| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | [**Class**](http://docs.google.com/java/awt/Container.html) | **Use** | [**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   NEXT | [**FRAMES**](http://docs.google.com/index.html?java/awt//class-useContainer.html)    [**NO FRAMES**](http://docs.google.com/Container.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |

**Uses of Class**

**java.awt.Container**

| Packages that use [Container](http://docs.google.com/java/awt/Container.html) | |
| --- | --- |
| [**java.applet**](#3znysh7) | Provides the classes necessary to create an applet and the classes an applet uses to communicate with its applet context. |
| [**java.awt**](#2et92p0) | Contains all of the classes for creating user interfaces and for painting graphics and images. |
| [**java.awt.event**](#tyjcwt) | Provides interfaces and classes for dealing with different types of events fired by AWT components. |
| [**java.beans.beancontext**](#3dy6vkm) | Provides classes and interfaces relating to bean context. |
| [**javax.swing**](#1t3h5sf) | Provides a set of "lightweight" (all-Java language) components that, to the maximum degree possible, work the same on all platforms. |
| [**javax.swing.colorchooser**](#2s8eyo1) | Contains classes and interfaces used by the JColorChooser component. |
| [**javax.swing.event**](#17dp8vu) | Provides for events fired by Swing components. |
| [**javax.swing.plaf.basic**](#3rdcrjn) | Provides user interface objects built according to the Basic look and feel. |
| [**javax.swing.plaf.metal**](#26in1rg) | Provides user interface objects built according to the Java look and feel (once codenamed *Metal*), which is the default look and feel. |
| [**javax.swing.table**](#lnxbz9) | Provides classes and interfaces for dealing with javax.swing.JTable. |
| [**javax.swing.text**](#35nkun2) | Provides classes and interfaces that deal with editable and noneditable text components. |
| [**javax.swing.tree**](#1ksv4uv) | Provides classes and interfaces for dealing with javax.swing.JTree. |

| Uses of [Container](http://docs.google.com/java/awt/Container.html) in [java.applet](http://docs.google.com/java/applet/package-summary.html) | |
| --- | --- |

| Subclasses of [Container](http://docs.google.com/java/awt/Container.html) in [java.applet](http://docs.google.com/java/applet/package-summary.html) | |
| --- | --- |
| class | [**Applet**](http://docs.google.com/java/applet/Applet.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. |

| Uses of [Container](http://docs.google.com/java/awt/Container.html) in [java.awt](http://docs.google.com/java/awt/package-summary.html) | |
| --- | --- |

| Subclasses of [Container](http://docs.google.com/java/awt/Container.html) in [java.awt](http://docs.google.com/java/awt/package-summary.html) | |
| --- | --- |
| class | [**Dialog**](http://docs.google.com/java/awt/Dialog.html)            A Dialog is a top-level window with a title and a border that is typically used to take some form of input from the user. |
| class | [**FileDialog**](http://docs.google.com/java/awt/FileDialog.html)            The FileDialog class displays a dialog window from which the user can select a file. |
| class | [**Frame**](http://docs.google.com/java/awt/Frame.html)            A Frame is a top-level window with a title and a border. |
| class | [**Panel**](http://docs.google.com/java/awt/Panel.html)            Panel is the simplest container class. |
| class | [**ScrollPane**](http://docs.google.com/java/awt/ScrollPane.html)            A container class which implements automatic horizontal and/or vertical scrolling for a single child component. |
| class | [**Window**](http://docs.google.com/java/awt/Window.html)            A Window object is a top-level window with no borders and no menubar. |

| Methods in [java.awt](http://docs.google.com/java/awt/package-summary.html) that return [Container](http://docs.google.com/java/awt/Container.html) | |
| --- | --- |
| [Container](http://docs.google.com/java/awt/Container.html) | **KeyboardFocusManager.**[**getCurrentFocusCycleRoot**](http://docs.google.com/java/awt/KeyboardFocusManager.html#getCurrentFocusCycleRoot())()            Returns the current focus cycle root, if the current focus cycle root is in the same context as the calling thread. |
| [Container](http://docs.google.com/java/awt/Container.html) | **Window.**[**getFocusCycleRootAncestor**](http://docs.google.com/java/awt/Window.html#getFocusCycleRootAncestor())()            Always returns null because Windows have no ancestors; they represent the top of the Component hierarchy. |
| [Container](http://docs.google.com/java/awt/Container.html) | **Component.**[**getFocusCycleRootAncestor**](http://docs.google.com/java/awt/Component.html#getFocusCycleRootAncestor())()            Returns the Container which is the focus cycle root of this Component's focus traversal cycle. |
| protected  [Container](http://docs.google.com/java/awt/Container.html) | **KeyboardFocusManager.**[**getGlobalCurrentFocusCycleRoot**](http://docs.google.com/java/awt/KeyboardFocusManager.html#getGlobalCurrentFocusCycleRoot())()            Returns the current focus cycle root, even if the calling thread is in a different context than the current focus cycle root. |
| protected static [Container](http://docs.google.com/java/awt/Container.html) | **Toolkit.**[**getNativeContainer**](http://docs.google.com/java/awt/Toolkit.html#getNativeContainer(java.awt.Component))([Component](http://docs.google.com/java/awt/Component.html) c)            Give native peers the ability to query the native container given a native component (eg the direct parent may be lightweight). |
| [Container](http://docs.google.com/java/awt/Container.html) | **Component.**[**getParent**](http://docs.google.com/java/awt/Component.html#getParent())()            Gets the parent of this component. |

