| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/Applet.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   [**NEXT CLASS**](http://docs.google.com/java/applet/Applet.AccessibleApplet.html) | [**FRAMES**](http://docs.google.com/index.html?java/applet/Applet.html)    [**NO FRAMES**](http://docs.google.com/Applet.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: [NESTED](#3znysh7) | [FIELD](#4d34og8) | [CONSTR](#17dp8vu) | [METHOD](#3rdcrjn) | DETAIL: FIELD | [CONSTR](#44sinio) | [METHOD](#z337ya) |

## **java.applet**

Class Applet

[java.lang.Object](http://docs.google.com/java/lang/Object.html)  
 [java.awt.Component](http://docs.google.com/java/awt/Component.html)  
 [java.awt.Container](http://docs.google.com/java/awt/Container.html)  
 [java.awt.Panel](http://docs.google.com/java/awt/Panel.html)  
 **java.applet.Applet**

**All Implemented Interfaces:** [ImageObserver](http://docs.google.com/java/awt/image/ImageObserver.html), [MenuContainer](http://docs.google.com/java/awt/MenuContainer.html), [Serializable](http://docs.google.com/java/io/Serializable.html), [Accessible](http://docs.google.com/javax/accessibility/Accessible.html) **Direct Known Subclasses:** [JApplet](http://docs.google.com/javax/swing/JApplet.html)

public class **Applet**extends [Panel](http://docs.google.com/java/awt/Panel.html)

An applet is a small program that is intended not to be run on its own, but rather to be embedded inside another application.

The Applet class must be the superclass of any applet that is to be embedded in a Web page or viewed by the Java Applet Viewer. The Applet class provides a standard interface between applets and their environment.

**Since:** JDK1.0 **See Also:**[Serialized Form](http://docs.google.com/serialized-form.html#java.applet.Applet)

| **Nested Class Summary** | |
| --- | --- |
| protected  class | [**Applet.AccessibleApplet**](http://docs.google.com/java/applet/Applet.AccessibleApplet.html)            This class implements accessibility support for the Applet class. |

| **Nested classes/interfaces inherited from class java.awt.**[**Panel**](http://docs.google.com/java/awt/Panel.html) |
| --- |
| [Panel.AccessibleAWTPanel](http://docs.google.com/java/awt/Panel.AccessibleAWTPanel.html) |

| **Nested classes/interfaces inherited from class java.awt.**[**Container**](http://docs.google.com/java/awt/Container.html) |
| --- |
| [Container.AccessibleAWTContainer](http://docs.google.com/java/awt/Container.AccessibleAWTContainer.html) |

| **Nested classes/interfaces inherited from class java.awt.**[**Component**](http://docs.google.com/java/awt/Component.html) |
| --- |
| [Component.AccessibleAWTComponent](http://docs.google.com/java/awt/Component.AccessibleAWTComponent.html), [Component.BaselineResizeBehavior](http://docs.google.com/java/awt/Component.BaselineResizeBehavior.html), [Component.BltBufferStrategy](http://docs.google.com/java/awt/Component.BltBufferStrategy.html), [Component.FlipBufferStrategy](http://docs.google.com/java/awt/Component.FlipBufferStrategy.html) |

| **Field Summary** | |
| --- | --- |

| **Fields inherited from class java.awt.**[**Component**](http://docs.google.com/java/awt/Component.html) |
| --- |
| [BOTTOM\_ALIGNMENT](http://docs.google.com/java/awt/Component.html#BOTTOM_ALIGNMENT), [CENTER\_ALIGNMENT](http://docs.google.com/java/awt/Component.html#CENTER_ALIGNMENT), [LEFT\_ALIGNMENT](http://docs.google.com/java/awt/Component.html#LEFT_ALIGNMENT), [RIGHT\_ALIGNMENT](http://docs.google.com/java/awt/Component.html#RIGHT_ALIGNMENT), [TOP\_ALIGNMENT](http://docs.google.com/java/awt/Component.html#TOP_ALIGNMENT) |

| **Fields inherited from interface java.awt.image.**[**ImageObserver**](http://docs.google.com/java/awt/image/ImageObserver.html) |
| --- |
| [ABORT](http://docs.google.com/java/awt/image/ImageObserver.html#ABORT), [ALLBITS](http://docs.google.com/java/awt/image/ImageObserver.html#ALLBITS), [ERROR](http://docs.google.com/java/awt/image/ImageObserver.html#ERROR), [FRAMEBITS](http://docs.google.com/java/awt/image/ImageObserver.html#FRAMEBITS), [HEIGHT](http://docs.google.com/java/awt/image/ImageObserver.html#HEIGHT), [PROPERTIES](http://docs.google.com/java/awt/image/ImageObserver.html#PROPERTIES), [SOMEBITS](http://docs.google.com/java/awt/image/ImageObserver.html#SOMEBITS), [WIDTH](http://docs.google.com/java/awt/image/ImageObserver.html#WIDTH) |

| **Constructor Summary** | |
| --- | --- |
| [**Applet**](http://docs.google.com/java/applet/Applet.html#Applet())()            Constructs a new Applet. |

