

CHUYÊN ĐỀ JAVA

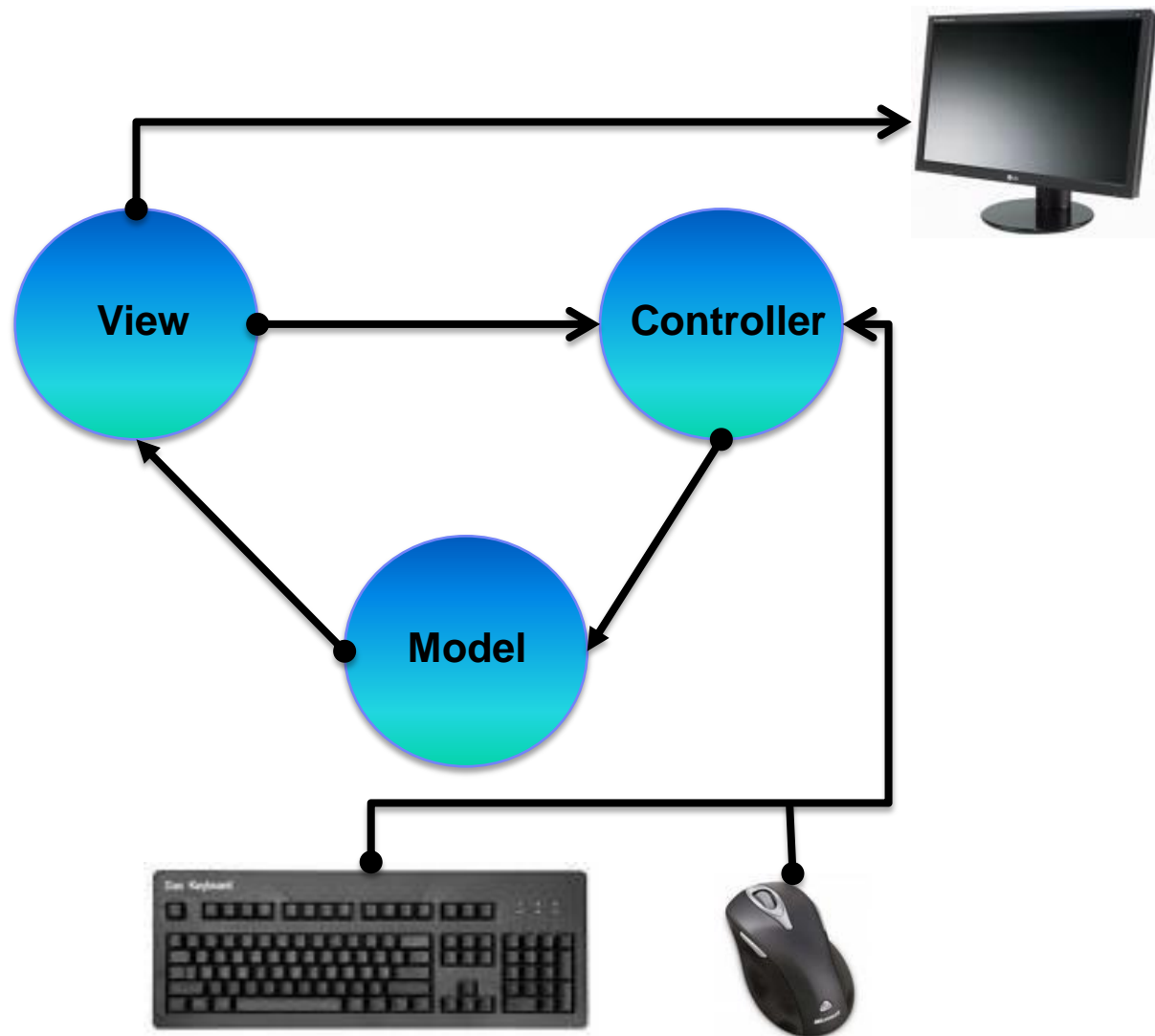
# SWING

Nguyễn Hoàng Anh – [nhanh@fit.hcmuns.edu.vn](mailto:nhanh@fit.hcmuns.edu.vn)

# Nội dung

- MVC
- JFrame
- JDesktopPane
- JInternalFrame
- JLabel
- JButton
- JTextField
- JCheckBox
- JRadioButton
- JPanel
- JComboBox, JList
- JTable
- JMenu
- JToolBar
- JOptionPane
- JFileChooser