| Methods in [java.awt](http://docs.google.com/java/awt/package-summary.html) with parameters of type [Container](http://docs.google.com/java/awt/Container.html) | |
| --- | --- |
| protected  void | **GridBagLayout.**[**arrangeGrid**](http://docs.google.com/java/awt/GridBagLayout.html#arrangeGrid(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent)            Lays out the grid. |
| protected  void | **GridBagLayout.**[**ArrangeGrid**](http://docs.google.com/java/awt/GridBagLayout.html#ArrangeGrid(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent)            This method is obsolete and supplied for backwards compatability only; new code should call [arrangeGrid](http://docs.google.com/java/awt/GridBagLayout.html#arrangeGrid(java.awt.Container)) instead. |
| void | **DefaultKeyboardFocusManager.**[**downFocusCycle**](http://docs.google.com/java/awt/DefaultKeyboardFocusManager.html#downFocusCycle(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) aContainer)            Moves the focus down one focus traversal cycle. |
| abstract  void | **KeyboardFocusManager.**[**downFocusCycle**](http://docs.google.com/java/awt/KeyboardFocusManager.html#downFocusCycle(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) aContainer)            Moves the focus down one focus traversal cycle. |
| void | **CardLayout.**[**first**](http://docs.google.com/java/awt/CardLayout.html#first(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent)            Flips to the first card of the container. |
| [Component](http://docs.google.com/java/awt/Component.html) | **ContainerOrderFocusTraversalPolicy.**[**getComponentAfter**](http://docs.google.com/java/awt/ContainerOrderFocusTraversalPolicy.html#getComponentAfter(java.awt.Container,%20java.awt.Component))([Container](http://docs.google.com/java/awt/Container.html) aContainer, [Component](http://docs.google.com/java/awt/Component.html) aComponent)            Returns the Component that should receive the focus after aComponent. |
| abstract  [Component](http://docs.google.com/java/awt/Component.html) | **FocusTraversalPolicy.**[**getComponentAfter**](http://docs.google.com/java/awt/FocusTraversalPolicy.html#getComponentAfter(java.awt.Container,%20java.awt.Component))([Container](http://docs.google.com/java/awt/Container.html) aContainer, [Component](http://docs.google.com/java/awt/Component.html) aComponent)            Returns the Component that should receive the focus after aComponent. |
| [Component](http://docs.google.com/java/awt/Component.html) | **ContainerOrderFocusTraversalPolicy.**[**getComponentBefore**](http://docs.google.com/java/awt/ContainerOrderFocusTraversalPolicy.html#getComponentBefore(java.awt.Container,%20java.awt.Component))([Container](http://docs.google.com/java/awt/Container.html) aContainer, [Component](http://docs.google.com/java/awt/Component.html) aComponent)            Returns the Component that should receive the focus before aComponent. |
| abstract  [Component](http://docs.google.com/java/awt/Component.html) | **FocusTraversalPolicy.**[**getComponentBefore**](http://docs.google.com/java/awt/FocusTraversalPolicy.html#getComponentBefore(java.awt.Container,%20java.awt.Component))([Container](http://docs.google.com/java/awt/Container.html) aContainer, [Component](http://docs.google.com/java/awt/Component.html) aComponent)            Returns the Component that should receive the focus before aComponent. |
| [Component](http://docs.google.com/java/awt/Component.html) | **ContainerOrderFocusTraversalPolicy.**[**getDefaultComponent**](http://docs.google.com/java/awt/ContainerOrderFocusTraversalPolicy.html#getDefaultComponent(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) aContainer)            Returns the default Component to focus. |
| abstract  [Component](http://docs.google.com/java/awt/Component.html) | **FocusTraversalPolicy.**[**getDefaultComponent**](http://docs.google.com/java/awt/FocusTraversalPolicy.html#getDefaultComponent(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) aContainer)            Returns the default Component to focus. |
| [Component](http://docs.google.com/java/awt/Component.html) | **ContainerOrderFocusTraversalPolicy.**[**getFirstComponent**](http://docs.google.com/java/awt/ContainerOrderFocusTraversalPolicy.html#getFirstComponent(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) aContainer)            Returns the first Component in the traversal cycle. |
| abstract  [Component](http://docs.google.com/java/awt/Component.html) | **FocusTraversalPolicy.**[**getFirstComponent**](http://docs.google.com/java/awt/FocusTraversalPolicy.html#getFirstComponent(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) aContainer)            Returns the first Component in the traversal cycle. |
| [Component](http://docs.google.com/java/awt/Component.html) | **ContainerOrderFocusTraversalPolicy.**[**getLastComponent**](http://docs.google.com/java/awt/ContainerOrderFocusTraversalPolicy.html#getLastComponent(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) aContainer)            Returns the last Component in the traversal cycle. |
| abstract  [Component](http://docs.google.com/java/awt/Component.html) | **FocusTraversalPolicy.**[**getLastComponent**](http://docs.google.com/java/awt/FocusTraversalPolicy.html#getLastComponent(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) aContainer)            Returns the last Component in the traversal cycle. |
| float | **GridBagLayout.**[**getLayoutAlignmentX**](http://docs.google.com/java/awt/GridBagLayout.html#getLayoutAlignmentX(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent)            Returns the alignment along the x axis. |
| float | **CardLayout.**[**getLayoutAlignmentX**](http://docs.google.com/java/awt/CardLayout.html#getLayoutAlignmentX(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent)            Returns the alignment along the x axis. |
| float | **LayoutManager2.**[**getLayoutAlignmentX**](http://docs.google.com/java/awt/LayoutManager2.html#getLayoutAlignmentX(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) target)            Returns the alignment along the x axis. |
| float | **BorderLayout.**[**getLayoutAlignmentX**](http://docs.google.com/java/awt/BorderLayout.html#getLayoutAlignmentX(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent)            Returns the alignment along the x axis. |
| float | **GridBagLayout.**[**getLayoutAlignmentY**](http://docs.google.com/java/awt/GridBagLayout.html#getLayoutAlignmentY(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent)            Returns the alignment along the y axis. |
| float | **CardLayout.**[**getLayoutAlignmentY**](http://docs.google.com/java/awt/CardLayout.html#getLayoutAlignmentY(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent)            Returns the alignment along the y axis. |
| float | **LayoutManager2.**[**getLayoutAlignmentY**](http://docs.google.com/java/awt/LayoutManager2.html#getLayoutAlignmentY(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) target)            Returns the alignment along the y axis. |
| float | **BorderLayout.**[**getLayoutAlignmentY**](http://docs.google.com/java/awt/BorderLayout.html#getLayoutAlignmentY(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent)            Returns the alignment along the y axis. |
| [Component](http://docs.google.com/java/awt/Component.html) | **BorderLayout.**[**getLayoutComponent**](http://docs.google.com/java/awt/BorderLayout.html#getLayoutComponent(java.awt.Container,%20java.lang.Object))([Container](http://docs.google.com/java/awt/Container.html) target, [Object](http://docs.google.com/java/lang/Object.html) constraints)            Returns the component that corresponds to the given constraint location based on the target Container's component orientation. |
| protected  [GridBagLayoutInfo](http://docs.google.com/java/awt/GridBagLayoutInfo.html) | **GridBagLayout.**[**getLayoutInfo**](http://docs.google.com/java/awt/GridBagLayout.html#getLayoutInfo(java.awt.Container,%20int))([Container](http://docs.google.com/java/awt/Container.html) parent, int sizeflag)            Fills in an instance of GridBagLayoutInfo for the current set of managed children. |
| protected  [GridBagLayoutInfo](http://docs.google.com/java/awt/GridBagLayoutInfo.html) | **GridBagLayout.**[**GetLayoutInfo**](http://docs.google.com/java/awt/GridBagLayout.html#GetLayoutInfo(java.awt.Container,%20int))([Container](http://docs.google.com/java/awt/Container.html) parent, int sizeflag)            This method is obsolete and supplied for backwards compatability only; new code should call [getLayoutInfo](http://docs.google.com/java/awt/GridBagLayout.html#getLayoutInfo(java.awt.Container,%20int)) instead. |
| protected  [Dimension](http://docs.google.com/java/awt/Dimension.html) | **GridBagLayout.**[**getMinSize**](http://docs.google.com/java/awt/GridBagLayout.html#getMinSize(java.awt.Container,%20java.awt.GridBagLayoutInfo))([Container](http://docs.google.com/java/awt/Container.html) parent, [GridBagLayoutInfo](http://docs.google.com/java/awt/GridBagLayoutInfo.html) info)            Figures out the minimum size of the master based on the information from getLayoutInfo. |
| protected  [Dimension](http://docs.google.com/java/awt/Dimension.html) | **GridBagLayout.**[**GetMinSize**](http://docs.google.com/java/awt/GridBagLayout.html#GetMinSize(java.awt.Container,%20java.awt.GridBagLayoutInfo))([Container](http://docs.google.com/java/awt/Container.html) parent, [GridBagLayoutInfo](http://docs.google.com/java/awt/GridBagLayoutInfo.html) info)            This method is obsolete and supplied for backwards compatability only; new code should call [getMinSize](http://docs.google.com/java/awt/GridBagLayout.html#getMinSize(java.awt.Container,%20java.awt.GridBagLayoutInfo)) instead. |
| void | **GridBagLayout.**[**invalidateLayout**](http://docs.google.com/java/awt/GridBagLayout.html#invalidateLayout(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) target)            Invalidates the layout, indicating that if the layout manager has cached information it should be discarded. |
| void | **CardLayout.**[**invalidateLayout**](http://docs.google.com/java/awt/CardLayout.html#invalidateLayout(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) target)            Invalidates the layout, indicating that if the layout manager has cached information it should be discarded. |
| void | **LayoutManager2.**[**invalidateLayout**](http://docs.google.com/java/awt/LayoutManager2.html#invalidateLayout(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) target)            Invalidates the layout, indicating that if the layout manager has cached information it should be discarded. |
| void | **BorderLayout.**[**invalidateLayout**](http://docs.google.com/java/awt/BorderLayout.html#invalidateLayout(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) target)            Invalidates the layout, indicating that if the layout manager has cached information it should be discarded. |
| boolean | **Container.**[**isFocusCycleRoot**](http://docs.google.com/java/awt/Container.html#isFocusCycleRoot(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) container)            Returns whether the specified Container is the focus cycle root of this Container's focus traversal cycle. |
| boolean | **Component.**[**isFocusCycleRoot**](http://docs.google.com/java/awt/Component.html#isFocusCycleRoot(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) container)            Returns whether the specified Container is the focus cycle root of this Component's focus traversal cycle. |
| void | **CardLayout.**[**last**](http://docs.google.com/java/awt/CardLayout.html#last(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent)            Flips to the last card of the container. |
| void | **GridLayout.**[**layoutContainer**](http://docs.google.com/java/awt/GridLayout.html#layoutContainer(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent)            Lays out the specified container using this layout. |
| void | **GridBagLayout.**[**layoutContainer**](http://docs.google.com/java/awt/GridBagLayout.html#layoutContainer(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent)            Lays out the specified container using this grid bag layout. |
| void | **FlowLayout.**[**layoutContainer**](http://docs.google.com/java/awt/FlowLayout.html#layoutContainer(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) target)            Lays out the container. |
| void | **CardLayout.**[**layoutContainer**](http://docs.google.com/java/awt/CardLayout.html#layoutContainer(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent)            Lays out the specified container using this card layout. |
| void | **BorderLayout.**[**layoutContainer**](http://docs.google.com/java/awt/BorderLayout.html#layoutContainer(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) target)            Lays out the container argument using this border layout. |
| void | **LayoutManager.**[**layoutContainer**](http://docs.google.com/java/awt/LayoutManager.html#layoutContainer(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent)            Lays out the specified container. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **GridBagLayout.**[**maximumLayoutSize**](http://docs.google.com/java/awt/GridBagLayout.html#maximumLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) target)            Returns the maximum dimensions for this layout given the components in the specified target container. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **CardLayout.**[**maximumLayoutSize**](http://docs.google.com/java/awt/CardLayout.html#maximumLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) target)            Returns the maximum dimensions for this layout given the components in the specified target container. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **LayoutManager2.**[**maximumLayoutSize**](http://docs.google.com/java/awt/LayoutManager2.html#maximumLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) target)            Calculates the maximum size dimensions for the specified container, given the components it contains. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **BorderLayout.**[**maximumLayoutSize**](http://docs.google.com/java/awt/BorderLayout.html#maximumLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) target)            Returns the maximum dimensions for this layout given the components in the specified target container. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **GridLayout.**[**minimumLayoutSize**](http://docs.google.com/java/awt/GridLayout.html#minimumLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent)            Determines the minimum size of the container argument using this grid layout. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **GridBagLayout.**[**minimumLayoutSize**](http://docs.google.com/java/awt/GridBagLayout.html#minimumLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent)            Determines the minimum size of the parent container using this grid bag layout. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **FlowLayout.**[**minimumLayoutSize**](http://docs.google.com/java/awt/FlowLayout.html#minimumLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) target)            Returns the minimum dimensions needed to layout the *visible* components contained in the specified target container. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **CardLayout.**[**minimumLayoutSize**](http://docs.google.com/java/awt/CardLayout.html#minimumLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent)            Calculates the minimum size for the specified panel. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **BorderLayout.**[**minimumLayoutSize**](http://docs.google.com/java/awt/BorderLayout.html#minimumLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) target)            Determines the minimum size of the target container using this layout manager. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **LayoutManager.**[**minimumLayoutSize**](http://docs.google.com/java/awt/LayoutManager.html#minimumLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent)            Calculates the minimum size dimensions for the specified container, given the components it contains. |
| void | **CardLayout.**[**next**](http://docs.google.com/java/awt/CardLayout.html#next(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent)            Flips to the next card of the specified container. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **GridLayout.**[**preferredLayoutSize**](http://docs.google.com/java/awt/GridLayout.html#preferredLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent)            Determines the preferred size of the container argument using this grid layout. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **GridBagLayout.**[**preferredLayoutSize**](http://docs.google.com/java/awt/GridBagLayout.html#preferredLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent)            Determines the preferred size of the parent container using this grid bag layout. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **FlowLayout.**[**preferredLayoutSize**](http://docs.google.com/java/awt/FlowLayout.html#preferredLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) target)            Returns the preferred dimensions for this layout given the *visible* components in the specified target container. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **CardLayout.**[**preferredLayoutSize**](http://docs.google.com/java/awt/CardLayout.html#preferredLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent)            Determines the preferred size of the container argument using this card layout. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **BorderLayout.**[**preferredLayoutSize**](http://docs.google.com/java/awt/BorderLayout.html#preferredLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) target)            Determines the preferred size of the target container using this layout manager, based on the components in the container. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **LayoutManager.**[**preferredLayoutSize**](http://docs.google.com/java/awt/LayoutManager.html#preferredLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent)            Calculates the preferred size dimensions for the specified container, given the components it contains. |
| void | **CardLayout.**[**previous**](http://docs.google.com/java/awt/CardLayout.html#previous(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent)            Flips to the previous card of the specified container. |
| void | **KeyboardFocusManager.**[**setGlobalCurrentFocusCycleRoot**](http://docs.google.com/java/awt/KeyboardFocusManager.html#setGlobalCurrentFocusCycleRoot(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) newFocusCycleRoot)            Sets the current focus cycle root. |
| void | **CardLayout.**[**show**](http://docs.google.com/java/awt/CardLayout.html#show(java.awt.Container,%20java.lang.String))([Container](http://docs.google.com/java/awt/Container.html) parent, [String](http://docs.google.com/java/lang/String.html) name)            Flips to the component that was added to this layout with the specified name, using addLayoutComponent. |

