Index of all Fields and Methods

A

ABORT. Static variable in interface java.awt.image.ImageObserver
An image which was being tracked asynchronously was aborted before production was complete.
ABORTED. Static variable in class java.awt.MediaTracker
Flag indicating the download of some media was aborted.
abs(double). Static method in class java.lang.Math
Returns the absolute double value of a.
abs(float). Static method in class java.lang.Math
Returns the absolute float value of a.
abs(int). Static method in class java.lang.Math
Returns the absolute integer value of a.
abs(long). Static method in class java.lang.Math
Returns the absolute long value of a.
AbstractMethodError(). Constructor for class java.lang.AbstractMethodError
Constructs an AbstractMethodError with no detail message.
AbstractMethodError(String). Constructor for class java.lang.AbstractMethodError
Constructs an AbstractMethodError with the specified detail message.
accept(). Method in class java.net.ServerSocket
Accepts a connection.
accept(File, String). Method in interface java.io.FilenameFilter
Determines whether a name should be included in a file list.
accept(SocketImpl). Method in class java.net.SocketImpl
Accepts a connection.
acos(double). Static method in class java.lang.Math
Returns the arc cosine of a, in the range of 0.0 through Pi.
action(Event, Object). Method in class java.awt.Component
Called if an action occurs in the Component.
ACTION_EVENT. Static variable in class java.awt.Event
An action event.
activeCount(). Static method in class java.lang.Thread
Returns the current number of active Threads in this Thread group.
activeCount(). Method in class java.lang.ThreadGroup
Returns an estimate of the number of active Threads in the Thread group.
activeGroupCount(). Method in class java.lang.ThreadGroup
Returns an estimate of the number of active groups in the Thread group.
add(Component). Method in class java.awt.Container
Adds the specified component to this container.

add(Component, int). Method in class java.awt.Container
Adds the specified component to this container at the given position.
add(int, int). Method in class java.awt.Rectangle
Add a point to a rectangle.
add(Menu). Method in class java.awt.MenuBar
Adds the specified menu to the menu bar.
add(MenuItem). Method in class java.awt.Menu
Adds the specified item to this menu.
add(Point). Method in class java.awt.Rectangle
Add a point to a rectangle.
add(Rectangle). Method in class java.awt.Rectangle
Add a rectangle to a rectangle.
add(String). Method in class java.awt.Menu
Add an item with with the specified label to this menu.
add(String, Component). Method in class java.awt.Container
Adds the specified component to this container.
addConsumer(ImageConsumer). Method in class java.awt.image.FilteredImageSource
Add an ImageConsumer to the list of consumers interested in data for this image.
addConsumer(ImageConsumer). Method in interface java.awt.image.ImageProducer
This method is used to register an ImageConsumer with the ImageProducer for access to the image data during a later reconstruction of the Image.
addConsumer(ImageConsumer). Method in class java.awt.image.MemoryImageSource
Add an ImageConsumer to the list of consumers interested in data for this image.
addElement(Object). Method in class java.util.Vector
Adds the specified object as the last element of the vector.
addHelpMenu(Menu). Method in interface java.awt.peer.MenuBarPeer
addImage(Image, int). Method in class java.awt.MediaTracker
Add an image to the list of images being tracked.
addImage(Image, int, int, int). Method in class java.awt.MediaTracker
Add a scaled image to the list of images being tracked.
addItem(MenuItem). Method in interface java.awt.peer.MenuPeer
addItem(String). Method in class java.awt.Choice
Adds an item to this Choice.
addItem(String). Method in class java.awt.List
Adds the specified item to the end of scrolling list.
addItem(String, int). Method in interface java.awt.peer.ChoicePeer
addItem(String, int). Method in class java.awt.List
Adds the specified item to the end of scrolling list.
addItem(String, int). Method in interface java.awt.peer.ListPeer
addLayoutComponent(String, Component). Method in class java.awt.BorderLayout
Adds the specified named component to the layout.
addLayoutComponent(String, Component). Method in class java.awt.CardLayout
Adds the specified component with the specified name to the layout.
addLayoutComponent(String, Component). Method in class java.awt.FlowLayout
Adds the specified component to the layout.
addLayoutComponent(String, Component). Method in class java.awt.GridBagLayout
Adds the specified component with the specified name to the layout.
addLayoutComponent(String, Component). Method in class java.awt.GridLayout
Adds the specified component with the specified name to the layout.

addLayoutComponent(String, Component). Method in interface java.awt.LayoutManager
Adds the specified component with the specified name to the layout.
addMenu(Menu). Method in interface java.awt.peer.MenuBarPeer
addNotify(). Method in class java.awt.Button
Creates the peer of the button.
addNotify(). Method in class java.awt.Canvas
Creates the peer of the canvas.
addNotify(). Method in class java.awt.Checkbox
Creates the peer of the Checkbox.
addNotify(). Method in class java.awt.CheckboxMenuItem
Creates the peer of the checkbox item.
addNotify(). Method in class java.awt.Choice
Creates the Choice's peer.
addNotify(). Method in class java.awt.Component
Notifies the Component to create a peer.
addNotify(). Method in class java.awt.Container
Notifies the container to create a peer.
addNotify(). Method in class java.awt.Dialog
Creates the frame's peer.
addNotify(). Method in class java.awt.FileDialog
Creates the frame's peer.
addNotify(). Method in class java.awt.Frame
Creates the Frame's peer.
addNotify(). Method in class java.awt.Label
Creates the peer for this label.
addNotify(). Method in class java.awt.List
Creates the peer for the list.
addNotify(). Method in class java.awt.Menu
Creates the menu's peer.
addNotify(). Method in class java.awt.MenuBar
Creates the menu bar's peer.
addNotify(). Method in class java.awt.MenuItem
Creates the menu item's peer.
addNotify(). Method in class java.awt.Panel
Creates the Panel's peer.
addNotify(). Method in class java.awt.Scrollbar
Create the Scrollbar's peer.
addNotify(). Method in class java.awt.TextArea
Creates the TextArea's peer.
addNotify(). Method in class java.awt.TextField
Creates the TextField's peer.
addNotify(). Method in class java.awt.Window
Creates the Window's peer.
addObserver(Observer). Method in class java.util.Observable
Adds an observer to the observer list.
addPoint(int, int). Method in class java.awt.Polygon
Appends a point to a polygon.
address. Variable in class java.net.SocketImpl
The internet address where the socket will make connection.

addSeparator(). Method in class java.awt.Menu
    Adds a separator line, or a hypen, to the menu at the current position.
addSeparator(). Method in interface java.awt.peer.MenuPeer
AdjustForGravity(GridBagConstraints, Rectangle). Method in class java.awt.GridBagLayout
after(Date). Method in class java.util.Date
    Checks whether this date comes after the specified date.
ALLBITS. Static variable in interface java.awt.image.ImageObserver
    A static image which was previously drawn is now complete and can be drawn again in its final form.
allowsMultipleSelections(). Method in class java.awt.List
    Returns true if this list allows multiple selections.
allowUserInteraction. Variable in class java.net.URLConnection
ALT_MASK. Static variable in class java.awt.Event
    The alt modifier constant.
anchor. Variable in class java.awt.GridBagConstraints
and(BitSet). Method in class java.util.BitSet
    Logically ANDs this bit set with the specified set of bits.
append(boolean). Method in class java.lang.StringBuffer
    Appends a boolean to the end of this buffer.
append(char). Method in class java.lang.StringBuffer
    Appends a character to the end of this buffer.
append(char[]). Method in class java.lang.StringBuffer
    Appends an array of characters to the end of this buffer.
append(char[], int, int). Method in class java.lang.StringBuffer
    Appends a part of an array of characters to the end of this buffer.
append(double). Method in class java.lang.StringBuffer
    Appends a double to the end of this buffer.
append(float). Method in class java.lang.StringBuffer
    Appends a float to the end of this buffer.
append(int). Method in class java.lang.StringBuffer
    Appends an integer to the end of this buffer.
append(long). Method in class java.lang.StringBuffer
    Appends a long to the end of this buffer.
append(Object). Method in class java.lang.StringBuffer
    Appends an object to the end of this buffer.
append(String). Method in class java.lang.StringBuffer
    Appends a String to the end of this buffer.
appendText(String). Method in class java.awt.TextArea
    Appends the given text to the end.
Applet(). Constructor for class java.applet.Applet
appletResize(int, int). Method in interface java.applet.AppletStub
    Is called when the applet wants to be resized.
arg. Variable in class java.awt.Event
    An arbitraty argument.
ArithmeticException(). Constructor for class java.lang.ArithmeticException
    Constructs an ArithmeticException with no detail message.
ArithmeticException(String). Constructor for class java.lang.ArithmeticException
    Constructs an ArithmeticException with the specified detail message.
ArrangeGrid(Container). Method in class java.awt.GridBagLayout

arraycopy(Object, int, Object, int, int). Static method in class java.lang.System
Copies an array from the source array, beginning at the specified position, to the specified position of the destination array.
ArrayIndexOutOfBoundsException(). Constructor for class java.lang.ArrayIndexOutOfBoundsException
Constructs an ArrayIndexOutOfBoundsException with no detail message.
ArrayIndexOutOfBoundsException(int). Constructor for class java.lang.ArrayIndexOutOfBoundsException
Constructs a new ArrayIndexOutOfBoundsException class initialized to the specific index.
ArrayIndexOutOfBoundsException(String). Constructor for class java.lang.ArrayIndexOutOfBoundsException
Constructs an ArrayIndexOutOfBoundsException class with the specified detail message.
ArrayStoreException(). Constructor for class java.lang.ArrayStoreException
Constructs a ArrayStoreException with no detail message.
ArrayStoreException(String). Constructor for class java.lang.ArrayStoreException
Constructs a ArrayStoreException with the specified detail message.
arrayTypeName(int). Method in class sun.tools.debug.RemoteArray
Return the element type as a string.
asin(double). Static method in class java.lang.Math
Returns the arc sine of a, in the range of −Pi/2 through Pi/2.
atan(double). Static method in class java.lang.Math
Returns the arc tangent of a, in the range of −Pi/2 through Pi/2.
atan2(double, double). Static method in class java.lang.Math
Converts rectangular coordinates (a, b) to polar (r, theta).
available(). Method in class java.io.BufferedInputStream
Returns the number of bytes that can be read without blocking.
available(). Method in class java.io.ByteArrayInputStream
Returns the number of available bytes in the buffer.
available(). Method in class java.io.FileInputStream
Returns the number of bytes that can be read without blocking.
available(). Method in class java.io.FilterInputStream
Returns the number of bytes that can be read.
available(). Method in class java.io.InputStream
Returns the number of bytes that can be read without blocking.
available(). Method in class java.io.LineNumberInputStream
Returns the number of bytes that can be read.
available(). Method in class java.io.PushbackInputStream
Returns the number of bytes that can be read.
available(). Method in class java.net.SocketImpl
Returns the number of bytes that can be read without blocking.
available(). Method in class java.io.StringBufferInputStream
Returns the number of available bytes in the buffer.
AWTError(String). Constructor for class java.awt.AWTError
AWTException(String). Constructor for class java.awt.AWTException
Constructs an AWTException with the specified detail message.

B

before(Date). Method in class java.util.Date
Checks whether this date comes before the specified date.
bind(InetAddress, int). Method in class java.net.SocketImpl
Binds the socket to the specified port on the specified host.
BitSet(). Constructor for class java.util.BitSet
Creates an empty set.
BitSet(int). Constructor for class java.util.BitSet
Creates an empty set with the specified size.
black. Static variable in class java.awt.Color
The color black.
blue. Static variable in class java.awt.Color
The color blue.
BOLD. Static variable in class java.awt.Font
The bold style constant.
Boolean(boolean). Constructor for class java.lang.Boolean
Constructs a Boolean object initialized to the specified boolean value.
Boolean(String). Constructor for class java.lang.Boolean
Constructs a Boolean object initialized to the value specified by the String parameter.
booleanValue(). Method in class java.lang.Boolean
Returns the value of this Boolean object as a boolean.
BorderLayout(). Constructor for class java.awt.BorderLayout
Constructs a new BorderLayout.
BorderLayout(int, int). Constructor for class java.awt.BorderLayout
Constructs a BorderLayout with the specified gaps.
BOTH. Static variable in class java.awt.GridBagConstraints
bottom. Variable in class java.awt.Insets
The inset from the bottom.
bounds(). Method in class java.awt.Component
Returns the current bounds of this component.
breakpointEvent(RemoteThread). Method in interface sun.tools.debug.DebuggerCallback
A breakpoint has been hit in the specified thread.
brighter(). Method in class java.awt.Color
Returns a brighter version of this color.
buf. Variable in class java.io.BufferedInputStream
The buffer where data is stored.
buf. Variable in class java.io.BufferedOutputStream
The buffer where data is stored.
buf. Variable in class java.io.ByteArrayInputStream
The buffer where data is stored.
buf. Variable in class java.io.ByteArrayOutputStream
The buffer where data is stored.
buffer. Variable in class java.io.StringBufferInputStream
The buffer where data is stored.
BufferedInputStream(InputStream). Constructor for class java.io.BufferedInputStream
Creates a new buffered stream with a default buffer size.
BufferedInputStream(InputStream, int). Constructor for class java.io.BufferedInputStream
Creates a new buffered stream with the specified buffer size.

BufferedOutputStream(OutputStream). Constructor for class java.io.BufferedOutputStream
Creates a new buffered stream with a default buffer size.
BufferedOutputStream(OutputStream, int). Constructor for class java.io.BufferedOutputStream
Creates a new buffered stream with the specified buffer size.
Button(). Constructor for class java.awt.Button
Constructs a Button with no label.
Button(String). Constructor for class java.awt.Button
Constructs a Button with a string label.
ByteArrayInputStream(byte[]). Constructor for class java.io.ByteArrayInputStream
Creates an ByteArrayInputStream from the specified array of bytes.
ByteArrayInputStream(byte[], int, int). Constructor for class java.io.ByteArrayInputStream
Creates an ByteArrayInputStream from the specified array of bytes.
ByteArrayOutputStream(). Constructor for class java.io.ByteArrayOutputStream
Creates a new ByteArrayOutputStream.
ByteArrayOutputStream(int). Constructor for class java.io.ByteArrayOutputStream
Creates a new ByteArrayOutputStream wit the specified initial size.
bytesTransferred. Variable in class java.io.InterruptedIOException
bytesTransferred reports how many bytes had been transferred as part of the IO operation before it was interrupted.
bytesWidth(byte[], int, int). Method in class java.awt.FontMetrics
Returns the width of the specified array of bytes in this Font.

C

canFilterIndexColorModel. Variable in class java.awt.image.RGBImageFilter
This boolean indicates whether or not it is acceptable to apply the color filtering of the filterRGB method to the color table entries of an IndexColorModel object in lieu of pixel by pixel filtering.
canRead(). Method in class java.io.File
Returns a boolean indicating whether or not a readable file exists.
Canvas(). Constructor for class java.awt.Canvas
canWrite(). Method in class java.io.File
Returns a boolean indicating whether or not a writable file exists.
capacity(). Method in class java.lang.StringBuffer
Returns the current capacity of the String buffer.
capacity(). Method in class java.util.Vector
Returns the current capacity of the vector.
capacityIncrement. Variable in class java.util.Vector
The size of the increment.
CardLayout(). Constructor for class java.awt.CardLayout
Creates a new card layout.
CardLayout(int, int). Constructor for class java.awt.CardLayout
Creates a card layout with the specified gaps.
catchExceptions(). Method in class sun.tools.debug.RemoteClass
Enter the debugger when an instance of this class is thrown.
ceil(double). Static method in class java.lang.Math

Returns the "ceiling" or smallest whole number greater than or equal to a.

CENTER. Static variable in class java.awt.FlowLayout

The right alignment variable.

CENTER. Static variable in class java.awt.GridBagConstraints

CENTER. Static variable in class java.awt.Label

The center alignment.

Character(char). Constructor for class java.lang.Character

Constructs a Character object with the specified value.

charAt(int). Method in class java.lang.String

Returns the character at the specified index.

charAt(int). Method in class java.lang.StringBuffer

Returns the character at the specified index.

charsWidth(char[], int, int). Method in class java.awt.FontMetrics

Returns the width of the specified character array in this Font.

charValue(). Method in class java.lang.Character

Returns the value of this Character object.

charWidth(char). Method in class java.awt.FontMetrics

Returns the width of the specified character in this Font.

charWidth(int). Method in class java.awt.FontMetrics

Returns the width of the specified character in this Font.

checkAccept(String, int). Method in class java.lang.SecurityManager

Checks to see if a socket connection to the specified port on the specified host has been accepted.

checkAccess(). Method in class java.lang.Thread

Checks whether the current Thread is allowed to modify this Thread.

checkAccess(). Method in class java.lang.ThreadGroup

Checks to see if the current Thread is allowed to modify this group.

checkAccess(Thread). Method in class java.lang.SecurityManager

Checks to see if the specified Thread is allowed to modify the Thread group.

checkAccess(ThreadGroup). Method in class java.lang.SecurityManager

Checks to see if the specified Thread group is allowed to modify this group.

checkAll(). Method in class java.awt.MediaTracker

Check to see if all images have finished loading, but do not start loading the images if they are not already loading.

checkAll(boolean). Method in class java.awt.MediaTracker

Check to see if all images have finished loading and start loading any images that are not yet being loaded if load is true.

Checkbox(). Constructor for class java.awt.Checkbox

Constructs a Checkbox with no label, no Checkbox group, and initialized to a false state.

Checkbox(String). Constructor for class java.awt.Checkbox

Constructs a Checkbox with the specified label, no Checkbox group, and initialized to a false state.

Checkbox(String, CheckboxGroup, boolean). Constructor for class java.awt.Checkbox

Constructs a Checkbox with the specified label, specified Checkbox group, and specified boolean state.

CheckboxGroup(). Constructor for class java.awt.CheckboxGroup

Creates a new CheckboxGroup.

CheckboxMenuItem(String). Constructor for class java.awt.CheckboxMenuItem

Creates the checkbox item with the specified label.

checkConnect(String, int). Method in class java.lang.SecurityManager
Checks to see if a socket has connected to the specified port on the the specified host.
checkConnect(String, int, Object). Method in class java.lang.SecurityManager
Checks to see if the current execution context and the indicated execution context are both allowed to connect to the indicated host and port.
checkCreateClassLoader(). Method in class java.lang.SecurityManager
Checks to see if the ClassLoader has been created.
checkDelete(String). Method in class java.lang.SecurityManager
Checks to see if a file with the specified system dependent file name can be deleted.
checkError(). Method in class java.io.PrintStream
Flushes the print stream and returns whether or not there was an error on the output stream.
checkExec(String). Method in class java.lang.SecurityManager
Checks to see if the system command is executed by trusted code.
checkExit(int). Method in class java.lang.SecurityManager
Checks to see if the system has exited the virtual machine with an exit code.
checkID(int). Method in class java.awt.MediaTracker
Check to see if all images tagged with the indicated ID have finished loading, but do not start loading the images if they are not already loading.
checkID(int, boolean). Method in class java.awt.MediaTracker
Check to see if all images tagged with the indicated ID have finished loading and start loading any images with that ID that are not yet being loaded if load is true.
checkImage(Image, ImageObserver). Method in class java.awt.Component
Returns the status of the construction of a screen representation of the specified image.
checkImage(Image, int, int, ImageObserver). Method in class java.awt.Component
Returns the status of the construction of a scaled screen representation of the specified image.
checkImage(Image, int, int, ImageObserver). Method in interface java.awt.peer.ComponentPeer
checkImage(Image, int, int, ImageObserver). Method in class java.awt.Toolkit
Returns the status of the construction of the indicated method at the indicated width and height for the default screen.
checkLink(String). Method in class java.lang.SecurityManager
Checks to see if the specified linked library exists.
checkListen(int). Method in class java.lang.SecurityManager
Checks to see if a server socket is listening to the specified local port that it is bounded to.
checkPackageAccess(String). Method in class java.lang.SecurityManager
Check if an applet can access a package.
checkPackageDefinition(String). Method in class java.lang.SecurityManager
Check if an applet can define classes in a package.
checkPropertiesAccess(). Method in class java.lang.SecurityManager
Checks to see who has access to the System properties.
checkPropertyAccess(String). Method in class java.lang.SecurityManager
Checks to see who has access to the System property named by *key*
checkPropertyAccess(String, String). Method in class java.lang.SecurityManager
Checks to see who has access to the System property named by *key* and *def*
checkRead(FileDescriptor). Method in class java.lang.SecurityManager
Checks to see if an input file with the specified file descriptor object gets created.
checkRead(String). Method in class java.lang.SecurityManager
Checks to see if an input file with the specified system dependent file name gets created.
checkRead(String, Object). Method in class java.lang.SecurityManager
Checks to see if the current context or the indicated context are both allowed to read the given

file name.
checkSetFactory(). Method in class java.lang.SecurityManager
Check if an applet can set a networking−related object factory.
checkTopLevelWindow(Object). Method in class java.lang.SecurityManager
Checks to see if top−level windows can be created by the caller.
checkWrite(FileDescriptor). Method in class java.lang.SecurityManager
Checks to see if an output file with the specified file descriptor object gets created.
checkWrite(String). Method in class java.lang.SecurityManager
Checks to see if an output file with the specified system dependent file name gets created.
Choice(). Constructor for class java.awt.Choice
Constructs a new Choice.
ClassCastException(). Constructor for class java.lang.ClassCastException
Constructs a ClassCastException with no detail message.
ClassCastException(String). Constructor for class java.lang.ClassCastException
Constructs a ClassCastException with the specified detail message.
ClassCircularityError(). Constructor for class java.lang.ClassCircularityError
Constructs a ClassCircularityError with no detail message.
ClassCircularityError(String). Constructor for class java.lang.ClassCircularityError
Constructs a ClassCircularityError with the specified detail message.
classDepth(String). Method in class java.lang.SecurityManager
Return the position of the stack frame containing the first occurrence of the named class.
ClassFormatError(). Constructor for class java.lang.ClassFormatError
Constructs a ClassFormatError with no detail message.
ClassFormatError(String). Constructor for class java.lang.ClassFormatError
Constructs a ClassFormatError with the specified detail message.
ClassLoader(). Constructor for class java.lang.ClassLoader
Constructs a new Class loader and initializes it.
classLoaderDepth(). Method in class java.lang.SecurityManager
ClassNotFoundException(). Constructor for class java.lang.ClassNotFoundException
Constructs a ClassNotFoundException with no detail message.
ClassNotFoundException(String). Constructor for class java.lang.ClassNotFoundException
Constructs a ClassNotFoundException with the specified detail message.
clear(). Method in class java.util.Hashtable
Clears the hash table so that it has no more elements in it.
clear(). Method in class java.awt.List
Clears the list.
clear(). Method in interface java.awt.peer.ListPeer
clear(int). Method in class java.util.BitSet
Clears a bit.
clearBreakpoint(int). Method in class sun.tools.debug.RemoteClass
Clear a breakpoint at a specific address in a class.
clearBreakpointLine(int). Method in class sun.tools.debug.RemoteClass
Clear a breakpoint at a specified line.
clearBreakpointMethod(RemoteField). Method in class sun.tools.debug.RemoteClass
Clear a breakpoint at the start of a specified method.
clearChanged(). Method in class java.util.Observable
Clears an observable change.
clearRect(int, int, int, int). Method in class java.awt.Graphics
Clears the specified rectangle by filling it with the current background color of the current

drawing surface.
clickCount. Variable in class java.awt.Event
The number of consecutive clicks.
clipRect(int, int, int, int). Method in class java.awt.Graphics
Clips to a rectangle.
clone(). Method in class java.util.BitSet
Clones the BitSet.
clone(). Method in class java.awt.GridBagConstraints
Creates a clone of the object.
clone(). Method in class java.util.Hashtable
Creates a clone of the hashtable.
clone(). Method in class java.awt.image.ImageFilter
Clone this object.
clone(). Method in class java.awt.Insets
Creates a clone of the object.
clone(). Method in class java.lang.Object
Creates a clone of the object.
clone(). Method in class java.util.Vector
Clones this vector.
CloneNotSupportedException(). Constructor for class java.lang.CloneNotSupportedException
Constructs an CloneNotSupportedException with no detail message.
CloneNotSupportedException(String). Constructor for class java.lang.CloneNotSupportedException
Constructs an CloneNotSupportedException with the specified detail message.
close(). Method in class java.net.DatagramSocket
Close the datagram socket.
close(). Method in class java.io.FileInputStream
Closes the input stream.
close(). Method in class java.io.FileOutputStream
Closes the stream.
close(). Method in class java.io.FilterInputStream
Closes the input stream.
close(). Method in class java.io.FilterOutputStream
Closes the stream.
close(). Method in class java.io.InputStream
Closes the input stream.
close(). Method in class java.io.OutputStream
Closes the stream.
close(). Method in class java.io.PipedInputStream
Closes the input stream.
close(). Method in class java.io.PipedOutputStream
Closes the stream.
close(). Method in class java.io.PrintStream
Closes the stream.
close(). Method in class java.io.RandomAccessFile
Closes the file.
close(). Method in class sun.tools.debug.RemoteDebugger
Close the connection to the remote debugging agent.
close(). Method in class java.io.SequenceInputStream
Closes the input stream; flipping to the next stream, if an EOF is reached.

close(). Method in class java.net.ServerSocket
Closes the server socket.
close(). Method in class java.net.Socket
Closes the socket.
close(). Method in class java.net.SocketImpl
Closes the socket.
Color(float, float, float). Constructor for class java.awt.Color
Creates a color with the specified red, green, and blue values in the range (0.0 – 1.0).
Color(int). Constructor for class java.awt.Color
Creates a color with the specified combined RGB value consisting of the red component in bits 16–23, the green component in bits 8–15, and the blue component in bits 0–7.
Color(int, int, int). Constructor for class java.awt.Color
Creates a color with the specified red, green, and blue values in the range (0 – 255).
ColorModel(int). Constructor for class java.awt.image.ColorModel
Construct a ColorModel which describes a pixel of the specified number of bits.
columnWeights. Variable in class java.awt.GridBagLayout
columnWidths. Variable in class java.awt.GridBagLayout
command(Object). Static method in class java.lang.Compiler
commentChar(int). Method in class java.io.StreamTokenizer
Specifies that this character starts a single line comment.
compareTo(String). Method in class java.lang.String
Compares this String to another specified String.
compileClass(Class). Static method in class java.lang.Compiler
compileClasses(String). Static method in class java.lang.Compiler
COMPLETE. Static variable in class java.awt.MediaTracker
Flag indicating the download of media completed successfully.
COMPLETESCANLINES. Static variable in interface java.awt.image.ImageConsumer
The pixels will be delivered in (multiples of) complete scanlines at a time.
comptable. Variable in class java.awt.GridBagLayout
concat(String). Method in class java.lang.String
Concatenates the specified string to the end of this String.
connect(). Method in class java.net.URLConnection
URLConnection objects go through two phases: first they are created, then they are connected.
connect(InetAddress, int). Method in class java.net.SocketImpl
Connects the socket to the specified address on the specified port.
connect(PipedInputStream). Method in class java.io.PipedOutputStream
Connect this output stream to a receiver.
connect(PipedOutputStream). Method in class java.io.PipedInputStream
Connects this input stream to a sender.
connect(String, int). Method in class java.net.SocketImpl
Connects the socket to the specified port on the specified host.
connected. Variable in class java.net.URLConnection
consumer. Variable in class java.awt.image.ImageFilter
The consumer of the particular image data stream that this instance of the ImageFilter is filtering data for.
cont(). Method in class sun.tools.debug.RemoteThread
Resume this thread from a breakpoint, unless it previously suspended.
contains(Object). Method in class java.util.Hashtable
Returns true if the specified object is an element of the hashtable.