| **Method Summary** | |
| --- | --- |
| void | [**destroy**](http://docs.google.com/java/applet/Applet.html#destroy())()            Called by the browser or applet viewer to inform this applet that it is being reclaimed and that it should destroy any resources that it has allocated. |
| [AccessibleContext](http://docs.google.com/javax/accessibility/AccessibleContext.html) | [**getAccessibleContext**](http://docs.google.com/java/applet/Applet.html#getAccessibleContext())()            Gets the AccessibleContext associated with this Applet. |
| [AppletContext](http://docs.google.com/java/applet/AppletContext.html) | [**getAppletContext**](http://docs.google.com/java/applet/Applet.html#getAppletContext())()            Determines this applet's context, which allows the applet to query and affect the environment in which it runs. |
| [String](http://docs.google.com/java/lang/String.html) | [**getAppletInfo**](http://docs.google.com/java/applet/Applet.html#getAppletInfo())()            Returns information about this applet. |
| [AudioClip](http://docs.google.com/java/applet/AudioClip.html) | [**getAudioClip**](http://docs.google.com/java/applet/Applet.html#getAudioClip(java.net.URL))([URL](http://docs.google.com/java/net/URL.html) url)            Returns the AudioClip object specified by the URL argument. |
| [AudioClip](http://docs.google.com/java/applet/AudioClip.html) | [**getAudioClip**](http://docs.google.com/java/applet/Applet.html#getAudioClip(java.net.URL,%20java.lang.String))([URL](http://docs.google.com/java/net/URL.html) url, [String](http://docs.google.com/java/lang/String.html) name)            Returns the AudioClip object specified by the URL and name arguments. |
| [URL](http://docs.google.com/java/net/URL.html) | [**getCodeBase**](http://docs.google.com/java/applet/Applet.html#getCodeBase())()            Gets the base URL. |
| [URL](http://docs.google.com/java/net/URL.html) | [**getDocumentBase**](http://docs.google.com/java/applet/Applet.html#getDocumentBase())()            Gets the URL of the document in which this applet is embedded. |
| [Image](http://docs.google.com/java/awt/Image.html) | [**getImage**](http://docs.google.com/java/applet/Applet.html#getImage(java.net.URL))([URL](http://docs.google.com/java/net/URL.html) url)            Returns an Image object that can then be painted on the screen. |
| [Image](http://docs.google.com/java/awt/Image.html) | [**getImage**](http://docs.google.com/java/applet/Applet.html#getImage(java.net.URL,%20java.lang.String))([URL](http://docs.google.com/java/net/URL.html) url, [String](http://docs.google.com/java/lang/String.html) name)            Returns an Image object that can then be painted on the screen. |
| [Locale](http://docs.google.com/java/util/Locale.html) | [**getLocale**](http://docs.google.com/java/applet/Applet.html#getLocale())()            Gets the locale of the applet. |
| [String](http://docs.google.com/java/lang/String.html) | [**getParameter**](http://docs.google.com/java/applet/Applet.html#getParameter(java.lang.String))([String](http://docs.google.com/java/lang/String.html) name)            Returns the value of the named parameter in the HTML tag. |
| [String](http://docs.google.com/java/lang/String.html)[][] | [**getParameterInfo**](http://docs.google.com/java/applet/Applet.html#getParameterInfo())()            Returns information about the parameters that are understood by this applet. |
| void | [**init**](http://docs.google.com/java/applet/Applet.html#init())()            Called by the browser or applet viewer to inform this applet that it has been loaded into the system. |
| boolean | [**isActive**](http://docs.google.com/java/applet/Applet.html#isActive())()            Determines if this applet is active. |
| static [AudioClip](http://docs.google.com/java/applet/AudioClip.html) | [**newAudioClip**](http://docs.google.com/java/applet/Applet.html#newAudioClip(java.net.URL))([URL](http://docs.google.com/java/net/URL.html) url)            Get an audio clip from the given URL. |
| void | [**play**](http://docs.google.com/java/applet/Applet.html#play(java.net.URL))([URL](http://docs.google.com/java/net/URL.html) url)            Plays the audio clip at the specified absolute URL. |
| void | [**play**](http://docs.google.com/java/applet/Applet.html#play(java.net.URL,%20java.lang.String))([URL](http://docs.google.com/java/net/URL.html) url, [String](http://docs.google.com/java/lang/String.html) name)            Plays the audio clip given the URL and a specifier that is relative to it. |
| void | [**resize**](http://docs.google.com/java/applet/Applet.html#resize(java.awt.Dimension))([Dimension](http://docs.google.com/java/awt/Dimension.html) d)            Requests that this applet be resized. |
| void | [**resize**](http://docs.google.com/java/applet/Applet.html#resize(int,%20int))(int width, int height)            Requests that this applet be resized. |
| void | [**setStub**](http://docs.google.com/java/applet/Applet.html#setStub(java.applet.AppletStub))([AppletStub](http://docs.google.com/java/applet/AppletStub.html) stub)            Sets this applet's stub. |
| void | [**showStatus**](http://docs.google.com/java/applet/Applet.html#showStatus(java.lang.String))([String](http://docs.google.com/java/lang/String.html) msg)            Requests that the argument string be displayed in the "status window". |
| void | [**start**](http://docs.google.com/java/applet/Applet.html#start())()            Called by the browser or applet viewer to inform this applet that it should start its execution. |
| void | [**stop**](http://docs.google.com/java/applet/Applet.html#stop())()            Called by the browser or applet viewer to inform this applet that it should stop its execution. |

| **Methods inherited from class java.awt.**[**Panel**](http://docs.google.com/java/awt/Panel.html) |
| --- |
| [addNotify](http://docs.google.com/java/awt/Panel.html#addNotify()) |

