| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/SystemTray.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV CLASS**](http://docs.google.com/java/awt/SystemColor.html)   [**NEXT CLASS**](http://docs.google.com/java/awt/TextArea.html) | [**FRAMES**](http://docs.google.com/index.html?java/awt/SystemTray.html)    [**NO FRAMES**](http://docs.google.com/SystemTray.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#3znysh7) | DETAIL: FIELD | CONSTR | [METHOD](#tyjcwt) |

## **java.awt**

Class SystemTray

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

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

The SystemTray class represents the system tray for a desktop. On Microsoft Windows it is referred to as the "Taskbar Status Area", on Gnome it is referred to as the "Notification Area", on KDE it is referred to as the "System Tray". The system tray is shared by all applications running on the desktop.

On some platforms the system tray may not be present or may not be supported, in this case [getSystemTray()](http://docs.google.com/java/awt/SystemTray.html#getSystemTray()) throws [UnsupportedOperationException](http://docs.google.com/java/lang/UnsupportedOperationException.html). To detect whether the system tray is supported, use [isSupported()](http://docs.google.com/java/awt/SystemTray.html#isSupported()).

The SystemTray may contain one or more [TrayIcons](http://docs.google.com/java/awt/TrayIcon.html), which are added to the tray using the [add(java.awt.TrayIcon)](http://docs.google.com/java/awt/SystemTray.html#add(java.awt.TrayIcon)) method, and removed when no longer needed, using the [remove(java.awt.TrayIcon)](http://docs.google.com/java/awt/SystemTray.html#remove(java.awt.TrayIcon)). TrayIcon consists of an image, a popup menu and a set of associated listeners. Please see the [TrayIcon](http://docs.google.com/java/awt/TrayIcon.html) class for details.

Every Java application has a single SystemTray instance that allows the app to interface with the system tray of the desktop while the app is running. The SystemTray instance can be obtained from the [getSystemTray()](http://docs.google.com/java/awt/SystemTray.html#getSystemTray()) method. An application may not create its own instance of SystemTray.

The following code snippet demonstrates how to access and customize the system tray:

[TrayIcon](http://docs.google.com/java/awt/TrayIcon.html) trayIcon = null;  
 if (SystemTray.isSupported()) {  
 // get the SystemTray instance  
 SystemTray tray = SystemTray.[getSystemTray()](http://docs.google.com/java/awt/SystemTray.html#getSystemTray());  
 // load an image  
 [Image](http://docs.google.com/java/awt/Image.html) image = [Toolkit.getDefaultToolkit.getImage](http://docs.google.com/java/awt/Toolkit.html#getImage(java.lang.String))(...);  
 // create a action listener to listen for default action executed on the tray icon  
 [ActionListener](http://docs.google.com/java/awt/event/ActionListener.html) listener = new [ActionListener](http://docs.google.com/java/awt/event/ActionListener.html)() {  
 public void [actionPerformed](http://docs.google.com/java/awt/event/ActionListener.html#actionPerformed(java.awt.event.ActionEvent))([ActionEvent](http://docs.google.com/java/awt/event/ActionEvent.html) e) {  
 // execute default action of the application  
 // ...  
 }  
 };  
 // create a popup menu  
 [PopupMenu](http://docs.google.com/java/awt/PopupMenu.html) popup = new [PopupMenu](http://docs.google.com/java/awt/PopupMenu.html#PopupMenu(java.lang.String))();  
 // create menu item for the default action  
 MenuItem defaultItem = new MenuItem(...);  
 defaultItem.addActionListener(listener);  
 popup.add(defaultItem);  
 /// ... add other items  
 // construct a TrayIcon  
 trayIcon = new [TrayIcon](http://docs.google.com/java/awt/TrayIcon.html#TrayIcon(java.awt.Image,%20java.lang.String,%20java.awt.PopupMenu))(image, "Tray Demo", popup);  
 // set the TrayIcon properties  
 trayIcon.[addActionListener](http://docs.google.com/java/awt/TrayIcon.html#addActionListener(java.awt.event.ActionListener))(listener);  
 // ...  
 // add the tray image  
 try {  
 tray.[add](http://docs.google.com/java/awt/SystemTray.html#add(java.awt.TrayIcon))(trayIcon);  
 } catch (AWTException e) {  
 System.err.println(e);  
 }  
 // ...  
 } else {  
 // disable tray option in your application or  
 // perform other actions  
 ...  
 }  
 // ...  
 // some time later  
 // the application state has changed - update the image  
 if (trayIcon != null) {  
 trayIcon.[setImage](http://docs.google.com/java/awt/TrayIcon.html#setImage(java.awt.Image))(updatedImage);  
 }  
 // ...

