| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/Rectangle2D.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/QuadCurve2D.Float.html)   [**NEXT CLASS**](http://docs.google.com/java/awt/geom/Rectangle2D.Double.html) | [**FRAMES**](http://docs.google.com/index.html?java/awt/geom/Rectangle2D.html)    [**NO FRAMES**](http://docs.google.com/Rectangle2D.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: [NESTED](#3znysh7) | [FIELD](#2et92p0) | [CONSTR](#tyjcwt) | [METHOD](#3dy6vkm) | DETAIL: [FIELD](#2s8eyo1) | [CONSTR](#35nkun2) | [METHOD](#44sinio) |

## **java.awt.geom**

Class Rectangle2D

[java.lang.Object](http://docs.google.com/java/lang/Object.html)  
 [java.awt.geom.RectangularShape](http://docs.google.com/java/awt/geom/RectangularShape.html)  
 **java.awt.geom.Rectangle2D**

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

public abstract class **Rectangle2D**extends [RectangularShape](http://docs.google.com/java/awt/geom/RectangularShape.html)

The Rectangle2D class describes a rectangle defined by a location (x,y) and dimension (w x h).

This class is only the abstract superclass for all objects that store a 2D rectangle. The actual storage representation of the coordinates is left to the subclass.

**Since:** 1.2

| **Nested Class Summary** | |
| --- | --- |
| static class | [**Rectangle2D.Double**](http://docs.google.com/java/awt/geom/Rectangle2D.Double.html)            The Double class defines a rectangle specified in double coordinates. |
| static class | [**Rectangle2D.Float**](http://docs.google.com/java/awt/geom/Rectangle2D.Float.html)            The Float class defines a rectangle specified in float coordinates. |

| **Field Summary** | |
| --- | --- |
| static int | [**OUT\_BOTTOM**](http://docs.google.com/java/awt/geom/Rectangle2D.html#OUT_BOTTOM)            The bitmask that indicates that a point lies below this Rectangle2D. |
| static int | [**OUT\_LEFT**](http://docs.google.com/java/awt/geom/Rectangle2D.html#OUT_LEFT)            The bitmask that indicates that a point lies to the left of this Rectangle2D. |
| static int | [**OUT\_RIGHT**](http://docs.google.com/java/awt/geom/Rectangle2D.html#OUT_RIGHT)            The bitmask that indicates that a point lies to the right of this Rectangle2D. |
| static int | [**OUT\_TOP**](http://docs.google.com/java/awt/geom/Rectangle2D.html#OUT_TOP)            The bitmask that indicates that a point lies above this Rectangle2D. |

| **Constructor Summary** | |
| --- | --- |
| protected | [**Rectangle2D**](http://docs.google.com/java/awt/geom/Rectangle2D.html#Rectangle2D())()            This is an abstract class that cannot be instantiated directly. |