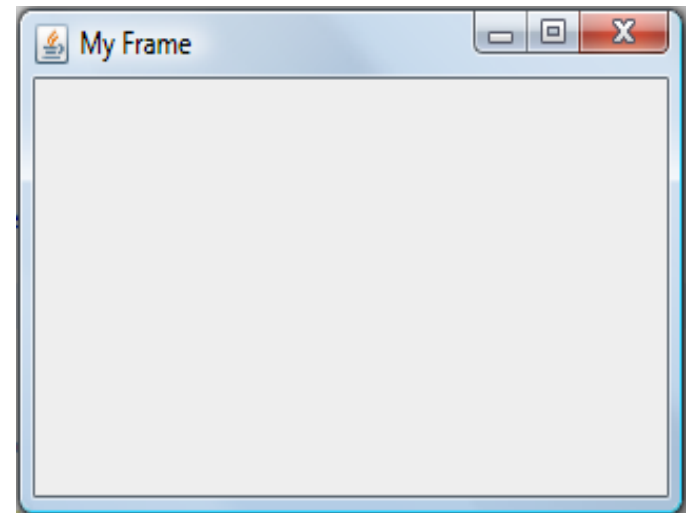
# MVC



# JFrame

```
public class MyFrame extends javax.swing.JFrame {  
  
}
```

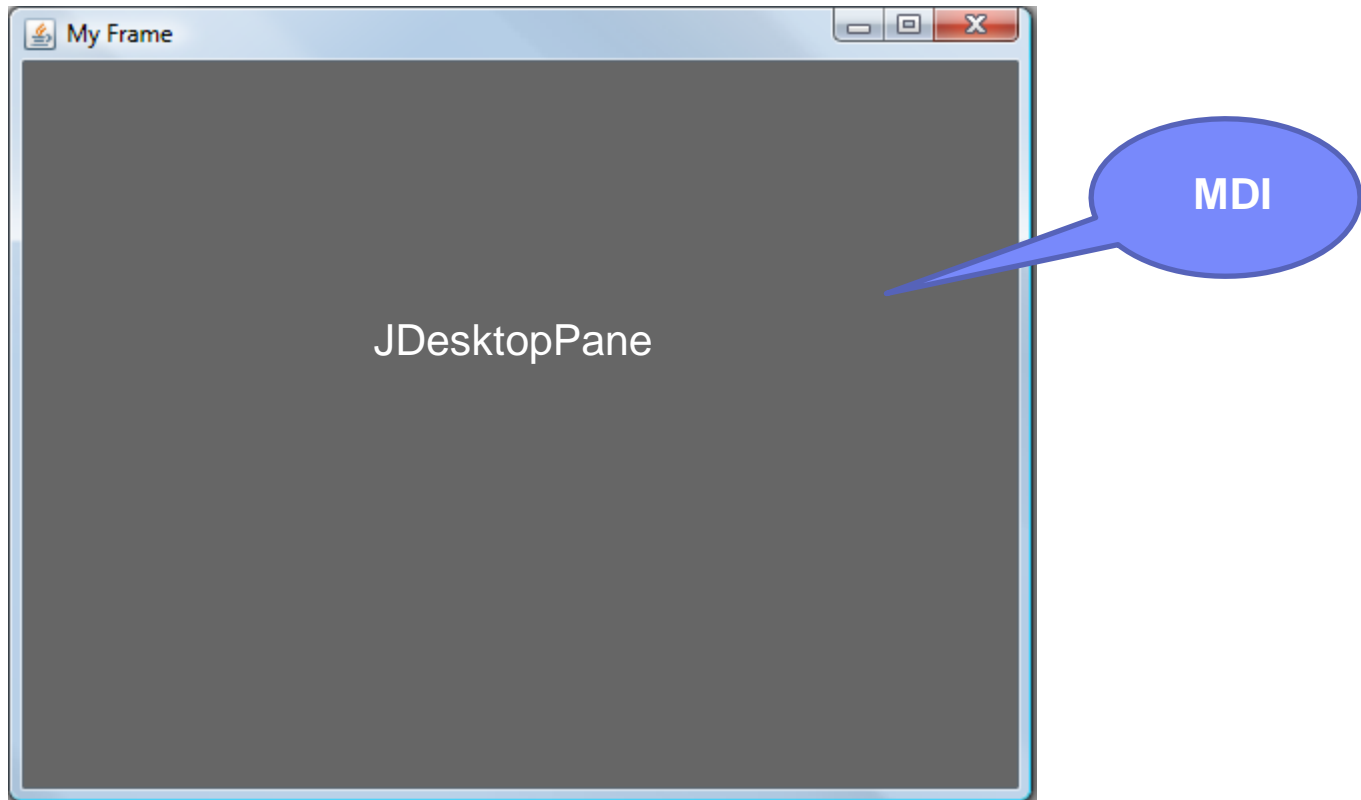
```
public class Main {  
    /**  
     * @param args the command line arguments  
     */  
    public static void main(String[] args) {  
        // TODO code application logic here  
        MyFrame frm=new MyFrame();  
        frm.show();  
    }  
}
```