contains(Object). Method in class java.util.Vector
Returns true if the specified object is a value of the collection.
containsKey(Object). Method in class java.util.Hashtable
Returns true if the collection contains an element for the key.
ContentHandler(). Constructor for class java.net.ContentHandler
controlDown(). Method in class java.awt.Event
Checks if the control key is down.
copyArea(int, int, int, int, int, int). Method in class java.awt.Graphics
Copies an area of the screen.
copyInto(Object[]). Method in class java.util.Vector
Copies the elements of this vector into the specified array.
copyValueOf(char[]). Static method in class java.lang.String
Returns a String that is equivalent to the specified character array.
copyValueOf(char[], int, int). Static method in class java.lang.String
Returns a String that is equivalent to the specified character array.
copyWhenShared(). Method in class java.lang.StringBuffer
Copies the buffer value if it is shared.
cos(double). Static method in class java.lang.Math
Returns the trigonometric cosine of an angle.
count. Variable in class java.io.BufferedInputStream
The number of bytes in the buffer.
count. Variable in class java.io.BufferedOutputStream
The number of bytes in the buffer.
count. Variable in class java.io.ByteArrayInputStream
The number of characters to use in the buffer.
count. Variable in class java.io.ByteArrayOutputStream
The number of bytes in the buffer.
count. Variable in class java.io.StringBufferInputStream
The number of characters to use in the buffer.
countComponents(). Method in class java.awt.Container
Returns the number of components in this panel.
countItems(). Method in class java.awt.Choice
Returns the number of items in this Choice.
countItems(). Method in class java.awt.List
Returns the number of items in the list.
countItems(). Method in class java.awt.Menu
Returns the number of elements in this menu.
countMenus(). Method in class java.awt.MenuBar
Counts the number of menus on the menu bar.
countObservers(). Method in class java.util.Observable
Counts the number of observers.
countStackFrames(). Method in class java.lang.Thread
Returns the number of stack frames in this Thread.
countTokens(). Method in class java.util.StringTokenizer
Returns the next number of tokens in the String using the current deliminter set.
create(). Method in class java.awt.Graphics
Creates a new Graphics Object that is a copy of the original Graphics Object.
create(boolean). Method in class java.net.SocketImpl
Creates a socket with a boolean that specifies whether this is a stream socket or a datagram

socket.

create(int, int, int, int). Method in class java.awt.Graphics
Creates a new Graphics Object with the specified parameters, based on the original Graphics Object.

createButton(Button). Method in class java.awt.Toolkit
Uses the specified Peer interface to create a new Button.

createCanvas(Canvas). Method in class java.awt.Toolkit
Uses the specified Peer interface to create a new Canvas.

createCheckbox(Checkbox). Method in class java.awt.Toolkit
Uses the specified Peer interface to create a new Checkbox.

createCheckboxMenuItem(CheckboxMenuItem). Method in class java.awt.Toolkit
Uses the specified Peer interface to create a new CheckboxMenuItem.

createChoice(Choice). Method in class java.awt.Toolkit
Uses the specified Peer interface to create a new Choice.

createContentHandler(String). Method in interface java.net.ContentHandlerFactory
Creates a new ContentHandler to read an object from a URLStreamHandler.

createDialog(Dialog). Method in class java.awt.Toolkit
Uses the specified Peer interface to create a new Dialog.

createFileDialog(FileDialog). Method in class java.awt.Toolkit
Uses the specified Peer interface to create a new FileDialog.

createFrame(Frame). Method in class java.awt.Toolkit
Uses the specified Peer interface to create a new Frame.

createImage(ImageProducer). Method in class java.awt.Component
Creates an image from the specified image producer.

createImage(ImageProducer). Method in interface java.awt.peer.ComponentPeer

createImage(ImageProducer). Method in class java.awt.Toolkit
Creates an image with the specified image producer.

createImage(int, int). Method in class java.awt.Component
Creates an off−screen drawable Image to be used for double buffering.

createImage(int, int). Method in interface java.awt.peer.ComponentPeer

createLabel(Label). Method in class java.awt.Toolkit
Uses the specified Peer interface to create a new Label.

createList(List). Method in class java.awt.Toolkit
Uses the specified Peer interface to create a new List.

createMenu(Menu). Method in class java.awt.Toolkit
Uses the specified Peer interface to create a new Menu.

createMenuBar(MenuBar). Method in class java.awt.Toolkit
Uses the specified Peer interface to create a new MenuBar.

createMenuItem(MenuItem). Method in class java.awt.Toolkit
Uses the specified Peer interface to create a new MenuItem.

createPanel(Panel). Method in class java.awt.Toolkit
Uses the specified Peer interface to create a new Panel.

createScrollbar(Scrollbar). Method in class java.awt.Toolkit
Uses the specified Peer interface to create a new Scrollbar.

createSocketImpl(). Method in interface java.net.SocketImplFactory
Creates a new SocketImpl instance.

createTextArea(TextArea). Method in class java.awt.Toolkit
Uses the specified Peer interface to create a new TextArea.

createTextField(TextField). Method in class java.awt.Toolkit

Uses the specified Peer interface to create a new TextField.

createURLStreamHandler(String). Method in interface java.net.URLStreamHandlerFactory
Creates a new URLStreamHandler instance with the specified protocol.

createWindow(Window). Method in class java.awt.Toolkit
Uses the specified Peer interface to create a new Window.

CropImageFilter(int, int, int, int). Constructor for class java.awt.image.CropImageFilter
Construct a CropImageFilter that extracts the absolute rectangular region of pixels from its source Image as specified by the x, y, w, and h parameters.

CROSSHAIR_CURSOR. Static variable in class java.awt.Frame

CTRL_MASK. Static variable in class java.awt.Event
The control modifier constant.

currentClassLoader(). Method in class java.lang.SecurityManager
The current ClassLoader on the execution stack.

currentThread(). Static method in class java.lang.Thread
Returns a reference to the currently executing Thread object.

currentTimeMillis(). Static method in class java.lang.System
Returns the current time in milliseconds GMT since the epoch (00:00:00 UTC, January 1, 1970).

cyan. Static variable in class java.awt.Color
The color cyan.

D

darker(). Method in class java.awt.Color
Returns a darker version of this color.

darkGray. Static variable in class java.awt.Color
The color dark gray.

DatagramPacket(byte[], int). Constructor for class java.net.DatagramPacket
This constructor is used to create a DatagramPacket object used for receiving datagrams.

DatagramPacket(byte[], int, InetAddress, int). Constructor for class java.net.DatagramPacket
This constructor is used construct the DatagramPacket to be sent.

DatagramSocket(). Constructor for class java.net.DatagramSocket
Creates a datagram socket

DatagramSocket(int). Constructor for class java.net.DatagramSocket
Creates a datagram socket

DataInputStream(InputStream). Constructor for class java.io.DataInputStream
Creates a new DataInputStream.

DataOutputStream(OutputStream). Constructor for class java.io.DataOutputStream
Creates a new DataOutputStream.

Date(). Constructor for class java.util.Date
Creates today's date/time.

Date(int, int, int). Constructor for class java.util.Date
Creates a date.

Date(int, int, int, int, int). Constructor for class java.util.Date
Creates a date.

Date(int, int, int, int, int, int). Constructor for class java.util.Date

Creates a date.
Date(long). Constructor for class java.util.Date
Creates a date.
Date(String). Constructor for class java.util.Date
Creates a date from a string according to the syntax accepted by parse().
DEFAULT_CURSOR. Static variable in class java.awt.Frame
defaultConstraints. Variable in class java.awt.GridBagLayout
defaults. Variable in class java.util.Properties
defineClass(byte[], int, int). Method in class java.lang.ClassLoader
Converts an array of bytes to an instance of class Class.
delete(). Method in class java.io.File
Deletes the specified file.
deleteObserver(Observer). Method in class java.util.Observable
Deletes an observer from the observer list.
deleteObservers(). Method in class java.util.Observable
Deletes observers from the observer list.
delItem(int). Method in class java.awt.List
Delete an item from the list.
delItem(int). Method in interface java.awt.peer.MenuPeer
delItems(int, int). Method in class java.awt.List
Delete multiple items from the list.
delItems(int, int). Method in interface java.awt.peer.ListPeer
deliverEvent(Event). Method in class java.awt.Component
Delivers an event to this component or one of its sub components.
deliverEvent(Event). Method in class java.awt.Container
Delivers an event.
delMenu(int). Method in interface java.awt.peer.MenuBarPeer
description(). Method in class sun.tools.debug.RemoteArray
Return a description of the array.
description(). Method in class sun.tools.debug.RemoteClass
Return a (somewhat verbose) description.
description(). Method in class sun.tools.debug.RemoteObject
Return a description of the object.
description(). Method in class sun.tools.debug.RemoteString
Return the string value, or "null"
description(). Method in class sun.tools.debug.RemoteValue
Return a description of the RemoteValue.
deselect(int). Method in class java.awt.List
Deselects the item at the specified index.
deselect(int). Method in interface java.awt.peer.ListPeer
destroy(). Method in class java.applet.Applet
Cleans up whatever resources are being held.
destroy(). Method in class java.lang.Process
Kills the subprocess.
destroy(). Method in class java.lang.Thread
Destroy a thread, without any cleanup, i.e.
destroy(). Method in class java.lang.ThreadGroup
Destroys a Thread group.
Dialog(Frame, boolean). Constructor for class java.awt.Dialog

Constructs an initially invisible Dialog.
Dialog(Frame, String, boolean). Constructor for class java.awt.Dialog
Constructs an initially invisible Dialog with a title.
Dictionary(). Constructor for class java.util.Dictionary
digit(char, int). Static method in class java.lang.Character
Returns the numeric value of the character digit using the specified radix.
Dimension(). Constructor for class java.awt.Dimension
Constructs a Dimension with a 0 width and 0 height.
Dimension(Dimension). Constructor for class java.awt.Dimension
Constructs a Dimension and initializes it to the specified value.
Dimension(int, int). Constructor for class java.awt.Dimension
Constructs a Dimension and initializes it to the specified width and specified height.
DirectColorModel(int, int, int, int). Constructor for class java.awt.image.DirectColorModel
Construct a DirectColorModel from the given masks specifying which bits in the pixel contain the red, green and blue color components.
DirectColorModel(int, int, int, int, int). Constructor for class java.awt.image.DirectColorModel
Construct a DirectColorModel from the given masks specifying which bits in the pixel contain the alhpa, red, green and blue color components.
disable(). Static method in class java.lang.Compiler
disable(). Method in class java.awt.Component
Disables a component.
disable(). Method in interface java.awt.peer.ComponentPeer
disable(). Method in class java.awt.MenuItem
Makes this menu item unselectable by the user.
disable(). Method in interface java.awt.peer.MenuItemPeer
dispose(). Method in interface java.awt.peer.ComponentPeer
dispose(). Method in class java.awt.Frame
Disposes of the Frame.
dispose(). Method in class java.awt.Graphics
Disposes of this graphics context.
dispose(). Method in interface java.awt.peer.MenuComponentPeer
dispose(). Method in class java.awt.Window
Disposes of the Window.
doInput. Variable in class java.net.URLConnection
doOutput. Variable in class java.net.URLConnection
Double(double). Constructor for class java.lang.Double
Constructs a Double wrapper for the specified double value.
Double(String). Constructor for class java.lang.Double
Constructs a Double object initialized to the value specified by the String parameter.
doubleToLongBits(double). Static method in class java.lang.Double
Returns the bit represention of a double−float value
doubleValue(). Method in class java.lang.Double
Returns the double value of this Double.
doubleValue(). Method in class java.lang.Float
Returns the double value of this Float.
doubleValue(). Method in class java.lang.Integer
Returns the value of this Integer as a double.
doubleValue(). Method in class java.lang.Long
Returns the value of this Long as a double.

doubleValue(). Method in class java.lang.Number
Returns the value of the number as a double.
DOWN. Static variable in class java.awt.Event
The down arrow key.
down(int). Method in class sun.tools.debug.RemoteThread
Change the current stackframe to be one or more frames lower (as in, toward the current program counter).
draw3DRect(int, int, int, int, boolean). Method in class java.awt.Graphics
Draws a highlighted 3−D rectangle.
drawArc(int, int, int, int, int, int). Method in class java.awt.Graphics
Draws an arc bounded by the specified rectangle from startAngle to endAngle.
drawBytes(byte[], int, int, int, int). Method in class java.awt.Graphics
Draws the specified bytes using the current font and color.
drawChars(char[], int, int, int, int). Method in class java.awt.Graphics
Draws the specified characters using the current font and color.
drawImage(Image, int, int, Color, ImageObserver). Method in class java.awt.Graphics
Draws the specified image at the specified coordinate (x, y), with the given solid background Color.
drawImage(Image, int, int, ImageObserver). Method in class java.awt.Graphics
Draws the specified image at the specified coordinate (x, y).
drawImage(Image, int, int, int, int, Color, ImageObserver). Method in class java.awt.Graphics
Draws the specified image inside the specified rectangle, with the given solid background Color.
drawImage(Image, int, int, int, int, ImageObserver). Method in class java.awt.Graphics
Draws the specified image inside the specified rectangle.
drawLine(int, int, int, int). Method in class java.awt.Graphics
Draws a line between the coordinates (x1,y1) and (x2,y2).
drawOval(int, int, int, int). Method in class java.awt.Graphics
Draws an oval inside the specified rectangle using the current color.
drawPolygon(int[], int[], int). Method in class java.awt.Graphics
Draws a polygon defined by an array of x points and y points.
drawPolygon(Polygon). Method in class java.awt.Graphics
Draws a polygon defined by the specified point.
drawRect(int, int, int, int). Method in class java.awt.Graphics
Draws the outline of the specified rectangle using the current color.
drawRoundRect(int, int, int, int, int, int). Method in class java.awt.Graphics
Draws an outlined rounded corner rectangle using the current color.
drawString(String, int, int). Method in class java.awt.Graphics
Draws the specified String using the current font and color.
DumpConstraints(GridBagConstraints). Method in class java.awt.GridBagLayout
Print the layout constraints.
DumpLayoutInfo(GridBagLayoutInfo). Method in class java.awt.GridBagLayout
Print the layout information.
dumpStack(). Method in class sun.tools.debug.RemoteThread
Dump the stack.
dumpStack(). Static method in class java.lang.Thread
A debugging procedure to print a stack trace for the current Thread.

E

E. Static variable in class java.lang.Math
The float representation of the value E.
E_RESIZE_CURSOR. Static variable in class java.awt.Frame
EAST. Static variable in class java.awt.GridBagConstraints
echoCharIsSet(). Method in class java.awt.TextField
Returns true if this TextField has a character set for echoing.
elementAt(int). Method in class java.util.Vector
Returns the element at the specified index.
elementCount. Variable in class java.util.Vector
The number of elements in the buffer.
elementData. Variable in class java.util.Vector
The buffer where elements are stored.
elements(). Method in class java.util.Dictionary
Returns an enumeration of the elements.
elements(). Method in class java.util.Hashtable
Returns an enumeration of the elements.
elements(). Method in class java.util.Vector
Returns an enumeration of the elements.
empty(). Method in class java.util.Stack
Returns true if the stack is empty.
EmptyStackException(). Constructor for class java.util.EmptyStackException
Constructs a new EmptySackException with no detail message.
enable(). Static method in class java.lang.Compiler
enable(). Method in class java.awt.Component
Enables a component.
enable(). Method in interface java.awt.peer.ComponentPeer
enable(). Method in class java.awt.MenuItem
Makes this menu item selectable by the user.
enable(). Method in interface java.awt.peer.MenuItemPeer
enable(boolean). Method in class java.awt.Component
Conditionally enables a component.
enable(boolean). Method in class java.awt.MenuItem
Conditionally enables a component.
encode(String). Static method in class java.net.URLEncoder
encode − translate String into x−www−form−urlencoded format.
END. Static variable in class java.awt.Event
The end key.
endsWith(String). Method in class java.lang.String
Determines whether the String ends with some suffix.
ensureCapacity(int). Method in class java.lang.StringBuffer
Ensures that the capacity of the buffer is at least equal to the specified minimum.
ensureCapacity(int). Method in class java.util.Vector
Ensures that the vector has at least the specified capacity.
enumerate(Thread[]). Static method in class java.lang.Thread
Copies, into the specified array, references to every active Thread in this Thread's group.
enumerate(Thread[]). Method in class java.lang.ThreadGroup
Copies, into the specified array, references to every active Thread in this Thread group.

enumerate(Thread[], boolean). Method in class java.lang.ThreadGroup
Copies, into the specified array, references to every active Thread in this Thread group.
enumerate(ThreadGroup[]). Method in class java.lang.ThreadGroup
Copies, into the specified array, references to every active Thread group in this Thread group.
enumerate(ThreadGroup[], boolean). Method in class java.lang.ThreadGroup
Copies, into the specified array, references to every active Thread group in this Thread group.
EOFException(). Constructor for class java.io.EOFException
Constructs an EOFException with no detail message.
EOFException(String). Constructor for class java.io.EOFException
Constructs an EOFException with the specified detail message.
eolIsSignificant(boolean). Method in class java.io.StreamTokenizer
If the flag is true, end−of−lines are significant (TT_EOL will be returned by nexttoken).
equals(Object). Method in class java.util.BitSet
Compares this object against the specified object.
equals(Object). Method in class java.lang.Boolean
Compares this object against the specified object.
equals(Object). Method in class java.lang.Character
Compares this object against the specified object.
equals(Object). Method in class java.awt.Color
Compares this object against the specified object.
equals(Object). Method in class java.util.Date
Compares this object against the specified object.
equals(Object). Method in class java.lang.Double
Compares this object against the specified object.
equals(Object). Method in class java.io.File
Compares this object against the specified object.
equals(Object). Method in class java.lang.Float
Compares this object against some other object.
equals(Object). Method in class java.awt.Font
Compares this object to the specifed object.
equals(Object). Method in class java.net.InetAddress
Compares this object against the specified object.
equals(Object). Method in class java.lang.Integer
Compares this object to the specified object.
equals(Object). Method in class java.lang.Long
Compares this object against the specified object.
equals(Object). Method in class java.lang.Object
Compares two Objects for equality.
equals(Object). Method in class java.awt.Point
Check if two pointers are equal.
equals(Object). Method in class java.awt.Rectangle
Check if two rectangles are equal.
equals(Object). Method in class java.lang.String
Compares this String to the specified object.
equals(Object). Method in class java.net.URL
Compares two URLs.
equalsIgnoreCase(String). Method in class java.lang.String
Compares this String to another object.
err. Static variable in class java.io.FileDescriptor

err. Static variable in class java.lang.System
Standard error stream.
ERROR. Static variable in interface java.awt.image.ImageObserver
An image which was being tracked asynchronously has encountered an error.
Error(). Constructor for class java.lang.Error
Constructs an Error with no specified detail message.
Error(String). Constructor for class java.lang.Error
Constructs an Error with the specified detail message.
ERRORED. Static variable in class java.awt.MediaTracker
Flag indicating the download of some media encountered an error.
Event(Object, int, Object). Constructor for class java.awt.Event
Constructs an event with the specified target component, event type, and argument.
Event(Object, long, int, int, int, int, int). Constructor for class java.awt.Event
Constructs an event with the specified target component, time stamp, event type, x and y coordinates, keyboard key, state of the modifier keys and an argument set to null.
Event(Object, long, int, int, int, int, int, Object). Constructor for class java.awt.Event
Constructs an event with the specified target component, time stamp, event type, x and y coordinates, keyboard key, state of the modifier keys and argument.
evt. Variable in class java.awt.Event
The next event.
Exception(). Constructor for class java.lang.Exception
Constructs an Exception with no specified detail message.
Exception(String). Constructor for class java.lang.Exception
Constructs a Exception with the specified detail message.
exceptionEvent(RemoteThread, String). Method in interface sun.tools.debug.DebuggerCallback
An exception has occurred.
exec(String). Method in class java.lang.Runtime
Executes the system command specified in the parameter.
exec(String, String[]). Method in class java.lang.Runtime
Executes the system command specified in the parameter.
exec(String[]). Method in class java.lang.Runtime
Executes the system command specified by cmdarray[0] with arguments specified by the strings in the rest of the array.
exec(String[], String[]). Method in class java.lang.Runtime
Executes the system command specified by cmdarray[0] with arguments specified by the strings in the rest of the array.
exists(). Method in class java.io.File
Returns a boolean indicating whether or not a file exists.
exit(int). Method in class java.lang.Runtime
Exits the virtual machine with an exit code.
exit(int). Static method in class java.lang.System
Exits the virtual machine with an exit code.
exitValue(). Method in class java.lang.Process
Returns the exit value for the subprocess.
exp(double). Static method in class java.lang.Math
Returns the exponential number e(2.718...) raised to the power of a.

F

F1. Static variable in class java.awt.Event
The F1 function key
F10. Static variable in class java.awt.Event
The F10 function key
F11. Static variable in class java.awt.Event
The F11 function key
F12. Static variable in class java.awt.Event
The F12 function key
F2. Static variable in class java.awt.Event
The F2 function key
F3. Static variable in class java.awt.Event
The F3 function key
F4. Static variable in class java.awt.Event
The F4 function key
F5. Static variable in class java.awt.Event
The F5 function key
F6. Static variable in class java.awt.Event
The F6 function key
F7. Static variable in class java.awt.Event
The F7 function key
F8. Static variable in class java.awt.Event
The F8 function key
F9. Static variable in class java.awt.Event
The F9 function key
FALSE. Static variable in class java.lang.Boolean
Assigns this Boolean to be false.
fd. Variable in class java.net.SocketImpl
The file descriptor object
File(File, String). Constructor for class java.io.File
Creates a File object (given a directory File object).
File(String). Constructor for class java.io.File
Creates a File object.
File(String, String). Constructor for class java.io.File
Creates a File object from the specified directory.
FileDescriptor(). Constructor for class java.io.FileDescriptor
FileDialog(Frame, String). Constructor for class java.awt.FileDialog
Creates a file dialog for loading a file.
FileDialog(Frame, String, int). Constructor for class java.awt.FileDialog
Creates a file dialog with the specified title and mode.
FileInputStream(File). Constructor for class java.io.FileInputStream
Creates an input file from the specified File object.
FileInputStream(FileDescriptor). Constructor for class java.io.FileInputStream
FileInputStream(String). Constructor for class java.io.FileInputStream
Creates an input file with the specified system dependent file name.
FileNotFoundException(). Constructor for class java.io.FileNotFoundException
Constructs a FileNotFoundException with no detail message.
FileNotFoundException(String). Constructor for class java.io.FileNotFoundException

Constructs a FileNotFoundException with the specified detail message.
FileOutputStream(File). Constructor for class java.io.FileOutputStream
Creates an output file with the specified File object.
FileOutputStream(FileDescriptor). Constructor for class java.io.FileOutputStream
FileOutputStream(String). Constructor for class java.io.FileOutputStream
Creates an output file with the specified system dependent file name.
fill. Variable in class java.awt.GridBagConstraints
fill3DRect(int, int, int, int, boolean). Method in class java.awt.Graphics
Paints a highlighted 3−D rectangle using the current color.
fillArc(int, int, int, int, int, int). Method in class java.awt.Graphics
Fills an arc using the current color.
fillInStackTrace(). Method in class java.lang.Throwable
Fills in the excecution stack trace.
fillOval(int, int, int, int). Method in class java.awt.Graphics
Fills an oval inside the specified rectangle using the current color.
fillPolygon(int[], int[], int). Method in class java.awt.Graphics
Fills a polygon with the current color using an even−odd fill rule (otherwise known as an alternating rule).
fillPolygon(Polygon). Method in class java.awt.Graphics
Fills the specified polygon with the current color using an even−odd fill rule (otherwise known as an alternating rule).
fillRect(int, int, int, int). Method in class java.awt.Graphics
Fills the specified rectangle with the current color.
fillRoundRect(int, int, int, int, int, int). Method in class java.awt.Graphics
Draws a rounded rectangle filled in with the current color.
FilteredImageSource(ImageProducer, ImageFilter). Constructor for class java.awt.image.FilteredImageSource
Construct an ImageProducer object from an existing ImageProducer and a filter object.
filterIndexColorModel(IndexColorModel). Method in class java.awt.image.RGBImageFilter
Filter an IndexColorModel object by running each entry in its color tables through the filterRGB function that RGBImageFilter subclasses must provide.
FilterInputStream(InputStream). Constructor for class java.io.FilterInputStream
Creates an input stream filter.
FilterOutputStream(OutputStream). Constructor for class java.io.FilterOutputStream
Creates an output stream filter.
filterRGB(int, int, int). Method in class java.awt.image.RGBImageFilter
Subclasses must specify a method to convert a single input pixel in the default RGB ColorModel to a single output pixel.
filterRGBPixels(int, int, int, int, int[], int, int). Method in class java.awt.image.RGBImageFilter
Filter a buffer of pixels in the default RGB ColorModel by passing them one by one through the filterRGB method.
finalize(). Method in class java.net.DatagramSocket
Code to perform when this object is garbage collected.
finalize(). Method in class java.io.FileInputStream
Closes the stream when garbage is collected.
finalize(). Method in class java.io.FileOutputStream
Closes the stream when garbage is collected.
finalize(). Method in class java.awt.Graphics
Disposes of this graphics context once it is no longer referenced.

finalize(). Method in class java.lang.Object
    Code to perform when this object is garbage collected.
findClass(String). Method in class sun.tools.debug.RemoteDebugger
    Find a specified class.
findSystemClass(String). Method in class java.lang.ClassLoader
    Loads a system Class.
first(Container). Method in class java.awt.CardLayout
    Flip to the first card.
firstElement(). Method in class java.util.Vector
    Returns the first element of the sequence.
Float(double). Constructor for class java.lang.Float
    Constructs a Float wrapper for the specified double value.
Float(float). Constructor for class java.lang.Float
    Constructs a Float wrapper for the specified float value.
Float(String). Constructor for class java.lang.Float
    Constructs a Float object initialized to the value specified by the String parameter.
floatToIntBits(float). Static method in class java.lang.Float
    Returns the bit represention of a single−float value
floatValue(). Method in class java.lang.Double
    Returns the float value of this Double.
floatValue(). Method in class java.lang.Float
    Returns the float value of this Float object.
floatValue(). Method in class java.lang.Integer
    Returns the value of this Integer as a float.
floatValue(). Method in class java.lang.Long
    Returns the value of this Long as a float.
floatValue(). Method in class java.lang.Number
    Returns the value of the number as a float.
floor(double). Static method in class java.lang.Math
    Returns the "floor" or largest whole number less than or equal to a.
FlowLayout(). Constructor for class java.awt.FlowLayout
    Constructs a new Flow Layout with a centered alignment.
FlowLayout(int). Constructor for class java.awt.FlowLayout
    Constructs a new Flow Layout with the specified alignment.
FlowLayout(int, int, int). Constructor for class java.awt.FlowLayout
    Constructs a new Flow Layout with the specified alignment and gap values.
flush(). Method in class java.io.BufferedOutputStream
    Flushes the stream.
flush(). Method in class java.io.DataOutputStream
    Flushes the stream.
flush(). Method in class java.io.FilterOutputStream
    Flushes the stream.
flush(). Method in class java.awt.Image
    Flushes all resources being used by this Image object.
flush(). Method in class java.io.OutputStream
    Flushes the stream.
flush(). Method in class java.io.PrintStream
    Flushes the stream.
font. Variable in class java.awt.FontMetrics

The actual font.
Font(String, int, int). Constructor for class java.awt.Font
Creates a new font with the specified name, style and point size.
FontMetrics(Font). Constructor for class java.awt.FontMetrics
Creates a new FontMetrics object with the specified font.
forDigit(int, int). Static method in class java.lang.Character
Returns the character value for the specified digit in the specified radix.
forName(String). Static method in class java.lang.Class
Returns the runtime Class descriptor for the specified Class.
Frame(). Constructor for class java.awt.Frame
Constructs a new Frame that is initially invisible.
Frame(String). Constructor for class java.awt.Frame
Constructs a new, initially invisible Frame with the specified title.
FRAMEBITS. Static variable in interface java.awt.image.ImageObserver
Another complete frame of a multi−frame image which was previously drawn is now available to be drawn again.
freeMemory(). Method in class sun.tools.debug.RemoteDebugger
Report the free memory available to the Java interpreter being debugged.
freeMemory(). Method in class java.lang.Runtime
Returns the number of free bytes in system memory.
fromHex(String). Static method in class sun.tools.debug.RemoteValue
Convert hexadecimal strings to ints.

