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

## **java.awt**

Class Desktop

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

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

The Desktop class allows a Java application to launch associated applications registered on the native desktop to handle a [URI](http://docs.google.com/java/net/URI.html) or a file.

Supported operations include:

* launching the user-default browser to show a specified URI;
* launching the user-default mail client with an optional mailto URI;
* launching a registered application to open, edit or print a specified file.

This class provides methods corresponding to these operations. The methods look for the associated application registered on the current platform, and launch it to handle a URI or file. If there is no associated application or the associated application fails to be launched, an exception is thrown.

An application is registered to a URI or file type; for example, the "sxi" file extension is typically registered to StarOffice. The mechanism of registereing, accessing, and launching the associated application is platform-dependent.

Each operation is an action type represented by the [Desktop.Action](http://docs.google.com/java/awt/Desktop.Action.html) class.

Note: when some action is invoked and the associated application is executed, it will be executed on the same system as the one on which the Java application was launched.

**Since:** 1.6

| **Nested Class Summary** | |
| --- | --- |
| static class | [**Desktop.Action**](http://docs.google.com/java/awt/Desktop.Action.html)            Represents an action type. |

| **Method Summary** | |
| --- | --- |
| void | [**browse**](http://docs.google.com/java/awt/Desktop.html#browse(java.net.URI))([URI](http://docs.google.com/java/net/URI.html) uri)            Launches the default browser to display a URI. |
| void | [**edit**](http://docs.google.com/java/awt/Desktop.html#edit(java.io.File))([File](http://docs.google.com/java/io/File.html) file)            Launches the associated editor application and opens a file for editing. |
| static [Desktop](http://docs.google.com/java/awt/Desktop.html) | [**getDesktop**](http://docs.google.com/java/awt/Desktop.html#getDesktop())()            Returns the Desktop instance of the current browser context. |
| static boolean | [**isDesktopSupported**](http://docs.google.com/java/awt/Desktop.html#isDesktopSupported())()            Tests whether this class is supported on the current platform. |
| boolean | [**isSupported**](http://docs.google.com/java/awt/Desktop.html#isSupported(java.awt.Desktop.Action))([Desktop.Action](http://docs.google.com/java/awt/Desktop.Action.html) action)            Tests whether an action is supported on the current platform. |
| void | [**mail**](http://docs.google.com/java/awt/Desktop.html#mail())()            Launches the mail composing window of the user default mail client. |
| void | [**mail**](http://docs.google.com/java/awt/Desktop.html#mail(java.net.URI))([URI](http://docs.google.com/java/net/URI.html) mailtoURI)            Launches the mail composing window of the user default mail client, filling the message fields specified by a mailto: URI. |
| void | [**open**](http://docs.google.com/java/awt/Desktop.html#open(java.io.File))([File](http://docs.google.com/java/io/File.html) file)            Launches the associated application to open the file. |
| void | [**print**](http://docs.google.com/java/awt/Desktop.html#print(java.io.File))([File](http://docs.google.com/java/io/File.html) file)            Prints a file with the native desktop printing facility, using the associated application's print command. |

| **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** |
| --- |

### getDesktop

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

Returns the Desktop instance of the current browser context. On some platforms the Desktop API may not be supported; use the [isDesktopSupported()](http://docs.google.com/java/awt/Desktop.html#isDesktopSupported()) method to determine if the current desktop is supported.

**Returns:**the Desktop instance of the current browser context **Throws:** [HeadlessException](http://docs.google.com/java/awt/HeadlessException.html) - if [GraphicsEnvironment.isHeadless()](http://docs.google.com/java/awt/GraphicsEnvironment.html#isHeadless()) returns true [UnsupportedOperationException](http://docs.google.com/java/lang/UnsupportedOperationException.html) - if this class is not supported on the current platform**See Also:**[isDesktopSupported()](http://docs.google.com/java/awt/Desktop.html#isDesktopSupported()), [GraphicsEnvironment.isHeadless()](http://docs.google.com/java/awt/GraphicsEnvironment.html#isHeadless())

### isDesktopSupported

public static boolean **isDesktopSupported**()

Tests whether this class is supported on the current platform. If it's supported, use [getDesktop()](http://docs.google.com/java/awt/Desktop.html#getDesktop()) to retrieve an instance.