| **Method Summary** | |
| --- | --- |
| void | [**add**](http://docs.google.com/java/awt/geom/Rectangle2D.html#add(double,%20double))(double newx, double newy)            Adds a point, specified by the double precision arguments newx and newy, to this Rectangle2D. |
| void | [**add**](http://docs.google.com/java/awt/geom/Rectangle2D.html#add(java.awt.geom.Point2D))([Point2D](http://docs.google.com/java/awt/geom/Point2D.html) pt)            Adds the Point2D object pt to this Rectangle2D. |
| void | [**add**](http://docs.google.com/java/awt/geom/Rectangle2D.html#add(java.awt.geom.Rectangle2D))([Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html) r)            Adds a Rectangle2D object to this Rectangle2D. |
| boolean | [**contains**](http://docs.google.com/java/awt/geom/Rectangle2D.html#contains(double,%20double))(double x, double y)            Tests if the specified coordinates are inside the boundary of the Shape. |
| boolean | [**contains**](http://docs.google.com/java/awt/geom/Rectangle2D.html#contains(double,%20double,%20double,%20double))(double x, double y, double w, double h)            Tests if the interior of the Shape entirely contains the specified rectangular area. |
| abstract  [Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html) | [**createIntersection**](http://docs.google.com/java/awt/geom/Rectangle2D.html#createIntersection(java.awt.geom.Rectangle2D))([Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html) r)            Returns a new Rectangle2D object representing the intersection of this Rectangle2D with the specified Rectangle2D. |
| abstract  [Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html) | [**createUnion**](http://docs.google.com/java/awt/geom/Rectangle2D.html#createUnion(java.awt.geom.Rectangle2D))([Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html) r)            Returns a new Rectangle2D object representing the union of this Rectangle2D with the specified Rectangle2D. |
| boolean | [**equals**](http://docs.google.com/java/awt/geom/Rectangle2D.html#equals(java.lang.Object))([Object](http://docs.google.com/java/lang/Object.html) obj)            Determines whether or not the specified Object is equal to this Rectangle2D. |
| [Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html) | [**getBounds2D**](http://docs.google.com/java/awt/geom/Rectangle2D.html#getBounds2D())()            Returns a high precision and more accurate bounding box of the Shape than the getBounds method. |
| [PathIterator](http://docs.google.com/java/awt/geom/PathIterator.html) | [**getPathIterator**](http://docs.google.com/java/awt/geom/Rectangle2D.html#getPathIterator(java.awt.geom.AffineTransform))([AffineTransform](http://docs.google.com/java/awt/geom/AffineTransform.html) at)            Returns an iteration object that defines the boundary of this Rectangle2D. |
| [PathIterator](http://docs.google.com/java/awt/geom/PathIterator.html) | [**getPathIterator**](http://docs.google.com/java/awt/geom/Rectangle2D.html#getPathIterator(java.awt.geom.AffineTransform,%20double))([AffineTransform](http://docs.google.com/java/awt/geom/AffineTransform.html) at, double flatness)            Returns an iteration object that defines the boundary of the flattened Rectangle2D. |
| int | [**hashCode**](http://docs.google.com/java/awt/geom/Rectangle2D.html#hashCode())()            Returns the hashcode for this Rectangle2D. |
| static void | [**intersect**](http://docs.google.com/java/awt/geom/Rectangle2D.html#intersect(java.awt.geom.Rectangle2D,%20java.awt.geom.Rectangle2D,%20java.awt.geom.Rectangle2D))([Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html) src1, [Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html) src2, [Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html) dest)            Intersects the pair of specified source Rectangle2D objects and puts the result into the specified destination Rectangle2D object. |
| boolean | [**intersects**](http://docs.google.com/java/awt/geom/Rectangle2D.html#intersects(double,%20double,%20double,%20double))(double x, double y, double w, double h)            Tests if the interior of the Shape intersects the interior of a specified rectangular area. |
| boolean | [**intersectsLine**](http://docs.google.com/java/awt/geom/Rectangle2D.html#intersectsLine(double,%20double,%20double,%20double))(double x1, double y1, double x2, double y2)            Tests if the specified line segment intersects the interior of this Rectangle2D. |
| boolean | [**intersectsLine**](http://docs.google.com/java/awt/geom/Rectangle2D.html#intersectsLine(java.awt.geom.Line2D))([Line2D](http://docs.google.com/java/awt/geom/Line2D.html) l)            Tests if the specified line segment intersects the interior of this Rectangle2D. |
| abstract  int | [**outcode**](http://docs.google.com/java/awt/geom/Rectangle2D.html#outcode(double,%20double))(double x, double y)            Determines where the specified coordinates lie with respect to this Rectangle2D. |
| int | [**outcode**](http://docs.google.com/java/awt/geom/Rectangle2D.html#outcode(java.awt.geom.Point2D))([Point2D](http://docs.google.com/java/awt/geom/Point2D.html) p)            Determines where the specified [Point2D](http://docs.google.com/java/awt/geom/Point2D.html) lies with respect to this Rectangle2D. |
| void | [**setFrame**](http://docs.google.com/java/awt/geom/Rectangle2D.html#setFrame(double,%20double,%20double,%20double))(double x, double y, double w, double h)            Sets the location and size of the outer bounds of this Rectangle2D to the specified rectangular values. |
| abstract  void | [**setRect**](http://docs.google.com/java/awt/geom/Rectangle2D.html#setRect(double,%20double,%20double,%20double))(double x, double y, double w, double h)            Sets the location and size of this Rectangle2D to the specified double values. |
| void | [**setRect**](http://docs.google.com/java/awt/geom/Rectangle2D.html#setRect(java.awt.geom.Rectangle2D))([Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html) r)            Sets this Rectangle2D to be the same as the specified Rectangle2D. |
| static void | [**union**](http://docs.google.com/java/awt/geom/Rectangle2D.html#union(java.awt.geom.Rectangle2D,%20java.awt.geom.Rectangle2D,%20java.awt.geom.Rectangle2D))([Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html) src1, [Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html) src2, [Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html) dest)            Unions the pair of source Rectangle2D objects and puts the result into the specified destination Rectangle2D object. |