G

gc(). Method in class java.lang.Runtime
Runs the garbage collector.
gc(). Static method in class java.lang.System
Runs the garbage collector.
gc(RemoteObject[]). Method in class sun.tools.debug.RemoteDebugger
Free all objects referenced by the debugger.
get(). Method in class sun.tools.debug.RemoteBoolean
Return the boolean's value.
get(). Method in class sun.tools.debug.RemoteByte
Return the byte's value.
get(). Method in class sun.tools.debug.RemoteChar
Return the char's value.
get(). Method in class sun.tools.debug.RemoteDouble
Return the double's value.
get(). Method in class sun.tools.debug.RemoteFloat
Return the float's value.
get(). Method in class sun.tools.debug.RemoteInt
Return the int's value.
get(). Method in class sun.tools.debug.RemoteLong
Return the long's value.
get(). Method in class sun.tools.debug.RemoteShort

Return the short's value.
get(int). Method in class java.util.BitSet
Gets a bit.
get(Integer). Method in class sun.tools.debug.RemoteDebugger
Get an object from the remote object cache.
get(Object). Method in class java.util.Dictionary
Gets the object associated with the specified key in the Dictionary.
get(Object). Method in class java.util.Hashtable
Gets the object associated with the specified key in the hashtable.
getAbsolutePath(). Method in class java.io.File
Gets the absolute path of the file.
getAddress(). Method in class java.net.DatagramPacket
getAddress(). Method in class java.net.InetAddress
Returns the raw IP address in network byte order.
getAlignment(). Method in class java.awt.Label
Gets the current alignment of this label.
getAllByName(String). Static method in class java.net.InetAddress
Given a hostname, return an array of all the corresponding InetAddresses.
getAllowUserInteraction(). Method in class java.net.URLConnection
getAlpha(int). Method in class java.awt.image.ColorModel
The subclass must provide a function which provides the alpha color compoment for the specified pixel.
getAlpha(int). Method in class java.awt.image.DirectColorModel
Return the alpha transparency value for the specified pixel in the range 0−255.
getAlpha(int). Method in class java.awt.image.IndexColorModel
Return the alpha transparency value for the specified pixel in the range 0−255.
getAlphaMask(). Method in class java.awt.image.DirectColorModel
Return the mask indicating which bits in a pixel contain the alpha transparency component.
getAlphas(byte[]). Method in class java.awt.image.IndexColorModel
Copy the array of alpha transparency values into the given array.
getApplet(String). Method in interface java.applet.AppletContext
Gets an applet by name.
getAppletContext(). Method in class java.applet.Applet
Gets a handle to the applet context.
getAppletContext(). Method in interface java.applet.AppletStub
Gets a handler to the applet's context.
getAppletInfo(). Method in class java.applet.Applet
Return a string containing information about the author, version and copyright of the applet.
getApplets(). Method in interface java.applet.AppletContext
Enumerate the applets in this context.
getAscent(). Method in class java.awt.FontMetrics
Gets the font ascent.
getAudioClip(URL). Method in class java.applet.Applet
Gets an audio clip.
getAudioClip(URL). Method in interface java.applet.AppletContext
Gets an audio clip.
getAudioClip(URL, String). Method in class java.applet.Applet
Gets an audio clip.
getBackground(). Method in class java.awt.Component

Gets the background color.

getBlue(). Method in class java.awt.Color

Gets the blue component.

getBlue(int). Method in class java.awt.image.ColorModel

The subclass must provide a function which provides the blue color compoment for the specified pixel.

getBlue(int). Method in class java.awt.image.DirectColorModel

Return the blue color compoment for the specified pixel in the range 0−255.

getBlue(int). Method in class java.awt.image.IndexColorModel

Return the blue color compoment for the specified pixel in the range 0−255.

getBlueMask(). Method in class java.awt.image.DirectColorModel

Return the mask indicating which bits in a pixel contain the blue color component.

getBlues(byte[]). Method in class java.awt.image.IndexColorModel

Copy the array of blue color components into the given array.

getBoolean(String). Static method in class java.lang.Boolean

Gets a Boolean from the properties.

getBoundingBox(). Method in class java.awt.Polygon

getBoundingBox() − what area does this polygon span?

getByName(String). Static method in class java.net.InetAddress

Returns a network address for the indicated host.

getBytes(int, int, byte[], int). Method in class java.lang.String

Copies characters from this String into the specified byte array.

getChars(int, int, char[], int). Method in class java.lang.String

Copies characters from this String into the specified character array.

getChars(int, int, char[], int). Method in class java.lang.StringBuffer

Copies the characters of the specified substring (determined by srcBegin and srcEnd) into the character array, starting at the array's dstBegin location.

getCheckboxGroup(). Method in class java.awt.Checkbox

Returns the checkbox group.

getClass(). Method in class java.lang.Object

Returns the Class of this Object.

getClassContext(). Method in class java.lang.SecurityManager

Gets the context of this Class.

getClassLoader(). Method in class java.lang.Class

Returns the Class loader of this Class.

getClassLoader(). Method in class sun.tools.debug.RemoteClass

Return the classloader for this class.

getClazz(). Method in class sun.tools.debug.RemoteObject

Returns the object's class.

getClipRect(). Method in class java.awt.Graphics

Returns the bounding rectangle of the current clipping area.

getCodeBase(). Method in class java.applet.Applet

Gets the base URL.

getCodeBase(). Method in interface java.applet.AppletStub

Gets the base URL.

getColor(). Method in class java.awt.Graphics

Gets the current color.

getColor(String). Static method in class java.awt.Color

Gets the specified Color property.

getColor(String, Color). Static method in class java.awt.Color
Gets the specified Color property of the specified Color.
getColor(String, int). Static method in class java.awt.Color
Gets the specified Color property of the color value.
getColorModel(). Method in class java.awt.Component
Gets the ColorModel used to display the component on the output device.
getColorModel(). Method in interface java.awt.peer.ComponentPeer
getColorModel(). Method in class java.awt.Toolkit
Returns the ColorModel of the screen.
getColumns(). Method in class java.awt.TextArea
Returns the number of columns in the TextArea.
getColumns(). Method in class java.awt.TextField
Returns the number of columns in this TextField.
getComponent(int). Method in class java.awt.Container
Gets the nth component in this container.
getComponents(). Method in class java.awt.Container
Gets all the components in this container.
getConstraints(Component). Method in class java.awt.GridBagLayout
Retrieves the constraints for the specified component.
getContent(). Method in class java.net.URL
Gets the contents from this opened connection.
getContent(). Method in class java.net.URLConnection
Gets the object referred to by this URL.
getContent(URLConnection). Method in class java.net.ContentHandler
Given an input stream positioned at the beginning of the representation of an object, read that stream and recreate the object from it
getContentEncoding(). Method in class java.net.URLConnection
Gets the content encoding.
getContentLength(). Method in class java.net.URLConnection
Gets the content length.
getContentType(). Method in class java.net.URLConnection
Gets the content type.
getCurrent(). Method in class java.awt.CheckboxGroup
Gets the current choice.
getCurrentFrame(). Method in class sun.tools.debug.RemoteThread
Get the current stack frame.
getCurrentFrameIndex(). Method in class sun.tools.debug.RemoteThread
Return the current stackframe index
getCursorType(). Method in class java.awt.Frame
Return the cursor type
getData(). Method in class java.net.DatagramPacket
getDate(). Method in class java.util.Date
Returns the day of the month.
getDate(). Method in class java.net.URLConnection
Gets the sending date of the object.
getDay(). Method in class java.util.Date
Returns the day of the week.
getDefaultAllowUserInteraction(). Static method in class java.net.URLConnection
getDefaultRequestProperty(String). Static method in class java.net.URLConnection

getDefaultToolkit(). Static method in class java.awt.Toolkit
Returns the default toolkit.
getDefaultUseCaches(). Method in class java.net.URLConnection
Set/get the default value of the UseCaches flag.
getDescent(). Method in class java.awt.FontMetrics
Gets the font descent.
getDirectory(). Method in class java.awt.FileDialog
Gets the directory of the Dialog.
getDocumentBase(). Method in class java.applet.Applet
Gets the document URL.
getDocumentBase(). Method in interface java.applet.AppletStub
Gets the document URL.
getDoInput(). Method in class java.net.URLConnection
getDoOutput(). Method in class java.net.URLConnection
getEchoChar(). Method in class java.awt.TextField
Returns the character to be used for echoing.
getElement(int). Method in class sun.tools.debug.RemoteArray
Return an array element.
getElements(). Method in class sun.tools.debug.RemoteArray
Returns a copy of the array as instances of RemoteValue.
getElements(int, int). Method in class sun.tools.debug.RemoteArray
Returns a copy of a portion of the array as instances of RemoteValue.
getElementType(). Method in class sun.tools.debug.RemoteArray
Return the element type as a "TC_" constant, such as "TC_CHAR".
getenv(String). Static method in class java.lang.System
Obsolete.
getErrorsAny(). Method in class java.awt.MediaTracker
Return a list of all media that have encountered an error.
getErrorsID(int). Method in class java.awt.MediaTracker
Return a list of media with the specified ID that have encountered an error.
getErrorStream(). Method in class java.lang.Process
Returns the an InputStream connected to the error stream of the child process.
getExceptionCatchList(). Method in class sun.tools.debug.RemoteDebugger
Return the list of the exceptions the debugger will stop on.
getExpiration(). Method in class java.net.URLConnection
Gets the expriation date of the object.
getFamily(). Method in class java.awt.Font
Gets the platform specific family name of the font.
getFD(). Method in class java.io.FileInputStream
Returns the opaque file descriptor object associated with this stream.
getFD(). Method in class java.io.FileOutputStream
Returns the file descriptor associated with this stream.
getFD(). Method in class java.io.RandomAccessFile
Returns the opaque file descriptor object.
getField(int). Method in class sun.tools.debug.RemoteClass
Return the static field, specified by index.
getField(int). Method in class sun.tools.debug.RemoteObject
Return an instance variable, specified by slot number.
getField(String). Method in class sun.tools.debug.RemoteClass

Return the static field, specified by name.
getField(String). Method in class sun.tools.debug.RemoteObject
Return an instance variable, specified by name.
getFields(). Method in class sun.tools.debug.RemoteClass
Return all the static fields for this class.
getFields(). Method in class sun.tools.debug.RemoteObject
Return the instance (non−static) fields of an object.
getFieldValue(int). Method in class sun.tools.debug.RemoteClass
Return the value of a static field, specified by its index
getFieldValue(int). Method in class sun.tools.debug.RemoteObject
Returns the value of an object's instance variable.
getFieldValue(String). Method in class sun.tools.debug.RemoteClass
Return the value of a static field, specified by name.
getFieldValue(String). Method in class sun.tools.debug.RemoteObject
Returns the value of an object's instance variable.
getFile(). Method in class java.awt.FileDialog
Gets the file of the Dialog.
getFile(). Method in class java.net.URL
Gets the file name.
getFileDescriptor(). Method in class java.net.SocketImpl
getFilenameFilter(). Method in class java.awt.FileDialog
Gets the filter.
getFilePointer(). Method in class java.io.RandomAccessFile
Returns the current location of the file pointer.
getFilterInstance(ImageConsumer). Method in class java.awt.image.ImageFilter
Return a unique instance of an ImageFilter object which will actually perform the filtering for the specified ImageConsumer.
getFont(). Method in class java.awt.Component
Gets the font of the component.
getFont(). Method in class java.awt.FontMetrics
Gets the font.
getFont(). Method in class java.awt.Graphics
Gets the current font.
getFont(). Method in class java.awt.MenuComponent
Gets the font used for this MenuItem.
getFont(). Method in interface java.awt.MenuContainer
getFont(String). Static method in class java.awt.Font
Gets a font from the system properties list.
getFont(String, Font). Static method in class java.awt.Font
Gets the specified font from the system properties list.
getFontList(). Method in class java.awt.Toolkit
Returns the names of the available fonts.
getFontMetrics(). Method in class java.awt.Graphics
Gets the current font metrics.
getFontMetrics(Font). Method in class java.awt.Component
Gets the font metrics for this component.
getFontMetrics(Font). Method in interface java.awt.peer.ComponentPeer
getFontMetrics(Font). Method in class java.awt.Graphics
Gets the current font metrics for the specified font.

getFontMetrics(Font). Method in class java.awt.Toolkit
Returns the screen metrics of the font.
getForeground(). Method in class java.awt.Component
Gets the foreground color.
getGraphics(). Method in class java.awt.Component
Gets a Graphics context for this component.
getGraphics(). Method in interface java.awt.peer.ComponentPeer
getGraphics(). Method in class java.awt.Image
Gets a graphics object to draw into this image.
getGreen(). Method in class java.awt.Color
Gets the green component.
getGreen(int). Method in class java.awt.image.ColorModel
The subclass must provide a function which provides the green color compoment for the specified pixel.
getGreen(int). Method in class java.awt.image.DirectColorModel
Return the green color compoment for the specified pixel in the range 0–255.
getGreen(int). Method in class java.awt.image.IndexColorModel
Return the green color compoment for the specified pixel in the range 0–255.
getGreenMask(). Method in class java.awt.image.DirectColorModel
Return the mask indicating which bits in a pixel contain the green color component.
getGreens(byte[]). Method in class java.awt.image.IndexColorModel
Copy the array of green color components into the given array.
getHeaderField(int). Method in class java.net.URLConnection
Return the value for the nth header field.
getHeaderField(String). Method in class java.net.URLConnection
Gets a header field by name.
getHeaderFieldDate(String, long). Method in class java.net.URLConnection
Gets a header field by name.
getHeaderFieldInt(String, int). Method in class java.net.URLConnection
Gets a header field by name.
getHeaderFieldKey(int). Method in class java.net.URLConnection
Return the key for the nth header field.
getHeight(). Method in class java.awt.FontMetrics
Gets the total height of the font.
getHeight(ImageObserver). Method in class java.awt.Image
Gets the actual height of the image.
getHelpMenu(). Method in class java.awt.MenuBar
Gets the help menu on the menu bar.
getHost(). Method in class java.net.URL
Gets the host name.
getHostName(). Method in class java.net.InetAddress
Gets the hostname for this address; also the key in the above hash table.
getHours(). Method in class java.util.Date
Returns the hour.
getHSBColor(float, float, float). Static method in class java.awt.Color
A static Color factory for generating a Color object from HSB values.
getIconImage(). Method in class java.awt.Frame
Returns the icon image for this Frame.
getId(). Method in class sun.tools.debug.RemoteObject

Returns the id of the object.
getIfModifiedSince(). Method in class java.net.URLConnection
getImage(String). Method in class java.awt.Toolkit
Returns an image which gets pixel data from the specified file.
getImage(URL). Method in class java.applet.Applet
Gets an image given a URL.
getImage(URL). Method in interface java.applet.AppletContext
Gets an image.
getImage(URL). Method in class java.awt.Toolkit
Returns an image which gets pixel data from the specified URL.
getImage(URL, String). Method in class java.applet.Applet
Gets an image relative to a URL.
getInCheck(). Method in class java.lang.SecurityManager
Returns whether there is a security check in progress.
getInetAddress(). Method in class java.net.ServerSocket
Gets the address to which the socket is connected.
getInetAddress(). Method in class java.net.Socket
Gets the address to which the socket is connected.
getInetAddress(). Method in class java.net.SocketImpl
getInputStream(). Method in class java.lang.Process
Returns a Stream connected to the output of the child process.
getInputStream(). Method in class java.net.Socket
Gets an InputStream for this socket.
getInputStream(). Method in class java.net.SocketImpl
Gets an InputStream for this socket.
getInputStream(). Method in class java.net.URLConnection
Calls this routine to get an InputStream that reads from the object.
getInstanceField(int). Method in class sun.tools.debug.RemoteClass
Return the instance field, specified by its index.
getInstanceFields(). Method in class sun.tools.debug.RemoteClass
Return all the instance fields for this class.
getInteger(String). Static method in class java.lang.Integer
Gets an Integer property.
getInteger(String, int). Static method in class java.lang.Integer
Gets an Integer property.
getInteger(String, Integer). Static method in class java.lang.Integer
Gets an Integer property.
getInterfaces(). Method in class java.lang.Class
Returns the interfaces of this Class.
getInterfaces(). Method in class sun.tools.debug.RemoteClass
Return the interfaces for this class.
getItem(int). Method in class java.awt.Choice
Returns the String at the specified index in the Choice.
getItem(int). Method in class java.awt.List
Gets the item associated with the specified index.
getItem(int). Method in class java.awt.Menu
Returns the item located at the specified index of this menu.
getLabel(). Method in class java.awt.Button
Gets the label of the button.

getLabel(). Method in class java.awt.Checkbox
    Gets the label of the button.
getLabel(). Method in class java.awt.MenuItem
    Gets the label for this menu item.
getLastModified(). Method in class java.net.URLConnection
    Gets the last modified date of the object.
getLayout(). Method in class java.awt.Container
    Gets the layout manager for this container.
getLayoutDimensions(). Method in class java.awt.GridBagLayout
GetLayoutInfo(Container, int). Method in class java.awt.GridBagLayout
getLayoutOrigin(). Method in class java.awt.GridBagLayout
getLayoutWeights(). Method in class java.awt.GridBagLayout
getLeading(). Method in class java.awt.FontMetrics
    Gets the standard leading, or line spacing, for the font.
getLength(). Method in class java.net.DatagramPacket
getLineIncrement(). Method in class java.awt.Scrollbar
    Gets the line increment for this scrollbar.
getLineNumber(). Method in class java.io.LineNumberInputStream
    Returns the current line number.
getLineNumber(). Method in class sun.tools.debug.RemoteStackFrame
    Return the source file line number.
getLocalHost(). Static method in class java.net.InetAddress
    Returns the local host.
getLocalizedInputStream(InputStream). Method in class java.lang.Runtime
    Localize an input stream.
getLocalizedOutputStream(OutputStream). Method in class java.lang.Runtime
    Localize an output stream.
getLocalPort(). Method in class java.net.DatagramSocket
    Returns the local port that this socket is bound to.
getLocalPort(). Method in class java.net.ServerSocket
    Gets the port to which the socket is listening on
getLocalPort(). Method in class java.net.Socket
    Gets the local port to which the socket is connected.
getLocalPort(). Method in class java.net.SocketImpl
getLocalVariable(String). Method in class sun.tools.debug.RemoteStackFrame
    Return a specific (named) stack variable.
getLocalVariables(). Method in class sun.tools.debug.RemoteStackFrame
    Return an array of all valid local variables and method arguments for this stack frame.
getLong(String). Static method in class java.lang.Long
    Get a Long property.
getLong(String, Long). Static method in class java.lang.Long
    Get a Long property.
getLong(String, long). Static method in class java.lang.Long
    Get a Long property.
getMapSize(). Method in class java.awt.image.IndexColorModel
    Returns the size of the color component arrays in this IndexColorModel.
getMaxAdvance(). Method in class java.awt.FontMetrics
    Gets the maximum advance width of any character in this Font.
getMaxAscent(). Method in class java.awt.FontMetrics

Gets the maximum ascent of all characters in this Font.
getMaxDecent(). Method in class java.awt.FontMetrics
For backward compatibility only.
getMaxDescent(). Method in class java.awt.FontMetrics
Gets the maximum descent of all characters.
getMaximum(). Method in class java.awt.Scrollbar
Returns the maximum value of this Scrollbar.
getMaxPriority(). Method in class java.lang.ThreadGroup
Gets the maximum priority of the group.
getMenu(int). Method in class java.awt.MenuBar
Gets the specified menu.
getMenuBar(). Method in class java.awt.Frame
Gets the menu bar for this Frame.
getMessage(). Method in class java.lang.Throwable
Gets the detail message of the Throwable.
getMethod(String). Method in class sun.tools.debug.RemoteClass
Return the method, specified by name.
getMethodName(). Method in class sun.tools.debug.RemoteStackFrame
Get the method name referenced by this stackframe.
getMethodNames(). Method in class sun.tools.debug.RemoteClass
Return the names of all methods supported by this class.
getMethods(). Method in class sun.tools.debug.RemoteClass
Return the class's methods.
getMinimum(). Method in class java.awt.Scrollbar
Returns the minimum value of this Scrollbar.
GetMinSize(Container, GridBagLayoutInfo). Method in class java.awt.GridBagLayout
getMinutes(). Method in class java.util.Date
Returns the minute.
getMode(). Method in class java.awt.FileDialog
Gets the mode of the file dialog.
getModifiers(). Method in class sun.tools.debug.RemoteField
Returns a string with the field's modifiers, such as "public", "static", "final", etc.
getMonth(). Method in class java.util.Date
Returns the month.
getName(). Method in class java.lang.Class
Returns the name of this Class.
getName(). Method in class java.io.File
Gets the name of the file.
getName(). Method in class java.awt.Font
Gets the logical name of the font.
getName(). Method in class sun.tools.debug.RemoteClass
Returns the name of the class.
getName(). Method in class sun.tools.debug.RemoteField
Returns the name of the field.
getName(). Method in class sun.tools.debug.RemoteStackVariable
Return the name of a stack variable or argument.
getName(). Method in class sun.tools.debug.RemoteThread
Return the name of the thread.
getName(). Method in class sun.tools.debug.RemoteThreadGroup

Return the threadgroup's name.
getName(). Method in class java.lang.Thread
Gets and returns this Thread's name.
getName(). Method in class java.lang.ThreadGroup
Gets the name of this Thread group.
getOrientation(). Method in class java.awt.Scrollbar
Returns the orientation for this Scrollbar.
getOutputStream(). Method in class java.lang.Process
Returns a Stream connected to the input of the child process.
getOutputStream(). Method in class java.net.Socket
Gets an OutputStream for this socket.
getOutputStream(). Method in class java.net.SocketImpl
Gets an OutputStream for this socket.
getOutputStream(). Method in class java.net.URLConnection
Calls this routine to get an OutputStream that writes to the object.
getPageIncrement(). Method in class java.awt.Scrollbar
Gets the page increment for this scrollbar.
getParameter(String). Method in class java.applet.Applet
Gets a parameter of the applet.
getParameter(String). Method in interface java.applet.AppletStub
Gets a paramater of the applet.
getParameterInfo(). Method in class java.applet.Applet
Returns an array of strings describing the parameters that are understoond by this applet.
getParent(). Method in class java.awt.Component
Gets the parent of the component.
getParent(). Method in class java.io.File
Gets the name of the parent directory.
getParent(). Method in class java.awt.MenuComponent
Returns the parent container.
getParent(). Method in class java.lang.ThreadGroup
Gets the parent of this Thread group.
getPath(). Method in class java.io.File
Gets the path of the file.
getPC(). Method in class sun.tools.debug.RemoteStackFrame
Get the program counter referenced by this stackframe.
getPeer(). Method in class java.awt.Component
Gets the peer of the component.
getPeer(). Method in class java.awt.MenuComponent
Gets the menucomponent's peer.
getPixelSize(). Method in class java.awt.image.ColorModel
Return the number of bits per pixel described by this ColorModel.
getPort(). Method in class java.net.DatagramPacket
getPort(). Method in class java.net.Socket
Gets the remote port to which the socket is connected.
getPort(). Method in class java.net.SocketImpl
getPort(). Method in class java.net.URL
Gets the port number.
getPriority(). Method in class java.lang.Thread
Gets and returns the Thread's priority.