| **Methods inherited from class java.awt.**[**Container**](http://docs.google.com/java/awt/Container.html) |
| --- |
| [add](http://docs.google.com/java/awt/Container.html#add(java.awt.Component)), [add](http://docs.google.com/java/awt/Container.html#add(java.awt.Component,%20int)), [add](http://docs.google.com/java/awt/Container.html#add(java.awt.Component,%20java.lang.Object)), [add](http://docs.google.com/java/awt/Container.html#add(java.awt.Component,%20java.lang.Object,%20int)), [add](http://docs.google.com/java/awt/Container.html#add(java.lang.String,%20java.awt.Component)), [addContainerListener](http://docs.google.com/java/awt/Container.html#addContainerListener(java.awt.event.ContainerListener)), [addImpl](http://docs.google.com/java/awt/Container.html#addImpl(java.awt.Component,%20java.lang.Object,%20int)), [addPropertyChangeListener](http://docs.google.com/java/awt/Container.html#addPropertyChangeListener(java.beans.PropertyChangeListener)), [addPropertyChangeListener](http://docs.google.com/java/awt/Container.html#addPropertyChangeListener(java.lang.String,%20java.beans.PropertyChangeListener)), [applyComponentOrientation](http://docs.google.com/java/awt/Container.html#applyComponentOrientation(java.awt.ComponentOrientation)), [areFocusTraversalKeysSet](http://docs.google.com/java/awt/Container.html#areFocusTraversalKeysSet(int)), [countComponents](http://docs.google.com/java/awt/Container.html#countComponents()), [deliverEvent](http://docs.google.com/java/awt/Container.html#deliverEvent(java.awt.Event)), [doLayout](http://docs.google.com/java/awt/Container.html#doLayout()), [findComponentAt](http://docs.google.com/java/awt/Container.html#findComponentAt(int,%20int)), [findComponentAt](http://docs.google.com/java/awt/Container.html#findComponentAt(java.awt.Point)), [getAlignmentX](http://docs.google.com/java/awt/Container.html#getAlignmentX()), [getAlignmentY](http://docs.google.com/java/awt/Container.html#getAlignmentY()), [getComponent](http://docs.google.com/java/awt/Container.html#getComponent(int)), [getComponentAt](http://docs.google.com/java/awt/Container.html#getComponentAt(int,%20int)), [getComponentAt](http://docs.google.com/java/awt/Container.html#getComponentAt(java.awt.Point)), [getComponentCount](http://docs.google.com/java/awt/Container.html#getComponentCount()), [getComponents](http://docs.google.com/java/awt/Container.html#getComponents()), [getComponentZOrder](http://docs.google.com/java/awt/Container.html#getComponentZOrder(java.awt.Component)), [getContainerListeners](http://docs.google.com/java/awt/Container.html#getContainerListeners()), [getFocusTraversalKeys](http://docs.google.com/java/awt/Container.html#getFocusTraversalKeys(int)), [getFocusTraversalPolicy](http://docs.google.com/java/awt/Container.html#getFocusTraversalPolicy()), [getInsets](http://docs.google.com/java/awt/Container.html#getInsets()), [getLayout](http://docs.google.com/java/awt/Container.html#getLayout()), [getListeners](http://docs.google.com/java/awt/Container.html#getListeners(java.lang.Class)), [getMaximumSize](http://docs.google.com/java/awt/Container.html#getMaximumSize()), [getMinimumSize](http://docs.google.com/java/awt/Container.html#getMinimumSize()), [getMousePosition](http://docs.google.com/java/awt/Container.html#getMousePosition(boolean)), [getPreferredSize](http://docs.google.com/java/awt/Container.html#getPreferredSize()), [insets](http://docs.google.com/java/awt/Container.html#insets()), [invalidate](http://docs.google.com/java/awt/Container.html#invalidate()), [isAncestorOf](http://docs.google.com/java/awt/Container.html#isAncestorOf(java.awt.Component)), [isFocusCycleRoot](http://docs.google.com/java/awt/Container.html#isFocusCycleRoot()), [isFocusCycleRoot](http://docs.google.com/java/awt/Container.html#isFocusCycleRoot(java.awt.Container)), [isFocusTraversalPolicyProvider](http://docs.google.com/java/awt/Container.html#isFocusTraversalPolicyProvider()), [isFocusTraversalPolicySet](http://docs.google.com/java/awt/Container.html#isFocusTraversalPolicySet()), [layout](http://docs.google.com/java/awt/Container.html#layout()), [list](http://docs.google.com/java/awt/Container.html#list(java.io.PrintStream,%20int)), [list](http://docs.google.com/java/awt/Container.html#list(java.io.PrintWriter,%20int)), [locate](http://docs.google.com/java/awt/Container.html#locate(int,%20int)), [minimumSize](http://docs.google.com/java/awt/Container.html#minimumSize()), [paint](http://docs.google.com/java/awt/Container.html#paint(java.awt.Graphics)), [paintComponents](http://docs.google.com/java/awt/Container.html#paintComponents(java.awt.Graphics)), [paramString](http://docs.google.com/java/awt/Container.html#paramString()), [preferredSize](http://docs.google.com/java/awt/Container.html#preferredSize()), [print](http://docs.google.com/java/awt/Container.html#print(java.awt.Graphics)), [printComponents](http://docs.google.com/java/awt/Container.html#printComponents(java.awt.Graphics)), [processContainerEvent](http://docs.google.com/java/awt/Container.html#processContainerEvent(java.awt.event.ContainerEvent)), [processEvent](http://docs.google.com/java/awt/Container.html#processEvent(java.awt.AWTEvent)), [remove](http://docs.google.com/java/awt/Container.html#remove(java.awt.Component)), [remove](http://docs.google.com/java/awt/Container.html#remove(int)), [removeAll](http://docs.google.com/java/awt/Container.html#removeAll()), [removeContainerListener](http://docs.google.com/java/awt/Container.html#removeContainerListener(java.awt.event.ContainerListener)), [removeNotify](http://docs.google.com/java/awt/Container.html#removeNotify()), [setComponentZOrder](http://docs.google.com/java/awt/Container.html#setComponentZOrder(java.awt.Component,%20int)), [setFocusCycleRoot](http://docs.google.com/java/awt/Container.html#setFocusCycleRoot(boolean)), [setFocusTraversalKeys](http://docs.google.com/java/awt/Container.html#setFocusTraversalKeys(int,%20java.util.Set)), [setFocusTraversalPolicy](http://docs.google.com/java/awt/Container.html#setFocusTraversalPolicy(java.awt.FocusTraversalPolicy)), [setFocusTraversalPolicyProvider](http://docs.google.com/java/awt/Container.html#setFocusTraversalPolicyProvider(boolean)), [setFont](http://docs.google.com/java/awt/Container.html#setFont(java.awt.Font)), [setLayout](http://docs.google.com/java/awt/Container.html#setLayout(java.awt.LayoutManager)), [transferFocusBackward](http://docs.google.com/java/awt/Container.html#transferFocusBackward()), [transferFocusDownCycle](http://docs.google.com/java/awt/Container.html#transferFocusDownCycle()), [update](http://docs.google.com/java/awt/Container.html#update(java.awt.Graphics)), [validate](http://docs.google.com/java/awt/Container.html#validate()), [validateTree](http://docs.google.com/java/awt/Container.html#validateTree()) |