# Frame

- Methods:
  - setTitle (String title)
  - setAlwaysOnTop (boolean alwaysOnTop)
  - show(), setVisible (boolean b), dispose()
  - setResizable (boolean resizable)
- Events:
  - mouseClicked, mouseEntered, mousePress, ....
  - windowClosed, windowOpened, windowIconified, ...
  - keyPressed, keyTyped, ....

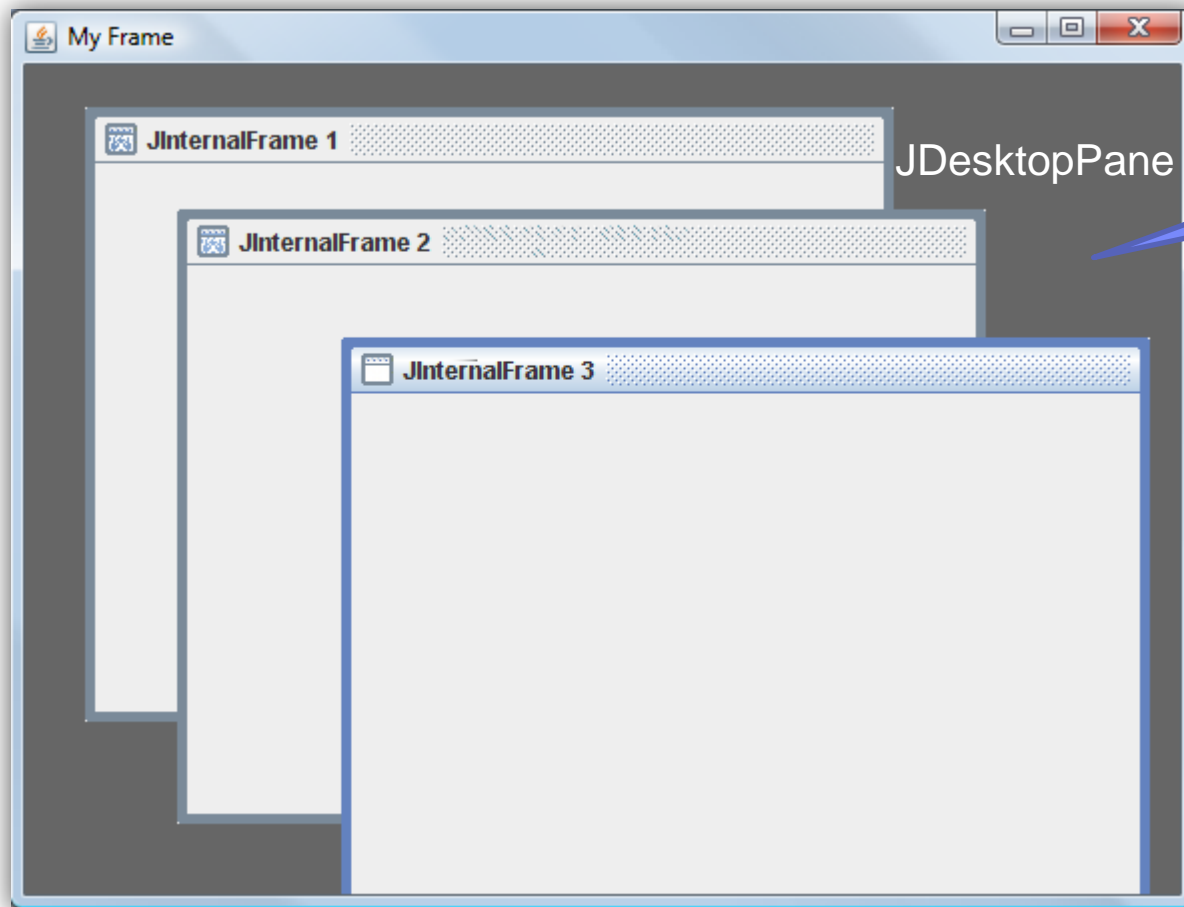
# JDesktopPane



# JDesktopPane

- Methods:
  - **add** (**Component** component)
  - **getAllFrames**()
  - **getSelectedFramed**()
