

## Class GTerm

```
java.lang.Object
  java.awt.Component
    java.awt.Container
      java.awt.Window
        java.awt.Frame
          javax.swing.JFrame
            GTerm
```

### All Implemented Interfaces:

```
java.awt.event.KeyListener, java.awt.event.MouseListener,
java.awt.image.ImageObserver, java.awt.MenuContainer, java.io.Serializable,
java.util.EventListener, javax.accessibility.Accessible,
javax.swing.RootPaneContainer, javax.swing.WindowConstants
```

---

```
public final class GTerm
extends javax.swing.JFrame
implements java.awt.event.KeyListener, java.awt.event.MouseListener
```

### Version:

2020.08.25

GTerm allows you to create a GUI application using System.out.println-like statements.

### Version History:

2020.02.29: Original version

2020.03.06: Added \n support for print and println. Added clear.

2020.03.10: Added textField, textArea, setTextInField, getTextInField, addButton support, repaint in clear.

2020.03.15: Does not exit on close; added close, addPasswordField, addList, addElementToList, removeElementFromList, getSelectedElementFromList

2020.03.21: Fixed non-removal in removeElementFromList, addList and addTextArea now has auto-scrollbars

2020.06.01: Lists of String type

2020.07.06: getElementFromList returns value of consistent type. Added getSelectedElementIndexFromList, setFilePath

2020.08.25: Lists replaced with tables (\t formats data in to columns. Supports optional header row.). Added addTable, addRowToTable, clearRowsOfTable, getSelectedRowIndexFromTable, getSelectedRowFromTable. Exceptions thrown from invoked supplied methods are more readable. Added addImageIcon(BufferedImage), showErrorDialog(String), showWarningDialog(String). Title of showMessageDialog(String) says "information" instead of "message" (to match icon). addPasswordField, addTextArea and addTextField now return the text field index.

### Author:

Gayan Wijesinghe

**See Also:**

[Serialized Form](#)

## ***Nested Class Summary***

### ***Nested classes/interfaces inherited from class java.awt.Window***

`java.awt.Window.Type`

### ***Nested classes/interfaces inherited from class java.awt.Component***

`java.awt.Component.BaselineResizeBehavior`

## ***Field Summary***

### **Fields inherited from class java.awt.Frame**

`ICONIFIED, MAXIMIZED_BOTH, MAXIMIZED_HORIZ, MAXIMIZED_VERT, NORMAL`

### **Fields inherited from class java.awt.Component**

`BOTTOM_ALIGNMENT, CENTER_ALIGNMENT, LEFT_ALIGNMENT, RIGHT_ALIGNMENT, TOP_ALIGNMENT`

### **Fields inherited from interface java.awt.image.ImageObserver**

`ABORT, ALLBITS, ERROR, FRAMEBITS, HEIGHT, PROPERTIES, SOMEBITS, WIDTH`

### **Fields inherited from interface javax.swing.WindowConstants**

`DISPOSE_ON_CLOSE, DO_NOTHING_ON_CLOSE, EXIT_ON_CLOSE, HIDE_ON_CLOSE`

## ***Constructor Summary***

### **Constructors**

Constructor	Description
<code>GTerm(int width,</code>	GTerm constructor taking a width and a height suggestion for