| **Methods inherited from class java.awt.**[**Component**](http://docs.google.com/java/awt/Component.html) |
| --- |
| [action](http://docs.google.com/java/awt/Component.html#action(java.awt.Event,%20java.lang.Object)), [add](http://docs.google.com/java/awt/Component.html#add(java.awt.PopupMenu)), [addComponentListener](http://docs.google.com/java/awt/Component.html#addComponentListener(java.awt.event.ComponentListener)), [addFocusListener](http://docs.google.com/java/awt/Component.html#addFocusListener(java.awt.event.FocusListener)), [addHierarchyBoundsListener](http://docs.google.com/java/awt/Component.html#addHierarchyBoundsListener(java.awt.event.HierarchyBoundsListener)), [addHierarchyListener](http://docs.google.com/java/awt/Component.html#addHierarchyListener(java.awt.event.HierarchyListener)), [addInputMethodListener](http://docs.google.com/java/awt/Component.html#addInputMethodListener(java.awt.event.InputMethodListener)), [addKeyListener](http://docs.google.com/java/awt/Component.html#addKeyListener(java.awt.event.KeyListener)), [addMouseListener](http://docs.google.com/java/awt/Component.html#addMouseListener(java.awt.event.MouseListener)), [addMouseMotionListener](http://docs.google.com/java/awt/Component.html#addMouseMotionListener(java.awt.event.MouseMotionListener)), [addMouseWheelListener](http://docs.google.com/java/awt/Component.html#addMouseWheelListener(java.awt.event.MouseWheelListener)), [bounds](http://docs.google.com/java/awt/Component.html#bounds()), [checkImage](http://docs.google.com/java/awt/Component.html#checkImage(java.awt.Image,%20java.awt.image.ImageObserver)), [checkImage](http://docs.google.com/java/awt/Component.html#checkImage(java.awt.Image,%20int,%20int,%20java.awt.image.ImageObserver)), [coalesceEvents](http://docs.google.com/java/awt/Component.html#coalesceEvents(java.awt.AWTEvent,%20java.awt.AWTEvent)), [contains](http://docs.google.com/java/awt/Component.html#contains(int,%20int)), [contains](http://docs.google.com/java/awt/Component.html#contains(java.awt.Point)), [createImage](http://docs.google.com/java/awt/Component.html#createImage(java.awt.image.ImageProducer)), [createImage](http://docs.google.com/java/awt/Component.html#createImage(int,%20int)), [createVolatileImage](http://docs.google.com/java/awt/Component.html#createVolatileImage(int,%20int)), [createVolatileImage](http://docs.google.com/java/awt/Component.html#createVolatileImage(int,%20int,%20java.awt.ImageCapabilities)), [disable](http://docs.google.com/java/awt/Component.html#disable()), [disableEvents](http://docs.google.com/java/awt/Component.html#disableEvents(long)), [dispatchEvent](http://docs.google.com/java/awt/Component.html#dispatchEvent(java.awt.AWTEvent)), [enable](http://docs.google.com/java/awt/Component.html#enable()), [enable](http://docs.google.com/java/awt/Component.html#enable(boolean)), [enableEvents](http://docs.google.com/java/awt/Component.html#enableEvents(long)), [enableInputMethods](http://docs.google.com/java/awt/Component.html#enableInputMethods(boolean)), [firePropertyChange](http://docs.google.com/java/awt/Component.html#firePropertyChange(java.lang.String,%20boolean,%20boolean)), [firePropertyChange](http://docs.google.com/java/awt/Component.html#firePropertyChange(java.lang.String,%20byte,%20byte)), [firePropertyChange](http://docs.google.com/java/awt/Component.html#firePropertyChange(java.lang.String,%20char,%20char)), [firePropertyChange](http://docs.google.com/java/awt/Component.html#firePropertyChange(java.lang.String,%20double,%20double)), [firePropertyChange](http://docs.google.com/java/awt/Component.html#firePropertyChange(java.lang.String,%20float,%20float)), [firePropertyChange](http://docs.google.com/java/awt/Component.html#firePropertyChange(java.lang.String,%20int,%20int)), [firePropertyChange](http://docs.google.com/java/awt/Component.html#firePropertyChange(java.lang.String,%20long,%20long)), [firePropertyChange](http://docs.google.com/java/awt/Component.html#firePropertyChange(java.lang.String,%20java.lang.Object,%20java.lang.Object)), [firePropertyChange](http://docs.google.com/java/awt/Component.html#firePropertyChange(java.lang.String,%20short,%20short)), [getBackground](http://docs.google.com/java/awt/Component.html#getBackground()), [getBaseline](http://docs.google.com/java/awt/Component.html#getBaseline(int,%20int)), [getBaselineResizeBehavior](http://docs.google.com/java/awt/Component.html#getBaselineResizeBehavior()), [getBounds](http://docs.google.com/java/awt/Component.html#getBounds()), [getBounds](http://docs.google.com/java/awt/Component.html#getBounds(java.awt.Rectangle)), [getColorModel](http://docs.google.com/java/awt/Component.html#getColorModel()), [getComponentListeners](http://docs.google.com/java/awt/Component.html#getComponentListeners()), [getComponentOrientation](http://docs.google.com/java/awt/Component.html#getComponentOrientation()), [getCursor](http://docs.google.com/java/awt/Component.html#getCursor()), [getDropTarget](http://docs.google.com/java/awt/Component.html#getDropTarget()), [getFocusCycleRootAncestor](http://docs.google.com/java/awt/Component.html#getFocusCycleRootAncestor()), [getFocusListeners](http://docs.google.com/java/awt/Component.html#getFocusListeners()), [getFocusTraversalKeysEnabled](http://docs.google.com/java/awt/Component.html#getFocusTraversalKeysEnabled()), [getFont](http://docs.google.com/java/awt/Component.html#getFont()), [getFontMetrics](http://docs.google.com/java/awt/Component.html#getFontMetrics(java.awt.Font)), [getForeground](http://docs.google.com/java/awt/Component.html#getForeground()), [getGraphics](http://docs.google.com/java/awt/Component.html#getGraphics()), [getGraphicsConfiguration](http://docs.google.com/java/awt/Component.html#getGraphicsConfiguration()), [getHeight](http://docs.google.com/java/awt/Component.html#getHeight()), [getHierarchyBoundsListeners](http://docs.google.com/java/awt/Component.html#getHierarchyBoundsListeners()), [getHierarchyListeners](http://docs.google.com/java/awt/Component.html#getHierarchyListeners()), [getIgnoreRepaint](http://docs.google.com/java/awt/Component.html#getIgnoreRepaint()), [getInputContext](http://docs.google.com/java/awt/Component.html#getInputContext()), [getInputMethodListeners](http://docs.google.com/java/awt/Component.html#getInputMethodListeners()), [getInputMethodRequests](http://docs.google.com/java/awt/Component.html#getInputMethodRequests()), [getKeyListeners](http://docs.google.com/java/awt/Component.html#getKeyListeners()), [getLocation](http://docs.google.com/java/awt/Component.html#getLocation()), [getLocation](http://docs.google.com/java/awt/Component.html#getLocation(java.awt.Point)), [getLocationOnScreen](http://docs.google.com/java/awt/Component.html#getLocationOnScreen()), [getMouseListeners](http://docs.google.com/java/awt/Component.html#getMouseListeners()), [getMouseMotionListeners](http://docs.google.com/java/awt/Component.html#getMouseMotionListeners()), [getMousePosition](http://docs.google.com/java/awt/Component.html#getMousePosition()), [getMouseWheelListeners](http://docs.google.com/java/awt/Component.html#getMouseWheelListeners()), [getName](http://docs.google.com/java/awt/Component.html#getName()), [getParent](http://docs.google.com/java/awt/Component.html#getParent()), [getPeer](http://docs.google.com/java/awt/Component.html#getPeer()), [getPropertyChangeListeners](http://docs.google.com/java/awt/Component.html#getPropertyChangeListeners()), [getPropertyChangeListeners](http://docs.google.com/java/awt/Component.html#getPropertyChangeListeners(java.lang.String)), [getSize](http://docs.google.com/java/awt/Component.html#getSize()), [getSize](http://docs.google.com/java/awt/Component.html#getSize(java.awt.Dimension)), [getToolkit](http://docs.google.com/java/awt/Component.html#getToolkit()), [getTreeLock](http://docs.google.com/java/awt/Component.html#getTreeLock()), [getWidth](http://docs.google.com/java/awt/Component.html#getWidth()), [getX](http://docs.google.com/java/awt/Component.html#getX()), [getY](http://docs.google.com/java/awt/Component.html#getY()), [gotFocus](http://docs.google.com/java/awt/Component.html#gotFocus(java.awt.Event,%20java.lang.Object)), [handleEvent](http://docs.google.com/java/awt/Component.html#handleEvent(java.awt.Event)), [hasFocus](http://docs.google.com/java/awt/Component.html#hasFocus()), [hide](http://docs.google.com/java/awt/Component.html#hide()), [imageUpdate](http://docs.google.com/java/awt/Component.html#imageUpdate(java.awt.Image,%20int,%20int,%20int,%20int,%20int)), [inside](http://docs.google.com/java/awt/Component.html#inside(int,%20int)), [isBackgroundSet](http://docs.google.com/java/awt/Component.html#isBackgroundSet()), [isCursorSet](http://docs.google.com/java/awt/Component.html#isCursorSet()), [isDisplayable](http://docs.google.com/java/awt/Component.html#isDisplayable()), [isDoubleBuffered](http://docs.google.com/java/awt/Component.html#isDoubleBuffered()), [isEnabled](http://docs.google.com/java/awt/Component.html#isEnabled()), [isFocusable](http://docs.google.com/java/awt/Component.html#isFocusable()), [isFocusOwner](http://docs.google.com/java/awt/Component.html#isFocusOwner()), [isFocusTraversable](http://docs.google.com/java/awt/Component.html#isFocusTraversable()), [isFontSet](http://docs.google.com/java/awt/Component.html#isFontSet()), [isForegroundSet](http://docs.google.com/java/awt/Component.html#isForegroundSet()), [isLightweight](http://docs.google.com/java/awt/Component.html#isLightweight()), [isMaximumSizeSet](http://docs.google.com/java/awt/Component.html#isMaximumSizeSet()), [isMinimumSizeSet](http://docs.google.com/java/awt/Component.html#isMinimumSizeSet()), [isOpaque](http://docs.google.com/java/awt/Component.html#isOpaque()), [isPreferredSizeSet](http://docs.google.com/java/awt/Component.html#isPreferredSizeSet()), [isShowing](http://docs.google.com/java/awt/Component.html#isShowing()), [isValid](http://docs.google.com/java/awt/Component.html#isValid()), [isVisible](http://docs.google.com/java/awt/Component.html#isVisible()), [keyDown](http://docs.google.com/java/awt/Component.html#keyDown(java.awt.Event,%20int)), [keyUp](http://docs.google.com/java/awt/Component.html#keyUp(java.awt.Event,%20int)), [list](http://docs.google.com/java/awt/Component.html#list()), [list](http://docs.google.com/java/awt/Component.html#list(java.io.PrintStream)), [list](http://docs.google.com/java/awt/Component.html#list(java.io.PrintWriter)), [location](http://docs.google.com/java/awt/Component.html#location()), [lostFocus](http://docs.google.com/java/awt/Component.html#lostFocus(java.awt.Event,%20java.lang.Object)), [mouseDown](http://docs.google.com/java/awt/Component.html#mouseDown(java.awt.Event,%20int,%20int)), [mouseDrag](http://docs.google.com/java/awt/Component.html#mouseDrag(java.awt.Event,%20int,%20int)), [mouseEnter](http://docs.google.com/java/awt/Component.html#mouseEnter(java.awt.Event,%20int,%20int)), [mouseExit](http://docs.google.com/java/awt/Component.html#mouseExit(java.awt.Event,%20int,%20int)), [mouseMove](http://docs.google.com/java/awt/Component.html#mouseMove(java.awt.Event,%20int,%20int)), [mouseUp](http://docs.google.com/java/awt/Component.html#mouseUp(java.awt.Event,%20int,%20int)), [move](http://docs.google.com/java/awt/Component.html#move(int,%20int)), [nextFocus](http://docs.google.com/java/awt/Component.html#nextFocus()), [paintAll](http://docs.google.com/java/awt/Component.html#paintAll(java.awt.Graphics)), [postEvent](http://docs.google.com/java/awt/Component.html#postEvent(java.awt.Event)), [prepareImage](http://docs.google.com/java/awt/Component.html#prepareImage(java.awt.Image,%20java.awt.image.ImageObserver)), [prepareImage](http://docs.google.com/java/awt/Component.html#prepareImage(java.awt.Image,%20int,%20int,%20java.awt.image.ImageObserver)), [printAll](http://docs.google.com/java/awt/Component.html