getProperties(). Static method in class java.lang.System
Gets the System properties.
getProperty(String). Method in class java.util.Properties
Gets a property with the specified key.
getProperty(String). Static method in class java.lang.System
Gets the System property indicated by the specified key.
getProperty(String, ImageObserver). Method in class java.awt.Image
Gets a property of the image by name.
getProperty(String, String). Method in class java.util.Properties
Gets a property with the specified key and default.
getProperty(String, String). Static method in class java.lang.System
Gets the System property indicated by the specified key and def.
getProtocol(). Method in class java.net.URL
Gets the protocol name.
getRed(). Method in class java.awt.Color
Gets the red component.
getRed(int). Method in class java.awt.image.ColorModel
The subclass must provide a function which provides the red color compoment for the specified pixel.
getRed(int). Method in class java.awt.image.DirectColorModel
Return the red color compoment for the specified pixel in the range 0−255.
getRed(int). Method in class java.awt.image.IndexColorModel
Return the red color compoment for the specified pixel in the range 0−255.
getRedMask(). Method in class java.awt.image.DirectColorModel
Return the mask indicating which bits in a pixel contain the red color component.
getReds(byte[]). Method in class java.awt.image.IndexColorModel
Copy the array of red color components into the given array.
getRef(). Method in class java.net.URL
Gets the ref.
getRemoteClass(). Method in class sun.tools.debug.RemoteStackFrame
Get the class this stackframe references.
getRequestProperty(String). Method in class java.net.URLConnection
getRGB(). Method in class java.awt.Color
Gets the RGB value representing the color in the default RGB ColorModel.
getRGB(int). Method in class java.awt.image.ColorModel
Return the color of the pixel in the default RGB color model.
getRGB(int). Method in class java.awt.image.DirectColorModel
Return the color of the pixel in the default RGB color model.
getRGB(int). Method in class java.awt.image.IndexColorModel
Return the color of the pixel in the default RGB color model.
getRGBdefault(). Static method in class java.awt.image.ColorModel
Return a ColorModel which describes the default format for integer RGB values used throughout the AWT image interfaces.
getRows(). Method in class java.awt.List
Returns the number of visible lines in this list.
getRows(). Method in class java.awt.TextArea
Returns the number of rows in the TextArea.
getRuntime(). Static method in class java.lang.Runtime
Return the runtime.

getScreenResolution(). Method in class java.awt.Toolkit
Returns the screen resolution in dots−per−inch.
getScreenSize(). Method in class java.awt.Toolkit
Gets the size of the screen.
getSeconds(). Method in class java.util.Date
Returns the second.
getSecurityContext(). Method in class java.lang.SecurityManager
Returns an implementation−dependent Object which encapsulates enough information about the current execution environment to perform some of the security checks later.
getSecurityManager(). Static method in class java.lang.System
Gets the system security interface.
getSelectedIndex(). Method in class java.awt.Choice
Returns the index of the currently selected item.
getSelectedIndex(). Method in class java.awt.List
Get the selected item on the list or −1 if no item is selected.
getSelectedIndexes(). Method in class java.awt.List
Returns the selected indexes on the list.
getSelectedIndexes(). Method in interface java.awt.peer.ListPeer
getSelectedItem(). Method in class java.awt.Choice
Returns a String representation of the current choice.
getSelectedItem(). Method in class java.awt.List
Returns the selected item on the list or null if no item is selected.
getSelectedItems(). Method in class java.awt.List
Returns the selected items on the list.
getSelectedText(). Method in class java.awt.TextComponent
Returns the selected text contained in this TextComponent.
getSelectionEnd(). Method in class java.awt.TextComponent
Returns the selected text's end position.
getSelectionEnd(). Method in interface java.awt.peer.TextComponentPeer
getSelectionStart(). Method in class java.awt.TextComponent
Returns the selected text's start position.
getSelectionStart(). Method in interface java.awt.peer.TextComponentPeer
getSize(). Method in class java.awt.Font
Gets the point size of the font.
getSize(). Method in class sun.tools.debug.RemoteArray
Return the number of elements in the array.
getSource(). Method in class java.awt.Image
Gets the object that produces the pixels for the image.
getSourceFile(). Method in class sun.tools.debug.RemoteClass
Get the source file referenced by this stackframe.
getSourceFileName(). Method in class sun.tools.debug.RemoteClass
Get the name of the source file referenced by this stackframe.
getSourcePath(). Method in class sun.tools.debug.RemoteDebugger
Return the source file path the Agent is currently using.
getStackVariable(String). Method in class sun.tools.debug.RemoteThread
Return a stack variable from the current stackframe.
getStackVariables(). Method in class sun.tools.debug.RemoteThread
Return the arguments and local variable from the current stackframe.
getState(). Method in class java.awt.Checkbox

Returns the boolean state of the Checkbox.
getState(). Method in class java.awt.CheckboxMenuItem
Returns the state of this MenuItem.
getStaticFields(). Method in class sun.tools.debug.RemoteClass
Return all the static fields for this class.
getStatus(). Method in class sun.tools.debug.RemoteThread
Return the thread status description
getStyle(). Method in class java.awt.Font
Gets the style of the font.
getSuperclass(). Method in class java.lang.Class
Returns the superclass of this Class.
getSuperclass(). Method in class sun.tools.debug.RemoteClass
Return the superclass for this class.
getText(). Method in class java.awt.Label
Gets the text of this label.
getText(). Method in class java.awt.TextComponent
Returns the text contained in this TextComponent.
getText(). Method in interface java.awt.peer.TextComponentPeer
getThreadGroup(). Method in class java.lang.Thread
Gets and returns this Thread group.
getTime(). Method in class java.util.Date
Returns the time in milliseconds since the epoch.
getTimezoneOffset(). Method in class java.util.Date
Return the time zone offset in minutes for the current locale that is appropriate for this time.
getTitle(). Method in class java.awt.Dialog
Gets the title of the Dialog.
getTitle(). Method in class java.awt.Frame
Gets the title of the Frame.
getToolkit(). Method in class java.awt.Component
Gets the toolkit of the component.
getToolkit(). Method in interface java.awt.peer.ComponentPeer
getToolkit(). Method in class java.awt.Window
Returns the toolkit of this frame.
getTransparentPixel(). Method in class java.awt.image.IndexColorModel
Returns the index of the transparent pixel in this IndexColorModel or −1 if there is no transparent pixel.
getType(). Method in class sun.tools.debug.RemoteField
Returns a type string describing the field.
getType(). Method in class sun.tools.debug.RemoteValue
Returns the RemoteValue's type.
getURL(). Method in class java.net.URLConnection
Gets the URL for this connection.
getUseCaches(). Method in class java.net.URLConnection
getValue(). Method in class sun.tools.debug.RemoteStackVariable
Return the value of a stack variable or argument.
getValue(). Method in class java.awt.Scrollbar
Returns the current value of this Scrollbar.
getVisible(). Method in class java.awt.Scrollbar
Returns the visible amount of the Scrollbar.

getVisibleIndex(). Method in class java.awt.List
Gets the index of the item that was last made visible by the method makeVisible.
getWarningString(). Method in class java.awt.Window
Gets the warning string for this window.
getWidth(ImageObserver). Method in class java.awt.Image
Gets the actual width of the image.
getWidths(). Method in class java.awt.FontMetrics
Gets the widths of the first 256 characters in the Font.
getYear(). Method in class java.util.Date
Returns the year after 1900.
GOT_FOCUS. Static variable in class java.awt.Event
A component gained the focus.
gotFocus(Event, Object). Method in class java.awt.Component
Indicates that this component has received the input focus.
grabPixels(). Method in class java.awt.image.PixelGrabber
Request the Image or ImageProducer to start delivering pixels and wait for all of the pixels in the rectangle of interest to be delivered.
grabPixels(long). Method in class java.awt.image.PixelGrabber
Request the Image or ImageProducer to start delivering pixels and wait for all of the pixels in the rectangle of interest to be delivered or until the specified timeout has elapsed.
Graphics(). Constructor for class java.awt.Graphics
Constructs a new Graphics Object.
gray. Static variable in class java.awt.Color
The color gray.
green. Static variable in class java.awt.Color
The color green.
GridBagConstraints(). Constructor for class java.awt.GridBagConstraints
GridBagLayout(). Constructor for class java.awt.GridBagLayout
Creates a gridbag layout.
gridheight. Variable in class java.awt.GridBagConstraints
GridLayout(int, int). Constructor for class java.awt.GridLayout
Creates a grid layout with the specified rows and specified columns.
GridLayout(int, int, int, int). Constructor for class java.awt.GridLayout
Creates a grid layout with the specified rows, columns, horizontal gap, and vertical gap.
gridwidth. Variable in class java.awt.GridBagConstraints
gridx. Variable in class java.awt.GridBagConstraints
gridy. Variable in class java.awt.GridBagConstraints
grow(int, int). Method in class java.awt.Rectangle
Grow the rectangle horizontally and vertically.
guessContentTypeFromName(String). Static method in class java.net.URLConnection
A useful utility routine that tries to guess the content−type of an object based upon its extension.
guessContentTypeFromStream(InputStream). Static method in class java.net.URLConnection
This disgusting hack is used to check for files have some type that can be determined by inspection.

H

HAND_CURSOR. Static variable in class java.awt.Frame
handleEvent(Event). Method in class java.awt.Component
Handles the event.
handleEvent(Event). Method in interface java.awt.peer.ComponentPeer
hasChanged(). Method in class java.util.Observable
Returns a true boolean if an observable change has occurred.
hashCode(). Method in class java.util.BitSet
Gets the hashcode.
hashCode(). Method in class java.lang.Boolean
Returns a hashcode for this Boolean.
hashCode(). Method in class java.lang.Character
Returns a hashcode for this Character.
hashCode(). Method in class java.awt.Color
Computes the hash code.
hashCode(). Method in class java.util.Date
Computes a hashCode.
hashCode(). Method in class java.lang.Double
Returns a hashcode for this Double.
hashCode(). Method in class java.io.File
Computes a hashcode for the file.
hashCode(). Method in class java.lang.Float
Returns a hashcode for this Float.
hashCode(). Method in class java.awt.Font
Returns a hashcode for this font.
hashCode(). Method in class java.net.InetAddress
Returns a hashcode for this InetAddress.
hashCode(). Method in class java.lang.Integer
Returns a hashcode for this Integer.
hashCode(). Method in class java.lang.Long
Computes a hashcode for this Long.
hashCode(). Method in class java.lang.Object
Returns a hashcode for this Object.
hashCode(). Method in class java.awt.Point
Hashcode
hashCode(). Method in class java.awt.Rectangle
Hashcode
hashCode(). Method in class java.lang.String
Returns a hashcode for this String.
hashCode(). Method in class java.net.URL
Creates an integer suitable for hash table indexing.
Hashtable(). Constructor for class java.util.Hashtable
Constructs a new, empty hashtable.
Hashtable(int). Constructor for class java.util.Hashtable
Constructs a new, empty hashtable with the specified initial capacity.
Hashtable(int, float). Constructor for class java.util.Hashtable
Constructs a new, empty hashtable with the specified initial capacity and the specified load factor.

hasMoreElements(). Method in interface java.util.Enumeration
Returns true if the enumeration contains more elements; false if its empty.
hasMoreElements(). Method in class java.util.StringTokenizer
Returns true if the Enumeration has more elements.
hasMoreTokens(). Method in class java.util.StringTokenizer
Returns true if more tokens exist.
height. Variable in class java.awt.Dimension
The height dimension.
HEIGHT. Static variable in interface java.awt.image.ImageObserver
The height of the base image is now available and can be taken from the height argument to the imageUpdate callback method.
height. Variable in class java.awt.Rectangle
The height of the rectangle.
hide(). Method in class java.awt.Component
Hides the component.
hide(). Method in interface java.awt.peer.ComponentPeer
HOME. Static variable in class java.awt.Event
The home key.
HORIZONTAL. Static variable in class java.awt.GridBagConstraints
HORIZONTAL. Static variable in class java.awt.Scrollbar
The horizontal Scrollbar variable.
HSBtoRGB(float, float, float). Static method in class java.awt.Color
Returns the RGB value defined by the default RGB ColorModel, of the color corresponding to the given HSB color components.

I

id. Variable in class java.awt.Event
The type of this event.
IEEEremainder(double, double). Static method in class java.lang.Math
Returns the remainder of f1 divided by f2 as defined by IEEE 754.
ifModifiedSince. Variable in class java.net.URLConnection
ignoreExceptions(). Method in class sun.tools.debug.RemoteClass
Don't enter the debugger when an instance of this class is thrown.
IllegalAccessError(). Constructor for class java.lang.IllegalAccessError
Constructs an IllegalAccessError with no detail message.
IllegalAccessError(String). Constructor for class java.lang.IllegalAccessError
Constructs an IllegalAccessError with the specified detail message.
IllegalAccessException(). Constructor for class java.lang.IllegalAccessException
Constructs a IllegalAccessException without a detail message.
IllegalAccessException(String). Constructor for class java.lang.IllegalAccessException
Constructs a IllegalAccessException with a detail message.
IllegalArgumentException(). Constructor for class java.lang.IllegalArgumentException
Constructs an IllegalArgumentException with no detail message.
IllegalArgumentException(String). Constructor for class java.lang.IllegalArgumentException
Constructs an IllegalArgumentException with the specified detail message.

IllegalMonitorStateException(). Constructor for class java.lang.IllegalMonitorStateException
Constructs an IllegalMonitorStateException with no detail message.
IllegalMonitorStateException(String). Constructor for class java.lang.IllegalMonitorStateException
Constructs an IllegalMonitorStateException with the specified detail message.
IllegalThreadStateException(). Constructor for class java.lang.IllegalThreadStateException
Constructs an IllegalThreadStateException with no detail message.
IllegalThreadStateException(String). Constructor for class java.lang.IllegalThreadStateException
Constructs an IllegalThreadStateException with the specified detail message.
Image(). Constructor for class java.awt.Image
IMAGEABORTED. Static variable in interface java.awt.image.ImageConsumer
The image creation process was deliberately aborted.
imageComplete(int). Method in interface java.awt.image.ImageConsumer
The imageComplete method is called when the ImageProducer is finished delivering all of the pixels that the source image contains, or when a single frame of a multi−frame animation has been completed, or when an error in loading or producing the image has occured.
imageComplete(int). Method in class java.awt.image.ImageFilter
Filter the information provided in the imageComplete method of the ImageConsumer interface.
imageComplete(int). Method in class java.awt.image.PixelGrabber
The imageComplete method is part of the ImageConsumer API which this class must implement to retrieve the pixels.
IMAGEERROR. Static variable in interface java.awt.image.ImageConsumer
An error was encountered while producing the image.
ImageFilter(). Constructor for class java.awt.image.ImageFilter
imageUpdate(Image, int, int, int, int, int). Method in class java.awt.Component
Repaints the component when the image has changed.
imageUpdate(Image, int, int, int, int, int). Method in interface java.awt.image.ImageObserver
This method is called when information about an image which was previously requested using an asynchronous interface becomes available.
in. Static variable in class java.io.FileDescriptor
in, out and err are handles to standard input, standard output and standard error respectively.
in. Variable in class java.io.FilterInputStream
The actual input stream.
in. Static variable in class java.lang.System
Standard input stream.
inCheck. Variable in class java.lang.SecurityManager
inClass(String). Method in class java.lang.SecurityManager
Returns true if the specified String is in this Class.
inClassLoader(). Method in class java.lang.SecurityManager
Returns a boolean indicating whether or not the current ClassLoader is equal to null.
IncompatibleClassChangeError(). Constructor for class java.lang.IncompatibleClassChangeError
Constructs an IncompatibleClassChangeError with no detail message.
IncompatibleClassChangeError(String). Constructor for class java.lang.IncompatibleClassChangeError
Constructs an IncompatibleClassChangeError with the specified detail message.
IndexColorModel(int, int, byte[], byte[], byte[]). Constructor for class java.awt.image.IndexColorModel
Construct an IndexColorModel from the given arrays of red, green, and blue components.
IndexColorModel(int, int, byte[], byte[], byte[], byte[]). Constructor for class java.awt.image.IndexColorModel

Construct an IndexColorModel from the given arrays of red, green, blue and alpha components.
IndexColorModel(int, int, byte[], byte[], byte[], int). Constructor for class java.awt.image.IndexColorModel
Construct an IndexColorModel from the given arrays of red, green, and blue components.
IndexColorModel(int, int, byte[], int, boolean). Constructor for class java.awt.image.IndexColorModel
Construct an IndexColorModel from a single arrays of packed red, green, blue and optional alpha components.
IndexColorModel(int, int, byte[], int, boolean, int). Constructor for class java.awt.image.IndexColorModel
Construct an IndexColorModel from a single arrays of packed red, green, blue and optional alpha components.
indexOf(int). Method in class java.lang.String
Returns the index within this String of the first occurrence of the specified character.
indexOf(int, int). Method in class java.lang.String
Returns the index within this String of the first occurrence of the specified character, starting the search at fromIndex.
indexOf(Object). Method in class java.util.Vector
Searches for the specified object, starting from the first position and returns an index to it.
indexOf(Object, int). Method in class java.util.Vector
Searches for the specified object, starting at the specified position and returns an index to it.
indexOf(String). Method in class java.lang.String
Returns the index within this String of the first occurrence of the specified substring.
indexOf(String, int). Method in class java.lang.String
Returns the index within this String of the first occurrence of the specified substring.
IndexOutOfBoundsException(). Constructor for class java.lang.IndexOutOfBoundsException
Constructs an IndexOutOfBoundsException with no detail message.
IndexOutOfBoundsException(String). Constructor for class java.lang.IndexOutOfBoundsException
Constructs a IndexOutOfBoundsException with the specified detail message.
init(). Method in class java.applet.Applet
Initializes the applet.
InputStream(). Constructor for class java.io.InputStream
inScope(). Method in class sun.tools.debug.RemoteStackVariable
Return whether variable is in scope.
insert(int, boolean). Method in class java.lang.StringBuffer
Inserts a boolean into the String buffer.
insert(int, char). Method in class java.lang.StringBuffer
Inserts a character into the String buffer.
insert(int, char[]). Method in class java.lang.StringBuffer
Inserts an array of characters into the String buffer.
insert(int, double). Method in class java.lang.StringBuffer
Inserts a double into the String buffer.
insert(int, float). Method in class java.lang.StringBuffer
Inserts a float into the String buffer.
insert(int, int). Method in class java.lang.StringBuffer
Inserts an integer into the String buffer.
insert(int, long). Method in class java.lang.StringBuffer
Inserts a long into the String buffer.
insert(int, Object). Method in class java.lang.StringBuffer

Inserts an object into the String buffer.
insert(int, String). Method in class java.lang.StringBuffer
Inserts a String into the String buffer.
insertElementAt(Object, int). Method in class java.util.Vector
Inserts the specified object as an element at the specified index.
insertText(String, int). Method in class java.awt.TextArea
Inserts the specified text at the specified position.
insertText(String, int). Method in interface java.awt.peer.TextAreaPeer
insets. Variable in class java.awt.GridBagConstraints
insets(). Method in class java.awt.Container
Returns the insets of the container.
insets(). Method in interface java.awt.peer.ContainerPeer
Insets(int, int, int, int). Constructor for class java.awt.Insets
Constructs and initializes a new Inset with the specified top, left, bottom, and right insets.
inside(int, int). Method in class java.awt.Component
Checks whether a specified x,y location is "inside" this Component.
inside(int, int). Method in class java.awt.Polygon
Is point (x, y) inside the polygon? Uses an even−odd insideness rule (otherwise known as an alternating rule).
inside(int, int). Method in class java.awt.Rectangle
Checks if the specified point lies inside a rectangle.
InstantiationError(). Constructor for class java.lang.InstantiationError
Constructs an InstantiationError with no detail message.
InstantiationError(String). Constructor for class java.lang.InstantiationError
Constructs an InstantiationError with the specified detail message.
InstantiationException(). Constructor for class java.lang.InstantiationException
Constructs an InstantiationException with no detail message.
InstantiationException(String). Constructor for class java.lang.InstantiationException
Constructs an InstantiationException with the specified detail message.
intBitsToFloat(int). Static method in class java.lang.Float
Returns the single−float corresponding to a given bit represention.
Integer(int). Constructor for class java.lang.Integer
Constructs an Integer object initialized to the specified int value.
Integer(String). Constructor for class java.lang.Integer
Constructs an Integer object initialized to the value specified by the String parameter.
intern(). Method in class java.lang.String
Returns a String that is equal to this String but which is guaranteed to be from the unique String pool.
InternalError(). Constructor for class java.lang.InternalError
Constructs an InternalError with no detail message.
InternalError(String). Constructor for class java.lang.InternalError
Constructs an InternalError with the specified detail message.
interrupt(). Method in class java.lang.Thread
Send an interrupt to a thread.
interrupted(). Static method in class java.lang.Thread
Ask if you have been interrupted.
InterruptedException(). Constructor for class java.lang.InterruptedException
Constructs an InterruptedException with no detail message.
InterruptedException(String). Constructor for class java.lang.InterruptedException

Constructs an InterruptedException with the specified detail message.
InterruptedIOException(). Constructor for class java.io.InterruptedIOException
Constructs an IOException with no detail message.
InterruptedIOException(String). Constructor for class java.io.InterruptedIOException
Constructs an IOException with the specified detail message.
intersection(Rectangle). Method in class java.awt.Rectangle
Compute the intersection of two rectangles.
intersects(Rectangle). Method in class java.awt.Rectangle
Check if two rectangles intersect.
intValue(). Method in class java.lang.Double
Returns the integer value of this Double (by casting to an int).
intValue(). Method in class java.lang.Float
Returns the integer value of this Float (by casting to an int).
intValue(). Method in class java.lang.Integer
Returns the value of this Integer as an int.
intValue(). Method in class java.lang.Long
Returns the value of this Long as an int.
intValue(). Method in class java.lang.Number
Returns the value of the number as an int.
invalidate(). Method in class java.awt.Component
Invalidates a component.
IOException(). Constructor for class java.io.IOException
Constructs an IOException with no detail message.
IOException(String). Constructor for class java.io.IOException
Constructs an IOException with the specified detail message.
ipadx. Variable in class java.awt.GridBagConstraints
ipady. Variable in class java.awt.GridBagConstraints
isAbsolute(). Method in class java.io.File
Returns a boolean indicating if the file name is absolute.
isActive(). Method in class java.applet.Applet
Returns true if the applet is active.
isActive(). Method in interface java.applet.AppletStub
Returns true if the applet is active.
isAlive(). Method in class java.lang.Thread
Returns a boolean indicating if the Thread is active.
isBold(). Method in class java.awt.Font
Returns true if the font is bold.
isConsumer(ImageConsumer). Method in class java.awt.image.FilteredImageSource
Determine if an ImageConsumer is on the list of consumers currently interested in data for this image.
isConsumer(ImageConsumer). Method in interface java.awt.image.ImageProducer
This method determines if a given ImageConsumer object is currently registered with this ImageProducer as one of its consumers.
isConsumer(ImageConsumer). Method in class java.awt.image.MemoryImageSource
Determine if an ImageConsumer is on the list of consumers currently interested in data for this image.
isDaemon(). Method in class java.lang.Thread
Returns the daemon flag of the Thread.
isDaemon(). Method in class java.lang.ThreadGroup

Returns the daemon flag of the Thread group.
isDigit(char). Static method in class java.lang.Character
Determines if the specified character is a ISO−LATIN−1 digit.
isDirectory(). Method in class java.io.File
Returns a boolean indicating whether or not a directory file exists.
isEditable(). Method in class java.awt.TextComponent
Returns the boolean indicating whether this TextComponent is editable or not.
isEmpty(). Method in class java.util.Dictionary
Returns true if the Dictionary contains no elements.
isEmpty(). Method in class java.util.Hashtable
Returns true if the hashtable contains no elements.
isEmpty(). Method in class java.awt.Rectangle
Check if the rectangle is empty.
isEmpty(). Method in class java.util.Vector
Returns true if the collection contains no values.
isEnabled(). Method in class java.awt.Component
Checks if this Component is enabled.
isEnabled(). Method in class java.awt.MenuItem
Checks if the menu item is enabled.
isErrorAny(). Method in class java.awt.MediaTracker
Check the error status of all of the images.
isErrorID(int). Method in class java.awt.MediaTracker
Check the error status of all of the images with the specified ID.
isFile(). Method in class java.io.File
Returns a boolean indicating whether or not a normal file exists.
isInfinite(). Method in class java.lang.Double
Returns true if this Double value is infinitely large in magnitude.
isInfinite(). Method in class java.lang.Float
Returns true if this Float value is infinitely large in magnitude.
isInfinite(double). Static method in class java.lang.Double
Returns true if the specified number is infinitely large in magnitude.
isInfinite(float). Static method in class java.lang.Float
Returns true if the specified number is infinitely large in magnitude.
isInterface(). Method in class java.lang.Class
Returns a boolean indicating whether or not this Class is an interface.
isInterface(). Method in class sun.tools.debug.RemoteClass
Is this RemoteClass an interface?
isInterrupted(). Method in class java.lang.Thread
Ask if another thread has been interrupted.
isItalic(). Method in class java.awt.Font
Returns true if the font is italic.
isLowerCase(char). Static method in class java.lang.Character
Determines if the specified character is ISO−LATIN−1 lower case.
isModal(). Method in class java.awt.Dialog
Returns true if the Dialog is modal.
isNaN(). Method in class java.lang.Double
Returns true if this Double value is the special Not−a−Number (NaN) value.
isNaN(). Method in class java.lang.Float
Returns true if this Float value is Not−a−Number (NaN).

isNaN(double). Static method in class java.lang.Double
Returns true if the specified number is the special Not−a−Number (NaN) value.
isNaN(float). Static method in class java.lang.Float
Returns true if the specified number is the special Not−a−Number (NaN) value.
isObject(). Method in class sun.tools.debug.RemoteValue
Returns whether the RemoteValue is an Object (as opposed to a primitive type, such as int).
isPlain(). Method in class java.awt.Font
Returns true if the font is plain.
isResizable(). Method in class java.awt.Dialog
Returns true if the user can resize the frame.
isResizable(). Method in class java.awt.Frame
Returns true if the user can resize the Frame.
isSelected(int). Method in class java.awt.List
Returns true if the item at the specified index has been selected; false otherwise.
isShowing(). Method in class java.awt.Component
Checks if this Component is showing on screen.
isSpace(char). Static method in class java.lang.Character
Determines if the specified character is ISO−LATIN−1 white space according to Java.
isStatic(). Method in class sun.tools.debug.RemoteField
Returns whether the field is static (a class variable or method).
isSuspended(). Method in class sun.tools.debug.RemoteThread
Return whether this thread is suspended.
isTearOff(). Method in class java.awt.Menu
Returns true if this is a tear−off menu.
isUpperCase(char). Static method in class java.lang.Character
Determines if the specified character is ISO−LATIN−1 upper case.
isValid(). Method in class java.awt.Component
Checks if this Component is valid.
isVisible(). Method in class java.awt.Component
Checks if this Component is visible.
ITALIC. Static variable in class java.awt.Font
The italicized style constant.
itrace(boolean). Method in class sun.tools.debug.RemoteDebugger
Turn on/off instruction tracing.