**Returns:**true if this class is supported on the current platform; false otherwise**See Also:**[getDesktop()](http://docs.google.com/java/awt/Desktop.html#getDesktop())

### isSupported

public boolean **isSupported**([Desktop.Action](http://docs.google.com/java/awt/Desktop.Action.html) action)

Tests whether an action is supported on the current platform.

Even when the platform supports an action, a file or URI may not have a registered application for the action. For example, most of the platforms support the [Desktop.Action.OPEN](http://docs.google.com/java/awt/Desktop.Action.html#OPEN) action. But for a specific file, there may not be an application registered to open it. In this case, [isSupported(java.awt.Desktop.Action)](http://docs.google.com/java/awt/Desktop.html#isSupported(java.awt.Desktop.Action)) may return true, but the corresponding action method will throw an [IOException](http://docs.google.com/java/io/IOException.html).

**Parameters:**action - the specified [Desktop.Action](http://docs.google.com/java/awt/Desktop.Action.html) **Returns:**true if the specified action is supported on the current platform; false otherwise**See Also:**[Desktop.Action](http://docs.google.com/java/awt/Desktop.Action.html)

### open

public void **open**([File](http://docs.google.com/java/io/File.html) file)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Launches the associated application to open the file.

If the specified file is a directory, the file manager of the current platform is launched to open it.

**Parameters:**file - the file to be opened with the associated application **Throws:** [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if file is null [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if the specified file dosen't exist [UnsupportedOperationException](http://docs.google.com/java/lang/UnsupportedOperationException.html) - if the current platform does not support the [Desktop.Action.OPEN](http://docs.google.com/java/awt/Desktop.Action.html#OPEN) action [IOException](http://docs.google.com/java/io/IOException.html) - if the specified file has no associated application or the associated application fails to be launched [SecurityException](http://docs.google.com/java/lang/SecurityException.html) - if a security manager exists and its [SecurityManager.checkRead(java.lang.String)](http://docs.google.com/java/lang/SecurityManager.html#checkRead(java.lang.String)) method denies read access to the file, or it denies the AWTPermission("showWindowWithoutWarningBanner") permission, or the calling thread is not allowed to create a subprocess**See Also:**[AWTPermission](http://docs.google.com/java/awt/AWTPermission.html)

### edit

public void **edit**([File](http://docs.google.com/java/io/File.html) file)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Launches the associated editor application and opens a file for editing.

**Parameters:**file - the file to be opened for editing **Throws:** [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if the specified file is null [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if the specified file doesn't exist [UnsupportedOperationException](http://docs.google.com/java/lang/UnsupportedOperationException.html) - if the current platform does not support the [Desktop.Action.EDIT](http://docs.google.com/java/awt/Desktop.Action.html#EDIT) action [IOException](http://docs.google.com/java/io/IOException.html) - if the specified file has no associated editor, or the associated application fails to be launched [SecurityException](http://docs.google.com/java/lang/SecurityException.html) - if a security manager exists and its [SecurityManager.checkRead(java.lang.String)](http://docs.google.com/java/lang/SecurityManager.html#checkRead(java.lang.String)) method denies read access to the file, or [SecurityManager.checkWrite(java.lang.String)](http://docs.google.com/java/lang/SecurityManager.html#checkWrite(java.lang.String)) method denies write access to the file, or it denies the AWTPermission("showWindowWithoutWarningBanner") permission, or the calling thread is not allowed to create a subprocess**See Also:**[AWTPermission](http://docs.google.com/java/awt/AWTPermission.html)

### print

public void **print**([File](http://docs.google.com/java/io/File.html) file)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Prints a file with the native desktop printing facility, using the associated application's print command.

