

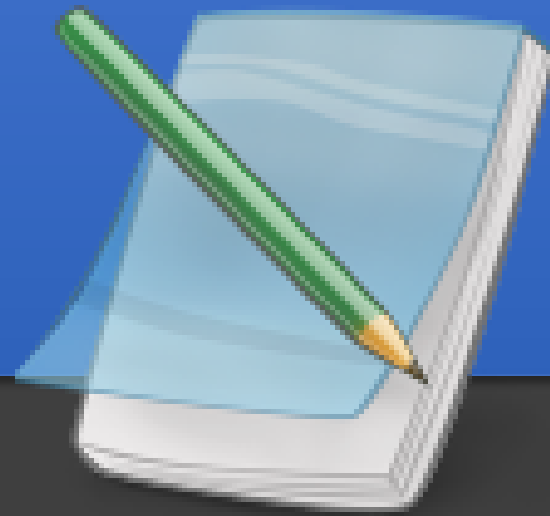
Lab 7: GUI and Event Handling

Lab Objectives



1. Create GUI using simple components & handle events
2. Use Eclipse 's WindowBuilder plugin
 - Create components, edit properties
 - Use nested panels and layout managers
 - Handle events

KEYNOTE

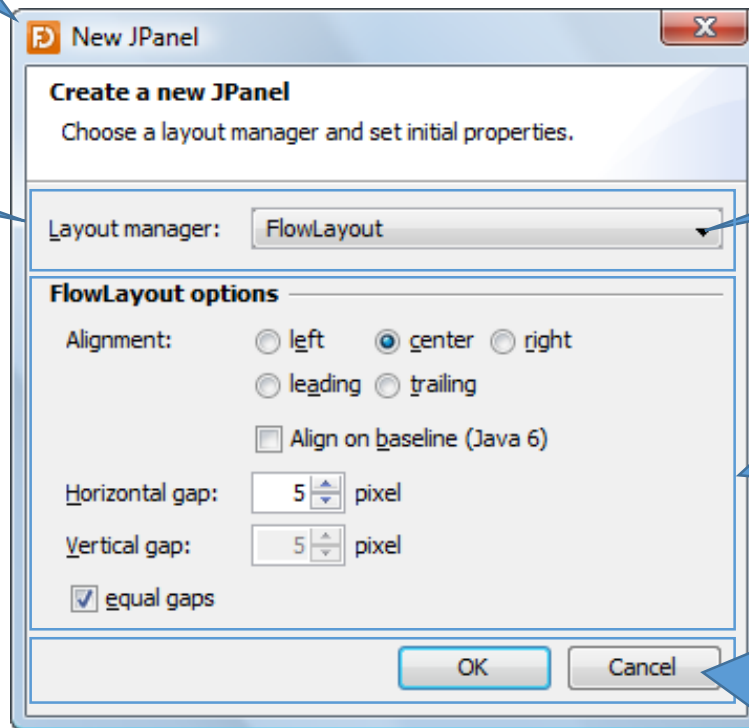


Java GUI



Window

Panel



Components have properties

Layout

- Flow Layout
- Border Layout
- Grid Layout
- ...

Listener:

- Action Listener
- Change Listener
- Item Listener
- Component Listener
- Focus Listener
- Key Listener
- Mouse Listener
- ...

TUTORIAL

Sample GUI application



1. “Hello World” application



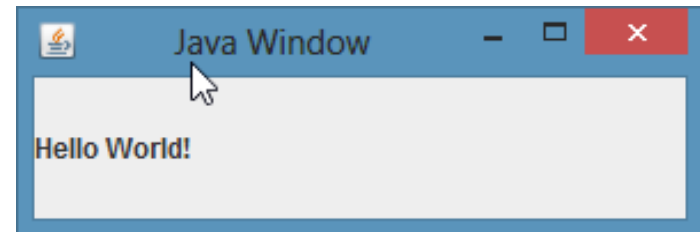
- **Problem:**

Create a small window with “Hello World” text

- **Do:**

3 easy steps to get a GUI window on screen:

1. Create and set up the window.
2. Add a label into window.
3. Show the window.



- **Note:**

- All of the following code lives inside a simple class with just a `main()` method.
- Actually the all the GUI code will be run in another thread called event-dispatching thread (EDT). (Will be explained in Multithread Class)

1. “Hello World” example

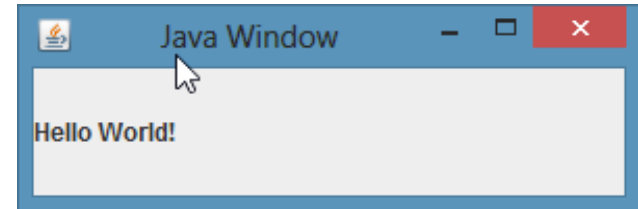


```
import java.awt.Dimension;
import javax.swing.JFrame;
import javax.swing.JLabel;

/**
 * Simple demo of Java Swing GUI toolkit.
 */
public class HelloWorldGUI {
    public static void main(String[] args) {
        // set up the window
        JFrame frame = new JFrame();
        frame.setTitle("Java Window");
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        // add a label
        JLabel label = new JLabel("Hello World!");
        frame.add(label);

        // show the window
        frame.setSize(300, 100);
        frame.setVisible(true);
    }
}
```



Set up the window:

- JFrame acts as window component
- Set title for the window
- Assign action to close button/operation

Add a label:

- JLabel contains our message
- Add label to frame's content pane

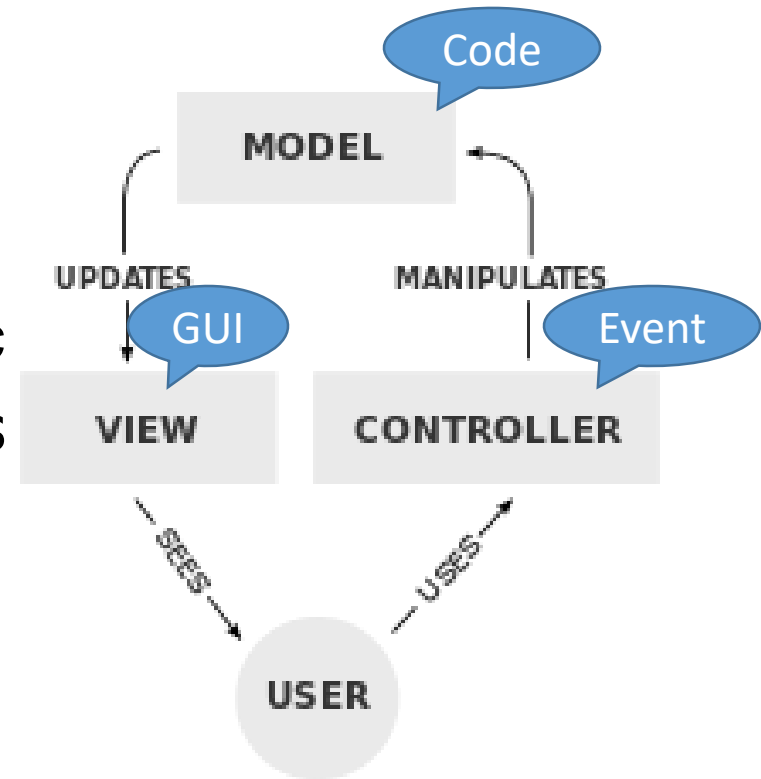
Show the window:

- Set size for the frame
- Make frame visible

2. MVC model



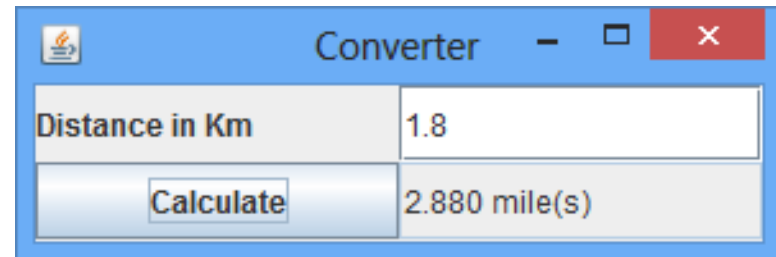
- **Model-View-Controller** design pattern
- Isolate data model from user interface from application logic
 - **Model**: data model classes
 - **View**: user interface (i.e. GUI, console, etc.)
 - **Controller**: interacts with View, manipulates Model