| **Methods inherited from class java.awt.geom.**[**RectangularShape**](http://docs.google.com/java/awt/geom/RectangularShape.html) |
| --- |
| [clone](http://docs.google.com/java/awt/geom/RectangularShape.html#clone()), [contains](http://docs.google.com/java/awt/geom/RectangularShape.html#contains(java.awt.geom.Point2D)), [contains](http://docs.google.com/java/awt/geom/RectangularShape.html#contains(java.awt.geom.Rectangle2D)), [getBounds](http://docs.google.com/java/awt/geom/RectangularShape.html#getBounds()), [getCenterX](http://docs.google.com/java/awt/geom/RectangularShape.html#getCenterX()), [getCenterY](http://docs.google.com/java/awt/geom/RectangularShape.html#getCenterY()), [getFrame](http://docs.google.com/java/awt/geom/RectangularShape.html#getFrame()), [getHeight](http://docs.google.com/java/awt/geom/RectangularShape.html#getHeight()), [getMaxX](http://docs.google.com/java/awt/geom/RectangularShape.html#getMaxX()), [getMaxY](http://docs.google.com/java/awt/geom/RectangularShape.html#getMaxY()), [getMinX](http://docs.google.com/java/awt/geom/RectangularShape.html#getMinX()), [getMinY](http://docs.google.com/java/awt/geom/RectangularShape.html#getMinY()), [getWidth](http://docs.google.com/java/awt/geom/RectangularShape.html#getWidth()), [getX](http://docs.google.com/java/awt/geom/RectangularShape.html#getX()), [getY](http://docs.google.com/java/awt/geom/RectangularShape.html#getY()), [intersects](http://docs.google.com/java/awt/geom/RectangularShape.html#intersects(java.awt.geom.Rectangle2D)), [isEmpty](http://docs.google.com/java/awt/geom/RectangularShape.html#isEmpty()), [setFrame](http://docs.google.com/java/awt/geom/RectangularShape.html#setFrame(java.awt.geom.Point2D,%20java.awt.geom.Dimension2D)), [setFrame](http://docs.google.com/java/awt/geom/RectangularShape.html#setFrame(java.awt.geom.Rectangle2D)), [setFrameFromCenter](http://docs.google.com/java/awt/geom/RectangularShape.html#setFrameFromCenter(double,%20double,%20double,%20double)), [setFrameFromCenter](http://docs.google.com/java/awt/geom/RectangularShape.html#setFrameFromCenter(java.awt.geom.Point2D,%20java.awt.geom.Point2D)), [setFrameFromDiagonal](http://docs.google.com/java/awt/geom/RectangularShape.html#setFrameFromDiagonal(double,%20double,%20double,%20double)), [setFrameFromDiagonal](http://docs.google.com/java/awt/geom/RectangularShape.html#setFrameFromDiagonal(java.awt.geom.Point2D,%20java.awt.geom.Point2D)) |

| **Methods inherited from class java.lang.**[**Object**](http://docs.google.com/java/lang/Object.html) |
| --- |
| [finalize](http://docs.google.com/java/lang/Object.html#finalize()), [getClass](http://docs.google.com/java/lang/Object.html#getClass()), [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)) |

| **Field Detail** |
| --- |

### OUT\_LEFT

public static final int **OUT\_LEFT**

The bitmask that indicates that a point lies to the left of this Rectangle2D.

**Since:** 1.2 **See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.awt.geom.Rectangle2D.OUT_LEFT)

### OUT\_TOP

public static final int **OUT\_TOP**

The bitmask that indicates that a point lies above this Rectangle2D.

**Since:** 1.2 **See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.awt.geom.Rectangle2D.OUT_TOP)

### OUT\_RIGHT

public static final int **OUT\_RIGHT**

The bitmask that indicates that a point lies to the right of this Rectangle2D.

**Since:** 1.2 **See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.awt.geom.Rectangle2D.OUT_RIGHT)

### OUT\_BOTTOM

public static final int **OUT\_BOTTOM**

The bitmask that indicates that a point lies below this Rectangle2D.

**Since:** 1.2 **See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.awt.geom.Rectangle2D.OUT_BOTTOM)

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

### Rectangle2D

protected **Rectangle2D**()

This is an abstract class that cannot be instantiated directly. Type-specific implementation subclasses are available for instantiation and provide a number of formats for storing the information necessary to satisfy the various accessor methods below.

