| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/PathIterator.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/Path2D.Float.html)   [**NEXT CLASS**](http://docs.google.com/java/awt/geom/Point2D.html) | [**FRAMES**](http://docs.google.com/index.html?java/awt/geom/PathIterator.html)    [**NO FRAMES**](http://docs.google.com/PathIterator.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#3znysh7) | CONSTR | [METHOD](#2et92p0) | DETAIL: [FIELD](#tyjcwt) | CONSTR | [METHOD](#lnxbz9) |

## **java.awt.geom**

Interface PathIterator

**All Known Implementing Classes:** [FlatteningPathIterator](http://docs.google.com/java/awt/geom/FlatteningPathIterator.html)

public interface **PathIterator**

The PathIterator interface provides the mechanism for objects that implement the [Shape](http://docs.google.com/java/awt/Shape.html) interface to return the geometry of their boundary by allowing a caller to retrieve the path of that boundary a segment at a time. This interface allows these objects to retrieve the path of their boundary a segment at a time by using 1st through 3rd order Bézier curves, which are lines and quadratic or cubic Bézier splines.

Multiple subpaths can be expressed by using a "MOVETO" segment to create a discontinuity in the geometry to move from the end of one subpath to the beginning of the next.

Each subpath can be closed manually by ending the last segment in the subpath on the same coordinate as the beginning "MOVETO" segment for that subpath or by using a "CLOSE" segment to append a line segment from the last point back to the first. Be aware that manually closing an outline as opposed to using a "CLOSE" segment to close the path might result in different line style decorations being used at the end points of the subpath. For example, the [BasicStroke](http://docs.google.com/java/awt/BasicStroke.html) object uses a line "JOIN" decoration to connect the first and last points if a "CLOSE" segment is encountered, whereas simply ending the path on the same coordinate as the beginning coordinate results in line "CAP" decorations being used at the ends.

**See Also:**[Shape](http://docs.google.com/java/awt/Shape.html), [BasicStroke](http://docs.google.com/java/awt/BasicStroke.html)

| **Field Summary** | |
| --- | --- |
| static int | [**SEG\_CLOSE**](http://docs.google.com/java/awt/geom/PathIterator.html#SEG_CLOSE)            The segment type constant that specifies that the preceding subpath should be closed by appending a line segment back to the point corresponding to the most recent SEG\_MOVETO. |
| static int | [**SEG\_CUBICTO**](http://docs.google.com/java/awt/geom/PathIterator.html#SEG_CUBICTO)            The segment type constant for the set of 3 points that specify a cubic parametric curve to be drawn from the most recently specified point. |
| static int | [**SEG\_LINETO**](http://docs.google.com/java/awt/geom/PathIterator.html#SEG_LINETO)            The segment type constant for a point that specifies the end point of a line to be drawn from the most recently specified point. |
| static int | [**SEG\_MOVETO**](http://docs.google.com/java/awt/geom/PathIterator.html#SEG_MOVETO)            The segment type constant for a point that specifies the starting location for a new subpath. |
| static int | [**SEG\_QUADTO**](http://docs.google.com/java/awt/geom/PathIterator.html#SEG_QUADTO)            The segment type constant for the pair of points that specify a quadratic parametric curve to be drawn from the most recently specified point. |
| static int | [**WIND\_EVEN\_ODD**](http://docs.google.com/java/awt/geom/PathIterator.html#WIND_EVEN_ODD)            The winding rule constant for specifying an even-odd rule for determining the interior of a path. |
| static int | [**WIND\_NON\_ZERO**](http://docs.google.com/java/awt/geom/PathIterator.html#WIND_NON_ZERO)            The winding rule constant for specifying a non-zero rule for determining the interior of a path. |

| **Method Summary** | |
| --- | --- |
| int | [**currentSegment**](http://docs.google.com/java/awt/geom/PathIterator.html#currentSegment(double%5B%5D))(double[] coords)            Returns the coordinates and type of the current path segment in the iteration. |
| int | [**currentSegment**](http://docs.google.com/java/awt/geom/PathIterator.html#currentSegment(float%5B%5D))(float[] coords)            Returns the coordinates and type of the current path segment in the iteration. |
| int | [**getWindingRule**](http://docs.google.com/java/awt/geom/PathIterator.html#getWindingRule())()            Returns the winding rule for determining the interior of the path. |
| boolean | [**isDone**](http://docs.google.com/java/awt/geom/PathIterator.html#isDone())()            Tests if the iteration is complete. |
| void | [**next**](http://docs.google.com/java/awt/geom/PathIterator.html#next())()            Moves the iterator to the next segment of the path forwards along the primary direction of traversal as long as there are more points in that direction. |

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

### WIND\_EVEN\_ODD

static final int **WIND\_EVEN\_ODD**

The winding rule constant for specifying an even-odd rule for determining the interior of a path. The even-odd rule specifies that a point lies inside the path if a ray drawn in any direction from that point to infinity is crossed by path segments an odd number of times.

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

### WIND\_NON\_ZERO

static final int **WIND\_NON\_ZERO**

The winding rule constant for specifying a non-zero rule for determining the interior of a path. The non-zero rule specifies that a point lies inside the path if a ray drawn in any direction from that point to infinity is crossed by path segments a different number of times in the counter-clockwise direction than the clockwise direction.

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

### SEG\_MOVETO

static final int **SEG\_MOVETO**

The segment type constant for a point that specifies the starting location for a new subpath.

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

### SEG\_LINETO

static final int **SEG\_LINETO**

The segment type constant for a point that specifies the end point of a line to be drawn from the most recently specified point.