| Uses of [Container](http://docs.google.com/java/awt/Container.html) in [java.awt.event](http://docs.google.com/java/awt/event/package-summary.html) | |
| --- | --- |

| Methods in [java.awt.event](http://docs.google.com/java/awt/event/package-summary.html) that return [Container](http://docs.google.com/java/awt/Container.html) | |
| --- | --- |
| [Container](http://docs.google.com/java/awt/Container.html) | **HierarchyEvent.**[**getChangedParent**](http://docs.google.com/java/awt/event/HierarchyEvent.html#getChangedParent())()            Returns the parent of the Component returned by getChanged(). |
| [Container](http://docs.google.com/java/awt/Container.html) | **ContainerEvent.**[**getContainer**](http://docs.google.com/java/awt/event/ContainerEvent.html#getContainer())()            Returns the originator of the event. |

| Constructors in [java.awt.event](http://docs.google.com/java/awt/event/package-summary.html) with parameters of type [Container](http://docs.google.com/java/awt/Container.html) | |
| --- | --- |
| [**HierarchyEvent**](http://docs.google.com/java/awt/event/HierarchyEvent.html#HierarchyEvent(java.awt.Component,%20int,%20java.awt.Component,%20java.awt.Container))([Component](http://docs.google.com/java/awt/Component.html) source, int id, [Component](http://docs.google.com/java/awt/Component.html) changed, [Container](http://docs.google.com/java/awt/Container.html) changedParent)            Constructs an HierarchyEvent object to identify a change in the Component hierarchy. |
| [**HierarchyEvent**](http://docs.google.com/java/awt/event/HierarchyEvent.html#HierarchyEvent(java.awt.Component,%20int,%20java.awt.Component,%20java.awt.Container,%20long))([Component](http://docs.google.com/java/awt/Component.html) source, int id, [Component](http://docs.google.com/java/awt/Component.html) changed, [Container](http://docs.google.com/java/awt/Container.html) changedParent, long changeFlags)            Constructs an HierarchyEvent object to identify a change in the Component hierarchy. |

| Uses of [Container](http://docs.google.com/java/awt/Container.html) in [java.beans.beancontext](http://docs.google.com/java/beans/beancontext/package-summary.html) | |
| --- | --- |

| Methods in [java.beans.beancontext](http://docs.google.com/java/beans/beancontext/package-summary.html) that return [Container](http://docs.google.com/java/awt/Container.html) | |
| --- | --- |
| [Container](http://docs.google.com/java/awt/Container.html) | **BeanContextContainerProxy.**[**getContainer**](http://docs.google.com/java/beans/beancontext/BeanContextContainerProxy.html#getContainer())()            Gets the java.awt.Container associated with this BeanContext. |

| Uses of [Container](http://docs.google.com/java/awt/Container.html) in [javax.swing](http://docs.google.com/javax/swing/package-summary.html) | |
| --- | --- |

| Subclasses of [Container](http://docs.google.com/java/awt/Container.html) in [javax.swing](http://docs.google.com/javax/swing/package-summary.html) | |
| --- | --- |
| class | [**AbstractButton**](http://docs.google.com/javax/swing/AbstractButton.html)            Defines common behaviors for buttons and menu items. |
| class | [**Box**](http://docs.google.com/javax/swing/Box.html)            A lightweight container that uses a BoxLayout object as its layout manager. |
| static class | [**Box.Filler**](http://docs.google.com/javax/swing/Box.Filler.html)            An implementation of a lightweight component that participates in layout but has no view. |
| class | [**CellRendererPane**](http://docs.google.com/javax/swing/CellRendererPane.html)            This class is inserted in between cell renderers and the components that use them. |
| class | [**DefaultListCellRenderer**](http://docs.google.com/javax/swing/DefaultListCellRenderer.html)            Renders an item in a list. |
| static class | [**DefaultListCellRenderer.UIResource**](http://docs.google.com/javax/swing/DefaultListCellRenderer.UIResource.html)            A subclass of DefaultListCellRenderer that implements UIResource. |
| class | [**JApplet**](http://docs.google.com/javax/swing/JApplet.html)            An extended version of java.applet.Applet that adds support for the JFC/Swing component architecture. |
| class | [**JButton**](http://docs.google.com/javax/swing/JButton.html)            An implementation of a "push" button. |
| class | [**JCheckBox**](http://docs.google.com/javax/swing/JCheckBox.html)            An implementation of a check box -- an item that can be selected or deselected, and which displays its state to the user. |
| class | [**JCheckBoxMenuItem**](http://docs.google.com/javax/swing/JCheckBoxMenuItem.html)            A menu item that can be selected or deselected. |
| class | [**JColorChooser**](http://docs.google.com/javax/swing/JColorChooser.html)            JColorChooser provides a pane of controls designed to allow a user to manipulate and select a color. |
| class | [**JComboBox**](http://docs.google.com/javax/swing/JComboBox.html)            A component that combines a button or editable field and a drop-down list. |
| class | [**JComponent**](http://docs.google.com/javax/swing/JComponent.html)            The base class for all Swing components except top-level containers. |
| class | [**JDesktopPane**](http://docs.google.com/javax/swing/JDesktopPane.html)            A container used to create a multiple-document interface or a virtual desktop. |
| class | [**JDialog**](http://docs.google.com/javax/swing/JDialog.html)            The main class for creating a dialog window. |
| class | [**JEditorPane**](http://docs.google.com/javax/swing/JEditorPane.html)            A text component to edit various kinds of content. |
| class | [**JFileChooser**](http://docs.google.com/javax/swing/JFileChooser.html)            JFileChooser provides a simple mechanism for the user to choose a file. |
| class | [**JFormattedTextField**](http://docs.google.com/javax/swing/JFormattedTextField.html)            JFormattedTextField extends JTextField adding support for formatting arbitrary values, as well as retrieving a particular object once the user has edited the text. |
| class | [**JFrame**](http://docs.google.com/javax/swing/JFrame.html)            An extended version of java.awt.Frame that adds support for the JFC/Swing component architecture. |
| class | [**JInternalFrame**](http://docs.google.com/javax/swing/JInternalFrame.html)            A lightweight object that provides many of the features of a native frame, including dragging, closing, becoming an icon, resizing, title display, and support for a menu bar. |
| static class | [**JInternalFrame.JDesktopIcon**](http://docs.google.com/javax/swing/JInternalFrame.JDesktopIcon.html)            This component represents an iconified version of a JInternalFrame. |
| class | [**JLabel**](http://docs.google.com/javax/swing/JLabel.html)            A display area for a short text string or an image, or both. |
| class | [**JLayeredPane**](http://docs.google.com/javax/swing/JLayeredPane.html)            JLayeredPane adds depth to a JFC/Swing container, allowing components to overlap each other when needed. |
| class | [**JList**](http://docs.google.com/javax/swing/JList.html)            A component that displays a list of objects and allows the user to select one or more items. |
| class | [**JMenu**](http://docs.google.com/javax/swing/JMenu.html)            An implementation of a menu -- a popup window containing JMenuItems that is displayed when the user selects an item on the JMenuBar. |
| class | [**JMenuBar**](http://docs.google.com/javax/swing/JMenuBar.html)            An implementation of a menu bar. |
| class | [**JMenuItem**](http://docs.google.com/javax/swing/JMenuItem.html)            An implementation of an item in a menu. |
| class | [**JOptionPane**](http://docs.google.com/javax/swing/JOptionPane.html)            JOptionPane makes it easy to pop up a standard dialog box that prompts users for a value or informs them of something. |
| class | [**JPanel**](http://docs.google.com/javax/swing/JPanel.html)            JPanel is a generic lightweight container. |
| class | [**JPasswordField**](http://docs.google.com/javax/swing/JPasswordField.html)            JPasswordField is a lightweight component that allows the editing of a single line of text where the view indicates something was typed, but does not show the original characters. |
| class | [**JPopupMenu**](http://docs.google.com/javax/swing/JPopupMenu.html)            An implementation of a popup menu -- a small window that pops up and displays a series of choices. |
| static class | [**JPopupMenu.Separator**](http://docs.google.com/javax/swing/JPopupMenu.Separator.html)            A popup menu-specific separator. |
| class | [**JProgressBar**](http://docs.google.com/javax/swing/JProgressBar.html)            A component that visually displays the progress of some task. |
| class | [**JRadioButton**](http://docs.google.com/javax/swing/JRadioButton.html)            An implementation of a radio button -- an item that can be selected or deselected, and which displays its state to the user. |
| class | [**JRadioButtonMenuItem**](http://docs.google.com/javax/swing/JRadioButtonMenuItem.html)            An implementation of a radio button menu item. |
| class | [**JRootPane**](http://docs.google.com/javax/swing/JRootPane.html)            A lightweight container used behind the scenes by JFrame, JDialog, JWindow, JApplet, and JInternalFrame. |
| class | [**JScrollBar**](http://docs.google.com/javax/swing/JScrollBar.html)            An implementation of a scrollbar. |
| class | [**JScrollPane**](http://docs.google.com/javax/swing/JScrollPane.html)            Provides a scrollable view of a lightweight component. |
| protected  class | [**JScrollPane.ScrollBar**](http://docs.google.com/javax/swing/JScrollPane.ScrollBar.html)            By default JScrollPane creates scrollbars that are instances of this class. |
| class | [**JSeparator**](http://docs.google.com/javax/swing/JSeparator.html)            JSeparator provides a general purpose component for implementing divider lines - most commonly used as a divider between menu items that breaks them up into logical groupings. |
| class | [**JSlider**](http://docs.google.com/javax/swing/JSlider.html)            A component that lets the user graphically select a value by sliding a knob within a bounded interval. |
| class | [**JSpinner**](http://docs.google.com/javax/swing/JSpinner.html)            A single line input field that lets the user select a number or an object value from an ordered sequence. |
| static class | [**JSpinner.DateEditor**](http://docs.google.com/javax/swing/JSpinner.DateEditor.html)            An editor for a JSpinner whose model is a SpinnerDateModel. |
| static class | [**JSpinner.DefaultEditor**](http://docs.google.com/javax/swing/JSpinner.DefaultEditor.html)            A simple base class for more specialized editors that displays a read-only view of the model's current value with a JFormattedTextField. |
| static class | [**JSpinner.ListEditor**](http://docs.google.com/javax/swing/JSpinner.ListEditor.html)            An editor for a JSpinner whose model is a SpinnerListModel. |
| static class | [**JSpinner.NumberEditor**](http://docs.google.com/javax/swing/JSpinner.NumberEditor.html)            An editor for a JSpinner whose model is a SpinnerNumberModel. |
| class | [**JSplitPane**](http://docs.google.com/javax/swing/JSplitPane.html)            JSplitPane is used to divide two (and only two) Components. |
| class | [**JTabbedPane**](http://docs.google.com/javax/swing/JTabbedPane.html)            A component that lets the user switch between a group of components by clicking on a tab with a given title and/or icon. |
| class | [**JTable**](http://docs.google.com/javax/swing/JTable.html)            The JTable is used to display and edit regular two-dimensional tables of cells. |
| class | [**JTextArea**](http://docs.google.com/javax/swing/JTextArea.html)            A JTextArea is a multi-line area that displays plain text. |
| class | [**JTextField**](http://docs.google.com/javax/swing/JTextField.html)            JTextField is a lightweight component that allows the editing of a single line of text. |
| class | [**JTextPane**](http://docs.google.com/javax/swing/JTextPane.html)            A text component that can be marked up with attributes that are represented graphically. |
| class | [**JToggleButton**](http://docs.google.com/javax/swing/JToggleButton.html)            An implementation of a two-state button. |
| class | [**JToolBar**](http://docs.google.com/javax/swing/JToolBar.html)            JToolBar provides a component that is useful for displaying commonly used Actions or controls. |
| static class | [**JToolBar.Separator**](http://docs.google.com/javax/swing/JToolBar.Separator.html)            A toolbar-specific separator. |
| class | [**JToolTip**](http://docs.google.com/javax/swing/JToolTip.html)            Used to display a "Tip" for a Component. |
| class | [**JTree**](http://docs.google.com/javax/swing/JTree.html)  A control that displays a set of hierarchical data as an outline. |
| class | [**JViewport**](http://docs.google.com/javax/swing/JViewport.html)            The "viewport" or "porthole" through which you see the underlying information. |
| class | [**JWindow**](http://docs.google.com/javax/swing/JWindow.html)            A JWindow is a container that can be displayed anywhere on the user's desktop. |