**Parameters:**file - the file to be printed **Throws:** [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if the specified file is null [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if the specified file doesn't exist [UnsupportedOperationException](http://docs.google.com/java/lang/UnsupportedOperationException.html) - if the current platform does not support the [Desktop.Action.PRINT](http://docs.google.com/java/awt/Desktop.Action.html#PRINT) action [IOException](http://docs.google.com/java/io/IOException.html) - if the specified file has no associated application that can be used to print it [SecurityException](http://docs.google.com/java/lang/SecurityException.html) - if a security manager exists and its [SecurityManager.checkRead(java.lang.String)](http://docs.google.com/java/lang/SecurityManager.html#checkRead(java.lang.String)) method denies read access to the file, or its [SecurityManager.checkPrintJobAccess()](http://docs.google.com/java/lang/SecurityManager.html#checkPrintJobAccess()) method denies the permission to print the file, or the calling thread is not allowed to create a subprocess

### browse

public void **browse**([URI](http://docs.google.com/java/net/URI.html) uri)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Launches the default browser to display a URI. If the default browser is not able to handle the specified URI, the application registered for handling URIs of the specified type is invoked. The application is determined from the protocol and path of the URI, as defined by the URI class.

If the calling thread does not have the necessary permissions, and this is invoked from within an applet, AppletContext.showDocument() is used. Similarly, if the calling does not have the necessary permissions, and this is invoked from within a Java Web Started application, BasicService.showDocument() is used.

**Parameters:**uri - the URI to be displayed in the user default browser **Throws:** [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if uri is null [UnsupportedOperationException](http://docs.google.com/java/lang/UnsupportedOperationException.html) - if the current platform does not support the [Desktop.Action.BROWSE](http://docs.google.com/java/awt/Desktop.Action.html#BROWSE) action [IOException](http://docs.google.com/java/io/IOException.html) - if the user default browser is not found, or it fails to be launched, or the default handler application failed to be launched [SecurityException](http://docs.google.com/java/lang/SecurityException.html) - if a security manager exists and it denies the AWTPermission("showWindowWithoutWarningBanner") permission, or the calling thread is not allowed to create a subprocess; and not invoked from within an applet or Java Web Started application [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if the necessary permissions are not available and the URI can not be converted to a URL**See Also:**[URI](http://docs.google.com/java/net/URI.html), [AWTPermission](http://docs.google.com/java/awt/AWTPermission.html), [AppletContext](http://docs.google.com/java/applet/AppletContext.html)

### mail

public void **mail**()  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Launches the mail composing window of the user default mail client.

**Throws:** [UnsupportedOperationException](http://docs.google.com/java/lang/UnsupportedOperationException.html) - if the current platform does not support the [Desktop.Action.MAIL](http://docs.google.com/java/awt/Desktop.Action.html#MAIL) action [IOException](http://docs.google.com/java/io/IOException.html) - if the user default mail client is not found, or it fails to be launched [SecurityException](http://docs.google.com/java/lang/SecurityException.html) - if a security manager exists and it denies the AWTPermission("showWindowWithoutWarningBanner") permission, or the calling thread is not allowed to create a subprocess**See Also:**[AWTPermission](http://docs.google.com/java/awt/AWTPermission.html)

### mail

public void **mail**([URI](http://docs.google.com/java/net/URI.html) mailtoURI)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Launches the mail composing window of the user default mail client, filling the message fields specified by a mailto: URI.

A mailto: URI can specify message fields including *"to"*, *"cc"*, *"subject"*, *"body"*, etc. See [The mailto URL scheme (RFC 2368)](http://www.ietf.org/rfc/rfc2368.txt) for the mailto: URI specification details.

**Parameters:**mailtoURI - the specified mailto: URI **Throws:** [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if the specified URI is null [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if the URI scheme is not "mailto" [UnsupportedOperationException](http://docs.google.com/java/lang/UnsupportedOperationException.html) - if the current platform does not support the [Desktop.Action.MAIL](http://docs.google.com/java/awt/Desktop.Action.html#MAIL) action [IOException](http://docs.google.com/java/io/IOException.html) - if the user default mail client is not found or fails to be launched [SecurityException](http://docs.google.com/java/lang/SecurityException.html) - if a security manager exists and it denies the AWTPermission("showWindowWithoutWarningBanner") permission, or the calling thread is not allowed to create a subprocess**See Also:**[URI](http://docs.google.com/java/net/URI.html), [AWTPermission](http://docs.google.com/java/awt/AWTPermission.html)

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

[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).