| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/InternalFrameListener.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/javax/swing/event/InternalFrameEvent.html)   [**NEXT CLASS**](http://docs.google.com/javax/swing/event/ListDataEvent.html) | [**FRAMES**](http://docs.google.com/index.html?javax/swing/event/InternalFrameListener.html)    [**NO FRAMES**](http://docs.google.com/InternalFrameListener.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#3znysh7) | DETAIL: FIELD | CONSTR | [METHOD](#2et92p0) |

## **javax.swing.event**

Interface InternalFrameListener

**All Superinterfaces:** [EventListener](http://docs.google.com/java/util/EventListener.html) **All Known Implementing Classes:** [BasicInternalFrameUI.BasicInternalFrameListener](http://docs.google.com/javax/swing/plaf/basic/BasicInternalFrameUI.BasicInternalFrameListener.html), [InternalFrameAdapter](http://docs.google.com/javax/swing/event/InternalFrameAdapter.html)

public interface **InternalFrameListener**extends [EventListener](http://docs.google.com/java/util/EventListener.html)

The listener interface for receiving internal frame events. This class is functionally equivalent to the WindowListener class in the AWT.

See [How to Write an Internal Frame Listener](http://java.sun.com/docs/books/tutorial/uiswing/events/internalframelistener.html) in *The Java Tutorial* and [The Java Class Libraries (update)](http://www.awl.com/cp/javaseries/jcl1_2.html) for further documentation.

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

| **Method Summary** | |
| --- | --- |
| void | [**internalFrameActivated**](http://docs.google.com/javax/swing/event/InternalFrameListener.html#internalFrameActivated(javax.swing.event.InternalFrameEvent))([InternalFrameEvent](http://docs.google.com/javax/swing/event/InternalFrameEvent.html) e)            Invoked when an internal frame is activated. |
| void | [**internalFrameClosed**](http://docs.google.com/javax/swing/event/InternalFrameListener.html#internalFrameClosed(javax.swing.event.InternalFrameEvent))([InternalFrameEvent](http://docs.google.com/javax/swing/event/InternalFrameEvent.html) e)            Invoked when an internal frame has been closed. |
| void | [**internalFrameClosing**](http://docs.google.com/javax/swing/event/InternalFrameListener.html#internalFrameClosing(javax.swing.event.InternalFrameEvent))([InternalFrameEvent](http://docs.google.com/javax/swing/event/InternalFrameEvent.html) e)            Invoked when an internal frame is in the process of being closed. |
| void | [**internalFrameDeactivated**](http://docs.google.com/javax/swing/event/InternalFrameListener.html#internalFrameDeactivated(javax.swing.event.InternalFrameEvent))([InternalFrameEvent](http://docs.google.com/javax/swing/event/InternalFrameEvent.html) e)            Invoked when an internal frame is de-activated. |
| void | [**internalFrameDeiconified**](http://docs.google.com/javax/swing/event/InternalFrameListener.html#internalFrameDeiconified(javax.swing.event.InternalFrameEvent))([InternalFrameEvent](http://docs.google.com/javax/swing/event/InternalFrameEvent.html) e)            Invoked when an internal frame is de-iconified. |
| void | [**internalFrameIconified**](http://docs.google.com/javax/swing/event/InternalFrameListener.html#internalFrameIconified(javax.swing.event.InternalFrameEvent))([InternalFrameEvent](http://docs.google.com/javax/swing/event/InternalFrameEvent.html) e)            Invoked when an internal frame is iconified. |
| void | [**internalFrameOpened**](http://docs.google.com/javax/swing/event/InternalFrameListener.html#internalFrameOpened(javax.swing.event.InternalFrameEvent))([InternalFrameEvent](http://docs.google.com/javax/swing/event/InternalFrameEvent.html) e)            Invoked when a internal frame has been opened. |

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

### internalFrameOpened

void **internalFrameOpened**([InternalFrameEvent](http://docs.google.com/javax/swing/event/InternalFrameEvent.html) e)

Invoked when a internal frame has been opened.

**See Also:**[JInternalFrame.show()](http://docs.google.com/javax/swing/JInternalFrame.html#show())

### internalFrameClosing

void **internalFrameClosing**([InternalFrameEvent](http://docs.google.com/javax/swing/event/InternalFrameEvent.html) e)

Invoked when an internal frame is in the process of being closed. The close operation can be overridden at this point.

**See Also:**[JInternalFrame.setDefaultCloseOperation(int)](http://docs.google.com/javax/swing/JInternalFrame.html#setDefaultCloseOperation(int))

### internalFrameClosed

void **internalFrameClosed**([InternalFrameEvent](http://docs.google.com/javax/swing/event/InternalFrameEvent.html) e)

Invoked when an internal frame has been closed.

**See Also:**[JInternalFrame.setClosed(boolean)](http://docs.google.com/javax/swing/JInternalFrame.html#setClosed(boolean))

### internalFrameIconified

void **internalFrameIconified**([InternalFrameEvent](http://docs.google.com/javax/swing/event/InternalFrameEvent.html) e)

Invoked when an internal frame is iconified.

**See Also:**[JInternalFrame.setIcon(boolean)](http://docs.google.com/javax/swing/JInternalFrame.html#setIcon(boolean))

### internalFrameDeiconified

void **internalFrameDeiconified**([InternalFrameEvent](http://docs.google.com/javax/swing/event/InternalFrameEvent.html) e)

Invoked when an internal frame is de-iconified.

**See Also:**[JInternalFrame.setIcon(boolean)](http://docs.google.com/javax/swing/JInternalFrame.html#setIcon(boolean))

### internalFrameActivated

void **internalFrameActivated**([InternalFrameEvent](http://docs.google.com/javax/swing/event/InternalFrameEvent.html) e)

Invoked when an internal frame is activated.

**See Also:**[JInternalFrame.setSelected(boolean)](http://docs.google.com/javax/swing/JInternalFrame.html#setSelected(boolean))

### internalFrameDeactivated

void **internalFrameDeactivated**([InternalFrameEvent](http://docs.google.com/javax/swing/event/InternalFrameEvent.html) e)

Invoked when an internal frame is de-activated.

**See Also:**[JInternalFrame.setSelected(boolean)](http://docs.google.com/javax/swing/JInternalFrame.html#setSelected(boolean))

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/InternalFrameListener.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/javax/swing/event/InternalFrameEvent.html)   [**NEXT CLASS**](http://docs.google.com/javax/swing/event/ListDataEvent.html) | [**FRAMES**](http://docs.google.com/index.html?javax/swing/event/InternalFrameListener.html)    [**NO FRAMES**](http://docs.google.com/InternalFrameListener.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#3znysh7) | DETAIL: FIELD | CONSTR | [METHOD](#2et92p0) |

[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).