  - **findComponentAt** (**Point** p)
  - **setBackground** (**Color** c)
- Events:
  - **componentAdded**
  - **componentRemoved**

# JInternalFrame



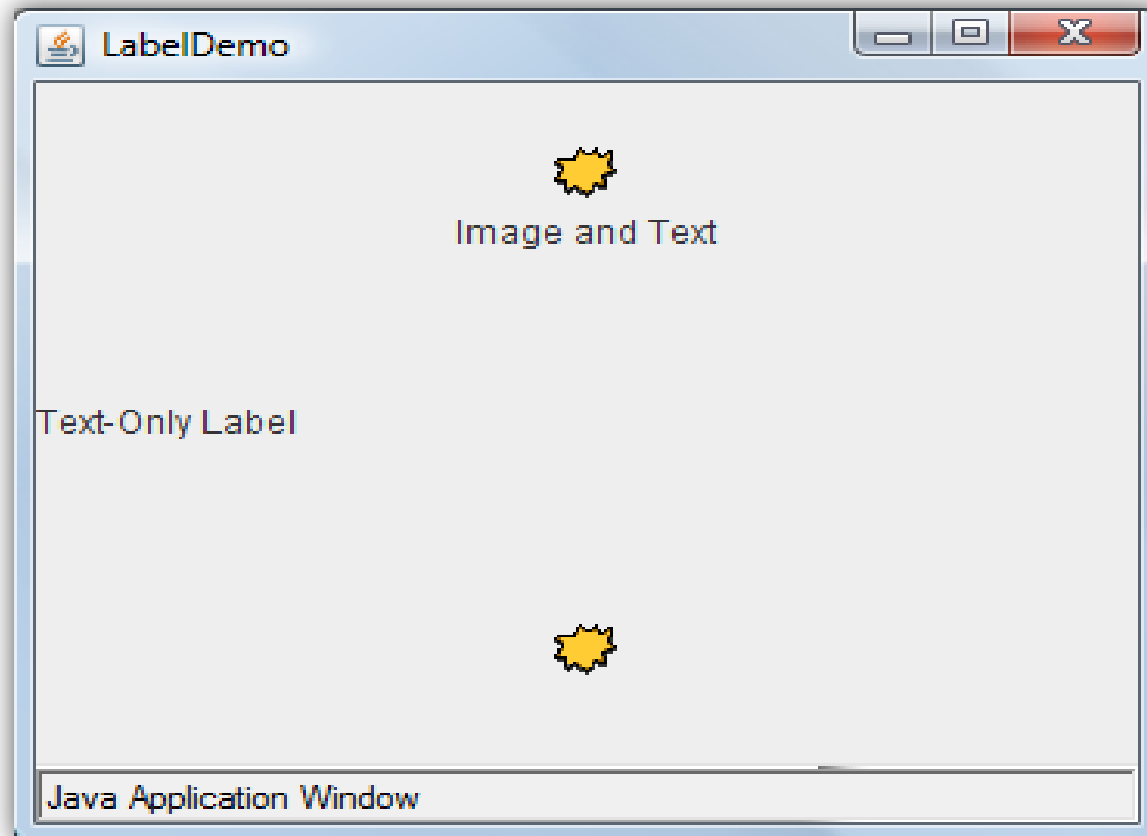
MDI



# JInternalFrame

- Methods:
  - setTitle (String title)
  - show(), setVisible (boolean b), dispose()
  - setClosable (boolean b)
  - setResizable (boolean resizable)
- Events:
  - mouseClicked, mouseEntered, mousePress, ....
  - windowClosed, windowOpened, windowIconified, ...
  - keyPressed, keyTyped, ....