3. Converter application



- **Problem:**
 - Create a GUI program to **convert** the distance from **kilometer to mile** (1 km = 0.621 mile)
- **Model (Data):**
 - We don't have to store any data so no need
- **View (GUI):**
 - 1 label
 - 1 input text
 - 1 button
 - 1 display text
- **Controller (User interactive)**
 - Click the button



To control more about the window and can be reusable, we create our own object extending JFrame

WINDOWBUILDER

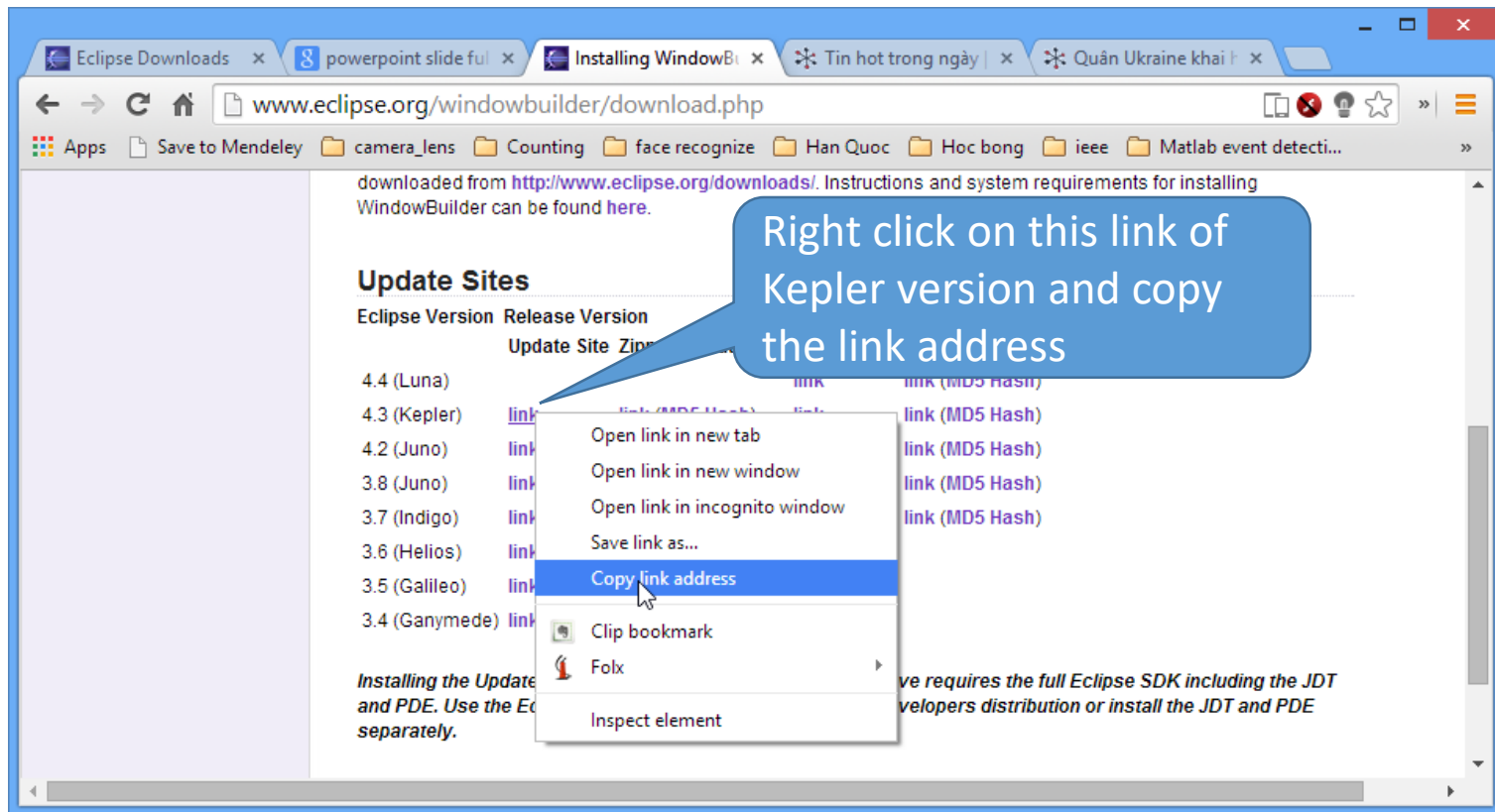
Tool to create Java GUI applications



1. Install WindowBuilder



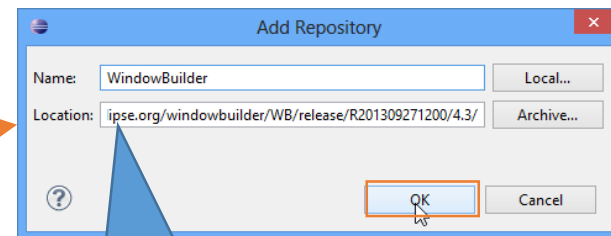
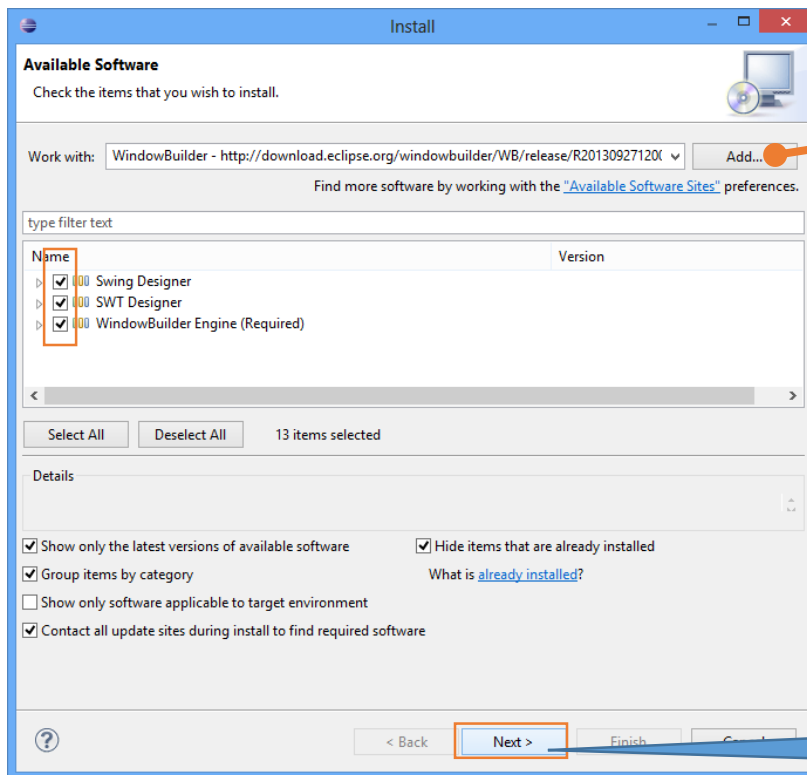
- Go to this website and copy the link based on your Eclipse version: <http://www.eclipse.org/windowbuilder/download.php>



