| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/ComponentOrientation.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/Component.FlipBufferStrategy.html)   [**NEXT CLASS**](http://docs.google.com/java/awt/Composite.html) | [**FRAMES**](http://docs.google.com/index.html?java/awt/ComponentOrientation.html)    [**NO FRAMES**](http://docs.google.com/ComponentOrientation.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#3znysh7) | CONSTR | [METHOD](#2et92p0) | DETAIL: [FIELD](#3dy6vkm) | CONSTR | [METHOD](#17dp8vu) |

## **java.awt**

Class ComponentOrientation

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

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

public final class **ComponentOrientation**extends [Object](http://docs.google.com/java/lang/Object.html)implements [Serializable](http://docs.google.com/java/io/Serializable.html)

The ComponentOrientation class encapsulates the language-sensitive orientation that is to be used to order the elements of a component or of text. It is used to reflect the differences in this ordering between Western alphabets, Middle Eastern (such as Hebrew), and Far Eastern (such as Japanese).

Fundamentally, this governs items (such as characters) which are laid out in lines, with the lines then laid out in a block. This also applies to items in a widget: for example, in a check box where the box is positioned relative to the text.

There are four different orientations used in modern languages as in the following table.

LT RT TL TR   
 A B C C B A A D G G D A  
 D E F F E D B E H H E B  
 G H I I H G C F I I F C

(In the header, the two-letter abbreviation represents the item direction in the first letter, and the line direction in the second. For example, LT means "items left-to-right, lines top-to-bottom", TL means "items top-to-bottom, lines left-to-right", and so on.)

The orientations are:

* LT - Western Europe (optional for Japanese, Chinese, Korean)
* RT - Middle East (Arabic, Hebrew)
* TR - Japanese, Chinese, Korean
* TL - Mongolian

Components whose view and controller code depends on orientation should use the isLeftToRight() and isHorizontal() methods to determine their behavior. They should not include switch-like code that keys off of the constants, such as:

if (orientation == LEFT\_TO\_RIGHT) {  
 ...  
 } else if (orientation == RIGHT\_TO\_LEFT) {  
 ...  
 } else {  
 // Oops  
 }

This is unsafe, since more constants may be added in the future and since it is not guaranteed that orientation objects will be unique.

**See Also:**[Serialized Form](http://docs.google.com/serialized-form.html#java.awt.ComponentOrientation)

| **Field Summary** | |
| --- | --- |
| static [ComponentOrientation](http://docs.google.com/java/awt/ComponentOrientation.html) | [**LEFT\_TO\_RIGHT**](http://docs.google.com/java/awt/ComponentOrientation.html#LEFT_TO_RIGHT)            Items run left to right and lines flow top to bottom Examples: English, French. |
| static [ComponentOrientation](http://docs.google.com/java/awt/ComponentOrientation.html) | [**RIGHT\_TO\_LEFT**](http://docs.google.com/java/awt/ComponentOrientation.html#RIGHT_TO_LEFT)            Items run right to left and lines flow top to bottom Examples: Arabic, Hebrew. |
| static [ComponentOrientation](http://docs.google.com/java/awt/ComponentOrientation.html) | [**UNKNOWN**](http://docs.google.com/java/awt/ComponentOrientation.html#UNKNOWN)            Indicates that a component's orientation has not been set. |

| **Method Summary** | |
| --- | --- |
| static [ComponentOrientation](http://docs.google.com/java/awt/ComponentOrientation.html) | [**getOrientation**](http://docs.google.com/java/awt/ComponentOrientation.html#getOrientation(java.util.Locale))([Locale](http://docs.google.com/java/util/Locale.html) locale)            Returns the orientation that is appropriate for the given locale. |
| static [ComponentOrientation](http://docs.google.com/java/awt/ComponentOrientation.html) | [**getOrientation**](http://docs.google.com/java/awt/ComponentOrientation.html#getOrientation(java.util.ResourceBundle))([ResourceBundle](http://docs.google.com/java/util/ResourceBundle.html) bdl)  **Deprecated.** *As of J2SE 1.4, use* [*getOrientation(java.util.Locale)*](http://docs.google.com/java/awt/ComponentOrientation.html#getOrientation(java.util.Locale))*.* |
| boolean | [**isHorizontal**](http://docs.google.com/java/awt/ComponentOrientation.html#isHorizontal())()            Are lines horizontal? This will return true for horizontal, left-to-right writing systems such as Roman. |
| boolean | [**isLeftToRight**](http://docs.google.com/java/awt/ComponentOrientation.html#isLeftToRight())()            HorizontalLines: Do items run left-to-right?  Vertical Lines: Do lines run left-to-right?  This will return true for horizontal, left-to-right writing systems such as Roman. |

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

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

### LEFT\_TO\_RIGHT

public static final [ComponentOrientation](http://docs.google.com/java/awt/ComponentOrientation.html) **LEFT\_TO\_RIGHT**

Items run left to right and lines flow top to bottom Examples: English, French.

### RIGHT\_TO\_LEFT

public static final [ComponentOrientation](http://docs.google.com/java/awt/ComponentOrientation.html) **RIGHT\_TO\_LEFT**

Items run right to left and lines flow top to bottom Examples: Arabic, Hebrew.

### UNKNOWN

public static final [ComponentOrientation](http://docs.google.com/java/awt/ComponentOrientation.html) **UNKNOWN**

Indicates that a component's orientation has not been set. To preserve the behavior of existing applications, isLeftToRight will return true for this value.

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

### isHorizontal

public boolean **isHorizontal**()

Are lines horizontal? This will return true for horizontal, left-to-right writing systems such as Roman.

### isLeftToRight

public boolean **isLeftToRight**()

HorizontalLines: Do items run left-to-right?

Vertical Lines: Do lines run left-to-right?

This will return true for horizontal, left-to-right writing systems such as Roman.

### getOrientation

public static [ComponentOrientation](http://docs.google.com/java/awt/ComponentOrientation.html) **getOrientation**([Locale](http://docs.google.com/java/util/Locale.html) locale)

Returns the orientation that is appropriate for the given locale.

**Parameters:**locale - the specified locale

### getOrientation

[@Deprecated](http://docs.google.com/java/lang/Deprecated.html)  
public static [ComponentOrientation](http://docs.google.com/java/awt/ComponentOrientation.html) **getOrientation**([ResourceBundle](http://docs.google.com/java/util/ResourceBundle.html) bdl)

**Deprecated.** *As of J2SE 1.4, use* [*getOrientation(java.util.Locale)*](http://docs.google.com/java/awt/ComponentOrientation.html#getOrientation(java.util.Locale))*.*

Returns the orientation appropriate for the given ResourceBundle's localization. Three approaches are tried, in the following order:

1. Retrieve a ComponentOrientation object from the ResourceBundle using the string "Orientation" as the key.
2. Use the ResourceBundle.getLocale to determine the bundle's locale, then return the orientation for that locale.
3. Return the default locale's orientation.

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/ComponentOrientation.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/Component.FlipBufferStrategy.html)   [**NEXT CLASS**](http://docs.google.com/java/awt/Composite.html) | [**FRAMES**](http://docs.google.com/index.html?java/awt/ComponentOrientation.html)    [**NO FRAMES**](http://docs.google.com/ComponentOrientation.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#3znysh7) | CONSTR | [METHOD](#2et92p0) | DETAIL: [FIELD](#3dy6vkm) | CONSTR | [METHOD](#17dp8vu) |

[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).