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

### SEG\_QUADTO

static final int **SEG\_QUADTO**

The segment type constant for the pair of points that specify a quadratic parametric curve to be drawn from the most recently specified point. The curve is interpolated by solving the parametric control equation in the range (t=[0..1]) using the most recently specified (current) point (CP), the first control point (P1), and the final interpolated control point (P2). The parametric control equation for this curve is:

P(t) = B(2,0)\*CP + B(2,1)\*P1 + B(2,2)\*P2  
 0 <= t <= 1  
  
 B(n,m) = mth coefficient of nth degree Bernstein polynomial  
 = C(n,m) \* t^(m) \* (1 - t)^(n-m)  
 C(n,m) = Combinations of n things, taken m at a time  
 = n! / (m! \* (n-m)!)

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

### SEG\_CUBICTO

static final int **SEG\_CUBICTO**

The segment type constant for the set of 3 points that specify a cubic parametric curve to be drawn from the most recently specified point. The curve is interpolated by solving the parametric control equation in the range (t=[0..1]) using the most recently specified (current) point (CP), the first control point (P1), the second control point (P2), and the final interpolated control point (P3). The parametric control equation for this curve is:

P(t) = B(3,0)\*CP + B(3,1)\*P1 + B(3,2)\*P2 + B(3,3)\*P3  
 0 <= t <= 1  
  
 B(n,m) = mth coefficient of nth degree Bernstein polynomial  
 = C(n,m) \* t^(m) \* (1 - t)^(n-m)  
 C(n,m) = Combinations of n things, taken m at a time  
 = n! / (m! \* (n-m)!)

This form of curve is commonly known as a Bézier curve.

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

### SEG\_CLOSE

static final int **SEG\_CLOSE**

The segment type constant that specifies that the preceding subpath should be closed by appending a line segment back to the point corresponding to the most recent SEG\_MOVETO.

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

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

### getWindingRule

int **getWindingRule**()

Returns the winding rule for determining the interior of the path.

**Returns:**the winding rule.**See Also:**[WIND\_EVEN\_ODD](http://docs.google.com/java/awt/geom/PathIterator.html#WIND_EVEN_ODD), [WIND\_NON\_ZERO](http://docs.google.com/java/awt/geom/PathIterator.html#WIND_NON_ZERO)

### isDone

boolean **isDone**()

Tests if the iteration is complete.

**Returns:**true if all the segments have been read; false otherwise.

### next

void **next**()

Moves the iterator to the next segment of the path forwards along the primary direction of traversal as long as there are more points in that direction.

### currentSegment

int **currentSegment**(float[] coords)

Returns the coordinates and type of the current path segment in the iteration. The return value is the path-segment type: SEG\_MOVETO, SEG\_LINETO, SEG\_QUADTO, SEG\_CUBICTO, or SEG\_CLOSE. A float array of length 6 must be passed in and can be used to store the coordinates of the point(s). Each point is stored as a pair of float x,y coordinates. SEG\_MOVETO and SEG\_LINETO types returns one point, SEG\_QUADTO returns two points, SEG\_CUBICTO returns 3 points and SEG\_CLOSE does not return any points.

**Parameters:**coords - an array that holds the data returned from this method **Returns:**the path-segment type of the current path segment.**See Also:**[SEG\_MOVETO](http://docs.google.com/java/awt/geom/PathIterator.html#SEG_MOVETO), [SEG\_LINETO](http://docs.google.com/java/awt/geom/PathIterator.html#SEG_LINETO), [SEG\_QUADTO](http://docs.google.com/java/awt/geom/PathIterator.html#SEG_QUADTO), [SEG\_CUBICTO](http://docs.google.com/java/awt/geom/PathIterator.html#SEG_CUBICTO), [SEG\_CLOSE](http://docs.google.com/java/awt/geom/PathIterator.html#SEG_CLOSE)

### currentSegment

int **currentSegment**(double[] coords)

Returns the coordinates and type of the current path segment in the iteration. The return value is the path-segment type: SEG\_MOVETO, SEG\_LINETO, SEG\_QUADTO, SEG\_CUBICTO, or SEG\_CLOSE. A double array of length 6 must be passed in and can be used to store the coordinates of the point(s). Each point is stored as a pair of double x,y coordinates. SEG\_MOVETO and SEG\_LINETO types returns one point, SEG\_QUADTO returns two points, SEG\_CUBICTO returns 3 points and SEG\_CLOSE does not return any points.

**Parameters:**coords - an array that holds the data returned from this method **Returns:**the path-segment type of the current path segment.**See Also:**[SEG\_MOVETO](http://docs.google.com/java/awt/geom/PathIterator.html#SEG_MOVETO), [SEG\_LINETO](http://docs.google.com/java/awt/geom/PathIterator.html#SEG_LINETO), [SEG\_QUADTO](http://docs.google.com/java/awt/geom/PathIterator.html#SEG_QUADTO), [SEG\_CUBICTO](http://docs.google.com/java/awt/geom/PathIterator.html#SEG_CUBICTO), [SEG\_CLOSE](http://docs.google.com/java/awt/geom/PathIterator.html#SEG_CLOSE)

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/PathIterator.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/Path2D.Float.html)   [**NEXT CLASS**](http://docs.google.com/java/awt/geom/Point2D.html) | [**FRAMES**](http://docs.google.com/index.html?java/awt/geom/PathIterator.html)    [**NO FRAMES**](http://docs.google.com/PathIterator.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#3znysh7) | CONSTR | [METHOD](#2et92p0) | DETAIL: [FIELD](#tyjcwt) | CONSTR | [METHOD](#lnxbz9) |

[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).