**Constructor**  
`int height()`

**Description**  
 the window

## Method Summary

### All Methods

### Instance Methods

### Concrete Methods

Modifier and Type	Method	Description
void	<b>addButton</b> ( <code>java.lang.String buttonText</code> , <code>java.lang.Object handlingObject</code> , <code>java.lang.String handlingMethodName</code> )	Adds a button.
void	<b>addImageIcon</b> ( <code>java.awt.image.BufferedImage image</code> )	Adds a <code>BufferedImage</code> object at current x,y coordinates.
void	<b>addImageIcon</b> ( <code>java.lang.String imageFilename</code> )	Adds an image at current x,y coordinates.
int	<b>addPasswordField</b> ( <code>java.lang.String defaultPassword</code> , <code>int widthInPixels</code> )	Adds a single-line password text field.
int	<b>addRowToTable</b> ( <code>int indexOfTable</code> , <code>java.lang.String rowData</code> )	Adds a row to an existing table.
int	<b>addTable</b> ( <code>int width</code> , <code>int height</code> , <code>java.lang.String columnNames</code> )	Adds a list-like table.
int	<b>addTextArea</b> ( <code>java.lang.String defaultText</code> , <code>int widthInPixels</code> , <code>int heightInPixels</code> )	Adds a multi-line, word-wrapping-capable, auto-scrolling text area.
int	<b>addTextField</b> ( <code>java.lang.String defaultText</code> , <code>int widthInPixels</code> )	Adds a single-line text field.
void	<b>clear</b> ()	Clears the GTerm like clearing a terminal.
void	<b>clearRowsOfTable</b> ( <code>int indexOfTable</code> )	Clears added rows of an existing table.
void	<b>close</b> ()	Closes this GTerm window but does not necessarily stop the

Modifier and Type	Method	Description
java.awt.Color	<b>getColorFromDialog</b> (java.lang.String message)	program from running.  Shows a colour chooser.
java.lang.String	<b>getFilePath()</b>	Displays a file open dialog.
int	<b>getIndexOfSelectedRowFromTable</b> (int indexofTable)	Retrieve the currently chosen row's index within an existing table.
java.lang.String	<b>getInputString</b> (java.lang.String message)	Allows the user to enter a String.
java.lang.String	<b>getKey()</b>	
int	<b>getMouseX()</b>	
int	<b>getMouseY()</b>	
java.lang.String	<b>getPasswordFromDialog</b> (java.lang.String message)	Shows a password entry dialog box and returns the password.
java.lang.String	<b>getSelectedRowFromTable</b> (int indexofTable)	Retrieve the currently chosen row String from an existing table.
java.lang.String	<b>getTextFromEntry</b> (int textFieldIndex)	Access the text entered in a text field or a text area.
void	<b>keyPressed</b> (java.awt.event.KeyEvent e)	Do not use.
void	<b>keyReleased</b> (java.awt.event.KeyEvent e)	Do not use.
void	<b>keyTyped</b> (java.awt.event.KeyEvent e)	Do not use.
void	<b>mouseClicked</b> (java.awt.event.MouseEvent e)	
void	<b>mouseEntered</b> (java.awt.event.MouseEvent e)	
void	<b>mouseExited</b> (java.awt.event.MouseEvent e)	

Modifier and Type	Method	Description
void	<b>mousePressed</b> (java.awt.event.MouseEvent e)	
void	<b>mouseReleased</b> (java.awt.event.MouseEvent e)	
void	<b>print</b> (java.lang.String message)	Similar to System.out.print.
void	<b>println</b> (java.lang.String message)	Similar to System.out.println.
void	<b>setBackgroundColor</b> (int r, int g, int b)	Change the RGB background colour of the window.
void	<b>setBackgroundColor</b> (java.awt.Color backgroundColor)	Change the background colour of the window.
java.lang.String	<b>setFilePath</b> ()	Displays a file save dialog.
void	<b>setFont</b> (java.lang.String fontName, int fontStyle, int fontSize)	Change the font size, style and size used for text that will be added next.
void	<b>setFontColor</b> (int r, int g, int b)	Change the font colour used for text that will be added next.
void	<b>setFontColor</b> (java.awt.Color fontColor)	Change the font colour used for text that will be added next.
void	<b>setFontName</b> (java.lang.String fontName)	Change the font name used for text that will be added next.
void	<b>setFontSize</b> (int fontSize)	Change the font size used for text that will be added next.
void	<b>setFontStyle</b> (int fontStyle)	Change the font style for text that will be added next.

