| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/Collator.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/text/CollationKey.html)   [**NEXT CLASS**](http://docs.google.com/java/text/DateFormat.html) | [**FRAMES**](http://docs.google.com/index.html?java/text/Collator.html)    [**NO FRAMES**](http://docs.google.com/Collator.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#3znysh7) | [CONSTR](#2et92p0) | [METHOD](#tyjcwt) | DETAIL: [FIELD](#1t3h5sf) | [CONSTR](#1ksv4uv) | [METHOD](#2jxsxqh) |

## **java.text**

Class Collator

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

**All Implemented Interfaces:** [Cloneable](http://docs.google.com/java/lang/Cloneable.html), [Comparator](http://docs.google.com/java/util/Comparator.html)<[Object](http://docs.google.com/java/lang/Object.html)> **Direct Known Subclasses:** [RuleBasedCollator](http://docs.google.com/java/text/RuleBasedCollator.html)

public abstract class **Collator**extends [Object](http://docs.google.com/java/lang/Object.html)implements [Comparator](http://docs.google.com/java/util/Comparator.html)<[Object](http://docs.google.com/java/lang/Object.html)>, [Cloneable](http://docs.google.com/java/lang/Cloneable.html)

The Collator class performs locale-sensitive String comparison. You use this class to build searching and sorting routines for natural language text.

Collator is an abstract base class. Subclasses implement specific collation strategies. One subclass, RuleBasedCollator, is currently provided with the Java Platform and is applicable to a wide set of languages. Other subclasses may be created to handle more specialized needs.

Like other locale-sensitive classes, you can use the static factory method, getInstance, to obtain the appropriate Collator object for a given locale. You will only need to look at the subclasses of Collator if you need to understand the details of a particular collation strategy or if you need to modify that strategy.

The following example shows how to compare two strings using the Collator for the default locale.

// Compare two strings in the default locale  
 Collator myCollator = Collator.getInstance();  
 if( myCollator.compare("abc", "ABC") < 0 )  
 System.out.println("abc is less than ABC");  
 else  
 System.out.println("abc is greater than or equal to ABC");

You can set a Collator's *strength* property to determine the level of difference considered significant in comparisons. Four strengths are provided: PRIMARY, SECONDARY, TERTIARY, and IDENTICAL. The exact assignment of strengths to language features is locale dependant. For example, in Czech, "e" and "f" are considered primary differences, while "e" and "ě" are secondary differences, "e" and "E" are tertiary differences and "e" and "e" are identical. The following shows how both case and accents could be ignored for US English.

//Get the Collator for US English and set its strength to PRIMARY  
 Collator usCollator = Collator.getInstance(Locale.US);  
 usCollator.setStrength(Collator.PRIMARY);  
 if( usCollator.compare("abc", "ABC") == 0 ) {  
 System.out.println("Strings are equivalent");  
 }

For comparing Strings exactly once, the compare method provides the best performance. When sorting a list of Strings however, it is generally necessary to compare each String multiple times. In this case, CollationKeys provide better performance. The CollationKey class converts a String to a series of bits that can be compared bitwise against other CollationKeys. A CollationKey is created by a Collator object for a given String.

**Note:** CollationKeys from different Collators can not be compared. See the class description for [CollationKey](http://docs.google.com/java/text/CollationKey.html) for an example using CollationKeys.

**See Also:**[RuleBasedCollator](http://docs.google.com/java/text/RuleBasedCollator.html), [CollationKey](http://docs.google.com/java/text/CollationKey.html), [CollationElementIterator](http://docs.google.com/java/text/CollationElementIterator.html), [Locale](http://docs.google.com/java/util/Locale.html)

| **Field Summary** | |
| --- | --- |
| static int | [**CANONICAL\_DECOMPOSITION**](http://docs.google.com/java/text/Collator.html#CANONICAL_DECOMPOSITION)            Decomposition mode value. |
| static int | [**FULL\_DECOMPOSITION**](http://docs.google.com/java/text/Collator.html#FULL_DECOMPOSITION)            Decomposition mode value. |
| static int | [**IDENTICAL**](http://docs.google.com/java/text/Collator.html#IDENTICAL)            Collator strength value. |
| static int | [**NO\_DECOMPOSITION**](http://docs.google.com/java/text/Collator.html#NO_DECOMPOSITION)            Decomposition mode value. |
| static int | [**PRIMARY**](http://docs.google.com/java/text/Collator.html#PRIMARY)            Collator strength value. |
| static int | [**SECONDARY**](http://docs.google.com/java/text/Collator.html#SECONDARY)            Collator strength value. |
| static int | [**TERTIARY**](http://docs.google.com/java/text/Collator.html#TERTIARY)            Collator strength value. |

