

Cheat Sheet based on:

Swing/GUI Cheat Sheet at Williams College

<http://eventfuljava.cs.williams.edu/s04/handouts/SwingGUICheatSheet.html>

Displaying a Swing component

Construct and initialize the component.

```
button = new JButton ("ButtonLabel");
```

Add it to the content pane of the window or to a JPanel that is added to the display.

```
getContentPane().add (button);
```

Import `javax.swing`. *and sometimes also java.awt.* at the beginning of the class creating the components.

```
import javax.swing.*; import java.awt.*;
```

Getting events from GUI component

Declare that the class handling the event implements the appropriate listener interface.

```
implements ActionListener
```

Define the method that the listener interface requires.

```
public void actionPerformed (ActionEvent event)
```

Add a listener appropriate for the component to the component.

```
button.addActionListener (this);
```

Import `java.awt.event`. (*and occasionally javax.swing.event.*) at the beginning of the class that is the listener.

```
import javax.swing.*; import java.awt.*;
```

Finding out which component sent the event

When the listener method is called, you can find out which component sent the event by calling `getSource()` on the event:

```
public void actionPerformed (ActionEvent event) {
    Object theButton = event.getSource();
    if (theButton == framedCircleButton) {
        // Create a framed circle
    }
}
```

If a method returns a String, remember to compare the result using the equals method, not `==`:

```
aMenu.getSelectedItems().equals ("A value");
```

Containers

JPanel constructor: `new JPanel ()`

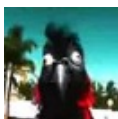
Define the type of layout: `void setLayout (LayoutManager lm)`

Add an object to a container: (FlowLayout or GridLayout) `void add (Component c)`

Add an object to a container: (BorderLayout) `void add (Component c, int position)`

Both JPanel and the object obtained by sending `getContentPane()` to a WindowController object are containers (and have type Container). These methods are available for all containers.

For BorderLayouts, position may be either `BorderLayout.NORTH`, `BorderLayout.SOUTH`, `BorderLayout.EAST`, `BorderLayout.WEST`, or `BorderLayout.CENTER`.



By NeonKnightOA

cheatography.com/neonknightoa/

Published 11th November, 2015.

Last updated 12th May, 2016.

Page 1 of 3.

Sponsored by ApolloPad.com

Everyone has a novel in them. Finish

Yours!

<https://apollopad.com>

Layout Managers

BorderLayout constructor: `new BorderLayout ()`

FlowLayout constructor: `new FlowLayout ()`

GridLayout constructor: `new GridLayout (int rows, int cols)`
`new GridLayout (int rows, int cols, int colSpacing, int rowSpacing)`

BorderLayout is the default layout for WindowController, whereas FlowLayout is default for JPanel.

GUI Components - General

The following methods can be applied to any Component:

```
void setFont (Font f)
void setForeground (Color c)
void setBackground (Color c)
```

To construct a font use:

```
new Font (String name, int style, int size)
```

Style can be one of the following:

```
Font.BOLD
Font.ITALIC
Font.PLAIN
Font.BOLD+Font.ITALIC
```

GUI Components - JButton

Constructor: `new JButton (String s)`

General `String getText ()`

Methods: `void setText (String s)`

Listener Interface: `ActionListener`

Adding the listener: `void addActionListener (ActionListener al)`

Listening Method: `void actionPerformed (ActionEvent e)`

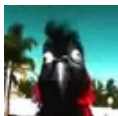
GUI Components - JComboBox

Constructor and Initialization: `new JComboBox ()`
`void addItem (Object item)`

General Methods: `Object getSelectedItem ()`
`String text= (String)menu.getSelectedItem();`
`int getSelectedIndex ()`

Listener Interface: `ItemListener`
`ActionListener`

Adding the listener: `void addItemListener (ItemListener il)`
`void addActionListener (ActionListener al)`



By NeonKnightOA

Published 11th November, 2015.

Last updated 12th May, 2016.

Page 2 of 3.

Sponsored by [ApolloPad.com](https://apollopad.com)

Everyone has a novel in them. Finish Yours!

<https://apollopad.com>

GUI Components - JComboBox (cont)

Listening Method: `void itemStateChanged (ItemEvent e)`
`void actionPerformed (ActionEvent e)`

About methods:

`getSelectedItem ()` returns the selected item
`(String) menu.getSelectedItem ()`; is a typecast which treats the above returned value as a String
`int getSelectedIndex ()` returns the index of the selected item.

About the listeners:

This component can hear the user making a menu selection dependong on the chosen interface. Be consistent in your choice of listener interface, adding method, and listening method.

GUI Components - JLabel

Constructors: `new JLabel (String s)`
`new JLabel (String s, int align)`

General Methods: `void setText (String s)`
`String getText ()`

Listener Interface: No listeners available.

`align` can be either `JLabel.RIGHT`, `JLabel.LEFT` or `JLabel.CENTER`.

GUI Components - JSlider

Constructor: `new JSlider (int orientation, int minimum, int maximum, int initialValue)`

General Method: `void setValue (int newVal)`

Methods: `int getValue ()`

Listener Interface: `ChangeListener`

Adding the Listener: `addChangeListener (ChangeListener al)`

Listening Method: `void stateChanged (ChangeEvent e)`

`orientation` can be either `JSlider.HORIZONTAL` or `JSlider.VERTICAL`.

GUI Components - JTextField

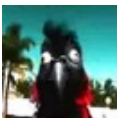
Constructors: `new JTextField (String s)`

General Methods: `void setText (String s)`
`String getText ()`

Listener Interface: `ActionListener`

Adding the Listener: `addActionListener (ActionListener al)`

Listening Method: `void actionPerformed (ActionEvent e)`



By NeonKnightOA

cheatography.com/neonknightoa/

Published 11th November, 2015.

Last updated 12th May, 2016.

Page 3 of 3.

Sponsored by [ApolloPad.com](https://apollopad.com)

Everyone has a novel in them. Finish Yours!

<https://apollopad.com>