

classes

Class GUIapp

java.lang.Object
 java.awt.event.WindowAdapter
 classes.GUIapp

All Implemented Interfaces:

java.awt.event.WindowFocusListener, java.awt.event.WindowListener,
java.awt.event.WindowStateListener, java.lang.Runnable, java.util.EventListener

```
public class GUIapp  
extends java.awt.event.WindowAdapter  
implements java.awt.event.WindowListener, java.lang.Runnable
```

Contains all GUI elements and calls the functions in the Worker class as needed. Taken from: <http://www.comweb.nl/java/Console/Console.html> and adapted for use with this application by Bryan Spahr.

Author:

bryan

Field Summary

Fields	
Modifier and Type	Field and Description
private javax.swing.JFrame	editorFrame Popup frame that can be opened to edit faculty info.
private javax.swing.JFileChooser	fc Save dialog for saving the results to a plaintext file.
private javax.swing.JFrame	mainFrame Main GUI window with buttons.
private java.io.PipedInputStream	pin For piping sysout text to the editor window.
private java.io.PipedInputStream	pin2 For piping sysout text to the editor window.
private boolean	quit Used for properly closing threads when the application is terminated.
private java.lang.Thread	reader For capturing the text stream from sysout.

<code>private java.lang.Thread</code>	reader2 For capturing the text stream from sysout.
<code>private java.io.File</code>	resultFile File pointer for saving the results to a plaintext file.
<code>private javax.swing.JFrame</code>	resultFrame Display window for showing calculated results.
<code>private java.lang.String</code>	resultString For storing the result string as a field in this class.
<code>private javax.swing.JTextArea</code>	resultText For displaying the results text to the user.
<code>private javax.swing.JTextArea</code>	textArea Recipient of sysout print statements in the editor window.

Constructor Summary

Constructors
Constructor and Description
<div> <div>GUIapp()</div> <div>Creates a new instance of GUIapp (duh), also calls the Worker class's init method and all GUI methods and init functions in this class.</div> </div>

Method Summary

All Methods	Instance Methods	Concrete Methods
Modifier and Type	Method and Description	
void	editorWindow (boolean show)	Creates the editor window with all swing elements.
void	mainWindow ()	Creates and shows the main GUI window, including all Swing elements.
java.lang.String	readLine (java.io.PipedInputStream in)	Threading manager for sysout piping from aforementioned example code.
void	resultWindow ()	Creates the window for results but doesn't show it.
void	run ()	Standard threaded GUI method.
void	textStreamInit ()	Taken from the example code referenced above.

void **windowClosed**(java.awt.event.WindowEvent evt)

Standard JSwing event method.

void **windowClosing**(java.awt.event.WindowEvent evt)

No idea what this does.

Methods inherited from class java.awt.event.WindowAdapter

windowActivated, windowDeactivated, windowDeiconified, windowGainedFocus, windowIconified, windowLostFocus, windowOpened, windowStateChanged

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface java.awt.event.WindowListener

windowActivated, windowDeactivated, windowDeiconified, windowIconified, windowOpened

Field Detail

editorFrame

```
private javax.swing.JFrame editorFrame
```

Popup frame that can be opened to edit faculty info.

mainFrame

```
private javax.swing.JFrame mainFrame
```

Main GUI window with buttons.

textArea

```
private javax.swing.JTextArea textArea
```

Recipient of sysout print statements in the editor window.

reader

```
private java.lang.Thread reader
```

For capturing the text stream from sysout.

reader2

```
private java.lang.Thread reader2
```

For capturing the text stream from sysout.

quit

```
private boolean quit
```

Used for properly closing threads when the application is terminated. I think.

resultFrame

```
private javax.swing.JFrame resultFrame
```

Display window for showing calculated results.

resultText

```
private javax.swing.JTextArea resultText
```

For displaying the results text to the user.

resultString

```
private java.lang.String resultString
```

For storing the result string as a field in this class. In case it's needed later.

resultFile

```
private java.io.File resultFile
```

File pointer for saving the results to a plaintext file.

fc

```
private javax.swing.JFileChooser fc
```

Save dialog for saving the results to a plaintext file.

pin

```
private final java.io.PipedInputStream pin
```

For piping sysout text to the editor window.

pin2

```
private final java.io.PipedInputStream pin2
```

For piping sysout text to the editor window.

Constructor Detail

GUIapp

```
public GUIapp()
```

Creates a new instance of GUIapp (duh), also calls the Worker class's init method and all GUI methods and init functions in this class.

Method Detail

mainWindow

```
public void mainWindow()
```

Creates and shows the main GUI window, including all Swing elements. Closing this windows is the only way to exit the application.

resultWindow

```
public void resultWindow()
```

Creates the window for results but doesn't show it.

editorWindow

```
public void editorWindow(boolean show)
```

Creates the editor window with all swing elements.

Parameters:

show - Indicates whether or not the editor window should be shown at the start.

textStreamInit

```
public void textStreamInit()
```

Taken from the example code referenced above. Evidently it sets up the text piping from sysout to the text area in the editor window.

windowClosed

```
public void windowClosed(java.awt.event.WindowEvent evt)
```

Standard JSwing event method. When the GUI window is closed, all threads are closed safely. Only the main window uses this method, so that only closing the main window closes the application.

Specified by:

windowClosed in interface `java.awt.event.WindowListener`

Overrides:

windowClosed in class java.awt.event.WindowAdapter

windowClosing

```
public void windowClosing(java.awt.event.WindowEvent evt)
```

No idea what this does. Google it.

Specified by:

windowClosing in interface java.awt.event.WindowListener

Overrides:

windowClosing in class java.awt.event.WindowAdapter

run

```
public void run()
```

Standard threaded GUI method. Here it's used to manage the sysout text threads.

Specified by:

run in interface java.lang.Runnable

readLine

```
public java.lang.String readLine(java.io.PipedInputStream in)
                             throws java.io.IOException
```

Threading manager for sysout piping from aforementioned example code. No, I don't know what it does.

Parameters:

in - A PipedInputStream, duh

Returns:

some sort of string obviously

Throws:

java.io.IOException - if there's an IOException, why do I have to describe this

[PACKAGE](#) **[CLASS](#)** [USE](#) [TREE](#) [DEPRECATED](#) [INDEX](#) [HELP](#)

[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [ALL CLASSES](#)

[SUMMARY: NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHOD](#) [DETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#)