| Fields in [javax.swing](http://docs.google.com/javax/swing/package-summary.html) declared as [Container](http://docs.google.com/java/awt/Container.html) | |
| --- | --- |
| protected  [Container](http://docs.google.com/java/awt/Container.html) | **JRootPane.**[**contentPane**](http://docs.google.com/javax/swing/JRootPane.html#contentPane)            The content pane. |

| Methods in [javax.swing](http://docs.google.com/javax/swing/package-summary.html) that return [Container](http://docs.google.com/java/awt/Container.html) | |
| --- | --- |
| protected  [Container](http://docs.google.com/java/awt/Container.html) | **JRootPane.**[**createContentPane**](http://docs.google.com/javax/swing/JRootPane.html#createContentPane())()            Called by the constructor methods to create the default contentPane. |
| static [Container](http://docs.google.com/java/awt/Container.html) | **SwingUtilities.**[**getAncestorNamed**](http://docs.google.com/javax/swing/SwingUtilities.html#getAncestorNamed(java.lang.String,%20java.awt.Component))([String](http://docs.google.com/java/lang/String.html) name, [Component](http://docs.google.com/java/awt/Component.html) comp)            Convenience method for searching above comp in the component hierarchy and returns the first object of name it finds. |
| static [Container](http://docs.google.com/java/awt/Container.html) | **SwingUtilities.**[**getAncestorOfClass**](http://docs.google.com/javax/swing/SwingUtilities.html#getAncestorOfClass(java.lang.Class,%20java.awt.Component))([Class](http://docs.google.com/java/lang/Class.html)<?> c, [Component](http://docs.google.com/java/awt/Component.html) comp)            Convenience method for searching above comp in the component hierarchy and returns the first object of class c it finds. |
| [Container](http://docs.google.com/java/awt/Container.html) | **JInternalFrame.**[**getContentPane**](http://docs.google.com/javax/swing/JInternalFrame.html#getContentPane())()            Returns the content pane for this internal frame. |
| [Container](http://docs.google.com/java/awt/Container.html) | **JDialog.**[**getContentPane**](http://docs.google.com/javax/swing/JDialog.html#getContentPane())()            Returns the contentPane object for this dialog. |
| [Container](http://docs.google.com/java/awt/Container.html) | **JApplet.**[**getContentPane**](http://docs.google.com/javax/swing/JApplet.html#getContentPane())()            Returns the contentPane object for this applet. |
| [Container](http://docs.google.com/java/awt/Container.html) | **JWindow.**[**getContentPane**](http://docs.google.com/javax/swing/JWindow.html#getContentPane())()            Returns the Container which is the contentPane for this window. |
| [Container](http://docs.google.com/java/awt/Container.html) | **JRootPane.**[**getContentPane**](http://docs.google.com/javax/swing/JRootPane.html#getContentPane())()            Returns the content pane -- the container that holds the components parented by the root pane. |
| [Container](http://docs.google.com/java/awt/Container.html) | **JFrame.**[**getContentPane**](http://docs.google.com/javax/swing/JFrame.html#getContentPane())()            Returns the contentPane object for this frame. |
| [Container](http://docs.google.com/java/awt/Container.html) | **RootPaneContainer.**[**getContentPane**](http://docs.google.com/javax/swing/RootPaneContainer.html#getContentPane())()            Returns the contentPane. |
| [Container](http://docs.google.com/java/awt/Container.html) | **JInternalFrame.**[**getFocusCycleRootAncestor**](http://docs.google.com/javax/swing/JInternalFrame.html#getFocusCycleRootAncestor())()            Always returns null because JInternalFrames must always be roots of a focus traversal cycle. |
| [Container](http://docs.google.com/java/awt/Container.html) | **BoxLayout.**[**getTarget**](http://docs.google.com/javax/swing/BoxLayout.html#getTarget())()            Returns the container that uses this layout manager. |
| [Container](http://docs.google.com/java/awt/Container.html) | **OverlayLayout.**[**getTarget**](http://docs.google.com/javax/swing/OverlayLayout.html#getTarget())()            Returns the container that uses this layout manager. |
| [Container](http://docs.google.com/java/awt/Container.html) | **JComponent.**[**getTopLevelAncestor**](http://docs.google.com/javax/swing/JComponent.html#getTopLevelAncestor())()            Returns the top-level ancestor of this component (either the containing Window or Applet), or null if this component has not been added to any container. |