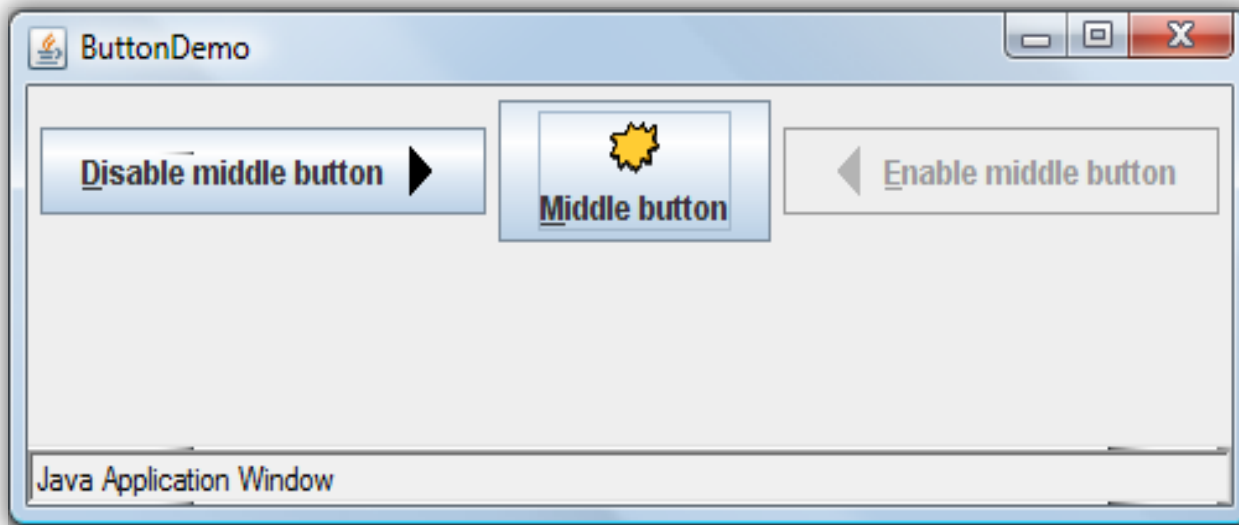
# JLabel



# JLabel

- Methods:
  - `setText (String text)`
  - `getText ()`
  - `setSize (Dimension d)`
  - `setForeground (Color fg)`
- Events:
  - `mouseClicked`