J

join(). Method in class java.lang.Thread
Waits forever for this Thread to die.
join(long). Method in class java.lang.Thread
Waits for this Thread to die.
join(long, int). Method in class java.lang.Thread
Waits for the Thread to die, with more precise time.

K

key. Variable in class java.awt.Event
The key that was pressed in a keyboard event.
KEY_ACTION. Static variable in class java.awt.Event
The key action keyboard event.
KEY_ACTION_RELEASE. Static variable in class java.awt.Event
The key action keyboard event.
KEY_PRESS. Static variable in class java.awt.Event
The key press keyboard event.
KEY_RELEASE. Static variable in class java.awt.Event
The key release keyboard event.
keyDown(Event, int). Method in class java.awt.Component
Called if a character is pressed.
keys(). Method in class java.util.Dictionary
Returns an enumeration of the Dictionary's keys.
keys(). Method in class java.util.Hashtable
Returns an enumeration of the hashtable's keys.
keyUp(Event, int). Method in class java.awt.Component
Called if a character is released.

L

Label(). Constructor for class java.awt.Label
Constructs an empty label.
Label(String). Constructor for class java.awt.Label
Constructs a new label with the specified String of text.
Label(String, int). Constructor for class java.awt.Label
Constructs a new label with the specified String of text and the specified alignment.
last(Container). Method in class java.awt.CardLayout
Flips to the last card of the specified container.
lastElement(). Method in class java.util.Vector
Returns the last element of the sequence.
lastIndexOf(int). Method in class java.lang.String
Returns the index within this String of the last occurrence of the specified character.
lastIndexOf(int, int). Method in class java.lang.String
Returns the index within this String of the last occurrence of the specified character.
lastIndexOf(Object). Method in class java.util.Vector
Searches backwards for the specified object, starting from the last position and returns an index to it.
lastIndexOf(Object, int). Method in class java.util.Vector
Searches backwards for the specified object, starting from the specified position and returns an index to it.
lastIndexOf(String). Method in class java.lang.String
Returns the index within this String of the last occurrence of the specified substring.
lastIndexOf(String, int). Method in class java.lang.String

Returns the index within this String of the last occurrence of the specified substring.
lastModified(). Method in class java.io.File
Returns the last modification time.
layout(). Method in class java.awt.Component
Lays out the component.
layout(). Method in class java.awt.Container
Does a layout on this Container.
layoutContainer(Container). Method in class java.awt.BorderLayout
Lays out the specified container.
layoutContainer(Container). Method in class java.awt.CardLayout
Performs a layout in the specified panel.
layoutContainer(Container). Method in class java.awt.FlowLayout
Lays out the container.
layoutContainer(Container). Method in class java.awt.GridBagLayout
Lays out the container in the specified panel.
layoutContainer(Container). Method in class java.awt.GridLayout
Lays out the container in the specified panel.
layoutContainer(Container). Method in interface java.awt.LayoutManager
Lays out the container in the specified panel.
layoutInfo. Variable in class java.awt.GridBagLayout
LEFT. Static variable in class java.awt.Event
The left arrow key.
LEFT. Static variable in class java.awt.FlowLayout
The left alignment variable.
left. Variable in class java.awt.Insets
The inset from the left.
LEFT. Static variable in class java.awt.Label
The left alignment.
length(). Method in class java.io.File
Returns the length of the file.
length(). Method in class java.io.RandomAccessFile
Returns the length of the file.
length(). Method in class java.lang.String
Returns the length of the String.
length(). Method in class java.lang.StringBuffer
Returns the length (character count) of the buffer.
lightGray. Static variable in class java.awt.Color
The color light gray.
lineno(). Method in class java.io.StreamTokenizer
Return the current line number.
LineNumberInputStream(InputStream). Constructor for class java.io.LineNumberInputStream
Constructs a new LineNumberInputStream initialized with the specified input stream
LinkageError(). Constructor for class java.lang.LinkageError
Constructs a LinkageError with no detail message.
LinkageError(String). Constructor for class java.lang.LinkageError
Constructs a LinkageError with the specified detail message.
list(). Method in class java.awt.Component
Prints a listing to a print stream.
list(). Method in class java.io.File

Lists the files in a directory.
List(). Constructor for class java.awt.List
Creates a new scrolling list initialized with no visible Lines or multiple selections.
list(). Method in class java.lang.ThreadGroup
Lists this Thread group.
list(FilenameFilter). Method in class java.io.File
Uses the specified filter to list files in a directory.
List(int, boolean). Constructor for class java.awt.List
Creates a new scrolling list initialized with the specified number of visible lines and a boolean stating whether multiple selections are allowed or not.
list(PrintStream). Method in class java.awt.Component
Prints a listing to the specified print out stream.
list(PrintStream). Method in class java.util.Properties
List properties, for debugging
list(PrintStream, int). Method in class java.awt.Component
Prints out a list, starting at the specified indention, to the specified print stream.
list(PrintStream, int). Method in class java.awt.Container
Prints out a list, starting at the specified indention, to the specified out stream.
LIST_DESELECT. Static variable in class java.awt.Event
LIST_SELECT. Static variable in class java.awt.Event
listBreakpoints(). Method in class sun.tools.debug.RemoteDebugger
Return a list of the breakpoints which are currently set.
listClasses(). Method in class sun.tools.debug.RemoteDebugger
List the currently known classes.
listen(int). Method in class java.net.SocketImpl
Listens for connections over a specified amount of time.
listThreadGroups(RemoteThreadGroup). Method in class sun.tools.debug.RemoteDebugger
List threadgroups
listThreads(boolean). Method in class sun.tools.debug.RemoteThreadGroup
List a threadgroup's threads
LOAD. Static variable in class java.awt.FileDialog
The file load variable.
load(InputStream). Method in class java.util.Properties
Loads properties from an InputStream.
load(String). Method in class java.lang.Runtime
Loads a dynamic library, given a complete path name.
load(String). Static method in class java.lang.System
Loads a dynamic library, given a complete path name.
LOAD_FILE. Static variable in class java.awt.Event
A file loading event.
loadClass(String, boolean). Method in class java.lang.ClassLoader
Resolves the specified name to a Class.
LOADING. Static variable in class java.awt.MediaTracker
Flag indicating some media is currently being loaded.
loadLibrary(String). Method in class java.lang.Runtime
Loads a dynamic library with the specified library name.
loadLibrary(String). Static method in class java.lang.System
Loads a dynamic library with the specified library name.
localport. Variable in class java.net.SocketImpl

locate(int, int). Method in class java.awt.Component
Returns the component or subcomponent that contains the x,y location.
locate(int, int). Method in class java.awt.Container
Locates the component that contains the x,y position.
location(). Method in class java.awt.Component
Returns the current location of this component.
location(int, int). Method in class java.awt.GridBagLayout
log(double). Static method in class java.lang.Math
Returns the natural logarithm (base e) of a.
Long(long). Constructor for class java.lang.Long
Constructs a Long object initialized to the specified value.
Long(String). Constructor for class java.lang.Long
Constructs a Long object initialized to the value specified by the String parameter.
longBitsToDouble(long). Static method in class java.lang.Double
Returns the double−float corresponding to a given bit represention.
longValue(). Method in class java.lang.Double
Returns the long value of this Double (by casting to a long).
longValue(). Method in class java.lang.Float
Returns the long value of this Float (by casting to a long).
longValue(). Method in class java.lang.Integer
Returns the value of this Integer as a long.
longValue(). Method in class java.lang.Long
Returns the value of this Long as a long.
longValue(). Method in class java.lang.Number
Returns the value of the number as a long.
lookupConstraints(Component). Method in class java.awt.GridBagLayout
Retrieves the constraints for the specified component.
loop(). Method in interface java.applet.AudioClip
Start playing the clip in a loop.
LOST_FOCUS. Static variable in class java.awt.Event
A component lost the focus.
lostFocus(Event, Object). Method in class java.awt.Component
Indicates that this component has lost the input focus.
lowerCaseMode(boolean). Method in class java.io.StreamTokenizer
Examines a boolean to decide whether TT_WORD tokens are forced to be lower case.

M

magenta. Static variable in class java.awt.Color
The color magneta.
makeVisible(int). Method in class java.awt.List
Forces the item at the specified index to be visible.
makeVisible(int). Method in interface java.awt.peer.ListPeer
MalformedURLException(). Constructor for class java.net.MalformedURLException
Constructs a MalformedURLException with no detail message.
MalformedURLException(String). Constructor for class java.net.MalformedURLException

Constructs a MalformedURLException with the specified detail message.
mark(int). Method in class java.io.BufferedInputStream
Marks the current position in the input stream.
mark(int). Method in class java.io.FilterInputStream
Marks the current position in the input stream.
mark(int). Method in class java.io.InputStream
Marks the current position in the input stream.
mark(int). Method in class java.io.LineNumberInputStream
Marks the current position in the input stream.
marklimit. Variable in class java.io.BufferedInputStream
The maximum readahead allowed after a mark() before subsequent calls to reset() fail.
markpos. Variable in class java.io.BufferedInputStream
The position in the buffer of the current mark.
markSupported(). Method in class java.io.BufferedInputStream
Returns a boolean indicating if this stream type supports mark/reset.
markSupported(). Method in class java.io.FilterInputStream
Returns true if this stream type supports mark/reset
markSupported(). Method in class java.io.InputStream
Returns a boolean indicating whether or not this stream type supports mark/reset.
markSupported(). Method in class java.io.PushbackInputStream
Returns true if this stream type supports mark/reset.
max(double, double). Static method in class java.lang.Math
Takes two double values, a and b, and returns the greater number of the two.
max(float, float). Static method in class java.lang.Math
Takes two float values, a and b, and returns the greater number of the two.
max(int, int). Static method in class java.lang.Math
Takes two int values, a and b, and returns the greater number of the two.
max(long, long). Static method in class java.lang.Math
Takes two long values, a and b, and returns the greater number of the two.
MAX_PRIORITY. Static variable in class java.lang.Thread
The maximum priority that a Thread can have.
MAX_RADIX. Static variable in class java.lang.Character
The maximum radix available for conversion to and from Strings.
MAX_VALUE. Static variable in class java.lang.Boolean
The maximum value a Character can have.
MAX_VALUE. Static variable in class java.lang.Double
The maximum value a double can have.
MAX_VALUE. Static variable in class java.lang.Float
The maximum value a float can have.
MAX_VALUE. Static variable in class java.lang.Integer
The maximum value an Integer can have.
MAX_VALUE. Static variable in class java.lang.Long
The maximum value a Long can have.
MAXGRIDSIZE. Static variable in class java.awt.GridBagLayout
MediaTracker(Component). Constructor for class java.awt.MediaTracker
Create a Media tracker to track images for a given Component.
MemoryImageSource(int, int, ColorModel, byte[], int, int). Constructor for class java.awt.image.MemoryImageSource
Construct an ImageProducer object which uses an array of bytes to produce data for an Image

object.
MemoryImageSource(int, int, ColorModel, byte[], int, int, Hashtable). Constructor for class java.awt.image.MemoryImageSource
Construct an ImageProducer object which uses an array of bytes to produce data for an Image object.
MemoryImageSource(int, int, ColorModel, int[], int, int). Constructor for class java.awt.image.MemoryImageSource
Construct an ImageProducer object which uses an array of integers to produce data for an Image object.
MemoryImageSource(int, int, ColorModel, int[], int, int, Hashtable). Constructor for class java.awt.image.MemoryImageSource
Construct an ImageProducer object which uses an array of integers to produce data for an Image object.
MemoryImageSource(int, int, int[], int, int). Constructor for class java.awt.image.MemoryImageSource
Construct an ImageProducer object which uses an array of integers in the default RGB ColorModel to produce data for an Image object.
MemoryImageSource(int, int, int[], int, int, Hashtable). Constructor for class java.awt.image.MemoryImageSource
Construct an ImageProducer object which uses an array of integers in the default RGB ColorModel to produce data for an Image object.
Menu(String). Constructor for class java.awt.Menu
Constructs a new Menu with the specified label.
Menu(String, boolean). Constructor for class java.awt.Menu
Constructs a new Menu with the specified label.
MenuBar(). Constructor for class java.awt.MenuBar
Creates a new menu bar.
MenuComponent(). Constructor for class java.awt.MenuComponent
MenuItem(String). Constructor for class java.awt.MenuItem
Constructs a new MenuItem with the specified label.
META_MASK. Static variable in class java.awt.Event
The meta modifier constant.
metaDown(). Method in class java.awt.Event
Checks if the meta key is down.
min(double, double). Static method in class java.lang.Math
Takes two double values, a and b, and returns the smallest number of the two.
min(float, float). Static method in class java.lang.Math
Takes two float values, a and b, and returns the smallest number of the two.
min(int, int). Static method in class java.lang.Math
Takes two integer values, a and b, and returns the smallest number of the two.
min(long, long). Static method in class java.lang.Math
Takes two long values, a and b, and returns the smallest number of the two.
MIN_PRIORITY. Static variable in class java.lang.Thread
The minimum priority that a Thread can have.
MIN_RADIX. Static variable in class java.lang.Character
The minimum radix available for conversion to and from Strings.
MIN_VALUE. Static variable in class java.lang.Boolean
The minimum value a Charater can have.
MIN_VALUE. Static variable in class java.lang.Double

The minimum value a double can have.
MIN_VALUE. Static variable in class java.lang.Float
The minimum value a float can have.
MIN_VALUE. Static variable in class java.lang.Integer
The minimum value an Integer can have.
MIN_VALUE. Static variable in class java.lang.Long
The minimum value a Long can have.
minimumLayoutSize(Container). Method in class java.awt.BorderLayout
Returns the minimum dimensions needed to layout the components contained in the specified target container.
minimumLayoutSize(Container). Method in class java.awt.CardLayout
Calculates the minimum size for the specified panel.
minimumLayoutSize(Container). Method in class java.awt.FlowLayout
Returns the minimum dimensions needed to layout the components contained in the specified target container.
minimumLayoutSize(Container). Method in class java.awt.GridBagLayout
Returns the minimum dimensions needed to layout the components contained in the specified panel.
minimumLayoutSize(Container). Method in class java.awt.GridLayout
Returns the minimum dimensions needed to layout the components contained in the specified panel.
minimumLayoutSize(Container). Method in interface java.awt.LayoutManager
Calculates the minimum size dimensions for the specified panel given the components in the specified parent container.
minimumSize(). Method in class java.awt.Component
Returns the minimum size of this component.
minimumSize(). Method in interface java.awt.peer.ComponentPeer
minimumSize(). Method in class java.awt.Container
Returns the minimum size of this container.
minimumSize(). Method in class java.awt.List
Returns the minimum dimensions needed for the list.
minimumSize(). Method in class java.awt.TextArea
Returns the minimum size Dimensions of the TextArea.
minimumSize(). Method in class java.awt.TextField
Returns the minimum size Dimensions needed for this TextField.
minimumSize(int). Method in class java.awt.List
Returns the minimum dimensions needed for the amount of rows in the list.
minimumSize(int). Method in interface java.awt.peer.ListPeer
minimumSize(int). Method in class java.awt.TextField
Returns the minimum size Dimensions needed for this TextField with the specified amount of columns.
minimumSize(int). Method in interface java.awt.peer.TextFieldPeer
minimumSize(int, int). Method in class java.awt.TextArea
Returns the specified minimum size Dimensions of the TextArea.
minimumSize(int, int). Method in interface java.awt.peer.TextAreaPeer
MINSIZE. Static variable in class java.awt.GridBagLayout
mkdir(). Method in class java.io.File
Creates a directory and returns a boolean indicating the success of the creation.
mkdirs(). Method in class java.io.File

Creates all directories in this path.
modifiers. Variable in class java.awt.Event
The state of the modifier keys.
MOUSE_DOWN. Static variable in class java.awt.Event
The mouse down event.
MOUSE_DRAG. Static variable in class java.awt.Event
The mouse drag event.
MOUSE_ENTER. Static variable in class java.awt.Event
The mouse enter event.
MOUSE_EXIT. Static variable in class java.awt.Event
The mouse exit event.
MOUSE_MOVE. Static variable in class java.awt.Event
The mouse move event.
MOUSE_UP. Static variable in class java.awt.Event
The mouse up event.
mouseDown(Event, int, int). Method in class java.awt.Component
Called if the mouse is down.
mouseDrag(Event, int, int). Method in class java.awt.Component
Called if the mouse is dragged (the mouse button is down).
mouseEnter(Event, int, int). Method in class java.awt.Component
Called when the mouse enters the component.
mouseExit(Event, int, int). Method in class java.awt.Component
Called when the mouse exits the component.
mouseMove(Event, int, int). Method in class java.awt.Component
Called if the mouse moves (the mouse button is up).
mouseUp(Event, int, int). Method in class java.awt.Component
Called if the mouse is up.
move(int, int). Method in class java.awt.Component
Moves the Component to a new location.
move(int, int). Method in class java.awt.Point
Move the point.
move(int, int). Method in class java.awt.Rectangle
Move the rectangle.
MOVE_CURSOR. Static variable in class java.awt.Frame

N

N_RESIZE_CURSOR. Static variable in class java.awt.Frame
name. Variable in class java.awt.Font
The logical name of this font.
NaN. Static variable in class java.lang.Double
Not−a−Number.
NaN. Static variable in class java.lang.Float
Not−a−Number.
NE_RESIZE_CURSOR. Static variable in class java.awt.Frame
NEGATIVE_INFINITY. Static variable in class java.lang.Double

Negative infinity.
NEGATIVE_INFINITY. Static variable in class java.lang.Float
Negative infinity.
NegativeArraySizeException(). Constructor for class java.lang.NegativeArraySizeException
Constructs a NegativeArraySizeException with no detail message.
NegativeArraySizeException(String). Constructor for class java.lang.NegativeArraySizeException
Constructs a NegativeArraySizeException with the specified detail message.
newInstance(). Method in class java.lang.Class
Creates a new instance of this Class.
newmodel. Variable in class java.awt.image.RGBImageFilter
next(). Method in class sun.tools.debug.RemoteThread
Continue execution of this thread to the next line, but don't step into a method call.
next(Container). Method in class java.awt.CardLayout
Flips to the next card of the specified container.
nextDouble(). Method in class java.util.Random
Generates a pseudorandom uniformally distributed `double` value between 0.0 and 1.0.
nextElement(). Method in interface java.util.Enumeration
Returns the next element of the enumeration.
nextElement(). Method in class java.util.StringTokenizer
Returns the next element in the Enumeration.
nextFloat(). Method in class java.util.Random
Generates a pseudorandom uniformally distributed `float` value between 0.0 and 1.0.
nextFocus(). Method in class java.awt.Component
Moves the focus to the next component.
nextFocus(). Method in interface java.awt.peer.ComponentPeer
nextGaussian(). Method in class java.util.Random
Generates a pseudorandom Gaussian distributed `double` value with mean 0.0 and standard deviation 1.0.
nextInt(). Method in class java.util.Random
Generates a pseudorandom uniformally distributed `int` value.
nextLong(). Method in class java.util.Random
Generate a pseudorandom uniformally distributed `long` value.
nextToken(). Method in class java.io.StreamTokenizer
Parses a token from the input stream.
nextToken(). Method in class java.util.StringTokenizer
Returns the next token of the String.
nextToken(String). Method in class java.util.StringTokenizer
Returns the next token, after switching to the new delimiter set.
NoClassDefFoundError(). Constructor for class java.lang.NoClassDefFoundError
Constructs a NoClassDefFoundError with no detail message.
NoClassDefFoundError(String). Constructor for class java.lang.NoClassDefFoundError
Constructs a NoClassDefFoundError with the specified detail message.
NONE. Static variable in class java.awt.GridBagConstraints
NORM_PRIORITY. Static variable in class java.lang.Thread
The default priority that is assigned to a Thread.
NORTH. Static variable in class java.awt.GridBagConstraints
NORTHEAST. Static variable in class java.awt.GridBagConstraints
NORTHWEST. Static variable in class java.awt.GridBagConstraints
NoSuchElementException(). Constructor for class java.util.NoSuchElementException

Constructs a NoSuchElementException with no detail message.
NoSuchElementException(String). Constructor for class java.util.NoSuchElementException
Constructs a NoSuchElementException with the specified detail message.
NoSuchFieldError(). Constructor for class java.lang.NoSuchFieldError
Constructs a NoSuchFieldException without a detail message.
NoSuchFieldError(String). Constructor for class java.lang.NoSuchFieldError
Constructs a NoSuchFieldException with a detail message.
NoSuchMethodError(). Constructor for class java.lang.NoSuchMethodError
NoSuchMethodError(String). Constructor for class java.lang.NoSuchMethodError
Constructs a NoSuchMethodException with a detail message.
NoSuchMethodException(). Constructor for class java.lang.NoSuchMethodException
Constructs a NoSuchMethodException without a detail message.
NoSuchMethodException(String). Constructor for class java.lang.NoSuchMethodException
Constructs a NoSuchMethodException with a detail message.
notify(). Method in class java.lang.Object
Notifies a single waiting thread on a change in condition of another thread.
notifyAll(). Method in class java.lang.Object
Notifies all of the threads waiting for a condition to change.
notifyObservers(). Method in class java.util.Observable
Notifies all observers if an observable change occurs.
notifyObservers(Object). Method in class java.util.Observable
Notifies all observers of the specified observable change which occurred.
npoints. Variable in class java.awt.Polygon
The total number of points.
NullPointerException(). Constructor for class java.lang.NullPointerException
Constructs a NullPointerException with no detail message.
NullPointerException(String). Constructor for class java.lang.NullPointerException
Constructs a NullPointerException with the specified detail message.
Number(). Constructor for class java.lang.Number
NumberFormatException(). Constructor for class java.lang.NumberFormatException
Constructs a NumberFormatException with no detail message.
NumberFormatException(String). Constructor for class java.lang.NumberFormatException
Constructs a NumberFormatException with the specified detail message.
nval. Variable in class java.io.StreamTokenizer
The number value.
NW_RESIZE_CURSOR. Static variable in class java.awt.Frame

O

Object(). Constructor for class java.lang.Object
Observable(). Constructor for class java.util.Observable
openConnection(). Method in class java.net.URL
Creates (if not already in existance) a URLConnection object that contains a connection to the remote object referred to by the URL.
openConnection(URL). Method in class java.net.URLStreamHandler
Opens an input stream to the object referenced by the URL.

openStream(). Method in class java.net.URL
Opens an input stream.
or(BitSet). Method in class java.util.BitSet
Logically ORs this bit set with the specified set of bits.
orange. Static variable in class java.awt.Color
The color orange.
ordinaryChar(int). Method in class java.io.StreamTokenizer
Specifies that this character is 'ordinary': it removes any significance as a word, comment, string, whitespace or number character.
ordinaryChars(int, int). Method in class java.io.StreamTokenizer
Specifies that characters in this range are 'ordinary'.
origmodel. Variable in class java.awt.image.RGBImageFilter
out. Static variable in class java.io.FileDescriptor
out. Variable in class java.io.FilterOutputStream
The actual output stream.
out. Static variable in class java.lang.System
Standard output stream.
OutOfMemoryError(). Constructor for class java.lang.OutOfMemoryError
Constructs an OutOfMemoryError with no detail message.
OutOfMemoryError(String). Constructor for class java.lang.OutOfMemoryError
Constructs an OutOfMemoryError with the specified detail message.
OutputStream(). Constructor for class java.io.OutputStream

P

pack(). Method in class java.awt.Window
Packs the components of the WIndow.
paint(Graphics). Method in class java.awt.Canvas
Paints the canvas in the default background color.
paint(Graphics). Method in class java.awt.Component
Paints the component.
paint(Graphics). Method in interface java.awt.peer.ComponentPeer
paintAll(Graphics). Method in class java.awt.Component
Paints the component and its subcomponents.
paintComponents(Graphics). Method in class java.awt.Container
Paints the components in this container.
Panel(). Constructor for class java.awt.Panel
Creates a new panel.
paramString(). Method in class java.awt.Button
Returns the parameter String of this button.
paramString(). Method in class java.awt.Checkbox
Returns the parameter String of this Checkbox.
paramString(). Method in class java.awt.CheckboxMenuItem
Returns the parameter String of this button.
paramString(). Method in class java.awt.Choice
Returns the parameter String of this Choice.

paramString(). Method in class java.awt.Component
Returns the parameter String of this Component.
paramString(). Method in class java.awt.Container
Returns the parameter String of this Container.
paramString(). Method in class java.awt.Dialog
Returns the parameter String of this Dialog.
paramString(). Method in class java.awt.Event
Returns the parameter String of this Event.
paramString(). Method in class java.awt.FileDialog
Returns the parameter String of this file dialog.
paramString(). Method in class java.awt.Frame
Returns the parameter String of this Frame.
paramString(). Method in class java.awt.Label
Returns the parameter String of this label.
paramString(). Method in class java.awt.List
Returns the parameter String of this list.
paramString(). Method in class java.awt.MenuComponent
Returns the String parameter of this MenuComponent.
paramString(). Method in class java.awt.MenuItem
Returns the String parameter of the menu item.
paramString(). Method in class java.awt.Scrollbar
Returns the String parameters for this Scrollbar.
paramString(). Method in class java.awt.TextArea
Returns the String of parameters for this TextArea.
paramString(). Method in class java.awt.TextComponent
Returns the String of parameters for this TextComponent.
paramString(). Method in class java.awt.TextField
Returns the String of parameters for this TExtField.
parentOf(ThreadGroup). Method in class java.lang.ThreadGroup
Checks to see if this Thread group is a parent of or is equal to another Thread group.
parse(String). Static method in class java.util.Date
Given a string representing a time, parse it and return the time value.
parseInt(String). Static method in class java.lang.Integer
Assuming the specified String represents an integer, returns that integer's value.
parseInt(String, int). Static method in class java.lang.Integer
Assuming the specified String represents an integer, returns that integer's value.
parseLong(String). Static method in class java.lang.Long
Assuming the specified String represents a long, return that long's value.
parseLong(String, int). Static method in class java.lang.Long
Assuming the specified String represents a long, returns that long's value.
parseNumbers(). Method in class java.io.StreamTokenizer
Specifies that numbers should be parsed.
parseURL(URL, String, int, int). Method in class java.net.URLStreamHandler
This method is called to parse the string spec into URL u.
pathSeparator. Static variable in class java.io.File
The system dependent path separator string.
pathSeparatorChar. Static variable in class java.io.File
The system dependent path separator character.
peek(). Method in class java.util.Stack

