| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/BasicStroke.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/AWTPermission.html)   [**NEXT CLASS**](http://docs.google.com/java/awt/BorderLayout.html) | [**FRAMES**](http://docs.google.com/index.html?java/awt/BasicStroke.html)    [**NO FRAMES**](http://docs.google.com/BasicStroke.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#3znysh7) | [CONSTR](#2et92p0) | [METHOD](#tyjcwt) | DETAIL: [FIELD](#1t3h5sf) | [CONSTR](#35nkun2) | [METHOD](#1y810tw) |

## **java.awt**

Class BasicStroke

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

**All Implemented Interfaces:** [Stroke](http://docs.google.com/java/awt/Stroke.html)

public class **BasicStroke**extends [Object](http://docs.google.com/java/lang/Object.html)implements [Stroke](http://docs.google.com/java/awt/Stroke.html)

The BasicStroke class defines a basic set of rendering attributes for the outlines of graphics primitives, which are rendered with a [Graphics2D](http://docs.google.com/java/awt/Graphics2D.html) object that has its Stroke attribute set to this BasicStroke. The rendering attributes defined by BasicStroke describe the shape of the mark made by a pen drawn along the outline of a [Shape](http://docs.google.com/java/awt/Shape.html) and the decorations applied at the ends and joins of path segments of the Shape. These rendering attributes include:

*width* The pen width, measured perpendicularly to the pen trajectory. *end caps* The decoration applied to the ends of unclosed subpaths and dash segments. Subpaths that start and end on the same point are still considered unclosed if they do not have a CLOSE segment. See [SEG\_CLOSE](http://docs.google.com/java/awt/geom/PathIterator.html#SEG_CLOSE) for more information on the CLOSE segment. The three different decorations are: [CAP\_BUTT](http://docs.google.com/java/awt/BasicStroke.html#CAP_BUTT), [CAP\_ROUND](http://docs.google.com/java/awt/BasicStroke.html#CAP_ROUND), and [CAP\_SQUARE](http://docs.google.com/java/awt/BasicStroke.html#CAP_SQUARE). *line joins* The decoration applied at the intersection of two path segments and at the intersection of the endpoints of a subpath that is closed using [SEG\_CLOSE](http://docs.google.com/java/awt/geom/PathIterator.html#SEG_CLOSE). The three different decorations are: [JOIN\_BEVEL](http://docs.google.com/java/awt/BasicStroke.html#JOIN_BEVEL), [JOIN\_MITER](http://docs.google.com/java/awt/BasicStroke.html#JOIN_MITER), and [JOIN\_ROUND](http://docs.google.com/java/awt/BasicStroke.html#JOIN_ROUND). *miter limit* The limit to trim a line join that has a JOIN\_MITER decoration. A line join is trimmed when the ratio of miter length to stroke width is greater than the miterlimit value. The miter length is the diagonal length of the miter, which is the distance between the inside corner and the outside corner of the intersection. The smaller the angle formed by two line segments, the longer the miter length and the sharper the angle of intersection. The default miterlimit value of 10.0f causes all angles less than 11 degrees to be trimmed. Trimming miters converts the decoration of the line join to bevel. *dash attributes* The definition of how to make a dash pattern by alternating between opaque and transparent sections. All attributes that specify measurements and distances controlling the shape of the returned outline are measured in the same coordinate system as the original unstroked Shape argument. When a Graphics2D object uses a Stroke object to redefine a path during the execution of one of its draw methods, the geometry is supplied in its original form before the Graphics2D transform attribute is applied. Therefore, attributes such as the pen width are interpreted in the user space coordinate system of the Graphics2D object and are subject to the scaling and shearing effects of the user-space-to-device-space transform in that particular Graphics2D. For example, the width of a rendered shape's outline is determined not only by the width attribute of this BasicStroke, but also by the transform attribute of the Graphics2D object. Consider this code:// sets the Graphics2D object's Tranform attribute g2d.scale(10, 10); // sets the Graphics2D object's Stroke attribute g2d.setStroke(new BasicStroke(1.5f));Assuming there are no other scaling transforms added to the Graphics2D object, the resulting line will be approximately 15 pixels wide. As the example code demonstrates, a floating-point line offers better precision, especially when large transforms are used with a Graphics2D object. When a line is diagonal, the exact width depends on how the rendering pipeline chooses which pixels to fill as it traces the theoretical widened outline. The choice of which pixels to turn on is affected by the antialiasing attribute because the antialiasing rendering pipeline can choose to color partially-covered pixels.