Modifier and Type	Method	Description
void	<b>setTabSize</b> (int tabSize)	Number of spaces used to replace a tab character by methods such as print and println.
void	<b>setTextInEntry</b> (int textFieldIndex, java.lang.String text)	Modify the text entered in a text field or a text area.
void	<b>setXY</b> (int x, int y)	Allows the change of placement within the window.
void	<b>showErrorDialog</b> (java.lang.String message)	Shows the supplied message to the user using an error message dialog box.
void	<b>showHelp</b> ()	Prints the version, credits and help file location.
void	<b>showMessageDialog</b> (java.lang.String message)	Shows the supplied message to the user using an information message dialog box.
void	<b>showWarningDialog</b> (java.lang.String message)	Shows the supplied message to the user using a warning message dialog box.

### Methods inherited from class javax.swing.JFrame

getAccessibleContext, getContentPane, getDefaultCloseOperation, getGlassPane, getGraphics, getJMenuBar, getLayeredPane, getRootPane, getTransferHandler, isDefaultLookAndFeelDecorated, remove, repaint, setContentPane, setDefaultCloseOperation, setDefaultCloseOperation, setGlassPane, setIconImage, setJMenuBar, setLayeredPane, setLayout, setTransferHandler, update

### Methods inherited from class java.awt.Frame

addNotify, getExtendedState, getFrames, getIconImage, getMaximizedBounds, getMenuBar, getState, getTitle, isResizable, isUndecorated, remove, removeNotify, setBackground, setExtendedState, setMaximizedBounds,

setMenuBar, setOpacity, setResizable, setShape, setState, setTitle, setUndecorated

### Methods inherited from class java.awt.Window

addPropertyChangeListener, addPropertyChangeListener,  
addWindowFocusListener, addWindowListener, addWindowStateListener,  
createBufferStrategy, createBufferStrategy, dispose, getBackground,  
getBufferStrategy, getFocusableWindowState, getFocusCycleRootAncestor,  
getFocusOwner, getFocusTraversalKeys, getIconImages, getInputContext,  
getListeners, getLocale, getModalExclusionType, getMostRecentFocusOwner,  
getOpacity, getOwnedWindows, getOwner, getOwnerlessWindows, getShape,  
getToolkit, getType, getWarningString, getWindowFocusListeners,  
getWindowListeners, getWindows, getWindowStateListeners, isActive,  
isAlwaysOnTop, isAlwaysOnTopSupported, isAutoRequestFocus,  
isFocusableWindow, isFocusCycleRoot, isFocused, isLocationByPlatform,  
isOpaque, isShowing, isValidRoot, pack, paint, removeWindowFocusListener,  
removeWindowListener, removeWindowStateListener, setAlwaysOnTop,  
setAutoRequestFocus, setBounds, setBounds, setCursor,  
setFocusableWindowState, setFocusCycleRoot, setIconImages, setLocation,  
setLocation, setLocationByPlatform, setLocationRelativeTo, setMinimumSize,  
setModalExclusionType, setSize, setSize, setType, setVisible, toBack,  
toFront

### Methods inherited from class java.awt.Container

add, add, add, add, add, addContainerListener, applyComponentOrientation,  
areFocusTraversalKeysSet, doLayout, findComponentAt, findComponentAt,  
getAlignmentX, getAlignmentY, getComponent, getComponentAt, getComponentAt,  
getComponentCount, getComponents, getComponentZOrder, getContainerListeners,  
getFocusTraversalPolicy, getInsets, getLayout, getMaximumSize,  
getMinimumSize, getMousePosition, getPreferredSize, invalidate,  
isAncestorOf, isFocusCycleRoot, isFocusTraversalPolicyProvider,  
isFocusTraversalPolicySet, list, list, paintComponents, print,  
printComponents, remove, removeAll, removeContainerListener,  
setComponentZOrder, setFocusTraversalKeys, setFocusTraversalPolicy,  
setFocusTraversalPolicyProvider, setFont, transferFocusDownCycle, validate