Peeks at the top of the stack.
PGDN. Static variable in class java.awt.Event
The page down key.
PGUP. Static variable in class java.awt.Event
The page up key.
PI. Static variable in class java.lang.Math
The float representation of the value Pi.
pink. Static variable in class java.awt.Color
The color pink.
PipedInputStream(). Constructor for class java.io.PipedInputStream
Creates an input file that isn't connected to anything (yet).
PipedInputStream(PipedOutputStream). Constructor for class java.io.PipedInputStream
Creates an input file from the specified PiledOutputStream.
PipedOutputStream(). Constructor for class java.io.PipedOutputStream
Creates an output file that isn't connected to anything (yet).
PipedOutputStream(PipedInputStream). Constructor for class java.io.PipedOutputStream
Creates an output file connected to the specified PipedInputStream.
pixel_bits. Variable in class java.awt.image.ColorModel
PixelGrabber(Image, int, int, int, int, int[], int, int). Constructor for class java.awt.image.PixelGrabber
Create a PixelGrabber object to grab the (x, y, w, h) rectangular section of pixels from the specified image into the given array.
PixelGrabber(ImageProducer, int, int, int, int, int[], int, int). Constructor for class java.awt.image.PixelGrabber
Create a PixelGrabber object to grab the (x, y, w, h) rectangular section of pixels from the image produced by the specified ImageProducer into the given array.
PLAIN. Static variable in class java.awt.Font
The plain style constant.
play(). Method in interface java.applet.AudioClip
Start playing the clip.
play(URL). Method in class java.applet.Applet
Play an audio clip.
play(URL, String). Method in class java.applet.Applet
Play an audio clip.
Point(int, int). Constructor for class java.awt.Point
Constructs and initializes a Point from the specified x and y coordinates.
Polygon(). Constructor for class java.awt.Polygon
Creates an empty polygon.
Polygon(int[], int[], int). Constructor for class java.awt.Polygon
Constructs and initializes a Polygon from the specified parameters.
pop(). Method in class java.util.Stack
Pops an item off the stack.
port. Variable in class java.net.SocketImpl
The port where the socket will make connection.
pos. Variable in class java.io.BufferedInputStream
The current position in the buffer.
pos. Variable in class java.io.ByteArrayInputStream
The current position in the buffer.
pos. Variable in class java.io.StringBufferInputStream

The position in the buffer.
POSITIVE_INFINITY. Static variable in class java.lang.Double
Positive infinity.
POSITIVE_INFINITY. Static variable in class java.lang.Float
Positive infinity.
postEvent(Event). Method in class java.awt.Component
Posts an event to this component.
postEvent(Event). Method in class java.awt.MenuComponent
Posts the specified event to the menu.
postEvent(Event). Method in interface java.awt.MenuContainer
pow(double, double). Static method in class java.lang.Math
Returns the number a raised to the power of b.
preferredLayoutSize(Container). Method in class java.awt.BorderLayout
Returns the preferred dimensions for this layout given the components in the specified target container.
preferredLayoutSize(Container). Method in class java.awt.CardLayout
Calculates the preferred size for the specified panel.
preferredLayoutSize(Container). Method in class java.awt.FlowLayout
Returns the preferred dimensions for this layout given the components in the specified target container.
preferredLayoutSize(Container). Method in class java.awt.GridBagLayout
Returns the preferred dimensions for this layout given the components in the specified panel.
preferredLayoutSize(Container). Method in class java.awt.GridLayout
Returns the preferred dimensions for this layout given the components int the specified panel.
preferredLayoutSize(Container). Method in interface java.awt.LayoutManager
Calculates the preferred size dimensions for the specified panel given the components in the specified parent container.
PREFERREDSIZE. Static variable in class java.awt.GridBagLayout
preferredSize(). Method in class java.awt.Component
Returns the preferred size of this component.
preferredSize(). Method in interface java.awt.peer.ComponentPeer
preferredSize(). Method in class java.awt.Container
Returns the preferred size of this container.
preferredSize(). Method in class java.awt.List
Returns the preferred dimensions needed for the list.
preferredSize(). Method in class java.awt.TextArea
Returns the preferred size Dimensions of the TextArea.
preferredSize(). Method in class java.awt.TextField
Returns the preferred size Dimensions needed for this TextField.
preferredSize(int). Method in class java.awt.List
Returns the preferred dimensions needed for the list with the specified amount of rows.
preferredSize(int). Method in interface java.awt.peer.ListPeer
preferredSize(int). Method in class java.awt.TextField
Returns the preferred size Dimensions needed for this TextField with the specified amount of columns.
preferredSize(int). Method in interface java.awt.peer.TextFieldPeer
preferredSize(int, int). Method in class java.awt.TextArea
Returns the specified row and column Dimensions of the TextArea.
preferredSize(int, int). Method in interface java.awt.peer.TextAreaPeer

prepareImage(Image, ImageObserver). Method in class java.awt.Component
Prepares an image for rendering on this Component.
prepareImage(Image, int, int, ImageObserver). Method in class java.awt.Component
Prepares an image for rendering on this Component at the specified width and height.
prepareImage(Image, int, int, ImageObserver). Method in interface java.awt.peer.ComponentPeer
prepareImage(Image, int, int, ImageObserver). Method in class java.awt.Toolkit
Prepares an image for rendering on the default screen at the specified width and height.
previous(Container). Method in class java.awt.CardLayout
Flips to the previous card of the specified container.
print(boolean). Method in class java.io.PrintStream
Prints a boolean.
print(char). Method in class java.io.PrintStream
Prints an character.
print(char[]). Method in class java.io.PrintStream
Prints an array of characters.
print(double). Method in class java.io.PrintStream
Prints a double.
print(float). Method in class java.io.PrintStream
Prints a float.
print(Graphics). Method in class java.awt.Component
Prints this component.
print(Graphics). Method in interface java.awt.peer.ComponentPeer
print(int). Method in class java.io.PrintStream
Prints an integer.
print(long). Method in class java.io.PrintStream
Prints a long.
print(Object). Method in class java.io.PrintStream
Prints an object.
print(String). Method in class java.io.PrintStream
Prints a String.
printAll(Graphics). Method in class java.awt.Component
Prints the component and its subcomponents.
printComponents(Graphics). Method in class java.awt.Container
Prints the components in this container.
println(). Method in class java.io.PrintStream
Prints a newline.
println(boolean). Method in class java.io.PrintStream
Prints a boolean followed by a newline.
println(char). Method in class java.io.PrintStream
Prints a character followed by a newline.
println(char[]). Method in class java.io.PrintStream
Prints an array of characters followed by a newline.
println(double). Method in class java.io.PrintStream
Prints a double followed by a newline.
println(float). Method in class java.io.PrintStream
Prints a float followed by a newline.
println(int). Method in class java.io.PrintStream
Prints an integer followed by a newline.
println(long). Method in class java.io.PrintStream

Prints a long followed by a newline.
println(Object). Method in class java.io.PrintStream
Prints an object followed by a newline.
println(String). Method in class java.io.PrintStream
Prints a string followed by a newline.
printStackTrace(). Method in class java.lang.Throwable
Prints the Throwable and the Throwable's stack trace.
printStackTrace(PrintStream). Method in class java.lang.Throwable
PrintStream(OutputStream). Constructor for class java.io.PrintStream
Creates a new PrintStream.
PrintStream(OutputStream, boolean). Constructor for class java.io.PrintStream
Creates a new PrintStream, with auto flushing.
printToConsole(String). Method in interface sun.tools.debug.DebuggerCallback
Print text to the debugger's console window.
Process(). Constructor for class java.lang.Process
PROPERTIES. Static variable in interface java.awt.image.ImageObserver
The properties of the image are now available.
Properties(). Constructor for class java.util.Properties
Creates an empty property list.
Properties(Properties). Constructor for class java.util.Properties
Creates an empty property list with specified defaults.
propertyNames(). Method in class java.util.Properties
Enumerates all the keys.
ProtocolException(). Constructor for class java.net.ProtocolException
Constructs a new ProtocolException with no detail message.
ProtocolException(String). Constructor for class java.net.ProtocolException
Constructs a new ProtocolException with the specified detail message.
push(Object). Method in class java.util.Stack
Pushes an item onto the stack.
pushBack. Variable in class java.io.PushbackInputStream
Push back character.
pushBack(). Method in class java.io.StreamTokenizer
Pushes back a stream token.
PushbackInputStream(InputStream). Constructor for class java.io.PushbackInputStream
Creates a PushbackInputStream.
put(Object, Object). Method in class java.util.Dictionary
Puts the specified element into the Dictionary, using the specified key.
put(Object, Object). Method in class java.util.Hashtable
Puts the specified element into the hashtable, using the specified key.

Q

quitEvent(). Method in interface sun.tools.debug.DebuggerCallback
The client interpreter has exited, either by returning from its main thread, or by calling System.exit().
quoteChar(int). Method in class java.io.StreamTokenizer

Specifies that matching pairs of this character delimit String constants.

R

random(). Static method in class java.lang.Math
Generates a random number between 0.0 and 1.0.
Random(). Constructor for class java.util.Random
Creates a new random number generator.
Random(long). Constructor for class java.util.Random
Creates a new random number generator using a single `long` seed.
RandomAccessFile(File, String). Constructor for class java.io.RandomAccessFile
Creates a RandomAccessFile from a specified File object and mode ("r" or "rw").
RandomAccessFile(String, String). Constructor for class java.io.RandomAccessFile
Creates a RandomAccessFile with the specified system dependent file name and the specified mode.
RANDOMPIXELORDER. Static variable in interface java.awt.image.ImageConsumer
The pixels will be delivered in a random order.
read(). Method in class java.io.BufferedInputStream
Reads a byte of data.
read(). Method in class java.io.ByteArrayInputStream
Reads a byte of data.
read(). Method in class java.io.FileInputStream
Reads a byte of data.
read(). Method in class java.io.FilterInputStream
Reads a byte.
read(). Method in class java.io.InputStream
Reads a byte of data.
read(). Method in class java.io.LineNumberInputStream
Reads a byte of data.
read(). Method in class java.io.PipedInputStream
Reads a byte of data.
read(). Method in class java.io.PushbackInputStream
Reads a byte of data.
read(). Method in class java.io.RandomAccessFile
Reads a byte of data.
read(). Method in class java.io.SequenceInputStream
Reads a stream, and upon reaching an EOF, flips to the next stream.
read(). Method in class java.io.StringBufferInputStream
Reads a byte of data.
read(byte[]). Method in class java.io.DataInputStream
Reads data into an array of bytes.
read(byte[]). Method in class java.io.FileInputStream
Reads data into an array of bytes.
read(byte[]). Method in class java.io.FilterInputStream
Reads into an array of bytes.
read(byte[]). Method in class java.io.InputStream

Reads into an array of bytes.
read(byte[]). Method in class java.io.RandomAccessFile
Reads data into an array of bytes.
read(byte[], int, int). Method in class java.io.BufferedInputStream
Reads into an array of bytes.
read(byte[], int, int). Method in class java.io.ByteArrayInputStream
Reads into an array of bytes.
read(byte[], int, int). Method in class java.io.DataInputStream
Reads data into an array of bytes.
read(byte[], int, int). Method in class java.io.FileInputStream
Reads data into an array of bytes.
read(byte[], int, int). Method in class java.io.FilterInputStream
Reads into an array of bytes.
read(byte[], int, int). Method in class java.io.InputStream
Reads into an array of bytes.
read(byte[], int, int). Method in class java.io.LineNumberInputStream
Reads into an array of bytes.
read(byte[], int, int). Method in class java.io.PipedInputStream
Reads into an array of bytes.
read(byte[], int, int). Method in class java.io.PushbackInputStream
Reads into an array of bytes.
read(byte[], int, int). Method in class java.io.RandomAccessFile
Reads a sub array as a sequence of bytes.
read(byte[], int, int). Method in class java.io.SequenceInputStream
Reads data into an array of bytes, and upon reaching an EOF, flips to the next stream.
read(byte[], int, int). Method in class java.io.StringBufferInputStream
Reads into an array of bytes.
readBoolean(). Method in interface java.io.DataInput
Reads in a boolean.
readBoolean(). Method in class java.io.DataInputStream
Reads in a boolean.
readBoolean(). Method in class java.io.RandomAccessFile
Reads a boolean.
readByte(). Method in interface java.io.DataInput
Reads an 8 bit byte.
readByte(). Method in class java.io.DataInputStream
Reads an 8 bit byte.
readByte(). Method in class java.io.RandomAccessFile
Reads a byte.
readChar(). Method in interface java.io.DataInput
Reads a 16 bit char.
readChar(). Method in class java.io.DataInputStream
Reads a 16 bit char.
readChar(). Method in class java.io.RandomAccessFile
Reads a 16 bit char.
readDouble(). Method in interface java.io.DataInput
Reads a 64 bit double.
readDouble(). Method in class java.io.DataInputStream
Reads a 64 bit double.

readDouble(). Method in class java.io.RandomAccessFile
    Reads a 64 bit double.
readFloat(). Method in interface java.io.DataInput
    Reads a 32 bit float.
readFloat(). Method in class java.io.DataInputStream
    Reads a 32 bit float.
readFloat(). Method in class java.io.RandomAccessFile
    Reads a 32 bit float.
readFully(byte[]). Method in interface java.io.DataInput
    Reads bytes, blocking until all bytes are read.
readFully(byte[]). Method in class java.io.DataInputStream
    Reads bytes, blocking until all bytes are read.
readFully(byte[]). Method in class java.io.RandomAccessFile
    Reads bytes, blocking until all bytes are read.
readFully(byte[], int, int). Method in interface java.io.DataInput
    Reads bytes, blocking until all bytes are read.
readFully(byte[], int, int). Method in class java.io.DataInputStream
    Reads bytes, blocking until all bytes are read.
readFully(byte[], int, int). Method in class java.io.RandomAccessFile
    Reads bytes, blocking until all bytes are read.
readInt(). Method in interface java.io.DataInput
    Reads a 32 bit int.
readInt(). Method in class java.io.DataInputStream
    Reads a 32 bit int.
readInt(). Method in class java.io.RandomAccessFile
    Reads a 32 bit int.
readLine(). Method in interface java.io.DataInput
readLine(). Method in class java.io.DataInputStream
    Reads in a line that has been terminated by a \n, \r, \r\n or EOF.
readLine(). Method in class java.io.RandomAccessFile
    Reads a line terminated by a '\n' or EOF.
readLong(). Method in interface java.io.DataInput
    Reads a 64 bit long.
readLong(). Method in class java.io.DataInputStream
    Reads a 64 bit long.
readLong(). Method in class java.io.RandomAccessFile
    Reads a 64 bit long.
readShort(). Method in interface java.io.DataInput
    Reads 16 bit short.
readShort(). Method in class java.io.DataInputStream
    Reads 16 bit short.
readShort(). Method in class java.io.RandomAccessFile
    Reads 16 bit short.
readUnsignedByte(). Method in interface java.io.DataInput
    Reads an unsigned 8 bit byte.
readUnsignedByte(). Method in class java.io.DataInputStream
    Reads an unsigned 8 bit byte.
readUnsignedByte(). Method in class java.io.RandomAccessFile
    Reads an unsigned 8 bit byte.

readUnsignedShort(). Method in interface java.io.DataInput
Reads an unsigned 16 bit short.
readUnsignedShort(). Method in class java.io.DataInputStream
Reads 16 bit short.
readUnsignedShort(). Method in class java.io.RandomAccessFile
Reads 16 bit short.
readUTF(). Method in interface java.io.DataInput
readUTF(). Method in class java.io.DataInputStream
Reads a UTF format String.
readUTF(). Method in class java.io.RandomAccessFile
Reads a UTF formatted String.
readUTF(DataInput). Static method in class java.io.DataInputStream
Reads a UTF format String from the given input stream.
receive(DatagramPacket). Method in class java.net.DatagramSocket
Receives datagram packet.
Rectangle(). Constructor for class java.awt.Rectangle
Constructs a new rectangle.
Rectangle(Dimension). Constructor for class java.awt.Rectangle
Constructs a rectangle and initializes it to the specified width and height.
Rectangle(int, int). Constructor for class java.awt.Rectangle
Constructs a rectangle and initializes it with the specified width and height parameters.
Rectangle(int, int, int, int). Constructor for class java.awt.Rectangle
Constructs and initializes a rectangle with the specified parameters.
Rectangle(Point). Constructor for class java.awt.Rectangle
Constructs a rectangle and initializes it to the specified point.
Rectangle(Point, Dimension). Constructor for class java.awt.Rectangle
Constructs a rectangle and initializes it to specified point and dimension.
red. Static variable in class java.awt.Color
The color red.
regionMatches(boolean, int, String, int, int). Method in class java.lang.String
Determines whether a region of this String matches the specified region of the specified String.
regionMatches(int, String, int, int). Method in class java.lang.String
Determines whether a region of this String matches the specified region of the specified String.
rehash(). Method in class java.util.Hashtable
Rehashes the content of the table into a bigger table.
RELATIVE. Static variable in class java.awt.GridBagConstraints
REMAINDER. Static variable in class java.awt.GridBagConstraints
RemoteDebugger(String, DebuggerCallback, boolean). Constructor for class sun.tools.debug.RemoteDebugger
Create a remote debugger, and connect it to a new client interpreter.

RemoteDebugger(String, String, DebuggerCallback, boolean). Constructor for class sun.tools.debug.RemoteDebugger
Create a remote debugger, connecting it with a running Java interpreter.

To connect to a running interpreter, it must be started with the "−debug" option, whereupon it will print out the password for that debugging session.
RemoteInt(int). Constructor for class sun.tools.debug.RemoteInt
remove(Component). Method in class java.awt.Container

Removes the specified component from this container.
remove(int). Method in class java.awt.Menu
Deletes the item at the specified index from this menu.
remove(int). Method in class java.awt.MenuBar
Removes the menu located at the specified index from the menu bar.
remove(MenuComponent). Method in class java.awt.Frame
Removes the specified menu bar from this Frame.
remove(MenuComponent). Method in class java.awt.Menu
Deletes the specified item from this menu.
remove(MenuComponent). Method in class java.awt.MenuBar
Removes the specified menu from the menu bar.
remove(MenuComponent). Method in interface java.awt.MenuContainer
remove(Object). Method in class java.util.Dictionary
Removes the element corresponding to the key.
remove(Object). Method in class java.util.Hashtable
Removes the element corresponding to the key.
removeAll(). Method in class java.awt.Container
Removes all the components from this container.
removeAllElements(). Method in class java.util.Vector
Removes all elements of the vector.
removeConsumer(ImageConsumer). Method in class java.awt.image.FilteredImageSource
Remove an ImageConsumer from the list of consumers interested in data for this image.
removeConsumer(ImageConsumer). Method in interface java.awt.image.ImageProducer
This method removes the given ImageConsumer object from the list of consumers currently registered to receive image data.
removeConsumer(ImageConsumer). Method in class java.awt.image.MemoryImageSource
Remove an ImageConsumer from the list of consumers interested in data for this image.
removeElement(Object). Method in class java.util.Vector
Removes the element from the vector.
removeElementAt(int). Method in class java.util.Vector
Deletes the element at the specified index.
removeLayoutComponent(Component). Method in class java.awt.BorderLayout
Removes the specified component from the layout.
removeLayoutComponent(Component). Method in class java.awt.CardLayout
Removes the specified component from the layout.
removeLayoutComponent(Component). Method in class java.awt.FlowLayout
Removes the specified component from the layout.
removeLayoutComponent(Component). Method in class java.awt.GridBagLayout
Removes the specified component from the layout.
removeLayoutComponent(Component). Method in class java.awt.GridLayout
Removes the specified component from the layout.
removeLayoutComponent(Component). Method in interface java.awt.LayoutManager
Removes the specified component from the layout.
removeNotify(). Method in class java.awt.Component
Notifies the Component to destroy the peer.
removeNotify(). Method in class java.awt.Container
Notifies the container to remove its peer.
removeNotify(). Method in class java.awt.List
Removes the peer for this list.

removeNotify(). Method in class java.awt.Menu
Removes the menu's peer.
removeNotify(). Method in class java.awt.MenuBar
Removes the menu bar's peer.
removeNotify(). Method in class java.awt.MenuComponent
Removes the menu component's peer.
removeNotify(). Method in class java.awt.TextComponent
Removes the TextComponent's peer.
renameTo(File). Method in class java.io.File
Renames a file and returns a boolean indicating whether or not this method was successful.
repaint(). Method in class java.awt.Component
Repaints the component.
repaint(int, int, int, int). Method in class java.awt.Component
Repaints part of the component.
repaint(long). Method in class java.awt.Component
Repaints the component.
repaint(long, int, int, int, int). Method in class java.awt.Component
Repaints part of the component.
repaint(long, int, int, int, int). Method in interface java.awt.peer.ComponentPeer
replace(char, char). Method in class java.lang.String
Converts this String by replacing all occurences of oldChar with newChar.
replaceItem(String, int). Method in class java.awt.List
Replaces the item at the given index.
replaceText(String, int, int). Method in class java.awt.TextArea
Replaces text from the indicated start to end position with the specified new text.
replaceText(String, int, int). Method in interface java.awt.peer.TextAreaPeer
requestFocus(). Method in class java.awt.Component
Requests the input focus.
requestFocus(). Method in interface java.awt.peer.ComponentPeer
requestTopDownLeftRightResend(ImageConsumer). Method in class java.awt.image.FilteredImageSource
Request that a given ImageConsumer have the image data delivered one more time in top−down, left−right order.
requestTopDownLeftRightResend(ImageConsumer). Method in interface java.awt.image.ImageProducer
This method is used by an ImageConsumer to request that the ImageProducer attempt to resend the image data one more time in TOPDOWNLEFTRIGHT order so that higher quality conversion algorithms which depend on receiving pixels in order can be used to produce a better output version of the image.
requestTopDownLeftRightResend(ImageConsumer). Method in class java.awt.image.MemoryImageSource
Request that a given ImageConsumer have the image data delivered one more time in top−down, left−right order.
resendTopDownLeftRight(ImageProducer). Method in class java.awt.image.ImageFilter
Respond to a request for a TopDownLeftRight (TDLR) ordered resend of the pixel data from an ImageConsumer.
reset(). Method in class java.io.BufferedInputStream
Repositions the stream to the last marked position.
reset(). Method in class java.io.ByteArrayInputStream

Resets the buffer to the beginning.

reset(). Method in class java.io.ByteArrayOutputStream
Resets the buffer so that you can use it again without throwing away the already allocated buffer.

reset(). Method in class java.io.FilterInputStream
Repositions the stream to the last marked position.

reset(). Method in class java.io.InputStream
Repositions the stream to the last marked position.

reset(). Method in class java.io.LineNumberInputStream
Repositions the stream to the last marked position.

reset(). Method in class java.io.StringBufferInputStream
Resets the buffer to the beginning.

resetCurrentFrameIndex(). Method in class sun.tools.debug.RemoteThread
Reset the current stackframe

resetSyntax(). Method in class java.io.StreamTokenizer
Resets the syntax table so that all characters are special.

reshape(int, int, int, int). Method in class java.awt.Component
Reshapes the Component to the specified bounding box.

reshape(int, int, int, int). Method in interface java.awt.peer.ComponentPeer

reshape(int, int, int, int). Method in class java.awt.Rectangle
Reshape the rectangle.

resize(Dimension). Method in class java.applet.Applet
Request for the applet to be resized.

resize(Dimension). Method in class java.awt.Component
Resizes the Component to the specified dimension.

resize(int, int). Method in class java.applet.Applet
Request for the applet to be resized.

resize(int, int). Method in class java.awt.Component
Resizes the Component to the specified width and height.

resize(int, int). Method in class java.awt.Rectangle
Resize the rectangle.

resolveClass(Class). Method in class java.lang.ClassLoader
Resolves classes referenced by this Class.

resume(). Method in class sun.tools.debug.RemoteThread
Resume execution of this thread.

resume(). Method in class java.lang.Thread
Resumes this Thread execution.

resume(). Method in class java.lang.ThreadGroup
Resumes all the Threads in this Thread group and all of its sub groups.

RGBImageFilter(). Constructor for class java.awt.image.RGBImageFilter

RGBtoHSB(int, int, int, float[]). Static method in class java.awt.Color
Returns the HSB values corresponding to the color defined by the red, green, and blue components.

RIGHT. Static variable in class java.awt.Event
The right arrow key.

RIGHT. Static variable in class java.awt.FlowLayout
The right alignment variable.

right. Variable in class java.awt.Insets
The inset from the right.

RIGHT. Static variable in class java.awt.Label
The right alignment.
rint(double). Static method in class java.lang.Math
Converts a double value into an integral value in double format.
round(double). Static method in class java.lang.Math
Rounds off a double value by first adding 0.5 to it and then returning the largest integer that is less than or equal to this new value.
round(float). Static method in class java.lang.Math
Rounds off a float value by first adding 0.5 to it and then returning the largest integer that is less than or equal to this new value.
rowHeights. Variable in class java.awt.GridBagLayout
rowWeights. Variable in class java.awt.GridBagLayout
run(). Method in interface java.lang.Runnable
The method that is executed when a Runnable object is activated.
run(). Method in class java.lang.Thread
The actual body of this Thread.
run(int, String[]). Method in class sun.tools.debug.RemoteDebugger
Load and run a runnable Java class, with any optional parameters.
runFinalization(). Method in class java.lang.Runtime
Runs the finalization methods of any objects pending finalization.
runFinalization(). Static method in class java.lang.System
Runs the finalization methods of any objects pending finalization.
RuntimeException(). Constructor for class java.lang.RuntimeException
Constructs a RuntimeException with no detail message.
RuntimeException(String). Constructor for class java.lang.RuntimeException
Constructs a RuntimeException with the specified detail message.