**Since:** 1.6 **See Also:**[TrayIcon](http://docs.google.com/java/awt/TrayIcon.html)

| **Method Summary** | |
| --- | --- |
| void | [**add**](http://docs.google.com/java/awt/SystemTray.html#add(java.awt.TrayIcon))([TrayIcon](http://docs.google.com/java/awt/TrayIcon.html) trayIcon)            Adds a TrayIcon to the SystemTray. |
| void | [**addPropertyChangeListener**](http://docs.google.com/java/awt/SystemTray.html#addPropertyChangeListener(java.lang.String,%20java.beans.PropertyChangeListener))([String](http://docs.google.com/java/lang/String.html) propertyName, [PropertyChangeListener](http://docs.google.com/java/beans/PropertyChangeListener.html) listener)            Adds a PropertyChangeListener to the listener list for a specific property. |
| [PropertyChangeListener](http://docs.google.com/java/beans/PropertyChangeListener.html)[] | [**getPropertyChangeListeners**](http://docs.google.com/java/awt/SystemTray.html#getPropertyChangeListeners(java.lang.String))([String](http://docs.google.com/java/lang/String.html) propertyName)            Returns an array of all the listeners that have been associated with the named property. |
| static [SystemTray](http://docs.google.com/java/awt/SystemTray.html) | [**getSystemTray**](http://docs.google.com/java/awt/SystemTray.html#getSystemTray())()            Gets the SystemTray instance that represents the desktop's tray area. |
| [TrayIcon](http://docs.google.com/java/awt/TrayIcon.html)[] | [**getTrayIcons**](http://docs.google.com/java/awt/SystemTray.html#getTrayIcons())()            Returns an array of all icons added to the tray by this application. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | [**getTrayIconSize**](http://docs.google.com/java/awt/SystemTray.html#getTrayIconSize())()            Returns the size, in pixels, of the space that a tray icon will occupy in the system tray. |
| static boolean | [**isSupported**](http://docs.google.com/java/awt/SystemTray.html#isSupported())()            Returns whether the system tray is supported on the current platform. |
| void | [**remove**](http://docs.google.com/java/awt/SystemTray.html#remove(java.awt.TrayIcon))([TrayIcon](http://docs.google.com/java/awt/TrayIcon.html) trayIcon)            Removes the specified TrayIcon from the SystemTray. |
| void | [**removePropertyChangeListener**](http://docs.google.com/java/awt/SystemTray.html#removePropertyChangeListener(java.lang.String,%20java.beans.PropertyChangeListener))([String](http://docs.google.com/java/lang/String.html) propertyName, [PropertyChangeListener](http://docs.google.com/java/beans/PropertyChangeListener.html) listener)            Removes a PropertyChangeListener from the listener list for a specific property. |

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

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

### getSystemTray

public static [SystemTray](http://docs.google.com/java/awt/SystemTray.html) **getSystemTray**()

Gets the SystemTray instance that represents the desktop's tray area. This always returns the same instance per application. On some platforms the system tray may not be supported. You may use the [isSupported()](http://docs.google.com/java/awt/SystemTray.html#isSupported()) method to check if the system tray is supported.

If a SecurityManager is installed, the AWTPermission accessSystemTray must be granted in order to get the SystemTray instance. Otherwise this method will throw a SecurityException.