**Since:** 1.2 **See Also:**[Rectangle2D.Float](http://docs.google.com/java/awt/geom/Rectangle2D.Float.html), [Rectangle2D.Double](http://docs.google.com/java/awt/geom/Rectangle2D.Double.html), [Rectangle](http://docs.google.com/java/awt/Rectangle.html)

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

### setRect

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

Sets the location and size of this Rectangle2D to the specified double values.

**Parameters:**x - the X coordinate of the upper-left corner of this Rectangle2Dy - the Y coordinate of the upper-left corner of this Rectangle2Dw - the width of this Rectangle2Dh - the height of this Rectangle2D**Since:** 1.2

### setRect

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

Sets this Rectangle2D to be the same as the specified Rectangle2D.

**Parameters:**r - the specified Rectangle2D**Since:** 1.2

### intersectsLine

public boolean **intersectsLine**(double x1,  
 double y1,  
 double x2,  
 double y2)

Tests if the specified line segment intersects the interior of this Rectangle2D.

**Parameters:**x1 - the X coordinate of the start point of the specified line segmenty1 - the Y coordinate of the start point of the specified line segmentx2 - the X coordinate of the end point of the specified line segmenty2 - the Y coordinate of the end point of the specified line segment **Returns:**true if the specified line segment intersects the interior of this Rectangle2D; false otherwise.**Since:** 1.2

### intersectsLine

public boolean **intersectsLine**([Line2D](http://docs.google.com/java/awt/geom/Line2D.html) l)

Tests if the specified line segment intersects the interior of this Rectangle2D.

**Parameters:**l - the specified [Line2D](http://docs.google.com/java/awt/geom/Line2D.html) to test for intersection with the interior of this Rectangle2D **Returns:**true if the specified Line2D intersects the interior of this Rectangle2D; false otherwise.**Since:** 1.2

### outcode

public abstract int **outcode**(double x,  
 double y)

Determines where the specified coordinates lie with respect to this Rectangle2D. This method computes a binary OR of the appropriate mask values indicating, for each side of this Rectangle2D, whether or not the specified coordinates are on the same side of the edge as the rest of this Rectangle2D.

**Parameters:**x - the specified X coordinatey - the specified Y coordinate **Returns:**the logical OR of all appropriate out codes.**Since:** 1.2 **See Also:**[OUT\_LEFT](http://docs.google.com/java/awt/geom/Rectangle2D.html#OUT_LEFT), [OUT\_TOP](http://docs.google.com/java/awt/geom/Rectangle2D.html#OUT_TOP), [OUT\_RIGHT](http://docs.google.com/java/awt/geom/Rectangle2D.html#OUT_RIGHT), [OUT\_BOTTOM](http://docs.google.com/java/awt/geom/Rectangle2D.html#OUT_BOTTOM)

### outcode

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

Determines where the specified [Point2D](http://docs.google.com/java/awt/geom/Point2D.html) lies with respect to this Rectangle2D. This method computes a binary OR of the appropriate mask values indicating, for each side of this Rectangle2D, whether or not the specified Point2D is on the same side of the edge as the rest of this Rectangle2D.

**Parameters:**p - the specified Point2D **Returns:**the logical OR of all appropriate out codes.**Since:** 1.2 **See Also:**[OUT\_LEFT](http://docs.google.com/java/awt/geom/Rectangle2D.html#OUT_LEFT), [OUT\_TOP](http://docs.google.com/java/awt/geom/Rectangle2D.html#OUT_TOP), [OUT\_RIGHT](http://docs.google.com/java/awt/geom/Rectangle2D.html#OUT_RIGHT), [OUT\_BOTTOM](http://docs.google.com/java/awt/geom/Rectangle2D.html#OUT_BOTTOM)

### setFrame

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

Sets the location and size of the outer bounds of this Rectangle2D to the specified rectangular values.

**Specified by:**[setFrame](http://docs.google.com/java/awt/geom/RectangularShape.html#setFrame(double,%20double,%20double,%20double)) in class [RectangularShape](http://docs.google.com/java/awt/geom/RectangularShape.html) **Parameters:**x - the X coordinate of the upper-left corner of this Rectangle2Dy - the Y coordinate of the upper-left corner of this Rectangle2Dw - the width of this Rectangle2Dh - the height of this Rectangle2D**Since:** 1.2 **See Also:**[RectangularShape.getFrame()](http://docs.google.com/java/awt/geom/RectangularShape.html#getFrame())

### getBounds2D

public [Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html) **getBounds2D**()

Returns a high precision and more accurate bounding box of the Shape than the getBounds method. Note that there is no guarantee that the returned [Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html) is the smallest bounding box that encloses the Shape, only that the Shape lies entirely within the indicated Rectangle2D. The bounding box returned by this method is usually tighter than that returned by the getBounds method and never fails due to overflow problems since the return value can be an instance of the Rectangle2D that uses double precision values to store the dimensions.