For more information on the user space coordinate system and the rendering process, see the Graphics2D class comments.

**See Also:**[Graphics2D](http://docs.google.com/java/awt/Graphics2D.html)

| **Field Summary** | |
| --- | --- |
| static int | [**CAP\_BUTT**](http://docs.google.com/java/awt/BasicStroke.html#CAP_BUTT)            Ends unclosed subpaths and dash segments with no added decoration. |
| static int | [**CAP\_ROUND**](http://docs.google.com/java/awt/BasicStroke.html#CAP_ROUND)            Ends unclosed subpaths and dash segments with a round decoration that has a radius equal to half of the width of the pen. |
| static int | [**CAP\_SQUARE**](http://docs.google.com/java/awt/BasicStroke.html#CAP_SQUARE)            Ends unclosed subpaths and dash segments with a square projection that extends beyond the end of the segment to a distance equal to half of the line width. |
| static int | [**JOIN\_BEVEL**](http://docs.google.com/java/awt/BasicStroke.html#JOIN_BEVEL)            Joins path segments by connecting the outer corners of their wide outlines with a straight segment. |
| static int | [**JOIN\_MITER**](http://docs.google.com/java/awt/BasicStroke.html#JOIN_MITER)            Joins path segments by extending their outside edges until they meet. |
| static int | [**JOIN\_ROUND**](http://docs.google.com/java/awt/BasicStroke.html#JOIN_ROUND)            Joins path segments by rounding off the corner at a radius of half the line width. |

| **Constructor Summary** | |
| --- | --- |
| [**BasicStroke**](http://docs.google.com/java/awt/BasicStroke.html#BasicStroke())()            Constructs a new BasicStroke with defaults for all attributes. |
| [**BasicStroke**](http://docs.google.com/java/awt/BasicStroke.html#BasicStroke(float))(float width)            Constructs a solid BasicStroke with the specified line width and with default values for the cap and join styles. |
| [**BasicStroke**](http://docs.google.com/java/awt/BasicStroke.html#BasicStroke(float,%20int,%20int))(float width, int cap, int join)            Constructs a solid BasicStroke with the specified attributes. |
| [**BasicStroke**](http://docs.google.com/java/awt/BasicStroke.html#BasicStroke(float,%20int,%20int,%20float))(float width, int cap, int join, float miterlimit)            Constructs a solid BasicStroke with the specified attributes. |
| [**BasicStroke**](http://docs.google.com/java/awt/BasicStroke.html#BasicStroke(float,%20int,%20int,%20float,%20float%5B%5D,%20float))(float width, int cap, int join, float miterlimit, float[] dash, float dash\_phase)            Constructs a new BasicStroke with the specified attributes. |