1. Install WindowBuilder



- Go to Eclipse, Menu Help -> Install New Software



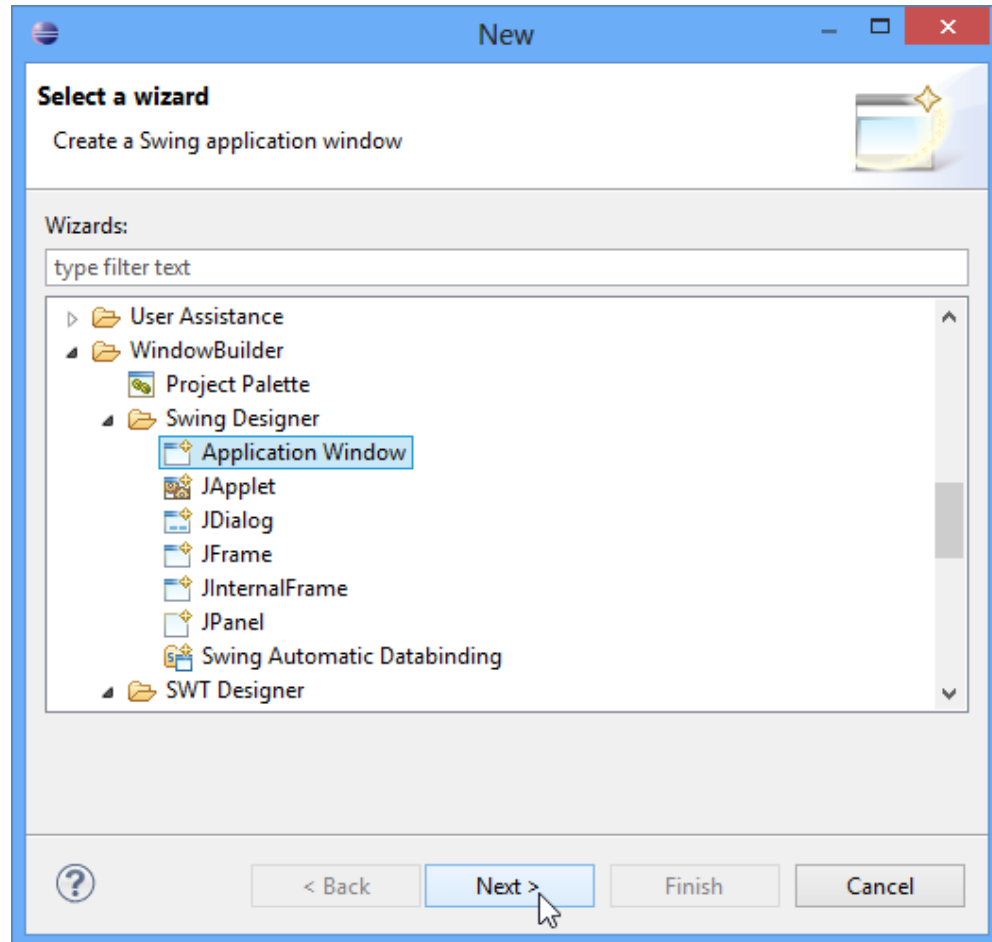
Paste your link here

Keep clicking next

2. Using WindowBuilder New Application



- File → New →
Other... →
WindowBuilder →
Swing Designer →
Application Window.



2. Using WindowBuilder Design Tab



WindowBuilder Design Tab Interface

Project: HelloWorldGU... | Files: AlignFrame.java, ConverterMai..., Align.java, Events.java, EventsFrame..., *EventTest.java

Structure

- frame
 - getContentPane()
 - panel
 - IbINewLabel - "New label"
 - textField

Properties

Variable	textField
Class	javax.swing.JTextField
background	255,255,255
columns	10
dropMode	USE_SELECTION
editable	<input checked="" type="checkbox"/> true
enabled	<input checked="" type="checkbox"/> true
font	Gulim 12
foreground	0,0,0
horizontalAlign...	LEADING
text	
toolTipText	

Palette

- System
 - Selection: Marquee
 - Choose co...: Tab Order
- Containers
 - JPanel, JScrollPane, JSplitPane, JTabbedPane, JToolBar, JLayeredPa..., JDesktopPa..., JInternalFra...
- Layouts
 - Absolute la..., FlowLayout, BorderLayout, GridLayout, GridBagLay..., CardLayout, BoxLayout, SpringLayout, FormLayout, MigLayout, GroupLayout
- Struts & Springs
- Components
 - JLabel, JTextField, JComboBox, JButton, JCheckBox, JRadioButton, JToggleButton, JTextArea, JFormatted..., JPasswordField, JTextPane, JEditorPane, JSpinner, JList

Design View

Visual representation of the GUI showing a window titled "New label" with a text field.

