2D()](http://docs.google.com/java/awt/Shape.html#getBounds2D())

### getPathIterator

public [PathIterator](http://docs.google.com/java/awt/geom/PathIterator.html) **getPathIterator**([AffineTransform](http://docs.google.com/java/awt/geom/AffineTransform.html) at,  
 double flatness)

Returns an iterator object that iterates along the Shape object's boundary and provides access to a flattened view of the outline of the Shape object's geometry.

Only SEG\_MOVETO, SEG\_LINETO, and SEG\_CLOSE point types will be returned by the iterator.

The amount of subdivision of the curved segments is controlled by the flatness parameter, which specifies the maximum distance that any point on the unflattened transformed curve can deviate from the returned flattened path segments. An optional [AffineTransform](http://docs.google.com/java/awt/geom/AffineTransform.html) can be specified so that the coordinates returned in the iteration are transformed accordingly.

**Specified by:**[getPathIterator](http://docs.google.com/java/awt/Shape.html#getPathIterator(java.awt.geom.AffineTransform,%20double)) in interface [Shape](http://docs.google.com/java/awt/Shape.html) **Parameters:**at - an optional AffineTransform to be applied to the coordinates as they are returned in the iteration, or null if untransformed coordinates are desired.flatness - the maximum distance that the line segments used to approximate the curved segments are allowed to deviate from any point on the original curve **Returns:**a PathIterator object that provides access to the Shape object's flattened geometry.**Since:** 1.2

### clone

public [Object](http://docs.google.com/java/lang/Object.html) **clone**()

Creates a new object of the same class and with the same contents as this object.

**Overrides:**[clone](http://docs.google.com/java/lang/Object.html#clone()) in class [Object](http://docs.google.com/java/lang/Object.html) **Returns:**a clone of this instance. **Throws:** [OutOfMemoryError](http://docs.google.com/java/lang/OutOfMemoryError.html) - if there is not enough memory.**Since:** 1.2 **See Also:**[Cloneable](http://docs.google.com/java/lang/Cloneable.html)

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/RectangularShape.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV CLASS**](http://docs.google.com/java/awt/geom/Rectangle2D.Float.html)   [**NEXT CLASS**](http://docs.google.com/java/awt/geom/RoundRectangle2D.html) | [**FRAMES**](http://docs.google.com/index.html?java/awt/geom/RectangularShape.html)    [**NO FRAMES**](http://docs.google.com/RectangularShape.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#3znysh7) | [METHOD](#2et92p0) | DETAIL: FIELD | [CONSTR](#1t3h5sf) | [METHOD](#2s8eyo1) |

[Submit a bug or feature](http://bugs.sun.com/services/bugreport/index.jsp)

For further API reference and developer documentation, see [Java SE Developer Documentation](http://docs.google.com/webnotes/devdocs-vs-specs.html). That documentation contains more detailed, developer-targeted descriptions, with conceptual overviews, definitions of terms, workarounds, and working code examples.

Copyright 2006 Sun Microsystems, Inc. All rights reserved. Use is subject to [license terms](http://docs.google.com/legal/license.html). Also see the [documentation redistribution policy](http://java.sun.com/docs/redist.html).