#printAll(java.awt.Graphics)), [processComponentEvent](http://docs.google.com/java/awt/Component.html#processComponentEvent(java.awt.event.ComponentEvent)), [processFocusEvent](http://docs.google.com/java/awt/Component.html#processFocusEvent(java.awt.event.FocusEvent)), [processHierarchyBoundsEvent](http://docs.google.com/java/awt/Component.html#processHierarchyBoundsEvent(java.awt.event.HierarchyEvent)), [processHierarchyEvent](http://docs.google.com/java/awt/Component.html#processHierarchyEvent(java.awt.event.HierarchyEvent)), [processInputMethodEvent](http://docs.google.com/java/awt/Component.html#processInputMethodEvent(java.awt.event.InputMethodEvent)), [processKeyEvent](http://docs.google.com/java/awt/Component.html#processKeyEvent(java.awt.event.KeyEvent)), [processMouseEvent](http://docs.google.com/java/awt/Component.html#processMouseEvent(java.awt.event.MouseEvent)), [processMouseMotionEvent](http://docs.google.com/java/awt/Component.html#processMouseMotionEvent(java.awt.event.MouseEvent)), [processMouseWheelEvent](http://docs.google.com/java/awt/Component.html#processMouseWheelEvent(java.awt.event.MouseWheelEvent)), [remove](http://docs.google.com/java/awt/Component.html#remove(java.awt.MenuComponent)), [removeComponentListener](http://docs.google.com/java/awt/Component.html#removeComponentListener(java.awt.event.ComponentListener)), [removeFocusListener](http://docs.google.com/java/awt/Component.html#removeFocusListener(java.awt.event.FocusListener)), [removeHierarchyBoundsListener](http://docs.google.com/java/awt/Component.html#removeHierarchyBoundsListener(java.awt.event.HierarchyBoundsListener)), [removeHierarchyListener](http://docs.google.com/java/awt/Component.html#removeHierarchyListener(java.awt.event.HierarchyListener)), [removeInputMethodListener](http://docs.google.com/java/awt/Component.html#removeInputMethodListener(java.awt.event.InputMethodListener)), [removeKeyListener](http://docs.google.com/java/awt/Component.html#removeKeyListener(java.awt.event.KeyListener)), [removeMouseListener](http://docs.google.com/java/awt/Component.html#removeMouseListener(java.awt.event.MouseListener)), [removeMouseMotionListener](http://docs.google.com/java/awt/Component.html#removeMouseMotionListener(java.awt.event.MouseMotionListener)), [removeMouseWheelListener](http://docs.google.com/java/awt/Component.html#removeMouseWheelListener(java.awt.event.MouseWheelListener)), [removePropertyChangeListener](http://docs.google.com/java/awt/Component.html#removePropertyChangeListener(java.beans.PropertyChangeListener)), [removePropertyChangeListener](http://docs.google.com/java/awt/Component.html#removePropertyChangeListener(java.lang.String,%20java.beans.PropertyChangeListener)), [repaint](http://docs.google.com/java/awt/Component.html#repaint()), [repaint](http://docs.google.com/java/awt/Component.html#repaint(int,%20int,%20int,%20int)), [repaint](http://docs.google.com/java/awt/Component.html#repaint(long)), [repaint](http://docs.google.com/java/awt/Component.html#repaint(long,%20int,%20int,%20int,%20int)), [requestFocus](http://docs.google.com/java/awt/Component.html#requestFocus()), [requestFocus](http://docs.google.com/java/awt/Component.html#requestFocus(boolean)), [requestFocusInWindow](http://docs.google.com/java/awt/Component.html#requestFocusInWindow()), [requestFocusInWindow](http://docs.google.com/java/awt/Component.html#requestFocusInWindow(boolean)), [reshape](http://docs.google.com/java/awt/Component.html#reshape(int,%20int,%20int,%20int)), [setBackground](http://docs.google.com/java/awt/Component.html#setBackground(java.awt.Color)), [setBounds](http://docs.google.com/java/awt/Component.html#setBounds(int,%20int,%20int,%20int)), [setBounds](http://docs.google.com/java/awt/Component.html#setBounds(java.awt.Rectangle)), [setComponentOrientation](http://docs.google.com/java/awt/Component.html#setComponentOrientation(java.awt.ComponentOrientation)), [setCursor](http://docs.google.com/java/awt/Component.html#setCursor(java.awt.Cursor)), [setDropTarget](http://docs.google.com/java/awt/Component.html#setDropTarget(java.awt.dnd.DropTarget)), [setEnabled](http://docs.google.com/java/awt/Component.html#setEnabled(boolean)), [setFocusable](http://docs.google.com/java/awt/Component.html#setFocusable(boolean)), [setFocusTraversalKeysEnabled](http://docs.google.com/java/awt/Component.html#setFocusTraversalKeysEnabled(boolean)), [setForeground](http://docs.google.com/java/awt/Component.html#setForeground(java.awt.Color)), [setIgnoreRepaint](http://docs.google.com/java/awt/Component.html#setIgnoreRepaint(boolean)), [setLocale](http://docs.google.com/java/awt/Component.html#setLocale(java.util.Locale)), [setLocation](http://docs.google.com/java/awt/Component.html#setLocation(int,%20int)), [setLocation](http://docs.google.com/java/awt/Component.html#setLocation(java.awt.Point)), [setMaximumSize](http://docs.google.com/java/awt/Component.html#setMaximumSize(java.awt.Dimension)), [setMinimumSize](http://docs.google.com/java/awt/Component.html#setMinimumSize(java.awt.Dimension)), [setName](http://docs.google.com/java/awt/Component.html#setName(java.lang.String)), [setPreferredSize](http://docs.google.com/java/awt/Component.html#setPreferredSize(java.awt.Dimension)), [setSize](http://docs.google.com/java/awt/Component.html#setSize(java.awt.Dimension)), [setSize](http://docs.google.com/java/awt/Component.html#setSize(int,%20int)), [setVisible](http://docs.google.com/java/awt/Component.html#setVisible(boolean)), [show](http://docs.google.com/java/awt/Component.html#show()), [show](http://docs.google.com/java/awt/Component.html#show(boolean)), [size](http://docs.google.com/java/awt/Component.html#size()), [toString](http://docs.google.com/java/awt/Component.html#toString()), [transferFocus](http://docs.google.com/java/awt/Component.html#transferFocus()), [transferFocusUpCycle](http://docs.google.com/java/awt/Component.html#transferFocusUpCycle()) |