### Methods inherited from class java.awt.Component

add, addComponentListener, addFocusListener, addHierarchyBoundsListener,  
addHierarchyListener, addInputMethodListener, addKeyListener,  
addMouseListener, addMouseMotionListener, addMouseWheelListener, checkImage,  
checkImage, contains, contains, createImage, createImage,  
createVolatileImage, createVolatileImage, dispatchEvent, enableInputMethods,  
firePropertyChange, firePropertyChange, firePropertyChange,

```
firePropertyChange, firePropertyChange, firePropertyChange, getBaseline,
getBaselineResizeBehavior, getBounds, getBounds, getColorModel,
getComponentListeners, getComponentOrientation, getCursor, getDropTarget,
getFocusListeners, getFocusTraversalKeysEnabled, getFont, getFontMetrics,
getForeground, getGraphicsConfiguration, getHeight,
getHierarchyBoundsListeners, getHierarchyListeners, getIgnoreRepaint,
getInputMethodListeners, getInputMethodRequests, getKeyListeners,
getLocation, getLocation, getLocationOnScreen, getMouseListeners,
getMouseMotionListeners, getMousePosition, getMouseWheelListeners, getName,
getParent, getPropertyChangeListeners, getPropertyChangeListeners, getSize,
getSize, getTreeLock, getWidth, getX, getY, hasFocus, imageUpdate,
isBackgroundSet, isCursorSet, isDisplayable, isDoubleBuffered, isEnabled,
isFocusable, isFocusOwner, isFontSet, isForegroundSet, isLightweight,
isMaximumSizeSet, isMinimumSizeSet, isPreferredSizeSet, isValid, isVisible,
list, list, list, paintAll, prepareImage, prepareImage, printAll,
removeComponentListener, removeFocusListener, removeHierarchyBoundsListener,
removeHierarchyListener, removeInputMethodListener, removeKeyListener,
removeMouseListener, removeMouseMotionListener, removeMouseWheelListener,
removePropertyChangeListener, removePropertyChangeListener, repaint,
repaint, repaint, requestFocus, requestFocus, requestFocusInWindow,
requestFocusInWindow, revalidate, setComponentOrientation, setDropTarget,
setEnabled, setFocusable, setFocusTraversalKeysEnabled, setForeground,
setIgnoreRepaint, setLocale, setMaximumSize, setMixingCutoutShape, setName,
setPreferredSize, toString, transferFocus, transferFocusBackward,
transferFocusUpCycle
```

### Methods inherited from class java.lang.Object

```
equals, getClass, hashCode, notify, notifyAll, wait, wait, wait
```

### Methods inherited from interface java.awt.MenuContainer

```
getFont
```

## Constructor Details

### GTerm

```
public GTerm(int width,
             int height)
```

GTerm constructor taking a width and a height suggestion for the window

**Parameters:**



width - Suggested width of content area as an int

height - Suggested height of content area as an int

## Method Details

### showHelp

```
public void showHelp()
```

Prints the version, credits and help file location.

### getPasswordFromDialog

```
public java.lang.String getPasswordFromDialog(java.lang.String message)
```

Shows a password entry dialog box and returns the password. Note: Password stored as a String with convenience, not security, in mind. Has no connection with addPasswordField method.

**Parameters:**

message - The message that will be shown to the user.

**Returns:**

Returns the entered password or null if cancelled.

### getColorFromDialog

```
public java.awt.Color getColorFromDialog(java.lang.String message)
```

Shows a colour chooser. Refer to documentation of javax.swing.JColorChooser's showDialog method for details.

**Parameters:**

message - The message that will be shown to the user.

**Returns:**

Returns the chosen colour or null if cancelled.

### addTable

```
public int addTable(int width,  
                   int height,  
                   java.lang.String columnNames)
```

Adds a list-like table. Also see `addRowToTable`, `getSelectedRowFromTable`, `getSelectedRowIndexFromTable` and `clearRowsOfTable` methods.