**Returns:**the SystemTray instance that represents the desktop's tray area **Throws:** [UnsupportedOperationException](http://docs.google.com/java/lang/UnsupportedOperationException.html) - if the system tray isn't supported by the current platform [HeadlessException](http://docs.google.com/java/awt/HeadlessException.html) - if GraphicsEnvironment.isHeadless() returns true [SecurityException](http://docs.google.com/java/lang/SecurityException.html) - if accessSystemTray permission is not granted**See Also:**[add(TrayIcon)](http://docs.google.com/java/awt/SystemTray.html#add(java.awt.TrayIcon)), [TrayIcon](http://docs.google.com/java/awt/TrayIcon.html), [isSupported()](http://docs.google.com/java/awt/SystemTray.html#isSupported()), [SecurityManager.checkPermission(java.security.Permission)](http://docs.google.com/java/lang/SecurityManager.html#checkPermission(java.security.Permission)), [AWTPermission](http://docs.google.com/java/awt/AWTPermission.html)

### isSupported

public static boolean **isSupported**()

Returns whether the system tray is supported on the current platform. In addition to displaying the tray icon, minimal system tray support includes either a popup menu (see [TrayIcon.setPopupMenu(PopupMenu)](http://docs.google.com/java/awt/TrayIcon.html#setPopupMenu(java.awt.PopupMenu))) or an action event (see [TrayIcon.addActionListener(ActionListener)](http://docs.google.com/java/awt/TrayIcon.html#addActionListener(java.awt.event.ActionListener))).

Developers should not assume that all of the system tray functionality is supported. To guarantee that the tray icon's default action is always accessible, add the default action to both the action listener and the popup menu. See the [example](http://docs.google.com/java/awt/SystemTray.html) for an example of how to do this.

**Note**: When implementing SystemTray and TrayIcon it is *strongly recommended* that you assign different gestures to the popup menu and an action event. Overloading a gesture for both purposes is confusing and may prevent the user from accessing one or the other.

**Returns:**false if no system tray access is supported; this method returns true if the minimal system tray access is supported but does not guarantee that all system tray functionality is supported for the current platform**See Also:**[getSystemTray()](http://docs.google.com/java/awt/SystemTray.html#getSystemTray())

### add

public void **add**([TrayIcon](http://docs.google.com/java/awt/TrayIcon.html) trayIcon)  
 throws [AWTException](http://docs.google.com/java/awt/AWTException.html)

Adds a TrayIcon to the SystemTray. The tray icon becomes visible in the system tray once it is added. The order in which icons are displayed in a tray is not specified - it is platform and implementation-dependent.

All icons added by the application are automatically removed from the SystemTray upon application exit and also when the desktop system tray becomes unavailable.

**Parameters:**trayIcon - the TrayIcon to be added **Throws:** [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if trayIcon is null [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if the same instance of a TrayIcon is added more than once [AWTException](http://docs.google.com/java/awt/AWTException.html) - if the desktop system tray is missing**See Also:**[remove(TrayIcon)](http://docs.google.com/java/awt/SystemTray.html#remove(java.awt.TrayIcon)), [getSystemTray()](http://docs.google.com/java/awt/SystemTray.html#getSystemTray()), [TrayIcon](http://docs.google.com/java/awt/TrayIcon.html), [Image](http://docs.google.com/java/awt/Image.html)

### remove

public void **remove**([TrayIcon](http://docs.google.com/java/awt/TrayIcon.html) trayIcon)

Removes the specified TrayIcon from the SystemTray.

All icons added by the application are automatically removed from the SystemTray upon application exit and also when the desktop system tray becomes unavailable.

If trayIcon is null or was not added to the system tray, no exception is thrown and no action is performed.

**Parameters:**trayIcon - the TrayIcon to be removed**See Also:**[add(TrayIcon)](http://docs.google.com/java/awt/SystemTray.html#add(java.awt.TrayIcon)), [TrayIcon](http://docs.google.com/java/awt/TrayIcon.html)

### getTrayIcons

public [TrayIcon](http://docs.google.com/java/awt/TrayIcon.html)[] **getTrayIcons**()

Returns an array of all icons added to the tray by this application. You can't access the icons added by another application. Some browsers partition applets in different code bases into separate contexts, and establish walls between these contexts. In such a scenario, only the tray icons added from this context will be returned.

The returned array is a copy of the actual array and may be modified in any way without affecting the system tray. To remove a TrayIcon from the SystemTray, use the [remove(TrayIcon)](http://docs.google.com/java/awt/SystemTray.html#remove(java.awt.TrayIcon)) method.

**Returns:**an array of all tray icons added to this tray, or an empty array if none has been added**See Also:**[add(TrayIcon)](http://docs.google.com/java/awt/SystemTray.html#add(java.awt.TrayIcon)), [TrayIcon](http://docs.google.com/java/awt/TrayIcon.html)

### getTrayIconSize

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

Returns the size, in pixels, of the space that a tray icon will occupy in the system tray. Developers may use this methods to acquire the preferred size for the image property of a tray icon before it is created. For convenience, there is a similar method [TrayIcon.getSize()](http://docs.google.com/java/awt/TrayIcon.html#getSize()) in the TrayIcon class.