**Returns:**an instance of Rectangle2D that is a high-precision bounding box of the Shape.**Since:** 1.2 **See Also:**[Shape.getBounds()](http://docs.google.com/java/awt/Shape.html#getBounds())

### contains

public boolean **contains**(double x,  
 double y)

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

**Parameters:**x - the specified X coordinate to be testedy - the specified Y coordinate to be tested **Returns:**true if the specified coordinates are inside the Shape boundary; false otherwise.**Since:** 1.2

### intersects

public boolean **intersects**(double x,  
 double y,  
 double w,  
 double h)

Tests if the interior of the Shape intersects the interior of a specified rectangular area. The rectangular area is considered to intersect the Shape if any point is contained in both the interior of the Shape and the specified rectangular area.

The Shape.intersects() method allows a Shape implementation to conservatively return true when:

* there is a high probability that the rectangular area 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 rectangular area 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.

**Parameters:**x - the X coordinate of the upper-left corner of the specified rectangular areay - the Y coordinate of the upper-left corner of the specified rectangular areaw - the width of the specified rectangular areah - the height of the specified rectangular area **Returns:**true if the interior of the Shape and the interior of the rectangular area intersect, or are both highly likely to intersect and intersection calculations would be too expensive to perform; false otherwise.**Since:** 1.2 **See Also:**[Area](http://docs.google.com/java/awt/geom/Area.html)

### contains

public boolean **contains**(double x,  
 double y,  
 double w,  
 double h)

Tests if the interior of the Shape entirely contains the specified rectangular area. All coordinates that lie inside the rectangular area must lie within the Shape for the entire rectanglar area to be considered contained within the Shape.

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 rectangular area are prohibitively expensive.

This means that for some Shapes this method might return false even though the Shape contains the rectangular area. 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.

**Parameters:**x - the X coordinate of the upper-left corner of the specified rectangular areay - the Y coordinate of the upper-left corner of the specified rectangular areaw - the width of the specified rectangular areah - the height of the specified rectangular area **Returns:**true if the interior of the Shape entirely contains the specified rectangular area; false otherwise or, if the Shape contains the rectangular area and the intersects method returns true and the containment calculations would be too expensive to perform.**Since:** 1.2 **See Also:**[Area](http://docs.google.com/java/awt/geom/Area.html), [Shape.intersects(double, double, double, double)](http://docs.google.com/java/awt/Shape.html#intersects(double,%20double,%20double,%20double))

### createIntersection

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

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

**Parameters:**r - the Rectangle2D to be intersected with this Rectangle2D **Returns:**the largest Rectangle2D contained in both the specified Rectangle2D and in this Rectangle2D.**Since:** 1.2

### intersect

public static void **intersect**([Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html) src1,  
 [Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html) src2,  
 [Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html) dest)

Intersects the pair of specified source Rectangle2D objects and puts the result into the specified destination Rectangle2D object. One of the source rectangles can also be the destination to avoid creating a third Rectangle2D object, but in this case the original points of this source rectangle will be overwritten by this method.

**Parameters:**src1 - the first of a pair of Rectangle2D objects to be intersected with each othersrc2 - the second of a pair of Rectangle2D objects to be intersected with each otherdest - the Rectangle2D that holds the results of the intersection of src1 and src2**Since:** 1.2

### createUnion

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

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

**Parameters:**r - the Rectangle2D to be combined with this Rectangle2D **Returns:**the smallest Rectangle2D containing both the specified Rectangle2D and this Rectangle2D.**Since:** 1.2

### union

public static void **union**([Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html) src1,  
 [Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html) src2,  
 [Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html) dest)

Unions the pair of source Rectangle2D objects and puts the result into the specified destination Rectangle2D object. One of the source rectangles can also be the destination to avoid creating a third Rectangle2D object, but in this case the original points of this source rectangle will be overwritten by this method.

**Parameters:**src1 - the first of a pair of Rectangle2D objects to be combined with each othersrc2 - the second of a pair of Rectangle2D objects to be combined with each otherdest - the Rectangle2D that holds the results of the union of src1 and src2**Since:** 1.2

### add

public void **add**(double newx,  
 double newy)

Adds a point, specified by the double precision arguments newx and newy, to this Rectangle2D. The resulting Rectangle2D is the smallest Rectangle2D that contains both the original Rectangle2D and the specified point.