**Parameters:**

`width` - Width in pixels

`height` - Height in pixels

`columnNames` - Tab-separated column names for header row. Use null for no header row.

**Returns:**

Returns the index of the table (use this when adding to, retrieving from, etc. this table.

## **addRowToTable**

```
public int addRowToTable(int indexOfTable,  
                        java.lang.String rowData)
```

Adds a row to an existing table. Use `addTable` method to add a table first.

**Parameters:**

`indexOfTable` - Index of an existing table (see `addTable`)

`rowData` - Data for this row, optionally separated by '\t'. Automatically adds columns as needed.

**Returns:**

Returns the index of the row (e.g. can be used with `getSelectedRowFromTable` method, etc.)

## **clearRowsOfTable**

```
public void clearRowsOfTable(int indexOfTable)
```

Clears added rows of an existing table. To remove a table or any other added component, use the `clear` method.

**Parameters:**

`indexOfTable` - Index of an existing table (see `addTable`)

## getIndexOfSelectedRowFromTable

```
public int getIndexOfSelectedRowFromTable(int indexOfTable)
```

Retrieve the currently chosen row's index within an existing table.

**Parameters:**

`indexOfTable` - Index of an existing table (see `addTable`)

**Returns:**

Returns the index of the selected row or -1 if no row selected (e.g. can be used with `getSelectedRowFromTable` method, etc.)

## getSelectedRowFromTable

```
public java.lang.String getSelectedRowFromTable(int indexOfTable)
```

Retrieve the currently chosen row String from an existing table.

**Parameters:**

`indexOfTable` - Index of an existing table (see `addTable`)

**Returns:**

Returns a String or -1 if no row selected

## close

```
public void close()
```

Closes this GTerm window but does not necessarily stop the program from running. Use the `clear` method if necessary to redraw (with removed, re-arranged components).

## setTextInEntry

```
public void setTextInEntry(int textFieldIndex,  
                           java.lang.String text)
```

Modify the text entered in a text field or a text area. Programmer must keep track of number of text areas currently visible. Indices reset to zero upon calls to the `clear()` method.

**Parameters:**

`textFieldIndex` - Index number of text field/area currently showing.

`text` - Text to be added

## getTextFromEntry

```
public java.lang.String getTextFromEntry(int textFieldIndex)
```

Access the text entered in a text field or a text area. Programmer must keep track of number of text areas currently visible. Indices reset to zero upon calls to the `clear()` method.

**Parameters:**

`textFieldIndex` - Index number of text field/area currently showing.

**Returns:**

The text that is currently in the specified text element.

## addPasswordField

```
public int addPasswordField(java.lang.String defaultPassword,  
                           int widthInPixels)
```

Adds a single-line password text field. (See also `addTextField`) Access its contents using `getTextFromEntry` method. Modify its contents using `setTextInEntry` method. Note: Password stored as a `String` with convenience, not security, in mind.

**Parameters:**

`defaultPassword` - Default text. Give an empty `String` to keep blank.

`widthInPixels` - Width in pixels

**Returns:**

Returns the index of this text component (to be used with `getTextFromEntry`, etc.)

## addTextField

```
public int addTextField(java.lang.String defaultText,  
                       int widthInPixels)
```

Adds a single-line text field. (See also `addTextArea` and `addPasswordField`). Access its contents using `getTextFromEntry` method. Modify its contents using `setTextInEntry` method.

**Parameters:**

`defaultText` - Default text. Give an empty `String` to keep blank.

`widthInPixels` - Width in pixels

**Returns:**

Returns the index of this text component (to be used with `getTextFromEntry`, etc.)

## addTextArea

```
public int addTextArea(java.lang.String defaultText,  
                      int widthInPixels,  
                      int heightInPixels)
```

Adds a multi-line, word-wrapping-capable, auto-scrolling text area. (See also `addTextField`) Access its contents using `getTextFromEntry` method. Modify its contents using `setTextInEntry` method.

**Parameters:**

`defaultText` - Default text. Give an empty String to keep blank.