**Returns:**the default size of a tray icon, in pixels**See Also:**[TrayIcon.setImageAutoSize(boolean)](http://docs.google.com/java/awt/TrayIcon.html#setImageAutoSize(boolean)), [Image](http://docs.google.com/java/awt/Image.html), [TrayIcon.getSize()](http://docs.google.com/java/awt/TrayIcon.html#getSize())

### addPropertyChangeListener

public void **addPropertyChangeListener**([String](http://docs.google.com/java/lang/String.html) propertyName,  
 [PropertyChangeListener](http://docs.google.com/java/beans/PropertyChangeListener.html) listener)

Adds a PropertyChangeListener to the listener list for a specific property. Currently supported property:

* trayIcons  
    
  This SystemTray's array of TrayIcons. The array is accessed via [getTrayIcons()](http://docs.google.com/java/awt/SystemTray.html#getTrayIcons()).  
  This property is changed when a TrayIcon is added to (or removed from) the SystemTray.  
  For example, this property is changed when the native SystemTray becomes unavailable on the desktop  
  and the TrayIcons are automatically removed.

The listener listens to property changes only in this context.

If listener is null, no exception is thrown and no action is performed.

**Parameters:**propertyName - the specified propertylistener - the property change listener to be added**See Also:**[removePropertyChangeListener(java.lang.String, java.beans.PropertyChangeListener)](http://docs.google.com/java/awt/SystemTray.html#removePropertyChangeListener(java.lang.String,%20java.beans.PropertyChangeListener)), [getPropertyChangeListeners(java.lang.String)](http://docs.google.com/java/awt/SystemTray.html#getPropertyChangeListeners(java.lang.String))

### removePropertyChangeListener

public void **removePropertyChangeListener**([String](http://docs.google.com/java/lang/String.html) propertyName,  
 [PropertyChangeListener](http://docs.google.com/java/beans/PropertyChangeListener.html) listener)

Removes a PropertyChangeListener from the listener list for a specific property.

The PropertyChangeListener must be from this context.

If propertyName or listener is null or invalid, no exception is thrown and no action is taken.

**Parameters:**propertyName - the specified propertylistener - the PropertyChangeListener to be removed**See Also:**[addPropertyChangeListener(java.lang.String, java.beans.PropertyChangeListener)](http://docs.google.com/java/awt/SystemTray.html#addPropertyChangeListener(java.lang.String,%20java.beans.PropertyChangeListener)), [getPropertyChangeListeners(java.lang.String)](http://docs.google.com/java/awt/SystemTray.html#getPropertyChangeListeners(java.lang.String))

### getPropertyChangeListeners

public [PropertyChangeListener](http://docs.google.com/java/beans/PropertyChangeListener.html)[] **getPropertyChangeListeners**([String](http://docs.google.com/java/lang/String.html) propertyName)

Returns an array of all the listeners that have been associated with the named property.

Only the listeners in this context are returned.

**Parameters:**propertyName - the specified property **Returns:**all of the PropertyChangeListeners associated with the named property; if no such listeners have been added or if propertyName is null or invalid, an empty array is returned**See Also:**[addPropertyChangeListener(java.lang.String, java.beans.PropertyChangeListener)](http://docs.google.com/java/awt/SystemTray.html#addPropertyChangeListener(java.lang.String,%20java.beans.PropertyChangeListener)), [removePropertyChangeListener(java.lang.String, java.beans.PropertyChangeListener)](http://docs.google.com/java/awt/SystemTray.html#removePropertyChangeListener(java.lang.String,%20java.beans.PropertyChangeListener))

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/SystemTray.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV CLASS**](http://docs.google.com/java/awt/SystemColor.html)   [**NEXT CLASS**](http://docs.google.com/java/awt/TextArea.html) | [**FRAMES**](http://docs.google.com/index.html?java/awt/SystemTray.html)    [**NO FRAMES**](http://docs.google.com/SystemTray.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#3znysh7) | DETAIL: FIELD | CONSTR | [METHOD](#tyjcwt) |

[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).