| **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()), [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)) |

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

### Applet

public **Applet**()  
 throws [HeadlessException](http://docs.google.com/java/awt/HeadlessException.html)

Constructs a new Applet.

Note: Many methods in java.applet.Applet may be invoked by the applet only after the applet is fully constructed; applet should avoid calling methods in java.applet.Applet in the constructor.

**Throws:** [HeadlessException](http://docs.google.com/java/awt/HeadlessException.html) - if GraphicsEnvironment.isHeadless() returns true.**Since:** 1.4 **See Also:**[GraphicsEnvironment.isHeadless()](http://docs.google.com/java/awt/GraphicsEnvironment.html#isHeadless())

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

### setStub

public final void **setStub**([AppletStub](http://docs.google.com/java/applet/AppletStub.html) stub)

Sets this applet's stub. This is done automatically by the system.

If there is a security manager, its checkPermission method is called with the AWTPermission("setAppletStub") permission if a stub has already been set.

**Parameters:**stub - the new stub. **Throws:** [SecurityException](http://docs.google.com/java/lang/SecurityException.html) - if the caller cannot set the stub

### isActive

public boolean **isActive**()

Determines if this applet is active. An applet is marked active just before its start method is called. It becomes inactive just before its stop method is called.

**Returns:**true if the applet is active; false otherwise.**See Also:**[start()](http://docs.google.com/java/applet/Applet.html#start()), [stop()](http://docs.google.com/java/applet/Applet.html#stop())

### getDocumentBase

public [URL](http://docs.google.com/java/net/URL.html) **getDocumentBase**()

Gets the URL of the document in which this applet is embedded. For example, suppose an applet is contained within the document:

http://java.sun.com/products/jdk/1.2/index.html

The document base is:

http://java.sun.com/products/jdk/1.2/index.html

**Returns:**the [URL](http://docs.google.com/java/net/URL.html) of the document that contains this applet.**See Also:**[getCodeBase()](http://docs.google.com/java/applet/Applet.html#getCodeBase())

### getCodeBase

public [URL](http://docs.google.com/java/net/URL.html) **getCodeBase**()

Gets the base URL. This is the URL of the directory which contains this applet.

**Returns:**the base [URL](http://docs.google.com/java/net/URL.html) of the directory which contains this applet.**See Also:**[getDocumentBase()](http://docs.google.com/java/applet/Applet.html#getDocumentBase())

### getParameter

public [String](http://docs.google.com/java/lang/String.html) **getParameter**([String](http://docs.google.com/java/lang/String.html) name)

Returns the value of the named parameter in the HTML tag. For example, if this applet is specified as

<applet code="Clock" width=50 height=50>  
 <param name=Color value="blue">  
 </applet>

then a call to getParameter("Color") returns the value "blue".

The name argument is case insensitive.

**Parameters:**name - a parameter name. **Returns:**the value of the named parameter, or null if not set.

### getAppletContext

public [AppletContext](http://docs.google.com/java/applet/AppletContext.html) **getAppletContext**()

Determines this applet's context, which allows the applet to query and affect the environment in which it runs.

This environment of an applet represents the document that contains the applet.

**Returns:**the applet's context.

### resize

public void **resize**(int width,  
 int height)

Requests that this applet be resized.

**Overrides:**[resize](http://docs.google.com/java/awt/Component.html#resize(int,%20int)) in class [Component](http://docs.google.com/java/awt/Component.html) **Parameters:**width - the new requested width for the applet.height - the new requested height for the applet.

### resize

public void **resize**([Dimension](http://docs.google.com/java/awt/Dimension.html) d)

Requests that this applet be resized.

**Overrides:**[resize](http://docs.google.com/java/awt/Component.html#resize(java.awt.Dimension)) in class [Component](http://docs.google.com/java/awt/Component.html) **Parameters:**d - an object giving the new width and height.

### showStatus

public void **showStatus**([String](http://docs.google.com/java/lang/String.html) msg)

Requests that the argument string be displayed in the "status window". Many browsers and applet viewers provide such a window, where the application can inform users of its current state.

**Parameters:**msg - a string to display in the status window.

### getImage

public [Image](http://docs.google.com/java/awt/Image.html) **getImage**([URL](http://docs.google.com/java/net/URL.html) url)

Returns an Image object that can then be painted on the screen. The url that is passed as an argument must specify an absolute URL.

This method always returns immediately, whether or not the image exists. When this applet attempts to draw the image on the screen, the data will be loaded. The graphics primitives that draw the image will incrementally paint on the screen.

**Parameters:**url - an absolute URL giving the location of the image. **Returns:**the image at the specified URL.**See Also:**[Image](http://docs.google.com/java/awt/Image.html)

### getImage

public [Image](http://docs.google.com/java/awt/Image.html) **getImage**([URL](http://docs.google.com/java/net/URL.html) url,  
 [String](http://docs.google.com/java/lang/String.html) name)

Returns an Image object that can then be painted on the screen. The url argument must specify an absolute URL. The name argument is a specifier that is relative to the url argument.

This method always returns immediately, whether or not the image exists. When this applet attempts to draw the image on the screen, the data will be loaded. The graphics primitives that draw the image will incrementally paint on the screen.

**Parameters:**url - an absolute URL giving the base location of the image.name - the location of the image, relative to the url argument. **Returns:**the image at the specified URL.**See Also:**[Image](http://docs.google.com/java/awt/Image.html)

### newAudioClip

public static final [AudioClip](http://docs.google.com/java/applet/AudioClip.html) **newAudioClip**([URL](http://docs.google.com/java/net/URL.html) url)

Get an audio clip from the given URL.

**Parameters:**url - points to the audio clip **Returns:**the audio clip at the specified URL.**Since:** 1.2

### getAudioClip

public [AudioClip](http://docs.google.com/java/applet/AudioClip.html) **getAudioClip**([URL](http://docs.google.com/java/net/URL.html) url)

Returns the AudioClip object specified by the URL argument.

This method always returns immediately, whether or not the audio clip exists. When this applet attempts to play the audio clip, the data will be loaded.

**Parameters:**url - an absolute URL giving the location of the audio clip. **Returns:**the audio clip at the specified URL.**See Also:**[AudioClip](http://docs.google.com/java/applet/AudioClip.html)

### getAudioClip

public [AudioClip](http://docs.google.com/java/applet/AudioClip.html) **getAudioClip**([URL](http://docs.google.com/java/net/URL.html) url,  
 [String](http://docs.google.com/java/lang/String.html) name)

Returns the AudioClip object specified by the URL and name arguments.

This method always returns immediately, whether or not the audio clip exists. When this applet attempts to play the audio clip, the data will be loaded.