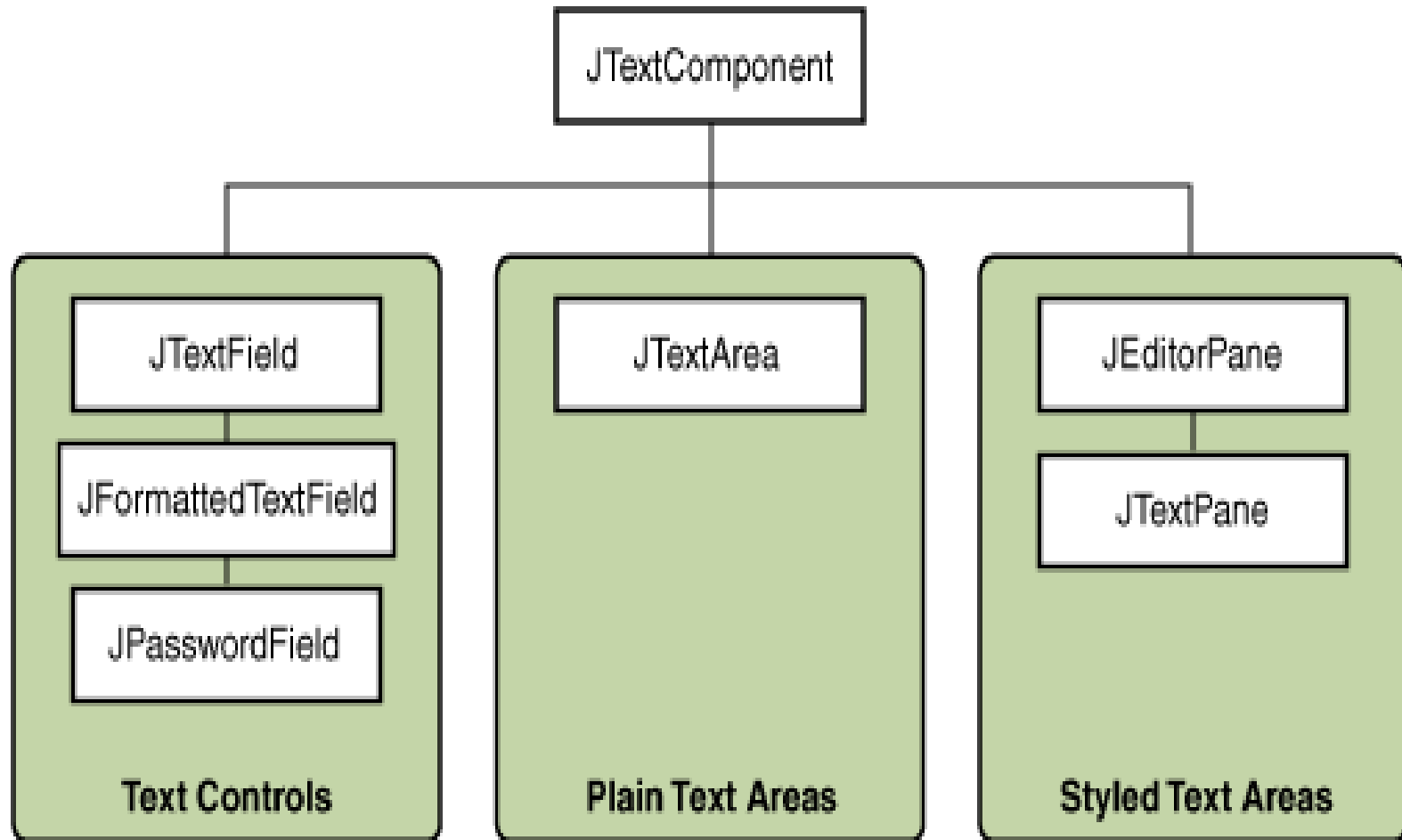
# JButton



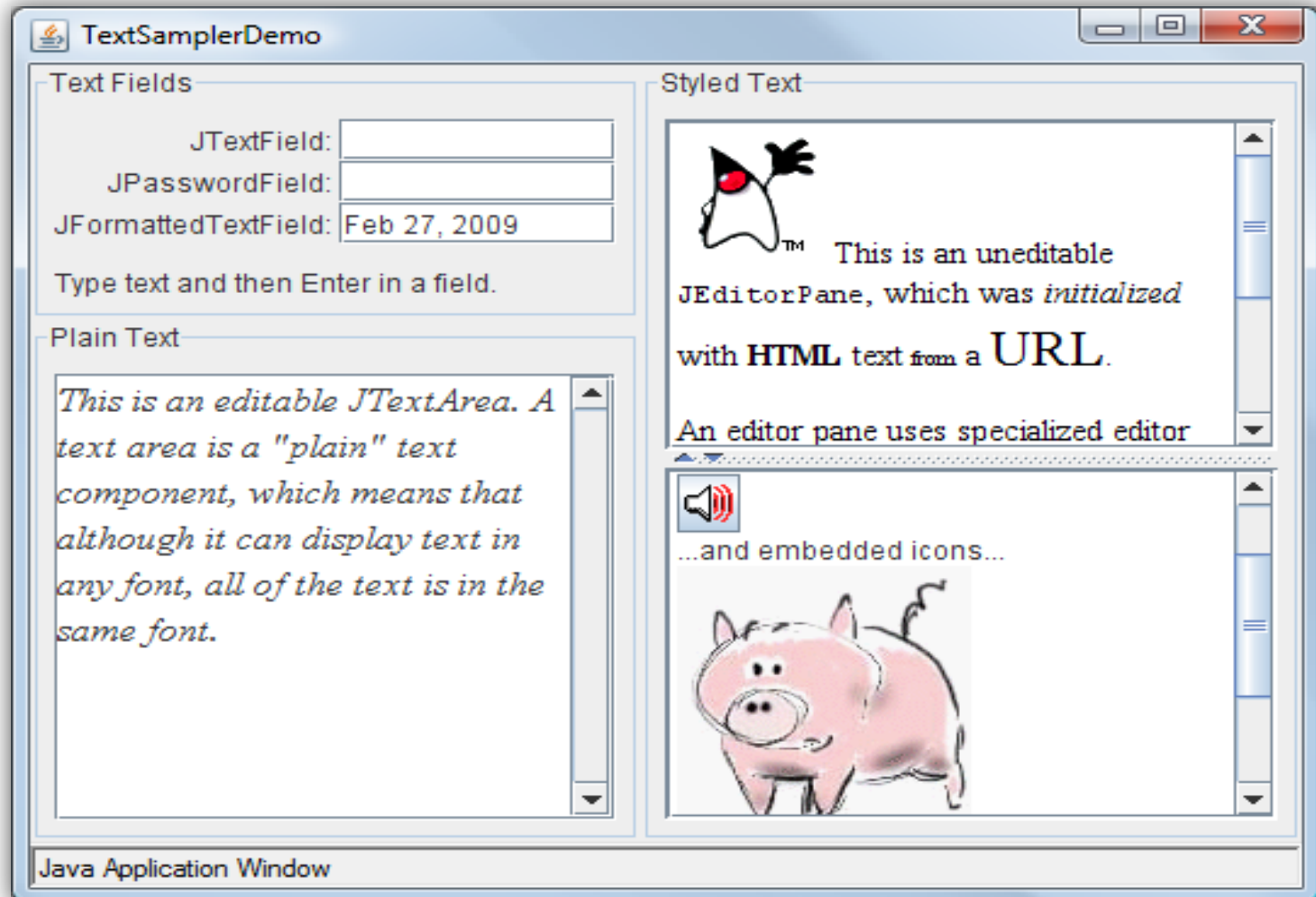
# JButton

- Methods:
  - `setText (String text)`
  - `getText ()`
  - `setForeground (Color fg)`
  - `setFocusCycleRoot (boolean b)`
- Events