After adding a point, a call to contains with the added point as an argument does not necessarily return true. The contains method does not return true for points on the right or bottom edges of a rectangle. Therefore, if the added point falls on the left or bottom edge of the enlarged rectangle, contains returns false for that point.

**Parameters:**newx - the X coordinate of the new pointnewy - the Y coordinate of the new point**Since:** 1.2

### add

public void **add**([Point2D](http://docs.google.com/java/awt/geom/Point2D.html) pt)

Adds the Point2D object pt to this Rectangle2D. The resulting Rectangle2D is the smallest Rectangle2D that contains both the original Rectangle2D and the specified Point2D.

After adding a point, a call to contains with the added point as an argument does not necessarily return true. The contains method does not return true for points on the right or bottom edges of a rectangle. Therefore, if the added point falls on the left or bottom edge of the enlarged rectangle, contains returns false for that point.

**Parameters:**pt - the new Point2D to add to this Rectangle2D.**Since:** 1.2

### add

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

Adds a Rectangle2D object to this Rectangle2D. The resulting Rectangle2D is the union of the two Rectangle2D objects.

**Parameters:**r - the Rectangle2D to add to this Rectangle2D.**Since:** 1.2

### getPathIterator

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

Returns an iteration object that defines the boundary of this Rectangle2D. The iterator for this class is multi-threaded safe, which means that this Rectangle2D class guarantees that modifications to the geometry of this Rectangle2D object do not affect any iterations of that geometry that are already in process.

**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 **Returns:**the PathIterator object that returns the geometry of the outline of this Rectangle2D, one segment at a time.**Since:** 1.2

### 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 iteration object that defines the boundary of the flattened Rectangle2D. Since rectangles are already flat, the flatness parameter is ignored. The iterator for this class is multi-threaded safe, which means that this Rectangle2D class guarantees that modifications to the geometry of this Rectangle2D object do not affect any iterations of that geometry that are already in process.

**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)**Overrides:**[getPathIterator](http://docs.google.com/java/awt/geom/RectangularShape.html#getPathIterator(java.awt.geom.AffineTransform,%20double)) in class [RectangularShape](http://docs.google.com/java/awt/geom/RectangularShape.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 desiredflatness - the maximum distance that the line segments used to approximate the curved segments are allowed to deviate from any point on the original curve. Since rectangles are already flat, the flatness parameter is ignored. **Returns:**the PathIterator object that returns the geometry of the outline of this Rectangle2D, one segment at a time.**Since:** 1.2

### hashCode

public int **hashCode**()

Returns the hashcode for this Rectangle2D.

**Overrides:**[hashCode](http://docs.google.com/java/lang/Object.html#hashCode()) in class [Object](http://docs.google.com/java/lang/Object.html) **Returns:**the hashcode for this Rectangle2D.**Since:** 1.2 **See Also:**[Object.equals(java.lang.Object)](http://docs.google.com/java/lang/Object.html#equals(java.lang.Object)), [Hashtable](http://docs.google.com/java/util/Hashtable.html)

### equals

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

Determines whether or not the specified Object is equal to this Rectangle2D. The specified Object is equal to this Rectangle2D if it is an instance of Rectangle2D and if its location and size are the same as this Rectangle2D.

**Overrides:**[equals](http://docs.google.com/java/lang/Object.html#equals(java.lang.Object)) in class [Object](http://docs.google.com/java/lang/Object.html) **Parameters:**obj - an Object to be compared with this Rectangle2D. **Returns:**true if obj is an instance of Rectangle2D and has the same values; false otherwise.**Since:** 1.2 **See Also:**[Object.hashCode()](http://docs.google.com/java/lang/Object.html#hashCode()), [Hashtable](http://docs.google.com/java/util/Hashtable.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/Rectangle2D.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/QuadCurve2D.Float.html)   [**NEXT CLASS**](http://docs.google.com/java/awt/geom/Rectangle2D.Double.html) | [**FRAMES**](http://docs.google.com/index.html?java/awt/geom/Rectangle2D.html)    [**NO FRAMES**](http://docs.google.com/Rectangle2D.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: [NESTED](#3znysh7) | [FIELD](#2et92p0) | [CONSTR](#tyjcwt) | [METHOD](#3dy6vkm) | DETAIL: [FIELD](#2s8eyo1) | [CONSTR](#35nkun2) | [METHOD](#44sinio) |

[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).