| **Method Summary** | |
| --- | --- |
| [Shape](http://docs.google.com/java/awt/Shape.html) | [**createStrokedShape**](http://docs.google.com/java/awt/BasicStroke.html#createStrokedShape(java.awt.Shape))([Shape](http://docs.google.com/java/awt/Shape.html) s)            Returns a Shape whose interior defines the stroked outline of a specified Shape. |
| boolean | [**equals**](http://docs.google.com/java/awt/BasicStroke.html#equals(java.lang.Object))([Object](http://docs.google.com/java/lang/Object.html) obj)            Tests if a specified object is equal to this BasicStroke by first testing if it is a BasicStroke and then comparing its width, join, cap, miter limit, dash, and dash phase attributes with those of this BasicStroke. |
| float[] | [**getDashArray**](http://docs.google.com/java/awt/BasicStroke.html#getDashArray())()            Returns the array representing the lengths of the dash segments. |
| float | [**getDashPhase**](http://docs.google.com/java/awt/BasicStroke.html#getDashPhase())()            Returns the current dash phase. |
| int | [**getEndCap**](http://docs.google.com/java/awt/BasicStroke.html#getEndCap())()            Returns the end cap style. |
| int | [**getLineJoin**](http://docs.google.com/java/awt/BasicStroke.html#getLineJoin())()            Returns the line join style. |
| float | [**getLineWidth**](http://docs.google.com/java/awt/BasicStroke.html#getLineWidth())()            Returns the line width. |
| float | [**getMiterLimit**](http://docs.google.com/java/awt/BasicStroke.html#getMiterLimit())()            Returns the limit of miter joins. |
| int | [**hashCode**](http://docs.google.com/java/awt/BasicStroke.html#hashCode())()            Returns the hashcode for this stroke. |

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

### JOIN\_MITER

public static final int **JOIN\_MITER**

Joins path segments by extending their outside edges until they meet.

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

### JOIN\_ROUND

public static final int **JOIN\_ROUND**

Joins path segments by rounding off the corner at a radius of half the line width.

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

### JOIN\_BEVEL

public static final int **JOIN\_BEVEL**

Joins path segments by connecting the outer corners of their wide outlines with a straight segment.

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

### CAP\_BUTT

public static final int **CAP\_BUTT**

Ends unclosed subpaths and dash segments with no added decoration.

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

### CAP\_ROUND

public static final int **CAP\_ROUND**

Ends unclosed subpaths and dash segments with a round decoration that has a radius equal to half of the width of the pen.

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

### CAP\_SQUARE

public static final int **CAP\_SQUARE**

Ends unclosed subpaths and dash segments with a square projection that extends beyond the end of the segment to a distance equal to half of the line width.

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

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

### BasicStroke

public **BasicStroke**(float width,  
 int cap,  
 int join,  
 float miterlimit,  
 float[] dash,  
 float dash\_phase)

Constructs a new BasicStroke with the specified attributes.

**Parameters:**width - the width of this BasicStroke. The width must be greater than or equal to 0.0f. If width is set to 0.0f, the stroke is rendered as the thinnest possible line for the target device and the antialias hint setting.cap - the decoration of the ends of a BasicStrokejoin - the decoration applied where path segments meetmiterlimit - the limit to trim the miter join. The miterlimit must be greater than or equal to 1.0f.dash - the array representing the dashing patterndash\_phase - the offset to start the dashing pattern **Throws:** [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if width is negative [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if cap is not either CAP\_BUTT, CAP\_ROUND or CAP\_SQUARE [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if miterlimit is less than 1 and join is JOIN\_MITER [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if join is not either JOIN\_ROUND, JOIN\_BEVEL, or JOIN\_MITER [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if dash\_phase is negative and dash is not null [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if the length of dash is zero [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if dash lengths are all zero.

### BasicStroke

public **BasicStroke**(float width,  
 int cap,  
 int join,  
 float miterlimit)

Constructs a solid BasicStroke with the specified attributes.

**Parameters:**width - the width of the BasicStrokecap - the decoration of the ends of a BasicStrokejoin - the decoration applied where path segments meetmiterlimit - the limit to trim the miter join **Throws:** [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if width is negative [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if cap is not either CAP\_BUTT, CAP\_ROUND or CAP\_SQUARE [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if miterlimit is less than 1 and join is JOIN\_MITER [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if join is not either JOIN\_ROUND, JOIN\_BEVEL, or JOIN\_MITER

### BasicStroke

public **BasicStroke**(float width,  
 int cap,  
 int join)

Constructs a solid BasicStroke with the specified attributes. The miterlimit parameter is unnecessary in cases where the default is allowable or the line joins are not specified as JOIN\_MITER.