| Methods in [javax.swing](http://docs.google.com/javax/swing/package-summary.html) with parameters of type [Container](http://docs.google.com/java/awt/Container.html) | |
| --- | --- |
| [Component](http://docs.google.com/java/awt/Component.html) | **SortingFocusTraversalPolicy.**[**getComponentAfter**](http://docs.google.com/javax/swing/SortingFocusTraversalPolicy.html#getComponentAfter(java.awt.Container,%20java.awt.Component))([Container](http://docs.google.com/java/awt/Container.html) aContainer, [Component](http://docs.google.com/java/awt/Component.html) aComponent)            Returns the Component that should receive the focus after aComponent. |
| [Component](http://docs.google.com/java/awt/Component.html) | **LayoutFocusTraversalPolicy.**[**getComponentAfter**](http://docs.google.com/javax/swing/LayoutFocusTraversalPolicy.html#getComponentAfter(java.awt.Container,%20java.awt.Component))([Container](http://docs.google.com/java/awt/Container.html) aContainer, [Component](http://docs.google.com/java/awt/Component.html) aComponent)            Returns the Component that should receive the focus after aComponent. |
| [Component](http://docs.google.com/java/awt/Component.html) | **DefaultFocusManager.**[**getComponentAfter**](http://docs.google.com/javax/swing/DefaultFocusManager.html#getComponentAfter(java.awt.Container,%20java.awt.Component))([Container](http://docs.google.com/java/awt/Container.html) aContainer, [Component](http://docs.google.com/java/awt/Component.html) aComponent) |
| [Component](http://docs.google.com/java/awt/Component.html) | **SortingFocusTraversalPolicy.**[**getComponentBefore**](http://docs.google.com/javax/swing/SortingFocusTraversalPolicy.html#getComponentBefore(java.awt.Container,%20java.awt.Component))([Container](http://docs.google.com/java/awt/Container.html) aContainer, [Component](http://docs.google.com/java/awt/Component.html) aComponent)            Returns the Component that should receive the focus before aComponent. |
| [Component](http://docs.google.com/java/awt/Component.html) | **LayoutFocusTraversalPolicy.**[**getComponentBefore**](http://docs.google.com/javax/swing/LayoutFocusTraversalPolicy.html#getComponentBefore(java.awt.Container,%20java.awt.Component))([Container](http://docs.google.com/java/awt/Container.html) aContainer, [Component](http://docs.google.com/java/awt/Component.html) aComponent)            Returns the Component that should receive the focus before aComponent. |
| [Component](http://docs.google.com/java/awt/Component.html) | **DefaultFocusManager.**[**getComponentBefore**](http://docs.google.com/javax/swing/DefaultFocusManager.html#getComponentBefore(java.awt.Container,%20java.awt.Component))([Container](http://docs.google.com/java/awt/Container.html) aContainer, [Component](http://docs.google.com/java/awt/Component.html) aComponent) |
| abstract  int | **LayoutStyle.**[**getContainerGap**](http://docs.google.com/javax/swing/LayoutStyle.html#getContainerGap(javax.swing.JComponent,%20int,%20java.awt.Container))([JComponent](http://docs.google.com/javax/swing/JComponent.html) component, int position, [Container](http://docs.google.com/java/awt/Container.html) parent)            Returns the amount of space to place between the component and specified edge of its parent. |
| [Component](http://docs.google.com/java/awt/Component.html) | **SortingFocusTraversalPolicy.**[**getDefaultComponent**](http://docs.google.com/javax/swing/SortingFocusTraversalPolicy.html#getDefaultComponent(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) aContainer)            Returns the default Component to focus. |
| [Component](http://docs.google.com/java/awt/Component.html) | **SortingFocusTraversalPolicy.**[**getFirstComponent**](http://docs.google.com/javax/swing/SortingFocusTraversalPolicy.html#getFirstComponent(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) aContainer)            Returns the first Component in the traversal cycle. |
| [Component](http://docs.google.com/java/awt/Component.html) | **LayoutFocusTraversalPolicy.**[**getFirstComponent**](http://docs.google.com/javax/swing/LayoutFocusTraversalPolicy.html#getFirstComponent(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) aContainer)            Returns the first Component in the traversal cycle. |
| [Component](http://docs.google.com/java/awt/Component.html) | **DefaultFocusManager.**[**getFirstComponent**](http://docs.google.com/javax/swing/DefaultFocusManager.html#getFirstComponent(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) aContainer) |
| [Component](http://docs.google.com/java/awt/Component.html) | **SortingFocusTraversalPolicy.**[**getLastComponent**](http://docs.google.com/javax/swing/SortingFocusTraversalPolicy.html#getLastComponent(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) aContainer)            Returns the last Component in the traversal cycle. |
| [Component](http://docs.google.com/java/awt/Component.html) | **LayoutFocusTraversalPolicy.**[**getLastComponent**](http://docs.google.com/javax/swing/LayoutFocusTraversalPolicy.html#getLastComponent(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) aContainer)            Returns the last Component in the traversal cycle. |
| [Component](http://docs.google.com/java/awt/Component.html) | **DefaultFocusManager.**[**getLastComponent**](http://docs.google.com/javax/swing/DefaultFocusManager.html#getLastComponent(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) aContainer) |
| float | **SpringLayout.**[**getLayoutAlignmentX**](http://docs.google.com/javax/swing/SpringLayout.html#getLayoutAlignmentX(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) p)            Returns 0.5f (centered). |
| float | **GroupLayout.**[**getLayoutAlignmentX**](http://docs.google.com/javax/swing/GroupLayout.html#getLayoutAlignmentX(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent)            Returns the alignment along the x axis. |
| float | **BoxLayout.**[**getLayoutAlignmentX**](http://docs.google.com/javax/swing/BoxLayout.html#getLayoutAlignmentX(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) target)            Returns the alignment along the X axis for the container. |
| float | **OverlayLayout.**[**getLayoutAlignmentX**](http://docs.google.com/javax/swing/OverlayLayout.html#getLayoutAlignmentX(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) target)            Returns the alignment along the x axis for the container. |
| float | **JRootPane.RootLayout.**[**getLayoutAlignmentX**](http://docs.google.com/javax/swing/JRootPane.RootLayout.html#getLayoutAlignmentX(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) target) |
| float | **SpringLayout.**[**getLayoutAlignmentY**](http://docs.google.com/javax/swing/SpringLayout.html#getLayoutAlignmentY(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) p)            Returns 0.5f (centered). |
| float | **GroupLayout.**[**getLayoutAlignmentY**](http://docs.google.com/javax/swing/GroupLayout.html#getLayoutAlignmentY(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent)            Returns the alignment along the y axis. |
| float | **BoxLayout.**[**getLayoutAlignmentY**](http://docs.google.com/javax/swing/BoxLayout.html#getLayoutAlignmentY(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) target)            Returns the alignment along the Y axis for the container. |
| float | **OverlayLayout.**[**getLayoutAlignmentY**](http://docs.google.com/javax/swing/OverlayLayout.html#getLayoutAlignmentY(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) target)            Returns the alignment along the y axis for the container. |
| float | **JRootPane.RootLayout.**[**getLayoutAlignmentY**](http://docs.google.com/javax/swing/JRootPane.RootLayout.html#getLayoutAlignmentY(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) target) |
| abstract  int | **LayoutStyle.**[**getPreferredGap**](http://docs.google.com/javax/swing/LayoutStyle.html#getPreferredGap(javax.swing.JComponent,%20javax.swing.JComponent,%20javax.swing.LayoutStyle.ComponentPlacement,%20int,%20java.awt.Container))([JComponent](http://docs.google.com/javax/swing/JComponent.html) component1, [JComponent](http://docs.google.com/javax/swing/JComponent.html) component2, [LayoutStyle.ComponentPlacement](http://docs.google.com/javax/swing/LayoutStyle.ComponentPlacement.html) type, int position, [Container](http://docs.google.com/java/awt/Container.html) parent)            Returns the amount of space to use between two components. |
| void | **SpringLayout.**[**invalidateLayout**](http://docs.google.com/javax/swing/SpringLayout.html#invalidateLayout(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) p) |
| void | **GroupLayout.**[**invalidateLayout**](http://docs.google.com/javax/swing/GroupLayout.html#invalidateLayout(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent)            Invalidates the layout, indicating that if the layout manager has cached information it should be discarded. |
| void | **BoxLayout.**[**invalidateLayout**](http://docs.google.com/javax/swing/BoxLayout.html#invalidateLayout(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) target)            Indicates that a child has changed its layout related information, and thus any cached calculations should be flushed. |
| void | **OverlayLayout.**[**invalidateLayout**](http://docs.google.com/javax/swing/OverlayLayout.html#invalidateLayout(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) target)            Indicates a child has changed its layout related information, which causes any cached calculations to be flushed. |
| void | **JRootPane.RootLayout.**[**invalidateLayout**](http://docs.google.com/javax/swing/JRootPane.RootLayout.html#invalidateLayout(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) target) |
| void | **SpringLayout.**[**layoutContainer**](http://docs.google.com/javax/swing/SpringLayout.html#layoutContainer(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent) |
| void | **GroupLayout.**[**layoutContainer**](http://docs.google.com/javax/swing/GroupLayout.html#layoutContainer(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent)            Lays out the specified container. |
| void | **JSpinner.DefaultEditor.**[**layoutContainer**](http://docs.google.com/javax/swing/JSpinner.DefaultEditor.html#layoutContainer(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent)            Resize the one (and only) child to completely fill the area within the parents insets. |
| void | **BoxLayout.**[**layoutContainer**](http://docs.google.com/javax/swing/BoxLayout.html#layoutContainer(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) target)            Called by the AWT when the specified container needs to be laid out. |
| void | **ViewportLayout.**[**layoutContainer**](http://docs.google.com/javax/swing/ViewportLayout.html#layoutContainer(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent)            Called by the AWT when the specified container needs to be laid out. |
| void | **ScrollPaneLayout.**[**layoutContainer**](http://docs.google.com/javax/swing/ScrollPaneLayout.html#layoutContainer(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent)            Lays out the scrollpane. |
| void | **OverlayLayout.**[**layoutContainer**](http://docs.google.com/javax/swing/OverlayLayout.html#layoutContainer(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) target)            Called by the AWT when the specified container needs to be laid out. |
| void | **JRootPane.RootLayout.**[**layoutContainer**](http://docs.google.com/javax/swing/JRootPane.RootLayout.html#layoutContainer(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent)            Instructs the layout manager to perform the layout for the specified container. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **SpringLayout.**[**maximumLayoutSize**](http://docs.google.com/javax/swing/SpringLayout.html#maximumLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent) |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **GroupLayout.**[**maximumLayoutSize**](http://docs.google.com/javax/swing/GroupLayout.html#maximumLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent)            Returns the maximum size for the specified container. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **BoxLayout.**[**maximumLayoutSize**](http://docs.google.com/javax/swing/BoxLayout.html#maximumLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) target)            Returns the maximum dimensions the target container can use to lay out the components it contains. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **OverlayLayout.**[**maximumLayoutSize**](http://docs.google.com/javax/swing/OverlayLayout.html#maximumLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) target)            Returns the maximum dimensions needed to lay out the components contained in the specified target container. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **JRootPane.RootLayout.**[**maximumLayoutSize**](http://docs.google.com/javax/swing/JRootPane.RootLayout.html#maximumLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) target)            Returns the maximum amount of space the layout can use. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **SpringLayout.**[**minimumLayoutSize**](http://docs.google.com/javax/swing/SpringLayout.html#minimumLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent) |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **GroupLayout.**[**minimumLayoutSize**](http://docs.google.com/javax/swing/GroupLayout.html#minimumLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent)            Returns the minimum size for the specified container. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **JSpinner.DefaultEditor.**[**minimumLayoutSize**](http://docs.google.com/javax/swing/JSpinner.DefaultEditor.html#minimumLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent)            Returns the minimum size of first (and only) child plus the size of the parents insets. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **BoxLayout.**[**minimumLayoutSize**](http://docs.google.com/javax/swing/BoxLayout.html#minimumLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) target)            Returns the minimum dimensions needed to lay out the components contained in the specified target container. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **ViewportLayout.**[**minimumLayoutSize**](http://docs.google.com/javax/swing/ViewportLayout.html#minimumLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent)            Returns the minimum dimensions needed to layout the components contained in the specified target container. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **ScrollPaneLayout.**[**minimumLayoutSize**](http://docs.google.com/javax/swing/ScrollPaneLayout.html#minimumLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent)            The minimum size of a ScrollPane is the size of the insets plus minimum size of the viewport, plus the scrollpane's viewportBorder insets, plus the minimum size of the visible headers, plus the minimum size of the scrollbars whose displayPolicy isn't NEVER. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **OverlayLayout.**[**minimumLayoutSize**](http://docs.google.com/javax/swing/OverlayLayout.html#minimumLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) target)            Returns the minimum dimensions needed to lay out the components contained in the specified target container. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **JRootPane.RootLayout.**[**minimumLayoutSize**](http://docs.google.com/javax/swing/JRootPane.RootLayout.html#minimumLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent)            Returns the minimum amount of space the layout needs. |
| void | **CellRendererPane.**[**paintComponent**](http://docs.google.com/javax/swing/CellRendererPane.html#paintComponent(java.awt.Graphics,%20java.awt.Component,%20java.awt.Container,%20int,%20int,%20int,%20int))([Graphics](http://docs.google.com/java/awt/Graphics.html) g, [Component](http://docs.google.com/java/awt/Component.html) c, [Container](http://docs.google.com/java/awt/Container.html) p, int x, int y, int w, int h)            Calls this.paintComponent(g, c, p, x, y, w, h, false). |
| static void | **SwingUtilities.**[**paintComponent**](http://docs.google.com/javax/swing/SwingUtilities.html#paintComponent(java.awt.Graphics,%20java.awt.Component,%20java.awt.Container,%20int,%20int,%20int,%20int))([Graphics](http://docs.google.com/java/awt/Graphics.html) g, [Component](http://docs.google.com/java/awt/Component.html) c, [Container](http://docs.google.com/java/awt/Container.html) p, int x, int y, int w, int h)            Paints a component to the specified Graphics. |
| void | **CellRendererPane.**[**paintComponent**](http://docs.google.com/javax/swing/CellRendererPane.html#paintComponent(java.awt.Graphics,%20java.awt.Component,%20java.awt.Container,%20int,%20int,%20int,%20int,%20boolean))([Graphics](http://docs.google.com/java/awt/Graphics.html) g, [Component](http://docs.google.com/java/awt/Component.html) c, [Container](http://docs.google.com/java/awt/Container.html) p, int x, int y, int w, int h, boolean shouldValidate)            Paint a cell renderer component c on graphics object g. |
| void | **CellRendererPane.**[**paintComponent**](http://docs.google.com/javax/swing/CellRendererPane.html#paintComponent(java.awt.Graphics,%20java.awt.Component,%20java.awt.Container,%20java.awt.Rectangle))([Graphics](http://docs.google.com/java/awt/Graphics.html) g, [Component](http://docs.google.com/java/awt/Component.html) c, [Container](http://docs.google.com/java/awt/Container.html) p, [Rectangle](http://docs.google.com/java/awt/Rectangle.html) r)            Calls this.paintComponent() with the rectangles x,y,width,height fields. |
| static void | **SwingUtilities.**[**paintComponent**](http://docs.google.com/javax/swing/SwingUtilities.html#paintComponent(java.awt.Graphics,%20java.awt.Component,%20java.awt.Container,%20java.awt.Rectangle))([Graphics](http://docs.google.com/java/awt/Graphics.html) g, [Component](http://docs.google.com/java/awt/Component.html) c, [Container](http://docs.google.com/java/awt/Container.html) p, [Rectangle](http://docs.google.com/java/awt/Rectangle.html) r)            Paints a component to the specified Graphics. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **SpringLayout.**[**preferredLayoutSize**](http://docs.google.com/javax/swing/SpringLayout.html#preferredLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent) |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **GroupLayout.**[**preferredLayoutSize**](http://docs.google.com/javax/swing/GroupLayout.html#preferredLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent)            Returns the preferred size for the specified container. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **JSpinner.DefaultEditor.**[**preferredLayoutSize**](http://docs.google.com/javax/swing/JSpinner.DefaultEditor.html#preferredLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent)            Returns the preferred size of first (and only) child plus the size of the parents insets. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **BoxLayout.**[**preferredLayoutSize**](http://docs.google.com/javax/swing/BoxLayout.html#preferredLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) target)            Returns the preferred dimensions for this layout, given the components in the specified target container. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **ViewportLayout.**[**preferredLayoutSize**](http://docs.google.com/javax/swing/ViewportLayout.html#preferredLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent)            Returns the preferred dimensions for this layout given the components in the specified target container. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **ScrollPaneLayout.**[**preferredLayoutSize**](http://docs.google.com/javax/swing/ScrollPaneLayout.html#preferredLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent)            The preferred size of a ScrollPane is the size of the insets, plus the preferred size of the viewport, plus the preferred size of the visible headers, plus the preferred size of the scrollbars that will appear given the current view and the current scrollbar displayPolicies. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **OverlayLayout.**[**preferredLayoutSize**](http://docs.google.com/javax/swing/OverlayLayout.html#preferredLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) target)            Returns the preferred dimensions for this layout given the components in the specified target container. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **JRootPane.RootLayout.**[**preferredLayoutSize**](http://docs.google.com/javax/swing/JRootPane.RootLayout.html#preferredLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent)            Returns the amount of space the layout would like to have. |
| void | **JInternalFrame.**[**setContentPane**](http://docs.google.com/javax/swing/JInternalFrame.html#setContentPane(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) c)            Sets this JInternalFrame's contentPane property. |
| void | **JDialog.**[**setContentPane**](http://docs.google.com/javax/swing/JDialog.html#setContentPane(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) contentPane)            Sets the contentPane property. |
| void | **JApplet.**[**setContentPane**](http://docs.google.com/javax/swing/JApplet.html#setContentPane(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) contentPane)            Sets the contentPane property. |
| void | **JWindow.**[**setContentPane**](http://docs.google.com/javax/swing/JWindow.html#setContentPane(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) contentPane)            Sets the contentPane property for this window. |
| void | **JRootPane.**[**setContentPane**](http://docs.google.com/javax/swing/JRootPane.html#setContentPane(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) content)            Sets the content pane -- the container that holds the components parented by the root pane. |
| void | **JFrame.**[**setContentPane**](http://docs.google.com/javax/swing/JFrame.html#setContentPane(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) contentPane)            Sets the contentPane property. |
| void | **RootPaneContainer.**[**setContentPane**](http://docs.google.com/javax/swing/RootPaneContainer.html#setContentPane(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) contentPane)            The "contentPane" is the primary container for application specific components. |