| **Constructor Summary** | |
| --- | --- |
| protected | [**Collator**](http://docs.google.com/java/text/Collator.html#Collator())()            Default constructor. |

| **Method Summary** | |
| --- | --- |
| [Object](http://docs.google.com/java/lang/Object.html) | [**clone**](http://docs.google.com/java/text/Collator.html#clone())()            Overrides Cloneable |
| int | [**compare**](http://docs.google.com/java/text/Collator.html#compare(java.lang.Object,%20java.lang.Object))([Object](http://docs.google.com/java/lang/Object.html) o1, [Object](http://docs.google.com/java/lang/Object.html) o2)            Compares its two arguments for order. |
| abstract  int | [**compare**](http://docs.google.com/java/text/Collator.html#compare(java.lang.String,%20java.lang.String))([String](http://docs.google.com/java/lang/String.html) source, [String](http://docs.google.com/java/lang/String.html) target)            Compares the source string to the target string according to the collation rules for this Collator. |
| boolean | [**equals**](http://docs.google.com/java/text/Collator.html#equals(java.lang.Object))([Object](http://docs.google.com/java/lang/Object.html) that)            Compares the equality of two Collators. |
| boolean | [**equals**](http://docs.google.com/java/text/Collator.html#equals(java.lang.String,%20java.lang.String))([String](http://docs.google.com/java/lang/String.html) source, [String](http://docs.google.com/java/lang/String.html) target)            Convenience method for comparing the equality of two strings based on this Collator's collation rules. |
| static [Locale](http://docs.google.com/java/util/Locale.html)[] | [**getAvailableLocales**](http://docs.google.com/java/text/Collator.html#getAvailableLocales())()            Returns an array of all locales for which the getInstance methods of this class can return localized instances. |
| abstract  [CollationKey](http://docs.google.com/java/text/CollationKey.html) | [**getCollationKey**](http://docs.google.com/java/text/Collator.html#getCollationKey(java.lang.String))([String](http://docs.google.com/java/lang/String.html) source)            Transforms the String into a series of bits that can be compared bitwise to other CollationKeys. |
| int | [**getDecomposition**](http://docs.google.com/java/text/Collator.html#getDecomposition())()            Get the decomposition mode of this Collator. |
| static [Collator](http://docs.google.com/java/text/Collator.html) | [**getInstance**](http://docs.google.com/java/text/Collator.html#getInstance())()            Gets the Collator for the current default locale. |
| static [Collator](http://docs.google.com/java/text/Collator.html) | [**getInstance**](http://docs.google.com/java/text/Collator.html#getInstance(java.util.Locale))([Locale](http://docs.google.com/java/util/Locale.html) desiredLocale)            Gets the Collator for the desired locale. |
| int | [**getStrength**](http://docs.google.com/java/text/Collator.html#getStrength())()            Returns this Collator's strength property. |
| abstract  int | [**hashCode**](http://docs.google.com/java/text/Collator.html#hashCode())()            Generates the hash code for this Collator. |
| void | [**setDecomposition**](http://docs.google.com/java/text/Collator.html#setDecomposition(int))(int decompositionMode)            Set the decomposition mode of this Collator. |
| void | [**setStrength**](http://docs.google.com/java/text/Collator.html#setStrength(int))(int newStrength)            Sets this Collator's strength property. |

| **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** |
| --- |

### PRIMARY

public static final int **PRIMARY**

Collator strength value. When set, only PRIMARY differences are considered significant during comparison. The assignment of strengths to language features is locale dependant. A common example is for different base letters ("a" vs "b") to be considered a PRIMARY difference.

**See Also:**[setStrength(int)](http://docs.google.com/java/text/Collator.html#setStrength(int)), [getStrength()](http://docs.google.com/java/text/Collator.html#getStrength()), [Constant Field Values](http://docs.google.com/constant-values.html#java.text.Collator.PRIMARY)

### SECONDARY

public static final int **SECONDARY**

Collator strength value. When set, only SECONDARY and above differences are considered significant during comparison. The assignment of strengths to language features is locale dependant. A common example is for different accented forms of the same base letter ("a" vs "�") to be considered a SECONDARY difference.