**Parameters:**width - the width of the BasicStrokecap - the decoration of the ends of a BasicStrokejoin - the decoration applied where path segments meet **Throws:** [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if width is negative [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if cap is not either CAP\_BUTT, CAP\_ROUND or CAP\_SQUARE [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if join is not either JOIN\_ROUND, JOIN\_BEVEL, or JOIN\_MITER

### BasicStroke

public **BasicStroke**(float width)

Constructs a solid BasicStroke with the specified line width and with default values for the cap and join styles.

**Parameters:**width - the width of the BasicStroke **Throws:** [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if width is negative

### BasicStroke

public **BasicStroke**()

Constructs a new BasicStroke with defaults for all attributes. The default attributes are a solid line of width 1.0, CAP\_SQUARE, JOIN\_MITER, a miter limit of 10.0.

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

### createStrokedShape

public [Shape](http://docs.google.com/java/awt/Shape.html) **createStrokedShape**([Shape](http://docs.google.com/java/awt/Shape.html) s)

Returns a Shape whose interior defines the stroked outline of a specified Shape.

**Specified by:**[createStrokedShape](http://docs.google.com/java/awt/Stroke.html#createStrokedShape(java.awt.Shape)) in interface [Stroke](http://docs.google.com/java/awt/Stroke.html) **Parameters:**s - the Shape boundary be stroked **Returns:**the Shape of the stroked outline.

### getLineWidth

public float **getLineWidth**()

Returns the line width. Line width is represented in user space, which is the default-coordinate space used by Java 2D. See the Graphics2D class comments for more information on the user space coordinate system.

**Returns:**the line width of this BasicStroke.**See Also:**[Graphics2D](http://docs.google.com/java/awt/Graphics2D.html)

### getEndCap

public int **getEndCap**()

Returns the end cap style.

**Returns:**the end cap style of this BasicStroke as one of the static int values that define possible end cap styles.

### getLineJoin

public int **getLineJoin**()

Returns the line join style.

**Returns:**the line join style of the BasicStroke as one of the static int values that define possible line join styles.

### getMiterLimit

public float **getMiterLimit**()

Returns the limit of miter joins.

**Returns:**the limit of miter joins of the BasicStroke.

### getDashArray

public float[] **getDashArray**()

Returns the array representing the lengths of the dash segments. Alternate entries in the array represent the user space lengths of the opaque and transparent segments of the dashes. As the pen moves along the outline of the Shape to be stroked, the user space distance that the pen travels is accumulated. The distance value is used to index into the dash array. The pen is opaque when its current cumulative distance maps to an even element of the dash array and transparent otherwise.

**Returns:**the dash array.

### getDashPhase

public float **getDashPhase**()

Returns the current dash phase. The dash phase is a distance specified in user coordinates that represents an offset into the dashing pattern. In other words, the dash phase defines the point in the dashing pattern that will correspond to the beginning of the stroke.

**Returns:**the dash phase as a float value.

### hashCode

public int **hashCode**()

Returns the hashcode for this stroke.

**Overrides:**[hashCode](http://docs.google.com/java/lang/Object.html#hashCode()) in class [Object](http://docs.google.com/java/lang/Object.html) **Returns:**a hash code for this stroke.**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)

Tests if a specified object is equal to this BasicStroke by first testing if it is a BasicStroke and then comparing its width, join, cap, miter limit, dash, and dash phase attributes with those of this BasicStroke.

**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 - the specified object to compare to this BasicStroke **Returns:**true if the width, join, cap, miter limit, dash, and dash phase are the same for both objects; false otherwise.**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/BasicStroke.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/AWTPermission.html)   [**NEXT CLASS**](http://docs.google.com/java/awt/BorderLayout.html) | [**FRAMES**](http://docs.google.com/index.html?java/awt/BasicStroke.html)    [**NO FRAMES**](http://docs.google.com/BasicStroke.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#3znysh7) | [CONSTR](#2et92p0) | [METHOD](#tyjcwt) | DETAIL: [FIELD](#1t3h5sf) | [CONSTR](#35nkun2) | [METHOD](#1y810tw) |

[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).