S

S_RESIZE_CURSOR. Static variable in class java.awt.Frame
sameFile(URL). Method in class java.net.URL
Compares two URLs, excluding the "ref" fields: sameFile is true if the true references the same remote object, but not necessarily the same subpiece of that object.
SAVE. Static variable in class java.awt.FileDialog
The file save variable.
save(OutputStream, String). Method in class java.util.Properties
Save properties to an OutputStream.
SAVE_FILE. Static variable in class java.awt.Event
A file saving event.
SCROLL_ABSOLUTE. Static variable in class java.awt.Event
The absolute scroll event.
SCROLL_LINE_DOWN. Static variable in class java.awt.Event
The line down scroll event.
SCROLL_LINE_UP. Static variable in class java.awt.Event
The line up scroll event.
SCROLL_PAGE_DOWN. Static variable in class java.awt.Event

The page down scroll event.
SCROLL_PAGE_UP. Static variable in class java.awt.Event
The page up scroll event.
Scrollbar(). Constructor for class java.awt.Scrollbar
Constructs a new vertical Scrollbar.
Scrollbar(int). Constructor for class java.awt.Scrollbar
Constructs a new Scrollbar with the specified orientation.
Scrollbar(int, int, int, int, int). Constructor for class java.awt.Scrollbar
Constructs a new Scrollbar with the specified orientation, value, page size, and minumum and maximum values.
SE_RESIZE_CURSOR. Static variable in class java.awt.Frame
search(Object). Method in class java.util.Stack
Sees if an object is on the stack.
SecurityException(). Constructor for class java.lang.SecurityException
Constructs a SecurityException with no detail message.
SecurityException(String). Constructor for class java.lang.SecurityException
Constructs a SecurityException with the specified detail message.
SecurityManager(). Constructor for class java.lang.SecurityManager
Constructs a new SecurityManager.
seek(long). Method in class java.io.RandomAccessFile
Sets the file pointer to the specified absolute position.
select(int). Method in class java.awt.Choice
Selects the item with the specified postion.
select(int). Method in interface java.awt.peer.ChoicePeer
select(int). Method in class java.awt.List
Selects the item at the specified index.
select(int). Method in interface java.awt.peer.ListPeer
select(int, int). Method in class java.awt.TextComponent
Selects the text found between the specified start and end locations.
select(int, int). Method in interface java.awt.peer.TextComponentPeer
select(String). Method in class java.awt.Choice
Selects the item with the specified String.
selectAll(). Method in class java.awt.TextComponent
Selects all the text in the TextComponent.
send(DatagramPacket). Method in class java.net.DatagramSocket
Sends Datagram Packet to the destination address
separator. Static variable in class java.io.File
The system dependent file separator String.
separatorChar. Static variable in class java.io.File
The system dependent file separator character.
SequenceInputStream(Enumeration). Constructor for class java.io.SequenceInputStream
Constructs a new SequenceInputStream initialized to the specified list.
SequenceInputStream(InputStream, InputStream). Constructor for class java.io.SequenceInputStream
Constructs a new SequenceInputStream initialized to the two specified input streams.
ServerSocket(int). Constructor for class java.net.ServerSocket
Creates a server socket on a specified port.
ServerSocket(int, int). Constructor for class java.net.ServerSocket
Creates a server socket, binds it to the specified local port and listens to it.

set(int). Method in class java.util.BitSet
Sets a bit.
set(String, String, int, String, String). Method in class java.net.URL
Sets the fields of the URL.
setAlignment(int). Method in class java.awt.Label
Sets the alignment for this label to the specified alignment.
setAlignment(int). Method in interface java.awt.peer.LabelPeer
setAllowUserInteraction(boolean). Method in class java.net.URLConnection
Some URL connections occasionally need to to interactions with the user.
setBackground(Color). Method in class java.awt.Component
Sets the background color.
setBackground(Color). Method in interface java.awt.peer.ComponentPeer
setBreakpointLine(int). Method in class sun.tools.debug.RemoteClass
Set a breakpoint at a specified source line number in a class.
setBreakpointMethod(RemoteField). Method in class sun.tools.debug.RemoteClass
Set a breakpoint at the first line of a class method.
setChanged(). Method in class java.util.Observable
Sets a flag to note an observable change.
setCharAt(int, char). Method in class java.lang.StringBuffer
Changes the character at the specified index to be ch.
setCheckboxGroup(CheckboxGroup). Method in class java.awt.Checkbox
Sets the CheckboxGroup to the specified group.
setCheckboxGroup(CheckboxGroup). Method in interface java.awt.peer.CheckboxPeer
setColor(Color). Method in class java.awt.Graphics
Sets the current color to the specified color.
setColorModel(ColorModel). Method in interface java.awt.image.ImageConsumer
The ColorModel object which will be used for the majority of the pixels that will be reported using the setPixels method calls.
setColorModel(ColorModel). Method in class java.awt.image.ImageFilter
Filter the information provided in the setColorModel method of the ImageConsumer interface.
setColorModel(ColorModel). Method in class java.awt.image.PixelGrabber
The setColorModel method is part of the ImageConsumer API which this class must implement to retrieve the pixels.
setColorModel(ColorModel). Method in class java.awt.image.RGBImageFilter
If the ColorModel is an IndexColorModel, and the subclass has set the canFilterIndexColorModel flag to true, then we substitute a filtered version of the color model here and whenever we see that original ColorModel object in the setPixels methods, otherwise we override the default ColorModel used by the ImageProducer and specify the default RGB ColorModel instead.
setConstraints(Component, GridBagConstraints). Method in class java.awt.GridBagLayout
Sets the constraints for the specified component.
setContentHandlerFactory(ContentHandlerFactory). Static method in class java.net.URLConnection
Sets the ContentHandler factory.
setCurrent(Checkbox). Method in class java.awt.CheckboxGroup
Sets the current choice to the specified Checkbox.
setCurrentFrameIndex(int). Method in class sun.tools.debug.RemoteThread
Set the current stackframe index
setCursor(int). Method in class java.awt.Frame
Set the cursor image to a predefined cursor.

setCursor(int). Method in interface java.awt.peer.FramePeer
setDaemon(boolean). Method in class java.lang.Thread
Marks this Thread as a daemon Thread or a user Thread.
setDaemon(boolean). Method in class java.lang.ThreadGroup
Changes the daemon status of this group.
setDate(int). Method in class java.util.Date
Sets the date.
setDefaultAllowUserInteraction(boolean). Static method in class java.net.URLConnection
Set/get the default value of the allowUserInteraction flag.
setDefaultRequestProperty(String, String). Static method in class java.net.URLConnection
Set/get the default value of a general request property.
setDefaultUseCaches(boolean). Method in class java.net.URLConnection
setDimensions(int, int). Method in class java.awt.image.CropImageFilter
Override the source image's dimensions and pass the dimensions of the rectangular cropped region to the ImageConsumer.
setDimensions(int, int). Method in interface java.awt.image.ImageConsumer
The dimensions of the source image are reported using the setDimensions method call.
setDimensions(int, int). Method in class java.awt.image.ImageFilter
Filter the information provided in the setDimensions method of the ImageConsumer interface.
setDimensions(int, int). Method in class java.awt.image.PixelGrabber
The setDimensions method is part of the ImageConsumer API which this class must implement to retrieve the pixels.
setDirectory(String). Method in class java.awt.FileDialog
Set the directory of the Dialog to the specified directory.
setDirectory(String). Method in interface java.awt.peer.FileDialogPeer
setDoInput(boolean). Method in class java.net.URLConnection
A URL connection can be used for input and/or output.
setDoOutput(boolean). Method in class java.net.URLConnection
A URL connection can be used for input and/or output.
setEchoCharacter(char). Method in class java.awt.TextField
Sets the echo character for this TextField.
setEchoCharacter(char). Method in interface java.awt.peer.TextFieldPeer
setEditable(boolean). Method in class java.awt.TextComponent
Sets the specified boolean to indicate whether or not this TextComponent should be editable.
setEditable(boolean). Method in interface java.awt.peer.TextComponentPeer
setElementAt(Object, int). Method in class java.util.Vector
Sets the element at the specified index to be the specified object.
setFile(String). Method in class java.awt.FileDialog
Sets the file for this dialog to the specified file.
setFile(String). Method in interface java.awt.peer.FileDialogPeer
setFilenameFilter(FilenameFilter). Method in class java.awt.FileDialog
Sets the filter for this dialog to the specified filter.
setFilenameFilter(FilenameFilter). Method in interface java.awt.peer.FileDialogPeer
setFont(Font). Method in class java.awt.Component
Sets the font of the component.
setFont(Font). Method in interface java.awt.peer.ComponentPeer
setFont(Font). Method in class java.awt.Graphics
Sets the font for all subsequent text−drawing operations.
setFont(Font). Method in class java.awt.MenuComponent

Sets the font to be used for this MenuItem to the specified font.
setForeground(Color). Method in class java.awt.Component
Sets the foreground color.
setForeground(Color). Method in interface java.awt.peer.ComponentPeer
setHelpMenu(Menu). Method in class java.awt.MenuBar
Sets the help menu to the specified menu on the menu bar.
setHints(int). Method in interface java.awt.image.ImageConsumer
The ImageProducer can deliver the pixels in any order, but the ImageConsumer may be able to scale or convert the pixels to the destination ColorModel more efficiently or with higher quality if it knows some information about how the pixels will be delivered up front.
setHints(int). Method in class java.awt.image.ImageFilter
Filter the information provided in the setHints method of the ImageConsumer interface.
setHints(int). Method in class java.awt.image.PixelGrabber
The setHints method is part of the ImageConsumer API which this class must implement to retrieve the pixels.
setHours(int). Method in class java.util.Date
Sets the hours.
setIconImage(Image). Method in class java.awt.Frame
Sets the image to display when this Frame is iconized.
setIconImage(Image). Method in interface java.awt.peer.FramePeer
setIfModifiedSince(long). Method in class java.net.URLConnection
Some protocols support skipping fetching unless the object is newer than some time.
setLabel(String). Method in class java.awt.Button
Sets the button with the specified label.
setLabel(String). Method in interface java.awt.peer.ButtonPeer
setLabel(String). Method in class java.awt.Checkbox
Sets the button with the specified label.
setLabel(String). Method in interface java.awt.peer.CheckboxPeer
setLabel(String). Method in class java.awt.MenuItem
Sets the label to be the specified label.
setLabel(String). Method in interface java.awt.peer.MenuItemPeer
setLayout(LayoutManager). Method in class java.awt.Container
Sets the layout manager for this container.
setLength(int). Method in class java.lang.StringBuffer
Sets the length of the String.
setLineIncrement(int). Method in class java.awt.Scrollbar
Sets the line increment for this scrollbar.
setLineIncrement(int). Method in interface java.awt.peer.ScrollbarPeer
setLineNumber(int). Method in class java.io.LineNumberInputStream
Sets the current line number.
setMaxPriority(int). Method in class java.lang.ThreadGroup
Sets the maximum priority of the group.
setMenuBar(MenuBar). Method in class java.awt.Frame
Sets the menubar for this Frame to the specified menubar.
setMenuBar(MenuBar). Method in interface java.awt.peer.FramePeer
setMinutes(int). Method in class java.util.Date
Sets the minutes.
setMonth(int). Method in class java.util.Date
Sets the month.

setMultipleSelections(boolean). Method in class java.awt.List
Sets whether this list should allow multiple selections or not.
setMultipleSelections(boolean). Method in interface java.awt.peer.ListPeer
setName(String). Method in class java.lang.Thread
Sets the Thread's name.
setPageIncrement(int). Method in class java.awt.Scrollbar
Sets the page increment for this scrollbar.
setPageIncrement(int). Method in interface java.awt.peer.ScrollbarPeer
setPaintMode(). Method in class java.awt.Graphics
Sets the paint mode to overwrite the destination with the current color.
setPixels(int, int, int, int, ColorModel, byte[], int, int). Method in class java.awt.image.CropImageFilter
Determine if the delivered byte pixels intersect the region to be extracted and pass through only that subset of pixels that appear in the output region.
setPixels(int, int, int, int, ColorModel, byte[], int, int). Method in interface java.awt.image.ImageConsumer
The pixels of the image are delivered using one or more calls to the setPixels method.
setPixels(int, int, int, int, ColorModel, byte[], int, int). Method in class java.awt.image.ImageFilter
Filter the information provided in the setPixels method of the ImageConsumer interface which takes an array of bytes.
setPixels(int, int, int, int, ColorModel, byte[], int, int). Method in class java.awt.image.PixelGrabber
The setPixels method is part of the ImageConsumer API which this class must implement to retrieve the pixels.
setPixels(int, int, int, int, ColorModel, byte[], int, int). Method in class java.awt.image.RGBImageFilter
If the ColorModel object is the same one that has already been converted, then simply pass the pixels through with the converted ColorModel, otherwise convert the buffer of byte pixels to the default RGB ColorModel and pass the converted buffer to the filterRGBPixels method to be converted one by one.
setPixels(int, int, int, int, ColorModel, int[], int, int). Method in class java.awt.image.CropImageFilter
Determine if the delivered int pixels intersect the region to be extracted and pass through only that subset of pixels that appear in the output region.
setPixels(int, int, int, int, ColorModel, int[], int, int). Method in interface java.awt.image.ImageConsumer
The pixels of the image are delivered using one or more calls to the setPixels method.
setPixels(int, int, int, int, ColorModel, int[], int, int). Method in class java.awt.image.ImageFilter
Filter the information provided in the setPixels method of the ImageConsumer interface which takes an array of integers.
setPixels(int, int, int, int, ColorModel, int[], int, int). Method in class java.awt.image.PixelGrabber
The setPixels method is part of the ImageConsumer API which this class must implement to retrieve the pixels.
setPixels(int, int, int, int, ColorModel, int[], int, int). Method in class java.awt.image.RGBImageFilter
If the ColorModel object is the same one that has already been converted, then simply pass the pixels through with the converted ColorModel, otherwise convert the buffer of integer pixels to the default RGB ColorModel and pass the converted buffer to the filterRGBPixels method to be converted one by one.
setPriority(int). Method in class java.lang.Thread
Sets the Thread's priority.
setProperties(Hashtable). Method in class java.awt.image.CropImageFilter

Pass the properties from the source object along after adding a property indicating the cropped region.

setProperties(Hashtable). Method in interface java.awt.image.ImageConsumer
The extensible list of properties associated with this image.

setProperties(Hashtable). Method in class java.awt.image.ImageFilter
Pass the properties from the source object along after adding a property indicating the stream of filters it has been run through.

setProperties(Hashtable). Method in class java.awt.image.PixelGrabber
The setProperties method is part of the ImageConsumer API which this class must implement to retrieve the pixels.

setProperties(Properties). Static method in class java.lang.System
Sets the System properties to the specified properties.

setRequestProperty(String, String). Method in class java.net.URLConnection
Set/get a general request property.

setResizable(boolean). Method in class java.awt.Dialog
Sets the resizable flag.

setResizable(boolean). Method in interface java.awt.peer.DialogPeer

setResizable(boolean). Method in class java.awt.Frame
Sets the resizable flag.

setResizable(boolean). Method in interface java.awt.peer.FramePeer

setSeconds(int). Method in class java.util.Date
Sets the seconds.

setSecurityManager(SecurityManager). Static method in class java.lang.System
Sets the System security.

setSeed(long). Method in class java.util.Random
Sets the seed of the random number generator using a single `long` seed.

setSize(int). Method in class java.util.Vector
Sets the size of the vector.

setSocketFactory(SocketImplFactory). Static method in class java.net.ServerSocket
Sets the system's server SocketImplFactory.

setSocketImplFactory(SocketImplFactory). Static method in class java.net.Socket
Sets the system's client SocketImplFactory.

setSourcePath(String). Method in class sun.tools.debug.RemoteDebugger
Specify the list of paths to use when searching for a source file.

setState(boolean). Method in class java.awt.Checkbox
Sets the Checkbox to the specifed boolean state.

setState(boolean). Method in class java.awt.CheckboxMenuItem
Sets the state of this MenuItem if it is a Checkbox.

setState(boolean). Method in interface java.awt.peer.CheckboxMenuItemPeer

setState(boolean). Method in interface java.awt.peer.CheckboxPeer

setStub(AppletStub). Method in class java.applet.Applet
Set the applet stub.

setText(String). Method in class java.awt.Label
Sets the text for this label to the specified text.

setText(String). Method in interface java.awt.peer.LabelPeer

setText(String). Method in class java.awt.TextComponent
Sets the text of this TextComponent to the specified text.

setText(String). Method in interface java.awt.peer.TextComponentPeer

setTime(long). Method in class java.util.Date

Sets the time.
setTitle(String). Method in class java.awt.Dialog
Sets the title of the Dialog.
setTitle(String). Method in interface java.awt.peer.DialogPeer
setTitle(String). Method in class java.awt.Frame
Sets the title for this Frame to the specified title.
setTitle(String). Method in interface java.awt.peer.FramePeer
setURL(URL, String, String, int, String, String). Method in class java.net.URLStreamHandler
Calls the (protected) set method out of the URL given.
setURLStreamHandlerFactory(URLStreamHandlerFactory). Static method in class java.net.URL
Sets the URLStreamHandler factory.
setUseCaches(boolean). Method in class java.net.URLConnection
Some protocols do caching of documents.
setValue(int). Method in class java.awt.Scrollbar
Sets the value of this Scrollbar to the specified value.
setValue(int). Method in interface java.awt.peer.ScrollbarPeer
setValues(int, int, int, int). Method in class java.awt.Scrollbar
Sets the values for this Scrollbar.
setValues(int, int, int, int). Method in interface java.awt.peer.ScrollbarPeer
setXORMode(Color). Method in class java.awt.Graphics
Sets the paint mode to alternate between the current color and the new specified color.
setYear(int). Method in class java.util.Date
Sets the year.
SHIFT_MASK. Static variable in class java.awt.Event
The shift modifier constant.
shiftDown(). Method in class java.awt.Event
Checks if the shift key is down.
show(). Method in class java.awt.Component
Shows the component.
show(). Method in interface java.awt.peer.ComponentPeer
show(). Method in class java.awt.Window
Shows the Window.
show(boolean). Method in class java.awt.Component
Conditionally shows the component.
show(Container, String). Method in class java.awt.CardLayout
Flips to the specified component name in the specified container.
showDocument(URL). Method in interface java.applet.AppletContext
Show a new document.
showDocument(URL, String). Method in interface java.applet.AppletContext
Show a new document in a target window or frame.
showStatus(String). Method in class java.applet.Applet
Show a status message in the Applet's context.
showStatus(String). Method in interface java.applet.AppletContext
Show a status string.
sin(double). Static method in class java.lang.Math
Returns the trigonometric sine of an angle.
SINGLEFRAME. Static variable in interface java.awt.image.ImageConsumer
The image contain a single static image.
SINGLEFRAMEDONE. Static variable in interface java.awt.image.ImageConsumer

One frame of the image is complete but there are more frames to be delivered.
SINGLEPASS. Static variable in interface java.awt.image.ImageConsumer
The pixels will be delivered in a single pass.
size. Variable in class java.awt.Font
The point size of this font.
size(). Method in class java.util.BitSet
Calculates and returns the set's size
size(). Method in class java.io.ByteArrayOutputStream
Returns the current size of the buffer.
size(). Method in class java.awt.Component
Returns the current size of this component.
size(). Method in class java.io.DataOutputStream
Returns the number of bytes written.
size(). Method in class java.util.Dictionary
Returns the number of elements contained within the Dictionary.
size(). Method in class java.util.Hashtable
Returns the number of elements contained in the hashtable.
size(). Method in class java.util.Vector
Returns the number of elements in the vector.
skip(long). Method in class java.io.BufferedInputStream
Skips n bytes of input.
skip(long). Method in class java.io.ByteArrayInputStream
Skips n bytes of input.
skip(long). Method in class java.io.FileInputStream
Skips n bytes of input.
skip(long). Method in class java.io.FilterInputStream
Skips bytes of input.
skip(long). Method in class java.io.InputStream
Skips n bytes of input.
skip(long). Method in class java.io.LineNumberInputStream
Skips n bytes of input.
skip(long). Method in class java.io.StringBufferInputStream
Skips n bytes of input.
skipBytes(int). Method in interface java.io.DataInput
Skips bytes, block until all bytes are skipped.
skipBytes(int). Method in class java.io.DataInputStream
Skips bytes, block until all bytes are skipped.
skipBytes(int). Method in class java.io.RandomAccessFile
slashSlashComments(boolean). Method in class java.io.StreamTokenizer
If the flag is true, recognize C++ style(//) comments.
slashStarComments(boolean). Method in class java.io.StreamTokenizer
If the flag is true, recognize C style(/*) comments.
sleep(long). Static method in class java.lang.Thread
Causes the currently executing Thread to sleep for the specified number of milliseconds.
sleep(long, int). Static method in class java.lang.Thread
Sleep, in milliseconds and additional nanosecond.
Socket(InetAddress, int). Constructor for class java.net.Socket
Creates a stream socket and connects it to the specified address on the specified port.
Socket(InetAddress, int, boolean). Constructor for class java.net.Socket

Creates a socket and connects it to the specified address on the specified port.
Socket(String, int). Constructor for class java.net.Socket
Creates a stream socket and connects it to the specified port on the specified host.
Socket(String, int, boolean). Constructor for class java.net.Socket
Creates a socket and connects it to the specified port on the specified host.
SocketException(). Constructor for class java.net.SocketException
Constructs a new SocketException with no detail message.
SocketException(String). Constructor for class java.net.SocketException
Constructs a new SocketException with the specified detail message.
SocketImpl(). Constructor for class java.net.SocketImpl
SOMEBITS. Static variable in interface java.awt.image.ImageObserver
More pixels needed for drawing a scaled variation of the image are available.
SOUTH. Static variable in class java.awt.GridBagConstraints
SOUTHEAST. Static variable in class java.awt.GridBagConstraints
SOUTHWEST. Static variable in class java.awt.GridBagConstraints
sqrt(double). Static method in class java.lang.Math
Returns the square root of a.
Stack(). Constructor for class java.util.Stack
StackFrame(). Constructor for class sun.tools.debug.StackFrame
StackOverflowError(). Constructor for class java.lang.StackOverflowError
Constructs a StackOverflowError with no detail message.
StackOverflowError(String). Constructor for class java.lang.StackOverflowError
Constructs a StackOverflowError with the specified detail message.
start(). Method in class java.applet.Applet
Called to start the applet.
start(). Method in class java.lang.Thread
Starts this Thread.
startProduction(ImageConsumer). Method in class java.awt.image.FilteredImageSource
Add an ImageConsumer to the list of consumers interested in data for this image, and immediately start delivery of the image data through the ImageConsumer interface.
startProduction(ImageConsumer). Method in interface java.awt.image.ImageProducer
This method both registers the given ImageConsumer object as a consumer and starts an immediate reconstruction of the image data which will then be delivered to this consumer and any other consumer which may have already been registered with the producer.
startProduction(ImageConsumer). Method in class java.awt.image.MemoryImageSource
Add an ImageConsumer to the list of consumers interested in data for this image, and immediately start delivery of the image data through the ImageConsumer interface.
startsWith(String). Method in class java.lang.String
Determines whether this String starts with some prefix.
startsWith(String, int). Method in class java.lang.String
Determines whether this String starts with some prefix.
STATICIMAGEDONE. Static variable in interface java.awt.image.ImageConsumer
The image is complete and there are no more pixels or frames to be delivered.
status(). Method in class java.awt.image.PixelGrabber
Return the status of the pixels.
statusAll(boolean). Method in class java.awt.MediaTracker
Return the boolean OR of the status of all of the media being tracked.
statusID(int, boolean). Method in class java.awt.MediaTracker
Return the boolean OR of the status of all of the media with a given ID.

step(boolean). Method in class sun.tools.debug.RemoteThread
Continue execution of this thread to the next instruction or line.
stop(). Method in class java.applet.Applet
Called to stop the applet.
stop(). Method in interface java.applet.AudioClip
Stop playing the clip.
stop(). Method in class sun.tools.debug.RemoteThread
Stop the remote thread.
stop(). Method in class sun.tools.debug.RemoteThreadGroup
Stop the remote threadgroup.
stop(). Method in class java.lang.Thread
Stops a Thread by tossing an object.
stop(). Method in class java.lang.ThreadGroup
Stops all the Threads in this Thread group and all of its sub groups.
stop(Throwable). Method in class java.lang.Thread
Stops a Thread by tossing an object.
StreamTokenizer(InputStream). Constructor for class java.io.StreamTokenizer
Creates a stream tokenizer that parses the specified input stream.
String(). Constructor for class java.lang.String
Constructs a new empty String.
String(byte[], int). Constructor for class java.lang.String
Constructs a new String whose value is the specified array of bytes.
String(byte[], int, int, int). Constructor for class java.lang.String
Constructs a new String whose initial value is the specified sub array of bytes.
String(char[]). Constructor for class java.lang.String
Constructs a new String whose initial value is the specified array of characters.
String(char[], int, int). Constructor for class java.lang.String
Constructs a new String whose initial value is the specified sub array of characters.
String(String). Constructor for class java.lang.String
Constructs a new String that is a copy of the specified String.
String(StringBuffer). Constructor for class java.lang.String
Construct a new string whose value is the current contents of the given string buffer
StringBuffer(). Constructor for class java.lang.StringBuffer
Constructs an empty String buffer.
StringBuffer(int). Constructor for class java.lang.StringBuffer
Constructs an empty String buffer with the specified initial length.
StringBuffer(String). Constructor for class java.lang.StringBuffer
Constructs a String buffer with the specified initial value.
StringBufferInputStream(String). Constructor for class java.io.StringBufferInputStream
Creates an StringBufferInputStream from the specified array of bytes.
StringIndexOutOfBoundsException(). Constructor for class java.lang.StringIndexOutOfBoundsException
Constructs a StringIndexOutOfBoundsException with no detail message.
StringIndexOutOfBoundsException(int). Constructor for class java.lang.StringIndexOutOfBoundsException
Constructs a StringIndexOutOfBoundsException initialized with the specified index.
StringIndexOutOfBoundsException(String). Constructor for class java.lang.StringIndexOutOfBoundsException
Constructs a StringIndexOutOfBoundsException with the specified detail message.

StringTokenizer(String). Constructor for class java.util.StringTokenizer
Constructs a StringTokenizer on the specified String, using the default delimiter set (which is " \t\n\r").
StringTokenizer(String, String). Constructor for class java.util.StringTokenizer
Constructs a StringTokenizer on the specified String, using the specified delimiter set.
StringTokenizer(String, String, boolean). Constructor for class java.util.StringTokenizer
Constructs a StringTokenizer on the specified String, using the specified delimiter set.
stringWidth(String). Method in class java.awt.FontMetrics
Returns the width of the specified String in this Font.
style. Variable in class java.awt.Font
The style of the font.
substituteColorModel(ColorModel, ColorModel). Method in class java.awt.image.RGBImageFilter
Register two ColorModel objects for substitution.
substring(int). Method in class java.lang.String
Returns the substring of this String.
substring(int, int). Method in class java.lang.String
Returns the substring of a String.
suspend(). Method in class sun.tools.debug.RemoteThread
Suspend execution of this thread.
suspend(). Method in class java.lang.Thread
Suspends this Thread's execution.
suspend(). Method in class java.lang.ThreadGroup
Suspends all the Threads in this Thread group and all of its sub groups.
sval. Variable in class java.io.StreamTokenizer
The Stream value.
SW_RESIZE_CURSOR. Static variable in class java.awt.Frame
sync(). Method in class java.awt.Toolkit
Syncs the graphics state, which is useful when doing animation.