`widthInPixels` - Width in pixels

`heightInPixels` - Height in pixels

**Returns:**

Returns the index of this text component (to be used with `getTextFromEntry`, etc.)

## addButton

```
public void addButton(java.lang.String buttonText,
                      java.lang.Object handlingObject,
                      java.lang.String handlingMethodName)
```

Adds a button.

**Parameters:**

`buttonText` - Button text.

`handlingObject` - The object containing the method mentioned in `handlingMethodName`.

`handlingMethodName` - The method that will be called when the button is pressed.

## clear

```
public void clear()
```

Clears the GTerm like clearing a terminal. To close with GTerm window, use the `close` method.

## setFontName

```
public void setFontName(java.lang.String fontName)
```

Change the font name used for text that will be added next. Does not change fonts of previously added text.

**Parameters:**

`fontName` - A String describing a logical Font. Default is `Font.MONOSPACED`. See API for more <https://docs.oracle.com/javase/10/docs/api/java/awt/Font.html>

## setFontSize

```
public void setFontSize(int fontSize)
```

Change the font size used for text that will be added next. Does not change size of previously added text.

**Parameters:**

`fontSize` - An int size. Default is 12.

## setFontStyle

```
public void setFontStyle(int fontStyle)
```

Change the font style for text that will be added next. Does not change the style of previously added text.

Examples: `setFontStyle(Font.BOLD)`, `setFontStyle(Font.ITALIC)`, `setFontStyle(Font.PLAIN)`,  
`setFontStyle(Font.BOLD | Font.ITALIC)`

**Parameters:**

`fontStyle` - One of the above statements (int)

## setBackgroundColor

```
public void setBackgroundColor(java.awt.Color backgroundColor)
```

Change the background colour of the window. Default is `Color.BLACK`.

**Parameters:**

`backgroundColor` - For available colours, refer to <https://docs.oracle.com/javase/7/docs/api/java/awt/Color.html>

## setBackgroundColor

```
public void setBackgroundColor(int r,  
                                int g,  
                                int b)
```

Change the RGB background colour of the window. Default is 0,0,0.

**Parameters:**

`r` - An int between 0-255 indicating the red value

`g` - An int between 0-255 indicating the green value

`b` - An int between 0-255 indicating the blue value

## setFont

```
public void setFont(java.lang.String fontName,  
                    int fontStyle,  
                    int fontSize)
```

Change the font size, style and size used for text that will be added next. Does not change previously added text.

**Parameters:**

`fontName` - See documentation of `setFontName` method for details

`fontStyle` - See documentation of `setFontStyle` method for details

`fontSize` - See documentation of `setFontSize` method for details

## setFontColor

```
public void setFontColor(java.awt.Color fontColor)
```

Change the font colour used for text that will be added next. Does not change colours of previously added text.

**Parameters:**

`fontColor` - For available colours, refer to <https://docs.oracle.com/javase/7/docs/api/java/awt/Color.html>

## setFontColor

```
public void setFontColor(int r,  
                        int g,  
                        int b)
```

Change the font colour used for text that will be added next. Does not change colours of previously added text.

**Parameters:**

`r` - An int between 0-255 indicating the red value

`g` - An int between 0-255 indicating the green value

`b` - An int between 0-255 indicating the blue value

## setTabSize

```
public void setTabSize(int tabSize)
```

Number of spaces used to replace a tab character by methods such as `print` and `println`. Does not affect tables. Does not affect previously added text.

**Parameters:**

tabSize - An int between 0-255 indicating the blue value

### showMessageDialog

```
public void showMessageDialog(java.lang.String message)
```

Shows the supplied message to the user using an information message dialog box.

**Parameters:**

message - The message String. If you want to display multiple variables, you will need to concatenate first.

### showErrorDialog

```
public void showErrorDialog(java.lang.String message)
```

Shows the supplied message to the user using an error message dialog box.

**Parameters:**

message - The message String. If you want to display multiple variables, you will need to concatenate first.