| Constructors in [javax.swing](http://docs.google.com/javax/swing/package-summary.html) with parameters of type [Container](http://docs.google.com/java/awt/Container.html) | |
| --- | --- |
| [**BoxLayout**](http://docs.google.com/javax/swing/BoxLayout.html#BoxLayout(java.awt.Container,%20int))([Container](http://docs.google.com/java/awt/Container.html) target, int axis)            Creates a layout manager that will lay out components along the given axis. |
| [**GroupLayout**](http://docs.google.com/javax/swing/GroupLayout.html#GroupLayout(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) host)            Creates a GroupLayout for the specified Container. |
| [**OverlayLayout**](http://docs.google.com/javax/swing/OverlayLayout.html#OverlayLayout(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) target)            Constructs a layout manager that performs overlay arrangement of the children. |

| Uses of [Container](http://docs.google.com/java/awt/Container.html) in [javax.swing.colorchooser](http://docs.google.com/javax/swing/colorchooser/package-summary.html) | |
| --- | --- |

| Subclasses of [Container](http://docs.google.com/java/awt/Container.html) in [javax.swing.colorchooser](http://docs.google.com/javax/swing/colorchooser/package-summary.html) | |
| --- | --- |
| class | [**AbstractColorChooserPanel**](http://docs.google.com/javax/swing/colorchooser/AbstractColorChooserPanel.html)            This is the abstract superclass for color choosers. |

| Uses of [Container](http://docs.google.com/java/awt/Container.html) in [javax.swing.event](http://docs.google.com/javax/swing/event/package-summary.html) | |
| --- | --- |