**See Also:**[setStrength(int)](http://docs.google.com/java/text/Collator.html#setStrength(int)), [getStrength()](http://docs.google.com/java/text/Collator.html#getStrength()), [Constant Field Values](http://docs.google.com/constant-values.html#java.text.Collator.SECONDARY)

### TERTIARY

public static final int **TERTIARY**

Collator strength value. When set, only TERTIARY and above differences are considered significant during comparison. The assignment of strengths to language features is locale dependant. A common example is for case differences ("a" vs "A") to be considered a TERTIARY difference.

**See Also:**[setStrength(int)](http://docs.google.com/java/text/Collator.html#setStrength(int)), [getStrength()](http://docs.google.com/java/text/Collator.html#getStrength()), [Constant Field Values](http://docs.google.com/constant-values.html#java.text.Collator.TERTIARY)

### IDENTICAL

public static final int **IDENTICAL**

Collator strength value. When set, all differences are considered significant during comparison. The assignment of strengths to language features is locale dependant. A common example is for control characters ("\u0001" vs "\u0002") to be considered equal at the PRIMARY, SECONDARY, and TERTIARY levels but different at the IDENTICAL level. Additionally, differences between pre-composed accents such as "\u00C0" (A-grave) and combining accents such as "A\u0300" (A, combining-grave) will be considered significant at the IDENTICAL level if decomposition is set to NO\_DECOMPOSITION.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.text.Collator.IDENTICAL)

### NO\_DECOMPOSITION

public static final int **NO\_DECOMPOSITION**

Decomposition mode value. With NO\_DECOMPOSITION set, accented characters will not be decomposed for collation. This is the default setting and provides the fastest collation but will only produce correct results for languages that do not use accents.

**See Also:**[getDecomposition()](http://docs.google.com/java/text/Collator.html#getDecomposition()), [setDecomposition(int)](http://docs.google.com/java/text/Collator.html#setDecomposition(int)), [Constant Field Values](http://docs.google.com/constant-values.html#java.text.Collator.NO_DECOMPOSITION)

### CANONICAL\_DECOMPOSITION

public static final int **CANONICAL\_DECOMPOSITION**

Decomposition mode value. With CANONICAL\_DECOMPOSITION set, characters that are canonical variants according to Unicode standard will be decomposed for collation. This should be used to get correct collation of accented characters.

CANONICAL\_DECOMPOSITION corresponds to Normalization Form D as described in [Unicode Technical Report #15](http://www.unicode.org/unicode/reports/tr15/tr15-23.html).

**See Also:**[getDecomposition()](http://docs.google.com/java/text/Collator.html#getDecomposition()), [setDecomposition(int)](http://docs.google.com/java/text/Collator.html#setDecomposition(int)), [Constant Field Values](http://docs.google.com/constant-values.html#java.text.Collator.CANONICAL_DECOMPOSITION)

### FULL\_DECOMPOSITION

public static final int **FULL\_DECOMPOSITION**

Decomposition mode value. With FULL\_DECOMPOSITION set, both Unicode canonical variants and Unicode compatibility variants will be decomposed for collation. This causes not only accented characters to be collated, but also characters that have special formats to be collated with their norminal form. For example, the half-width and full-width ASCII and Katakana characters are then collated together. FULL\_DECOMPOSITION is the most complete and therefore the slowest decomposition mode.

FULL\_DECOMPOSITION corresponds to Normalization Form KD as described in [Unicode Technical Report #15](http://www.unicode.org/unicode/reports/tr15/tr15-23.html).

**See Also:**[getDecomposition()](http://docs.google.com/java/text/Collator.html#getDecomposition()), [setDecomposition(int)](http://docs.google.com/java/text/Collator.html#setDecomposition(int)), [Constant Field Values](http://docs.google.com/constant-values.html#java.text.Collator.FULL_DECOMPOSITION)

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

### Collator

protected **Collator**()

Default constructor. This constructor is protected so subclasses can get access to it. Users typically create a Collator sub-class by calling the factory method getInstance.

**See Also:**[getInstance()](http://docs.google.com/java/text/Collator.html#getInstance())

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

### getInstance

public static [Collator](http://docs.google.com/java/text/Collator.html) **getInstance**()

Gets the Collator for the current default locale. The default locale is determined by java.util.Locale.getDefault.