### showWarningDialog

```
public void showWarningDialog(java.lang.String message)
```

Shows the supplied message to the user using a warning message dialog box.

**Parameters:**

message - The message String. If you want to display multiple variables, you will need to concatenate first.

### getInputString

```
public java.lang.String getInputString(java.lang.String message)
```

Allows the user to enter a String. Shows an input dialog box with the message String supplied as a prompt.

**Parameters:**

message - The message String. If you want to display multiple variables, you will need to concatenate first.

**Returns:**

Returns a String. If you need an int or double use Integer.parseInt or Double.parseDouble to convert the returned String.



## getFilePath

```
public java.lang.String getFilePath()
```

Displays a file open dialog.

**Returns:**

Returns a String of the absolute path of the chosen file.

## setFilePath

```
public java.lang.String setFilePath()
```

Displays a file save dialog.

**Returns:**

Returns a String of the absolute path of the chosen/entered file.

## addImageIcon

```
public void addImageIcon(java.lang.String imageFilename)
```

Adds an image at current x,y coordinates. Supports GIF, JPEG and PNG. Can be used with getFilePath method as `gt.addImageIcon(gt.getFilePath())` Also see setXY method.

**Parameters:**

`imageFilename` - Path to a image file.

## addImageIcon

```
public void addImageIcon(java.awt.image.BufferedImage image)
```

Adds a BufferedImage object at current x,y coordinates. Also see setXY method.

**Parameters:**

`image` - Existing BufferedImage object.

## print

```
public void print(java.lang.String message)
```

Similar to System.out.print. Text is added at current X,Y coordinates. Also see setXY method.

**Parameters:**

`message` - String message to display.

## println

```
public void println(java.lang.String message)
```

Similar to System.out.println. Text is added at current X,Y coordinates then cursor moves to "next line". Also see setXY method.

### Parameters:

message - String message to display.

## setXY

```
public void setXY(int x,  
                  int y)
```

Allows the change of placement within the window. Can be used with print, println, addImageIcon, etc. methods.

### Parameters:

x - An int x value within the window

y - An int y value within the window

## getMouseX

```
public int getMouseX()
```

### Returns:

Returns the x coordinate of the mouse within the window. -1 if outside window.

## getMouseY

```
public int getMouseY()
```

### Returns:

Returns the y coordinate of the mouse within the window. -1 if outside window.

## getKey

```
public java.lang.String getKey()
```

### Returns:

Returns the character of the key that was most recently pressed.

## keyReleased

```
public void keyReleased(java.awt.event.KeyEvent e)
```

Do not use. Internal method.

**Specified by:**

keyReleased in interface `java.awt.event.KeyListener`

## keyTyped

```
public void keyTyped(java.awt.event.KeyEvent e)
```

Do not use. Internal method.

**Specified by:**

keyTyped in interface `java.awt.event.KeyListener`

## keyPressed

```
public void keyPressed(java.awt.event.KeyEvent e)
```

Do not use. Internal method.

**Specified by:**

keyPressed in interface `java.awt.event.KeyListener`

## mouseClicked

```
public void mouseClicked(java.awt.event.MouseEvent e)
```

**Specified by:**

mouseClicked in interface `java.awt.event.MouseListener`

## mousePressed

```
public void mousePressed(java.awt.event.MouseEvent e)
```

**Specified by:**

mousePressed in interface `java.awt.event.MouseListener`

## mouseReleased

```
public void mouseReleased(java.awt.event.MouseEvent e)
```

**Specified by:**

mouseReleased in interface `java.awt.event.MouseListener`

**mouseEntered**

```
public void mouseEntered(java.awt.event.MouseEvent e)
```

**Specified by:**

mouseEntered in interface `java.awt.event.MouseListener`

**mouseExited**

```
public void mouseExited(java.awt.event.MouseEvent e)
```

**Specified by:**

mouseExited in interface `java.awt.event.MouseListener`