| Methods in [javax.swing.event](http://docs.google.com/javax/swing/event/package-summary.html) that return [Container](http://docs.google.com/java/awt/Container.html) | |
| --- | --- |
| [Container](http://docs.google.com/java/awt/Container.html) | **AncestorEvent.**[**getAncestor**](http://docs.google.com/javax/swing/event/AncestorEvent.html#getAncestor())()            Returns the ancestor that the event actually occurred on. |
| [Container](http://docs.google.com/java/awt/Container.html) | **AncestorEvent.**[**getAncestorParent**](http://docs.google.com/javax/swing/event/AncestorEvent.html#getAncestorParent())()            Returns the parent of the ancestor the event actually occurred on. |

| Constructors in [javax.swing.event](http://docs.google.com/javax/swing/event/package-summary.html) with parameters of type [Container](http://docs.google.com/java/awt/Container.html) | |
| --- | --- |
| [**AncestorEvent**](http://docs.google.com/javax/swing/event/AncestorEvent.html#AncestorEvent(javax.swing.JComponent,%20int,%20java.awt.Container,%20java.awt.Container))([JComponent](http://docs.google.com/javax/swing/JComponent.html) source, int id, [Container](http://docs.google.com/java/awt/Container.html) ancestor, [Container](http://docs.google.com/java/awt/Container.html) ancestorParent)            Constructs an AncestorEvent object to identify a change in an ancestor-component's display-status. |

| Uses of [Container](http://docs.google.com/java/awt/Container.html) in [javax.swing.plaf.basic](http://docs.google.com/javax/swing/plaf/basic/package-summary.html) | |
| --- | --- |

| Subclasses of [Container](http://docs.google.com/java/awt/Container.html) in [javax.swing.plaf.basic](http://docs.google.com/javax/swing/plaf/basic/package-summary.html) | |
| --- | --- |
| class | [**BasicArrowButton**](http://docs.google.com/javax/swing/plaf/basic/BasicArrowButton.html)            JButton object that draws a scaled Arrow in one of the cardinal directions. |
| class | [**BasicComboBoxRenderer**](http://docs.google.com/javax/swing/plaf/basic/BasicComboBoxRenderer.html)            ComboBox renderer |
| static class | [**BasicComboBoxRenderer.UIResource**](http://docs.google.com/javax/swing/plaf/basic/BasicComboBoxRenderer.UIResource.html)            A subclass of BasicComboBoxRenderer that implements UIResource. |
| class | [**BasicComboPopup**](http://docs.google.com/javax/swing/plaf/basic/BasicComboPopup.html)            This is a basic implementation of the ComboPopup interface. |
| class | [**BasicInternalFrameTitlePane**](http://docs.google.com/javax/swing/plaf/basic/BasicInternalFrameTitlePane.html)            The class that manages a basic title bar |
| class | [**BasicInternalFrameTitlePane.SystemMenuBar**](http://docs.google.com/javax/swing/plaf/basic/BasicInternalFrameTitlePane.SystemMenuBar.html)            This class should be treated as a "protected" inner class. |
| class | [**BasicSplitPaneDivider**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneDivider.html)            Divider used by BasicSplitPaneUI. |
| protected  class | [**BasicToolBarUI.DragWindow**](http://docs.google.com/javax/swing/plaf/basic/BasicToolBarUI.DragWindow.html) |

| Methods in [javax.swing.plaf.basic](http://docs.google.com/javax/swing/plaf/basic/package-summary.html) that return [Container](http://docs.google.com/java/awt/Container.html) | |
| --- | --- |
| protected  [Container](http://docs.google.com/java/awt/Container.html) | **BasicOptionPaneUI.**[**createButtonArea**](http://docs.google.com/javax/swing/plaf/basic/BasicOptionPaneUI.html#createButtonArea())()            Creates and returns a Container containing the buttons. |
| protected  [Container](http://docs.google.com/java/awt/Container.html) | **BasicOptionPaneUI.**[**createMessageArea**](http://docs.google.com/javax/swing/plaf/basic/BasicOptionPaneUI.html#createMessageArea())()            Messaged from installComponents to create a Container containing the body of the message. |
| protected  [Container](http://docs.google.com/java/awt/Container.html) | **BasicOptionPaneUI.**[**createSeparator**](http://docs.google.com/javax/swing/plaf/basic/BasicOptionPaneUI.html#createSeparator())() |

| Methods in [javax.swing.plaf.basic](http://docs.google.com/javax/swing/plaf/basic/package-summary.html) with parameters of type [Container](http://docs.google.com/java/awt/Container.html) | |
| --- | --- |
| protected  void | **BasicOptionPaneUI.**[**addButtonComponents**](http://docs.google.com/javax/swing/plaf/basic/BasicOptionPaneUI.html#addButtonComponents(java.awt.Container,%20java.lang.Object%5B%5D,%20int))([Container](http://docs.google.com/java/awt/Container.html) container, [Object](http://docs.google.com/java/lang/Object.html)[] buttons, int initialIndex)            Creates the appropriate object to represent each of the objects in buttons and adds it to container. |
| protected  void | **BasicOptionPaneUI.**[**addIcon**](http://docs.google.com/javax/swing/plaf/basic/BasicOptionPaneUI.html#addIcon(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) top)            Creates and adds a JLabel representing the icon returned from getIcon to top. |
| protected  void | **BasicOptionPaneUI.**[**addMessageComponents**](http://docs.google.com/javax/swing/plaf/basic/BasicOptionPaneUI.html#addMessageComponents(java.awt.Container,%20java.awt.GridBagConstraints,%20java.lang.Object,%20int,%20boolean))([Container](http://docs.google.com/java/awt/Container.html) container, [GridBagConstraints](http://docs.google.com/java/awt/GridBagConstraints.html) cons, [Object](http://docs.google.com/java/lang/Object.html) msg, int maxll, boolean internallyCreated)            Creates the appropriate object to represent msg and places it into container. |
| protected  void | **BasicOptionPaneUI.**[**burstStringInto**](http://docs.google.com/javax/swing/plaf/basic/BasicOptionPaneUI.html#burstStringInto(java.awt.Container,%20java.lang.String,%20int))([Container](http://docs.google.com/java/awt/Container.html) c, [String](http://docs.google.com/java/lang/String.html) d, int maxll)            Recursively creates new JLabel instances to represent d. |
| float | **BasicSplitPaneUI.BasicHorizontalLayoutManager.**[**getLayoutAlignmentX**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.BasicHorizontalLayoutManager.html#getLayoutAlignmentX(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) target)            Returns the alignment along the x axis. |
| float | **BasicSplitPaneUI.BasicHorizontalLayoutManager.**[**getLayoutAlignmentY**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.BasicHorizontalLayoutManager.html#getLayoutAlignmentY(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) target)            Returns the alignment along the y axis. |
| void | **BasicSplitPaneUI.BasicHorizontalLayoutManager.**[**invalidateLayout**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.BasicHorizontalLayoutManager.html#invalidateLayout(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) c)            Does nothing. |
| void | **BasicTabbedPaneUI.TabbedPaneLayout.**[**layoutContainer**](http://docs.google.com/javax/swing/plaf/basic/BasicTabbedPaneUI.TabbedPaneLayout.html#layoutContainer(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent) |
| void | **BasicScrollBarUI.**[**layoutContainer**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#layoutContainer(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) scrollbarContainer) |
| void | **BasicOptionPaneUI.ButtonAreaLayout.**[**layoutContainer**](http://docs.google.com/javax/swing/plaf/basic/BasicOptionPaneUI.ButtonAreaLayout.html#layoutContainer(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) container) |
| void | **BasicInternalFrameTitlePane.TitlePaneLayout.**[**layoutContainer**](http://docs.google.com/javax/swing/plaf/basic/BasicInternalFrameTitlePane.TitlePaneLayout.html#layoutContainer(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) c) |
| void | **BasicInternalFrameUI.InternalFrameLayout.**[**layoutContainer**](http://docs.google.com/javax/swing/plaf/basic/BasicInternalFrameUI.InternalFrameLayout.html#layoutContainer(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) c) |
| void | **BasicSplitPaneUI.BasicHorizontalLayoutManager.**[**layoutContainer**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.BasicHorizontalLayoutManager.html#layoutContainer(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) container)            Does the actual layout. |
| void | **BasicComboBoxUI.ComboBoxLayoutManager.**[**layoutContainer**](http://docs.google.com/javax/swing/plaf/basic/BasicComboBoxUI.ComboBoxLayoutManager.html#layoutContainer(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent) |
| void | **BasicSplitPaneDivider.DividerLayout.**[**layoutContainer**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneDivider.DividerLayout.html#layoutContainer(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) c) |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **BasicSplitPaneUI.BasicHorizontalLayoutManager.**[**maximumLayoutSize**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.BasicHorizontalLayoutManager.html#maximumLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) target)            Returns the maximum layout size, which is Integer.MAX\_VALUE in both directions. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **BasicTabbedPaneUI.TabbedPaneLayout.**[**minimumLayoutSize**](http://docs.google.com/javax/swing/plaf/basic/BasicTabbedPaneUI.TabbedPaneLayout.html#minimumLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent) |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **BasicScrollBarUI.**[**minimumLayoutSize**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#minimumLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) scrollbarContainer) |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **BasicOptionPaneUI.ButtonAreaLayout.**[**minimumLayoutSize**](http://docs.google.com/javax/swing/plaf/basic/BasicOptionPaneUI.ButtonAreaLayout.html#minimumLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) c) |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **BasicInternalFrameTitlePane.TitlePaneLayout.**[**minimumLayoutSize**](http://docs.google.com/javax/swing/plaf/basic/BasicInternalFrameTitlePane.TitlePaneLayout.html#minimumLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) c) |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **BasicInternalFrameUI.InternalFrameLayout.**[**minimumLayoutSize**](http://docs.google.com/javax/swing/plaf/basic/BasicInternalFrameUI.InternalFrameLayout.html#minimumLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) c) |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **BasicSplitPaneUI.BasicHorizontalLayoutManager.**[**minimumLayoutSize**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.BasicHorizontalLayoutManager.html#minimumLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) container)            Returns the minimum size needed to contain the children. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **BasicComboBoxUI.ComboBoxLayoutManager.**[**minimumLayoutSize**](http://docs.google.com/javax/swing/plaf/basic/BasicComboBoxUI.ComboBoxLayoutManager.html#minimumLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent) |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **BasicSplitPaneDivider.DividerLayout.**[**minimumLayoutSize**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneDivider.DividerLayout.html#minimumLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) c) |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **BasicTabbedPaneUI.TabbedPaneLayout.**[**preferredLayoutSize**](http://docs.google.com/javax/swing/plaf/basic/BasicTabbedPaneUI.TabbedPaneLayout.html#preferredLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent) |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **BasicScrollBarUI.**[**preferredLayoutSize**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.html#preferredLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) scrollbarContainer) |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **BasicOptionPaneUI.ButtonAreaLayout.**[**preferredLayoutSize**](http://docs.google.com/javax/swing/plaf/basic/BasicOptionPaneUI.ButtonAreaLayout.html#preferredLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) c) |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **BasicInternalFrameTitlePane.TitlePaneLayout.**[**preferredLayoutSize**](http://docs.google.com/javax/swing/plaf/basic/BasicInternalFrameTitlePane.TitlePaneLayout.html#preferredLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) c) |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **BasicInternalFrameUI.InternalFrameLayout.**[**preferredLayoutSize**](http://docs.google.com/javax/swing/plaf/basic/BasicInternalFrameUI.InternalFrameLayout.html#preferredLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) c) |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **BasicSplitPaneUI.BasicHorizontalLayoutManager.**[**preferredLayoutSize**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.BasicHorizontalLayoutManager.html#preferredLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) container)            Returns the preferred size needed to contain the children. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **BasicComboBoxUI.ComboBoxLayoutManager.**[**preferredLayoutSize**](http://docs.google.com/javax/swing/plaf/basic/BasicComboBoxUI.ComboBoxLayoutManager.html#preferredLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent) |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **DefaultMenuLayout.**[**preferredLayoutSize**](http://docs.google.com/javax/swing/plaf/basic/DefaultMenuLayout.html#preferredLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) target) |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | **BasicSplitPaneDivider.DividerLayout.**[**preferredLayoutSize**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneDivider.DividerLayout.html#preferredLayoutSize(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) c) |