**Returns:**the Collator for the default locale.(for example, en\_US)**See Also:**[Locale.getDefault()](http://docs.google.com/java/util/Locale.html#getDefault())

### getInstance

public static [Collator](http://docs.google.com/java/text/Collator.html) **getInstance**([Locale](http://docs.google.com/java/util/Locale.html) desiredLocale)

Gets the Collator for the desired locale.

**Parameters:**desiredLocale - the desired locale. **Returns:**the Collator for the desired locale.**See Also:**[Locale](http://docs.google.com/java/util/Locale.html), [ResourceBundle](http://docs.google.com/java/util/ResourceBundle.html)

### compare

public abstract int **compare**([String](http://docs.google.com/java/lang/String.html) source,  
 [String](http://docs.google.com/java/lang/String.html) target)

Compares the source string to the target string according to the collation rules for this Collator. Returns an integer less than, equal to or greater than zero depending on whether the source String is less than, equal to or greater than the target string. See the Collator class description for an example of use.

For a one time comparison, this method has the best performance. If a given String will be involved in multiple comparisons, CollationKey.compareTo has the best performance. See the Collator class description for an example using CollationKeys.

**Parameters:**source - the source string.target - the target string. **Returns:**Returns an integer value. Value is less than zero if source is less than target, value is zero if source and target are equal, value is greater than zero if source is greater than target.**See Also:**[CollationKey](http://docs.google.com/java/text/CollationKey.html), [getCollationKey(java.lang.String)](http://docs.google.com/java/text/Collator.html#getCollationKey(java.lang.String))

### compare

public int **compare**([Object](http://docs.google.com/java/lang/Object.html) o1,  
 [Object](http://docs.google.com/java/lang/Object.html) o2)

Compares its two arguments for order. Returns a negative integer, zero, or a positive integer as the first argument is less than, equal to, or greater than the second.

This implementation merely returns compare((String)o1, (String)o2) .

**Specified by:**[compare](http://docs.google.com/java/util/Comparator.html#compare(T,%20T)) in interface [Comparator](http://docs.google.com/java/util/Comparator.html)<[Object](http://docs.google.com/java/lang/Object.html)> **Parameters:**o1 - the first object to be compared.o2 - the second object to be compared. **Returns:**a negative integer, zero, or a positive integer as the first argument is less than, equal to, or greater than the second. **Throws:** [ClassCastException](http://docs.google.com/java/lang/ClassCastException.html) - the arguments cannot be cast to Strings.**Since:** 1.2 **See Also:**[Comparator](http://docs.google.com/java/util/Comparator.html)

### getCollationKey

public abstract [CollationKey](http://docs.google.com/java/text/CollationKey.html) **getCollationKey**([String](http://docs.google.com/java/lang/String.html) source)

Transforms the String into a series of bits that can be compared bitwise to other CollationKeys. CollationKeys provide better performance than Collator.compare when Strings are involved in multiple comparisons. See the Collator class description for an example using CollationKeys.

**Parameters:**source - the string to be transformed into a collation key. **Returns:**the CollationKey for the given String based on this Collator's collation rules. If the source String is null, a null CollationKey is returned.**See Also:**[CollationKey](http://docs.google.com/java/text/CollationKey.html), [compare(java.lang.String, java.lang.String)](http://docs.google.com/java/text/Collator.html#compare(java.lang.String,%20java.lang.String))

### equals

public boolean **equals**([String](http://docs.google.com/java/lang/String.html) source,  
 [String](http://docs.google.com/java/lang/String.html) target)

Convenience method for comparing the equality of two strings based on this Collator's collation rules.

**Parameters:**source - the source string to be compared with.target - the target string to be compared with. **Returns:**true if the strings are equal according to the collation rules. false, otherwise.**See Also:**[compare(java.lang.String, java.lang.String)](http://docs.google.com/java/text/Collator.html#compare(java.lang.String,%20java.lang.String))

### getStrength

public int **getStrength**()

Returns this Collator's strength property. The strength property determines the minimum level of difference considered significant during comparison. See the Collator class description for an example of use.

**Returns:**this Collator's current strength property.**See Also:**[setStrength(int)](http://docs.google.com/java/text/Collator.html#setStrength(int)), [PRIMARY](http://docs.google.com/java/text/Collator.html#PRIMARY), [SECONDARY](http://docs.google.com/java/text/Collator.html#SECONDARY), [TERTIARY](http://docs.google.com/java/text/Collator.html#TERTIARY), [IDENTICAL](http://docs.google.com/java/text/Collator.html#IDENTICAL)

### setStrength

public void **setStrength**(int newStrength)

Sets this Collator's strength property. The strength property determines the minimum level of difference considered significant during comparison. See the Collator class description for an example of use.