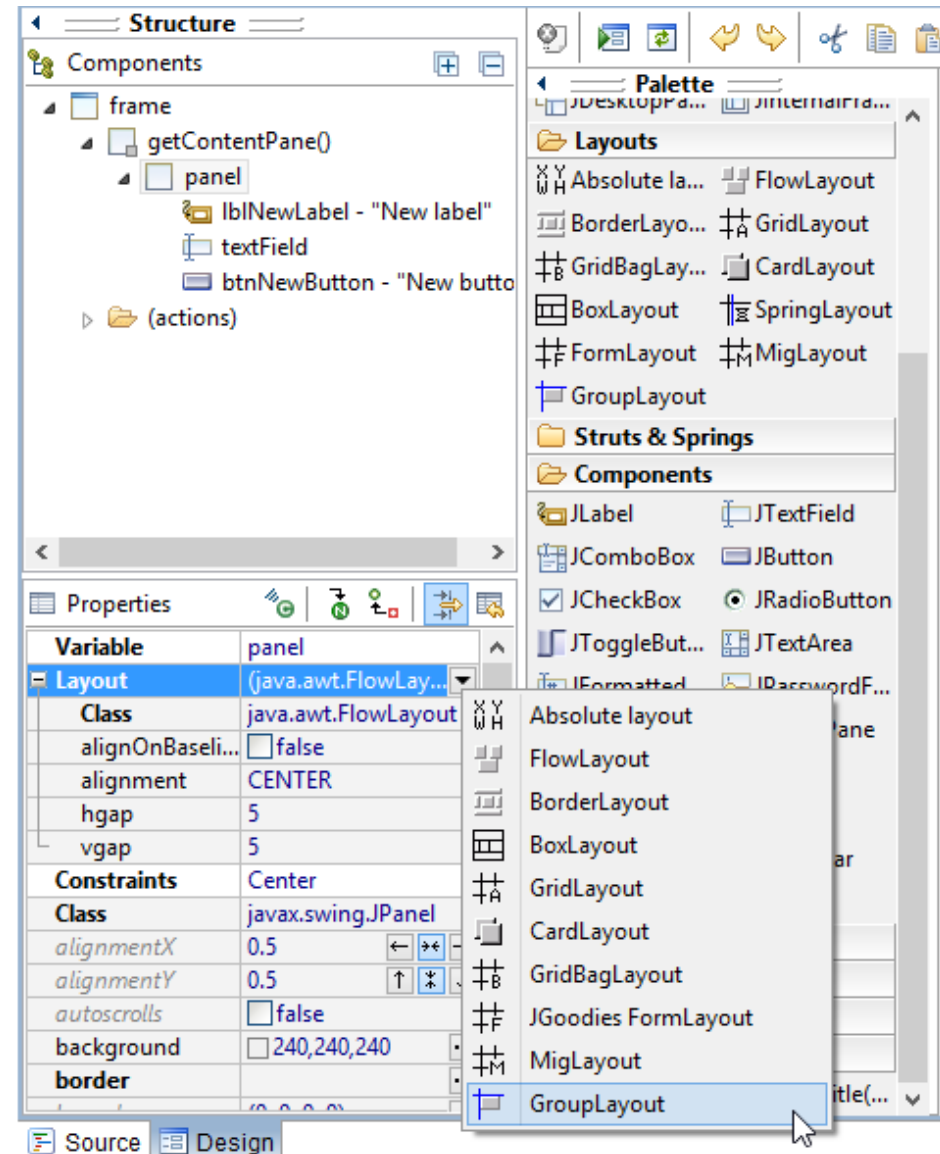
Source | **Design**

1.6.1.r43x201401161243

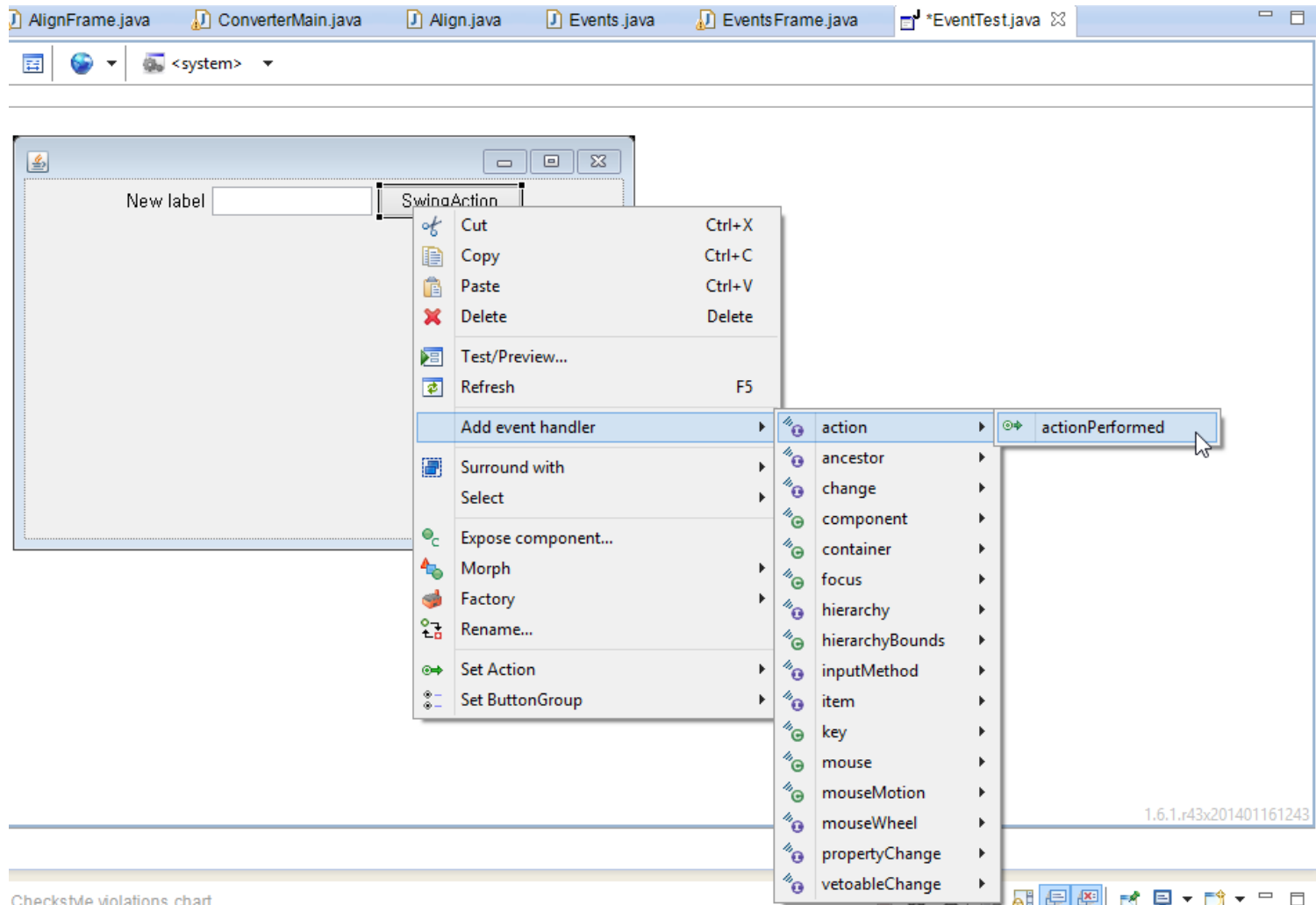
2. Using WindowBuilder Layout



- Change the layout in the Properties window
- Or right click on the panel object -> Set Layout