T

tan(double). Static method in class java.lang.Math
Returns the trigonometric tangent of an angle.
target. Variable in class java.awt.Event
The target component.
TEXT_CURSOR. Static variable in class java.awt.Frame
TextArea(). Constructor for class java.awt.TextArea
Constructs a new TextArea.
TextArea(int, int). Constructor for class java.awt.TextArea
Constructs a new TextArea with the specified number of rows and columns.
TextArea(String). Constructor for class java.awt.TextArea
Constructs a new TextArea with the specified text displayed.
TextArea(String, int, int). Constructor for class java.awt.TextArea
Constructs a new TextArea with the specified text and the specified number of rows and columns.
TextField(). Constructor for class java.awt.TextField

Constructs a new TextField.
TextField(int). Constructor for class java.awt.TextField
Constructs a new TextField initialized with the specified columns.
TextField(String). Constructor for class java.awt.TextField
Constructs a new TextField initialized with the specified text.
TextField(String, int). Constructor for class java.awt.TextField
Constructs a new TextField initialized with the specified text and columns.
Thread(). Constructor for class java.lang.Thread
Constructs a new Thread.
Thread(Runnable). Constructor for class java.lang.Thread
Constructs a new Thread which applies the run() method of the specified target.
Thread(Runnable, String). Constructor for class java.lang.Thread
Constructs a new Thread with the specified name and applies the run() method of the specified target.
Thread(String). Constructor for class java.lang.Thread
Constructs a new Thread with the specified name.
Thread(ThreadGroup, Runnable). Constructor for class java.lang.Thread
Constructs a new Thread in the specified Thread group that applies the run() method of the specified target.
Thread(ThreadGroup, Runnable, String). Constructor for class java.lang.Thread
Constructs a new Thread in the specified Thread group with the specified name and applies the run() method of the specified target.
Thread(ThreadGroup, String). Constructor for class java.lang.Thread
Constructs a new Thread in the specified Thread group with the specified name.
ThreadDeath(). Constructor for class java.lang.ThreadDeath
threadDeathEvent(RemoteThread). Method in interface sun.tools.debug.DebuggerCallback
A thread has died.
ThreadGroup(String). Constructor for class java.lang.ThreadGroup
Creates a new ThreadGroup.
ThreadGroup(ThreadGroup, String). Constructor for class java.lang.ThreadGroup
Creates a new ThreadGroup with a specified name in the specified Thread group.
Throwable(). Constructor for class java.lang.Throwable
Constructs a new Throwable with no detail message.
Throwable(String). Constructor for class java.lang.Throwable
Constructs a new Throwable with the specified detail message.
toBack(). Method in class java.awt.Window
Sends the frame to the back of the Window.
toBack(). Method in interface java.awt.peer.WindowPeer
toByteArray(). Method in class java.io.ByteArrayOutputStream
Returns a copy of the input data.
toCharArray(). Method in class java.lang.String
Converts this String to a character array.
toExternalForm(). Method in class java.net.URL
Reverses the parsing of the URL.
toExternalForm(URL). Method in class java.net.URLStreamHandler
Reverses the parsing of the URL.
toFront(). Method in class java.awt.Window
Brings the frame to the front of the Window.
toFront(). Method in interface java.awt.peer.WindowPeer

toGMTString(). Method in class java.util.Date
Converts a date to a String, using the Internet GMT conventions.
toHex(int). Static method in class sun.tools.debug.RemoteValue
Convert an int to a hexadecimal string.
toLocaleString(). Method in class java.util.Date
Converts a date to a String, using the locale conventions.
toLowerCase(). Method in class java.lang.String
Converts all of the characters in this String to lower case.
toLowerCase(char). Static method in class java.lang.Character
Returns the lower case character value of the specified ISO−LATIN−1 character.
Toolkit(). Constructor for class java.awt.Toolkit
top. Variable in class java.awt.Insets
The inset from the top.
TOPDOWNLEFTRIGHT. Static variable in interface java.awt.image.ImageConsumer
The pixels will be delivered in top−down, left−to−right order.
toString(). Method in class java.util.BitSet
Converts the BitSet to a String.
toString(). Method in class java.lang.Boolean
Returns a new String object representing this Boolean's value.
toString(). Method in class java.awt.BorderLayout
Returns the String representation of this BorderLayout's values.
toString(). Method in class java.io.ByteArrayOutputStream
Converts input data to a string.
toString(). Method in class java.awt.CardLayout
Returns the String representation of this CardLayout's values.
toString(). Method in class java.lang.Character
Returns a String object representing this character's value.
toString(). Method in class java.awt.CheckboxGroup
Returns the String representation of this CheckboxGroup's values.
toString(). Method in class java.lang.Class
Returns the name of this Class or this interface.
toString(). Method in class java.awt.Color
Returns the String representation of this Color's values.
toString(). Method in class java.awt.Component
Returns the String representation of this Component's values.
toString(). Method in class java.util.Date
Converts a date to a String, using the UNIX ctime conventions.
toString(). Method in class java.awt.Dimension
Returns the String representation of this Dimension's values.
toString(). Method in class java.lang.Double
Returns a String representation of this Double object.
toString(). Method in class java.awt.Event
Returns the String representation of this Event's values.
toString(). Method in class java.io.File
Returns a String object representing this file's path.
toString(). Method in class java.lang.Float
Returns a String representation of this Float object.
toString(). Method in class java.awt.FlowLayout
Returns the String representation of this FlowLayout's values.

toString(). Method in class java.awt.Font
    Converts this object to a String representation.
toString(). Method in class java.awt.FontMetrics
    Returns the String representation of this FontMetric's values.
toString(). Method in class java.awt.Graphics
    Returns a String object representing this Graphic's value.
toString(). Method in class java.awt.GridBagLayout
    Returns the String representation of this GridLayout's values.
toString(). Method in class java.awt.GridLayout
    Returns the String representation of this GridLayout's values.
toString(). Method in class java.util.Hashtable
    Converts to a rather lengthy String.
toString(). Method in class java.net.InetAddress
    Converts the InetAddress to a String.
toString(). Method in class java.awt.Insets
    Returns a String object representing this Inset's values.
toString(). Method in class java.lang.Integer
    Returns a String object representing this Integer's value.
toString(). Method in class java.lang.Long
    Returns a String object representing this Long's value.
toString(). Method in class java.awt.MenuComponent
    Returns the String representation of this MenuComponent's values.
toString(). Method in class java.lang.Object
    Returns a String that represents the value of this Object.
toString(). Method in class java.awt.Point
    Returns the String representation of this Point's coordinates.
toString(). Method in class java.awt.Rectangle
    Returns the String representation of this Rectangle's values.
toString(). Method in class sun.tools.debug.RemoteArray
    Return a string version of the array.
toString(). Method in class sun.tools.debug.RemoteBoolean
    Return the boolean's value as a string.
toString(). Method in class sun.tools.debug.RemoteByte
    Return the byte's value as a string.
toString(). Method in class sun.tools.debug.RemoteChar
    Return the char's value as a string.
toString(). Method in class sun.tools.debug.RemoteClass
    Return a (somewhat verbose) description.
toString(). Method in class sun.tools.debug.RemoteDouble
    Return the double's value as a string.
toString(). Method in class sun.tools.debug.RemoteField
    Returns a String that represents the value of this Object.
toString(). Method in class sun.tools.debug.RemoteFloat
    Return the float's value as a string.
toString(). Method in class sun.tools.debug.RemoteInt
    Return the int's value as a string.
toString(). Method in class sun.tools.debug.RemoteLong
    Return the long's value as a string.
toString(). Method in class sun.tools.debug.RemoteObject

Return object as a string.
toString(). Method in class sun.tools.debug.RemoteShort
Return the short's value as a string.
toString(). Method in class sun.tools.debug.RemoteString
Return the string value, or "null"
toString(). Method in class java.net.ServerSocket
Returns the implementation address and implementation port of this ServerSocket as a String.
toString(). Method in class java.net.Socket
Converts the Socket to a String.
toString(). Method in class java.net.SocketImpl
Returns the address and port of this Socket as a String.
toString(). Method in class sun.tools.debug.StackFrame
Returns a String that represents the value of this Object.
toString(). Method in class java.io.StreamTokenizer
Returns the String representation of the stream token.
toString(). Method in class java.lang.String
Converts this String to a String.
toString(). Method in class java.lang.StringBuffer
Converts to a String representing the data in the buffer.
toString(). Method in class java.lang.Thread
Returns a String representation of the Thread, including the thread's name, priority and thread group.
toString(). Method in class java.lang.ThreadGroup
Returns a String representation of the Thread group.
toString(). Method in class java.lang.Throwable
Returns a short description of the Throwable.
toString(). Method in class java.net.URL
Converts to a human−readable form.
toString(). Method in class java.net.URLConnection
Returns the String representation of the URL connection.
toString(). Method in class java.util.Vector
Converts the vector to a string.
toString(double). Static method in class java.lang.Double
Returns a String representation for the specified double value.
toString(float). Static method in class java.lang.Float
Returns a String representation for the specified float value.
toString(int). Method in class java.io.ByteArrayOutputStream
Converts input data to a string.
toString(int). Static method in class java.lang.Integer
Returns a new String object representing the specified integer.
toString(int, int). Static method in class java.lang.Integer
Returns a new String object representing the specified integer in the specified radix.
toString(long). Static method in class java.lang.Long
Returns a new String object representing the specified integer.
toString(long, int). Static method in class java.lang.Long
Returns a new String object representing the specified long in the specified radix.
totalMemory(). Method in class sun.tools.debug.RemoteDebugger
Report the total memory usage of the Java interpreter being debugged.
totalMemory(). Method in class java.lang.Runtime

Returns the total number of bytes in system memory.
toUpperCase(). Method in class java.lang.String
Converts all of the characters in this String to upper case.
toUpperCase(char). Static method in class java.lang.Character
Returns the upper case character value of the specified ISO−LATIN−1 character.
trace(boolean). Method in class sun.tools.debug.RemoteDebugger
Turn on/off method call tracing.
traceInstructions(boolean). Method in class java.lang.Runtime
Enables/Disables tracing of instructions.
traceMethodCalls(boolean). Method in class java.lang.Runtime
Enables/Disables tracing of method calls.
translate(int, int). Method in class java.awt.Event
Translates an event relative to the given component.
translate(int, int). Method in class java.awt.Graphics
Translates the specified parameters into the origin of the graphics context.
translate(int, int). Method in class java.awt.Point
Translate the point.
translate(int, int). Method in class java.awt.Rectangle
Translate the rectangle.
trim(). Method in class java.lang.String
Trims leading and trailing whitespace from this String.
trimToSize(). Method in class java.util.Vector
Trims the vector's capacity down to size.
TRUE. Static variable in class java.lang.Boolean
Assigns this Boolean to be true.
TT_EOF. Static variable in class java.io.StreamTokenizer
The End−of−file token.
TT_EOL. Static variable in class java.io.StreamTokenizer
The End−of−line token.
TT_NUMBER. Static variable in class java.io.StreamTokenizer
The number token.
TT_WORD. Static variable in class java.io.StreamTokenizer
The word token.
ttype. Variable in class java.io.StreamTokenizer
The type of the last token returned.
typeName(). Method in class sun.tools.debug.RemoteArray
Return this RemoteValue's type ("array").
typeName(). Method in class sun.tools.debug.RemoteBoolean
Print this RemoteValue's type ("boolean").
typeName(). Method in class sun.tools.debug.RemoteByte
Print this RemoteValue's type ("byte").
typeName(). Method in class sun.tools.debug.RemoteChar
Print this RemoteValue's type ("char").
typeName(). Method in class sun.tools.debug.RemoteClass
Returns the name of the class as its type.
typeName(). Method in class sun.tools.debug.RemoteDouble
Print this RemoteValue's type ("double").
typeName(). Method in class sun.tools.debug.RemoteFloat
Print this RemoteValue's type ("float").

typeName(). Method in class sun.tools.debug.RemoteInt
Print this RemoteValue's type ("int").
typeName(). Method in class sun.tools.debug.RemoteLong
Print this RemoteValue's type ("long").
typeName(). Method in class sun.tools.debug.RemoteObject
Returns the RemoteValue's type name ("Object").
typeName(). Method in class sun.tools.debug.RemoteShort
Print this RemoteValue's type ("short").
typeName(). Method in class sun.tools.debug.RemoteString
Print this RemoteValue's type ("String").
typeName(). Method in class sun.tools.debug.RemoteValue
Returns the RemoteValue's type as a string.

U

uncaughtException(Thread, Throwable). Method in class java.lang.ThreadGroup
Called when a thread in this group exists because of an uncaught exception.
UndefinedProperty. Static variable in class java.awt.Image
The UndefinedProperty object should be returned whenever a property which was not defined for a particular image is fetched.
union(Rectangle). Method in class java.awt.Rectangle
Compute the union of two rectangles.
UnknownError(). Constructor for class java.lang.UnknownError
Constructs an UnknownError with no detail message.
UnknownError(String). Constructor for class java.lang.UnknownError
Constructs an UnknownError with the specified detail message.
UnknownHostException(). Constructor for class java.net.UnknownHostException
Constructs a new UnknownHostException with no detail message.
UnknownHostException(String). Constructor for class java.net.UnknownHostException
Constructs a new UnknownHostException with the specified detail message.
UnknownServiceException(). Constructor for class java.net.UnknownServiceException
Constructs a new UnknownServiceException with no detail message.
UnknownServiceException(String). Constructor for class java.net.UnknownServiceException
Constructs a new UnknownServiceException with the specified detail message.
unread(int). Method in class java.io.PushbackInputStream
Pushes back a character.
UnsatisfiedLinkError(). Constructor for class java.lang.UnsatisfiedLinkError
Constructs an UnsatisfiedLinkError with no detail message.
UnsatisfiedLinkError(String). Constructor for class java.lang.UnsatisfiedLinkError
Constructs an UnsatisfiedLinkError with the specified detail message.
UP. Static variable in class java.awt.Event
The up arrow key.
up(int). Method in class sun.tools.debug.RemoteThread
Change the current stackframe to be one or more frames higher (as in, away from the current program counter).
update(Graphics). Method in class java.awt.Component

Updates the component.
update(Observable, Object). Method in interface java.util.Observer
This is called if observers in the observable list need to be updated.
url. Variable in class java.net.URLConnection
URL(String). Constructor for class java.net.URL
Creates a URL from the unparsed absolute URL.
URL(String, String, int, String). Constructor for class java.net.URL
Creates an absolute URL from the specified protocol, host, port and file.
URL(String, String, String). Constructor for class java.net.URL
Creates an absolute URL from the specified protocol, host, and file.
URL(URL, String). Constructor for class java.net.URL
Creates a URL from the unparsed URL in the context of the specified context.
URLConnection(URL). Constructor for class java.net.URLConnection
Constructs a URL connection to the specified URL.
URLStreamHandler(). Constructor for class java.net.URLStreamHandler
useCaches. Variable in class java.net.URLConnection
UTC(int, int, int, int, int, int). Static method in class java.util.Date
Calculates a UTC value from YMDHMS.
UTFDataFormatException(). Constructor for class java.io.UTFDataFormatException
Constructs an UTFDataFormatException with no detail message.
UTFDataFormatException(String). Constructor for class java.io.UTFDataFormatException
Constructs an UTFDataFormatException with the specified detail message.

V

valid(). Method in class java.io.FileDescriptor
This routine tells us if the file descriptor object is valid.
validate(). Method in class java.awt.Component
Validates a component.
validate(). Method in class java.awt.Container
Validates this Container and all of the components contained within it.
valueOf(boolean). Static method in class java.lang.String
Returns a String object that represents the state of the specified boolean.
valueOf(char). Static method in class java.lang.String
Returns a String object that contains a single character
valueOf(char[]). Static method in class java.lang.String
Returns a String that is equivalent to the specified character array.
valueOf(char[], int, int). Static method in class java.lang.String
Returns a String that is equivalent to the specified character array.
valueOf(double). Static method in class java.lang.String
Returns a String object that represents the value of the specified double.
valueOf(float). Static method in class java.lang.String
Returns a String object that represents the value of the specified float.
valueOf(int). Static method in class java.lang.String
Returns a String object that represents the value of the specified integer.
valueOf(long). Static method in class java.lang.String

Returns a String object that represents the value of the specified long.
valueOf(Object). Static method in class java.lang.String
Returns a String that represents the String value of the object.
valueOf(String). Static method in class java.lang.Boolean
Returns the boolean value represented by the specified String.
valueOf(String). Static method in class java.lang.Double
Returns a new Double value initialized to the value represented by the specified String.
valueOf(String). Static method in class java.lang.Float
Returns the floating point value represented by the specified String.
valueOf(String). Static method in class java.lang.Integer
Assuming the specified String represents an integer, returns a new Integer object initialized to that value.
valueOf(String). Static method in class java.lang.Long
Assuming the specified String represents a long, returns a new Long object initialized to that value.
valueOf(String, int). Static method in class java.lang.Integer
Assuming the specified String represents an integer, returns a new Integer object initialized to that value.
valueOf(String, int). Static method in class java.lang.Long
Assuming the specified String represents a long, returns a new Long object initialized to that value.
Vector(). Constructor for class java.util.Vector
Constructs an empty vector.
Vector(int). Constructor for class java.util.Vector
Constructs an empty vector with the specified storage capacity.
Vector(int, int). Constructor for class java.util.Vector
Constructs an empty vector with the specified storage capacity and the specified capacityIncrement.
VerifyError(). Constructor for class java.lang.VerifyError
Constructor.
VerifyError(String). Constructor for class java.lang.VerifyError
Constructor with a detail message.
VERTICAL. Static variable in class java.awt.GridBagConstraints
VERTICAL. Static variable in class java.awt.Scrollbar
The vertical Scrollbar variable.
VirtualMachineError(). Constructor for class java.lang.VirtualMachineError
Constructs a VirtualMachineError with no detail message.
VirtualMachineError(String). Constructor for class java.lang.VirtualMachineError
Constructs a VirtualMachineError with the specified detail message.

W

W_RESIZE_CURSOR. Static variable in class java.awt.Frame
wait(). Method in class java.lang.Object
Causes a thread to wait forever until it is notified.
wait(long). Method in class java.lang.Object

Causes a thread to wait until it is notified or the specified timeout expires.
wait(long, int). Method in class java.lang.Object
More accurate wait.
WAIT_CURSOR. Static variable in class java.awt.Frame
waitFor(). Method in class java.lang.Process
Waits for the subprocess to complete.
waitForAll(). Method in class java.awt.MediaTracker
Start loading all images and wait until they have finished loading, are aborted, or receive an error.
waitForAll(long). Method in class java.awt.MediaTracker
Start loading all images and wait until they have finished loading, are aborted, receive an error, or until the specified timeout has elapsed.
waitForID(int). Method in class java.awt.MediaTracker
Start loading all images with the specified ID and wait until they have finished loading or receive an error.
waitForID(int, long). Method in class java.awt.MediaTracker
Start loading all images with the specified ID and wait until they have finished loading or receive an error, or until the specified timeout has elapsed.
weightx. Variable in class java.awt.GridBagConstraints
weighty. Variable in class java.awt.GridBagConstraints
WEST. Static variable in class java.awt.GridBagConstraints
when. Variable in class java.awt.Event
The time stamp.
white. Static variable in class java.awt.Color
The color white.
whitespaceChars(int, int). Method in class java.io.StreamTokenizer
Specifies that characters in this range are whitespace characters.
width. Variable in class java.awt.Dimension
The width dimension.
WIDTH. Static variable in interface java.awt.image.ImageObserver
The width of the base image is now available and can be taken from the width argument to the imageUpdate callback method.
width. Variable in class java.awt.Rectangle
The width of the rectangle.
Window(Frame). Constructor for class java.awt.Window
Constructs a new Window initialized to an invisible state.
WINDOW_DEICONIFY. Static variable in class java.awt.Event
The de−iconify window event.
WINDOW_DESTROY. Static variable in class java.awt.Event
The destroy window event.
WINDOW_EXPOSE. Static variable in class java.awt.Event
The expose window event.
WINDOW_ICONIFY. Static variable in class java.awt.Event
The iconify window event.
WINDOW_MOVED. Static variable in class java.awt.Event
The move window event.
wordChars(int, int). Method in class java.io.StreamTokenizer
Specifies that characters in this range are word characters.
write(byte[]). Method in interface java.io.DataOutput

Writes an array of bytes.
write(byte[]). Method in class java.io.FileOutputStream
Writes an array of bytes.
write(byte[]). Method in class java.io.FilterOutputStream
Writes an array of bytes.
write(byte[]). Method in class java.io.OutputStream
Writes an array of bytes.
write(byte[]). Method in class java.io.RandomAccessFile
Writes an array of bytes.
write(byte[], int, int). Method in class java.io.BufferedOutputStream
Writes a sub array of bytes.
write(byte[], int, int). Method in class java.io.ByteArrayOutputStream
Writes bytes to the buffer.
write(byte[], int, int). Method in interface java.io.DataOutput
Writes a sub array of bytes.
write(byte[], int, int). Method in class java.io.DataOutputStream
Writes a sub array of bytes.
write(byte[], int, int). Method in class java.io.FileOutputStream
Writes a sub array of bytes.
write(byte[], int, int). Method in class java.io.FilterOutputStream
Writes a sub array of bytes.
write(byte[], int, int). Method in class java.io.OutputStream
Writes a sub array of bytes.
write(byte[], int, int). Method in class java.io.PipedOutputStream
Writes a sub array of bytes.
write(byte[], int, int). Method in class java.io.PrintStream
Writes a sub array of bytes.
write(byte[], int, int). Method in class java.io.RandomAccessFile
Wrotes a sub array of bytes.
write(int). Method in class java.io.BufferedOutputStream
Writes a byte.
write(int). Method in class java.io.ByteArrayOutputStream
Writes a byte to the buffer.
write(int). Method in interface java.io.DataOutput
Writes a byte.
write(int). Method in class java.io.DataOutputStream
Writes a byte.
write(int). Method in class java.io.FileOutputStream
Writes a byte of data.
write(int). Method in class java.io.FilterOutputStream
Writes a byte.
write(int). Method in class java.io.OutputStream
Writes a byte.
write(int). Method in class java.io.PipedOutputStream
Write a byte.
write(int). Method in class java.io.PrintStream
Writes a byte.
write(int). Method in class java.io.RandomAccessFile
Writes a byte of data.

writeBoolean(boolean). Method in interface java.io.DataOutput
Writes a boolean.
writeBoolean(boolean). Method in class java.io.DataOutputStream
Writes a boolean.
writeBoolean(boolean). Method in class java.io.RandomAccessFile
Writes a boolean.
writeByte(int). Method in interface java.io.DataOutput
Writes an 8 bit byte.
writeByte(int). Method in class java.io.DataOutputStream
Writes an 8 bit byte.
writeByte(int). Method in class java.io.RandomAccessFile
Writes a byte.
writeBytes(String). Method in interface java.io.DataOutput
Writes a String as a sequence of bytes.
writeBytes(String). Method in class java.io.DataOutputStream
Writes a String as a sequence of bytes.
writeBytes(String). Method in class java.io.RandomAccessFile
Writes a String as a sequence of bytes.
writeChar(int). Method in interface java.io.DataOutput
Writes a 16 bit char.
writeChar(int). Method in class java.io.DataOutputStream
Writes a 16 bit char.
writeChar(int). Method in class java.io.RandomAccessFile
Writes a character.
writeChars(String). Method in interface java.io.DataOutput
Writes a String as a sequence of chars.
writeChars(String). Method in class java.io.DataOutputStream
Writes a String as a sequence of chars.
writeChars(String). Method in class java.io.RandomAccessFile
Writes a String as a sequence of chars.
writeDouble(double). Method in interface java.io.DataOutput
Writes a 64 bit double.
writeDouble(double). Method in class java.io.DataOutputStream
Writes a 64 bit double.
writeDouble(double). Method in class java.io.RandomAccessFile
writeFloat(float). Method in interface java.io.DataOutput
Writes a 32 bit float.
writeFloat(float). Method in class java.io.DataOutputStream
Writes a 32 bit float.
writeFloat(float). Method in class java.io.RandomAccessFile
writeInt(int). Method in interface java.io.DataOutput
Writes a 32 bit int.
writeInt(int). Method in class java.io.DataOutputStream
Writes a 32 bit int.
writeInt(int). Method in class java.io.RandomAccessFile
Writes an integer.
writeLong(long). Method in interface java.io.DataOutput
Writes a 64 bit long.
writeLong(long). Method in class java.io.DataOutputStream

Writes a 64 bit long.
writeLong(long). Method in class java.io.RandomAccessFile
Writes a long.
writeShort(int). Method in interface java.io.DataOutput
Writes a 16 bit short.
writeShort(int). Method in class java.io.DataOutputStream
Writes a 16 bit short.
writeShort(int). Method in class java.io.RandomAccessFile
Writes a short.
writeTo(OutputStream). Method in class java.io.ByteArrayOutputStream
Writes the contents of the buffer to another stream.
writeUTF(String). Method in interface java.io.DataOutput
Writes a String in UTF format.
writeUTF(String). Method in class java.io.DataOutputStream
Writes a String in UTF format.
writeUTF(String). Method in class java.io.RandomAccessFile
Writes a String in UTF format.
written. Variable in class java.io.DataOutputStream
The number of bytes written so far.

X

x. Variable in class java.awt.Event
The x coordinate of the event.
x. Variable in class java.awt.Point
The x coordinate.
x. Variable in class java.awt.Rectangle
The x coordinate of the rectangle.
xor(BitSet). Method in class java.util.BitSet
Logically XORs this bit set with the specified set of bits.
xpoints. Variable in class java.awt.Polygon
The array of x coordinates.

Y

y. Variable in class java.awt.Event
The y coordinate of the event.
y. Variable in class java.awt.Point
The y coordinate.
y. Variable in class java.awt.Rectangle
The y coordinate of the rectangle.
yellow. Static variable in class java.awt.Color
The color yellow.

yield(). Static method in class java.lang.Thread

Causes the currently executing Thread object to yield.

ypoints. Variable in class java.awt.Polygon

The array of y coordinates.