**Parameters:**url - an absolute URL giving the base location of the audio clip.name - the location of the audio clip, relative to the url argument. **Returns:**the audio clip at the specified URL.**See Also:**[AudioClip](http://docs.google.com/java/applet/AudioClip.html)

### getAppletInfo

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

Returns information about this applet. An applet should override this method to return a String containing information about the author, version, and copyright of the applet.

The implementation of this method provided by the Applet class returns null.

**Returns:**a string containing information about the author, version, and copyright of the applet.

### getLocale

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

Gets the locale of the applet. It allows the applet to maintain its own locale separated from the locale of the browser or appletviewer.

**Overrides:**[getLocale](http://docs.google.com/java/awt/Component.html#getLocale()) in class [Component](http://docs.google.com/java/awt/Component.html) **Returns:**the locale of the applet; if no locale has been set, the default locale is returned.**Since:** JDK1.1 **See Also:**[Component.setLocale(java.util.Locale)](http://docs.google.com/java/awt/Component.html#setLocale(java.util.Locale))

### getParameterInfo

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

Returns information about the parameters that are understood by this applet. An applet should override this method to return an array of Strings describing these parameters.

Each element of the array should be a set of three Strings containing the name, the type, and a description. For example:

String pinfo[][] = {  
 {"fps", "1-10", "frames per second"},  
 {"repeat", "boolean", "repeat image loop"},  
 {"imgs", "url", "images directory"}  
 };

The implementation of this method provided by the Applet class returns null.

**Returns:**an array describing the parameters this applet looks for.

### play

public void **play**([URL](http://docs.google.com/java/net/URL.html) url)

Plays the audio clip at the specified absolute URL. Nothing happens if the audio clip cannot be found.

**Parameters:**url - an absolute URL giving the location of the audio clip.

### play

public void **play**([URL](http://docs.google.com/java/net/URL.html) url,  
 [String](http://docs.google.com/java/lang/String.html) name)

Plays the audio clip given the URL and a specifier that is relative to it. Nothing happens if the audio clip cannot be found.

**Parameters:**url - an absolute URL giving the base location of the audio clip.name - the location of the audio clip, relative to the url argument.

### init

public void **init**()

Called by the browser or applet viewer to inform this applet that it has been loaded into the system. It is always called before the first time that the start method is called.

A subclass of Applet should override this method if it has initialization to perform. For example, an applet with threads would use the init method to create the threads and the destroy method to kill them.

The implementation of this method provided by the Applet class does nothing.

**See Also:**[destroy()](http://docs.google.com/java/applet/Applet.html#destroy()), [start()](http://docs.google.com/java/applet/Applet.html#start()), [stop()](http://docs.google.com/java/applet/Applet.html#stop())

### start

public void **start**()

Called by the browser or applet viewer to inform this applet that it should start its execution. It is called after the init method and each time the applet is revisited in a Web page.

A subclass of Applet should override this method if it has any operation that it wants to perform each time the Web page containing it is visited. For example, an applet with animation might want to use the start method to resume animation, and the stop method to suspend the animation.

Note: some methods, such as getLocationOnScreen, can only provide meaningful results if the applet is showing. Because isShowing returns false when the applet's start is first called, methods requiring isShowing to return true should be called from a ComponentListener.

The implementation of this method provided by the Applet class does nothing.

**See Also:**[destroy()](http://docs.google.com/java/applet/Applet.html#destroy()), [init()](http://docs.google.com/java/applet/Applet.html#init()), [stop()](http://docs.google.com/java/applet/Applet.html#stop()), [Component.isShowing()](http://docs.google.com/java/awt/Component.html#isShowing()), [ComponentListener.componentShown(java.awt.event.ComponentEvent)](http://docs.google.com/java/awt/event/ComponentListener.html#componentShown(java.awt.event.ComponentEvent))

### stop

public void **stop**()

Called by the browser or applet viewer to inform this applet that it should stop its execution. It is called when the Web page that contains this applet has been replaced by another page, and also just before the applet is to be destroyed.

A subclass of Applet should override this method if it has any operation that it wants to perform each time the Web page containing it is no longer visible. For example, an applet with animation might want to use the start method to resume animation, and the stop method to suspend the animation.

The implementation of this method provided by the Applet class does nothing.

**See Also:**[destroy()](http://docs.google.com/java/applet/Applet.html#destroy()), [init()](http://docs.google.com/java/applet/Applet.html#init())

### destroy

public void **destroy**()

Called by the browser or applet viewer to inform this applet that it is being reclaimed and that it should destroy any resources that it has allocated. The stop method will always be called before destroy.

A subclass of Applet should override this method if it has any operation that it wants to perform before it is destroyed. For example, an applet with threads would use the init method to create the threads and the destroy method to kill them.

The implementation of this method provided by the Applet class does nothing.

**See Also:**[init()](http://docs.google.com/java/applet/Applet.html#init()), [start()](http://docs.google.com/java/applet/Applet.html#start()), [stop()](http://docs.google.com/java/applet/Applet.html#stop())

### getAccessibleContext

public [AccessibleContext](http://docs.google.com/javax/accessibility/AccessibleContext.html) **getAccessibleContext**()

Gets the AccessibleContext associated with this Applet. For applets, the AccessibleContext takes the form of an AccessibleApplet. A new AccessibleApplet instance is created if necessary.

**Specified by:**[getAccessibleContext](http://docs.google.com/javax/accessibility/Accessible.html#getAccessibleContext()) in interface [Accessible](http://docs.google.com/javax/accessibility/Accessible.html)**Overrides:**[getAccessibleContext](http://docs.google.com/java/awt/Panel.html#getAccessibleContext()) in class [Panel](http://docs.google.com/java/awt/Panel.html) **Returns:**an AccessibleApplet that serves as the AccessibleContext of this Applet**Since:** 1.3

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/Applet.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   [**NEXT CLASS**](http://docs.google.com/java/applet/Applet.AccessibleApplet.html) | [**FRAMES**](http://docs.google.com/index.html?java/applet/Applet.html)    [**NO FRAMES**](http://docs.google.com/Applet.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: [NESTED](#3znysh7) | [FIELD](#4d34og8) | [CONSTR](#17dp8vu) | [METHOD](#3rdcrjn) | DETAIL: FIELD | [CONSTR](#44sinio) | [METHOD](#z337ya) |

[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).