| Constructors in [javax.swing.plaf.basic](http://docs.google.com/javax/swing/plaf/basic/package-summary.html) with parameters of type [Container](http://docs.google.com/java/awt/Container.html) | |
| --- | --- |
| [**DefaultMenuLayout**](http://docs.google.com/javax/swing/plaf/basic/DefaultMenuLayout.html#DefaultMenuLayout(java.awt.Container,%20int))([Container](http://docs.google.com/java/awt/Container.html) target, int axis) |

| Uses of [Container](http://docs.google.com/java/awt/Container.html) in [javax.swing.plaf.metal](http://docs.google.com/javax/swing/plaf/metal/package-summary.html) | |
| --- | --- |

| Subclasses of [Container](http://docs.google.com/java/awt/Container.html) in [javax.swing.plaf.metal](http://docs.google.com/javax/swing/plaf/metal/package-summary.html) | |
| --- | --- |
| class | [**MetalComboBoxButton**](http://docs.google.com/javax/swing/plaf/metal/MetalComboBoxButton.html)            JButton subclass to help out MetalComboBoxUI |
| class | [**MetalComboBoxUI.MetalComboPopup**](http://docs.google.com/javax/swing/plaf/metal/MetalComboBoxUI.MetalComboPopup.html)  **Deprecated.** *As of Java 2 platform v1.4.* |
| protected  class | [**MetalFileChooserUI.FileRenderer**](http://docs.google.com/javax/swing/plaf/metal/MetalFileChooserUI.FileRenderer.html) |
| class | [**MetalFileChooserUI.FilterComboBoxRenderer**](http://docs.google.com/javax/swing/plaf/metal/MetalFileChooserUI.FilterComboBoxRenderer.html)            Render different type sizes and styles. |
| class | [**MetalInternalFrameTitlePane**](http://docs.google.com/javax/swing/plaf/metal/MetalInternalFrameTitlePane.html)            Class that manages a JLF title bar |
| class | [**MetalScrollButton**](http://docs.google.com/javax/swing/plaf/metal/MetalScrollButton.html)            JButton object for Metal scrollbar arrows. |

| Methods in [javax.swing.plaf.metal](http://docs.google.com/javax/swing/plaf/metal/package-summary.html) with parameters of type [Container](http://docs.google.com/java/awt/Container.html) | |
| --- | --- |
| void | **MetalComboBoxUI.**[**layoutComboBox**](http://docs.google.com/javax/swing/plaf/metal/MetalComboBoxUI.html#layoutComboBox(java.awt.Container,%20javax.swing.plaf.metal.MetalComboBoxUI.MetalComboBoxLayoutManager))([Container](http://docs.google.com/java/awt/Container.html) parent, [MetalComboBoxUI.MetalComboBoxLayoutManager](http://docs.google.com/javax/swing/plaf/metal/MetalComboBoxUI.MetalComboBoxLayoutManager.html) manager) |
| void | **MetalComboBoxUI.MetalComboBoxLayoutManager.**[**layoutContainer**](http://docs.google.com/javax/swing/plaf/metal/MetalComboBoxUI.MetalComboBoxLayoutManager.html#layoutContainer(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent) |
| void | **MetalComboBoxUI.MetalComboBoxLayoutManager.**[**superLayout**](http://docs.google.com/javax/swing/plaf/metal/MetalComboBoxUI.MetalComboBoxLayoutManager.html#superLayout(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) parent) |

| Uses of [Container](http://docs.google.com/java/awt/Container.html) in [javax.swing.table](http://docs.google.com/javax/swing/table/package-summary.html) | |
| --- | --- |

| Subclasses of [Container](http://docs.google.com/java/awt/Container.html) in [javax.swing.table](http://docs.google.com/javax/swing/table/package-summary.html) | |
| --- | --- |
| class | [**DefaultTableCellRenderer**](http://docs.google.com/javax/swing/table/DefaultTableCellRenderer.html)            The standard class for rendering (displaying) individual cells in a JTable. |
| static class | [**DefaultTableCellRenderer.UIResource**](http://docs.google.com/javax/swing/table/DefaultTableCellRenderer.UIResource.html)            A subclass of DefaultTableCellRenderer that implements UIResource. |
| class | [**JTableHeader**](http://docs.google.com/javax/swing/table/JTableHeader.html)            This is the object which manages the header of the JTable. |

| Uses of [Container](http://docs.google.com/java/awt/Container.html) in [javax.swing.text](http://docs.google.com/javax/swing/text/package-summary.html) | |
| --- | --- |

| Subclasses of [Container](http://docs.google.com/java/awt/Container.html) in [javax.swing.text](http://docs.google.com/javax/swing/text/package-summary.html) | |
| --- | --- |
| class | [**JTextComponent**](http://docs.google.com/javax/swing/text/JTextComponent.html)            JTextComponent is the base class for swing text components. |

| Methods in [javax.swing.text](http://docs.google.com/javax/swing/text/package-summary.html) that return [Container](http://docs.google.com/java/awt/Container.html) | |
| --- | --- |
| [Container](http://docs.google.com/java/awt/Container.html) | **View.**[**getContainer**](http://docs.google.com/javax/swing/text/View.html#getContainer())()            Fetches the container hosting the view. |

| Uses of [Container](http://docs.google.com/java/awt/Container.html) in [javax.swing.tree](http://docs.google.com/javax/swing/tree/package-summary.html) | |
| --- | --- |

| Subclasses of [Container](http://docs.google.com/java/awt/Container.html) in [javax.swing.tree](http://docs.google.com/javax/swing/tree/package-summary.html) | |
| --- | --- |
| class | [**DefaultTreeCellEditor.DefaultTextField**](http://docs.google.com/javax/swing/tree/DefaultTreeCellEditor.DefaultTextField.html)            TextField used when no editor is supplied. |
| class | [**DefaultTreeCellEditor.EditorContainer**](http://docs.google.com/javax/swing/tree/DefaultTreeCellEditor.EditorContainer.html)            Container responsible for placing the editingComponent. |
| class | [**DefaultTreeCellRenderer**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html)            Displays an entry in a tree. |

| Fields in [javax.swing.tree](http://docs.google.com/javax/swing/tree/package-summary.html) declared as [Container](http://docs.google.com/java/awt/Container.html) | |
| --- | --- |
| protected  [Container](http://docs.google.com/java/awt/Container.html) | **DefaultTreeCellEditor.**[**editingContainer**](http://docs.google.com/javax/swing/tree/DefaultTreeCellEditor.html#editingContainer)            Editing container, will contain the editorComponent. |

| Methods in [javax.swing.tree](http://docs.google.com/javax/swing/tree/package-summary.html) that return [Container](http://docs.google.com/java/awt/Container.html) | |
| --- | --- |
| protected  [Container](http://docs.google.com/java/awt/Container.html) | **DefaultTreeCellEditor.**[**createContainer**](http://docs.google.com/javax/swing/tree/DefaultTreeCellEditor.html#createContainer())()            Creates the container to manage placement of editingComponent. |

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | [**Class**](http://docs.google.com/java/awt/Container.html) | **Use** | [**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   NEXT | [**FRAMES**](http://docs.google.com/index.html?java/awt//class-useContainer.html)    [**NO FRAMES**](http://docs.google.com/Container.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |

[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).