2. Using WindowBuilder Event handling

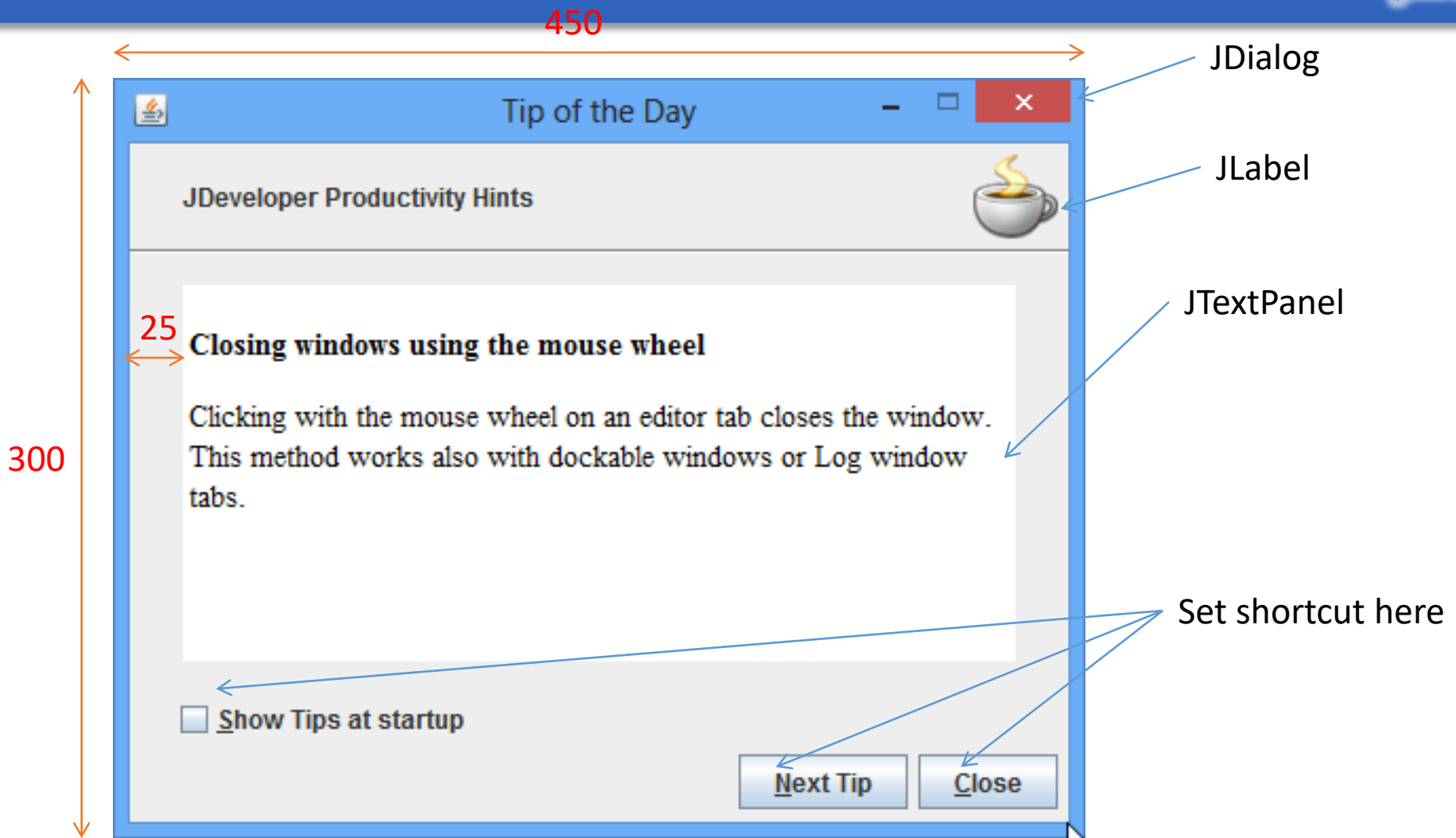


PRACTICE

Sample GUI application



Tip of the day



You can get the icon by:

```
ImageIcon icon = new ImageIcon(getClass().getResource("/package_name/image.png"));
```