**Parameters:**newStrength - the new strength value. **Throws:** [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - If the new strength value is not one of PRIMARY, SECONDARY, TERTIARY or IDENTICAL.**See Also:**[getStrength()](http://docs.google.com/java/text/Collator.html#getStrength()), [PRIMARY](http://docs.google.com/java/text/Collator.html#PRIMARY), [SECONDARY](http://docs.google.com/java/text/Collator.html#SECONDARY), [TERTIARY](http://docs.google.com/java/text/Collator.html#TERTIARY), [IDENTICAL](http://docs.google.com/java/text/Collator.html#IDENTICAL)

### getDecomposition

public int **getDecomposition**()

Get the decomposition mode of this Collator. Decomposition mode determines how Unicode composed characters are handled. Adjusting decomposition mode allows the user to select between faster and more complete collation behavior.

The three values for decomposition mode are:

* NO\_DECOMPOSITION,
* CANONICAL\_DECOMPOSITION
* FULL\_DECOMPOSITION.

See the documentation for these three constants for a description of their meaning.

**Returns:**the decomposition mode**See Also:**[setDecomposition(int)](http://docs.google.com/java/text/Collator.html#setDecomposition(int)), [NO\_DECOMPOSITION](http://docs.google.com/java/text/Collator.html#NO_DECOMPOSITION), [CANONICAL\_DECOMPOSITION](http://docs.google.com/java/text/Collator.html#CANONICAL_DECOMPOSITION), [FULL\_DECOMPOSITION](http://docs.google.com/java/text/Collator.html#FULL_DECOMPOSITION)

### setDecomposition

public void **setDecomposition**(int decompositionMode)

Set the decomposition mode of this Collator. See getDecomposition for a description of decomposition mode.

**Parameters:**decompositionMode - the new decomposition mode. **Throws:** [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - If the given value is not a valid decomposition mode.**See Also:**[getDecomposition()](http://docs.google.com/java/text/Collator.html#getDecomposition()), [NO\_DECOMPOSITION](http://docs.google.com/java/text/Collator.html#NO_DECOMPOSITION), [CANONICAL\_DECOMPOSITION](http://docs.google.com/java/text/Collator.html#CANONICAL_DECOMPOSITION), [FULL\_DECOMPOSITION](http://docs.google.com/java/text/Collator.html#FULL_DECOMPOSITION)

### getAvailableLocales

public static [Locale](http://docs.google.com/java/util/Locale.html)[] **getAvailableLocales**()

Returns an array of all locales for which the getInstance methods of this class can return localized instances. The returned array represents the union of locales supported by the Java runtime and by installed [CollatorProvider](http://docs.google.com/java/text/spi/CollatorProvider.html) implementations. It must contain at least a Locale instance equal to [Locale.US](http://docs.google.com/java/util/Locale.html#US).

**Returns:**An array of locales for which localized Collator instances are available.

### clone

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

Overrides Cloneable

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

### equals

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

Compares the equality of two Collators.

**Specified by:**[equals](http://docs.google.com/java/util/Comparator.html#equals(java.lang.Object)) in interface [Comparator](http://docs.google.com/java/util/Comparator.html)<[Object](http://docs.google.com/java/lang/Object.html)>**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:**that - the Collator to be compared with this. **Returns:**true if this Collator is the same as that Collator; false otherwise.**See Also:**[Object.hashCode()](http://docs.google.com/java/lang/Object.html#hashCode()), [Hashtable](http://docs.google.com/java/util/Hashtable.html)

### hashCode

public abstract int **hashCode**()

Generates the hash code for this Collator.

**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 value for this object.**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)

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/Collator.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/text/CollationKey.html)   [**NEXT CLASS**](http://docs.google.com/java/text/DateFormat.html) | [**FRAMES**](http://docs.google.com/index.html?java/text/Collator.html)    [**NO FRAMES**](http://docs.google.com/Collator.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#3znysh7) | [CONSTR](#2et92p0) | [METHOD](#tyjcwt) | DETAIL: [FIELD](#1t3h5sf) | [CONSTR](#1ksv4uv) | [METHOD](#2jxsxqh) |

[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).