  - `actionPerformed`
  - `mousePressed`

# JTextField



# JTextField

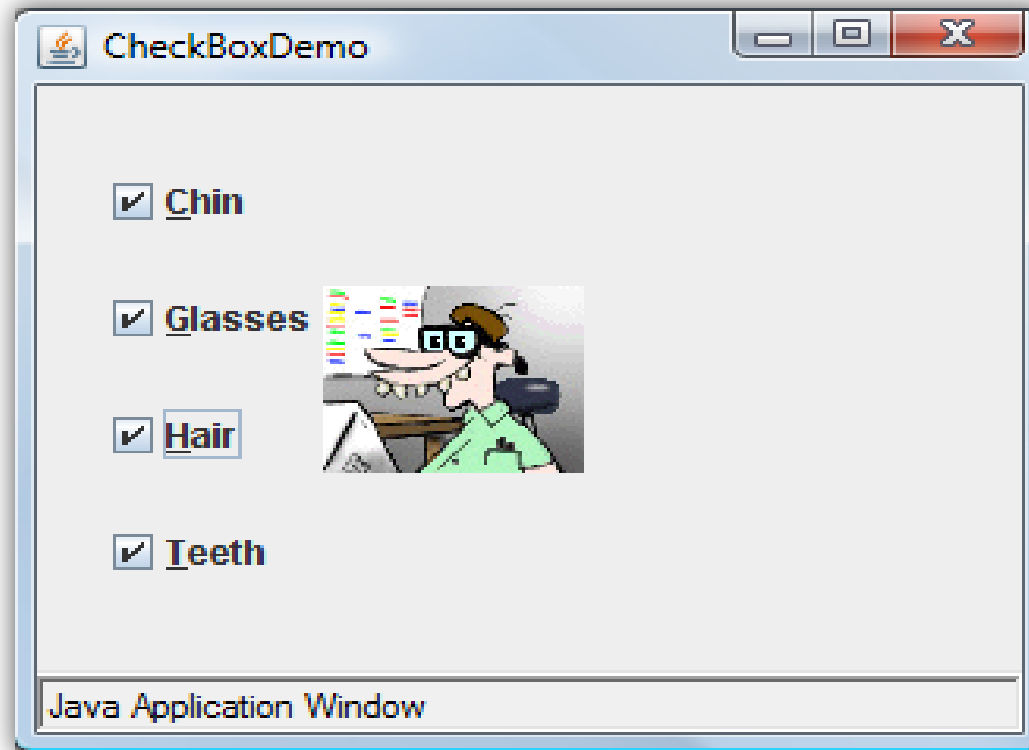


# JTextField

- Methods:
  - `setText (String text)`
  - `getText ()`
  - `setForeground (Color fg)`
- Events:
  - `caretUpdate`



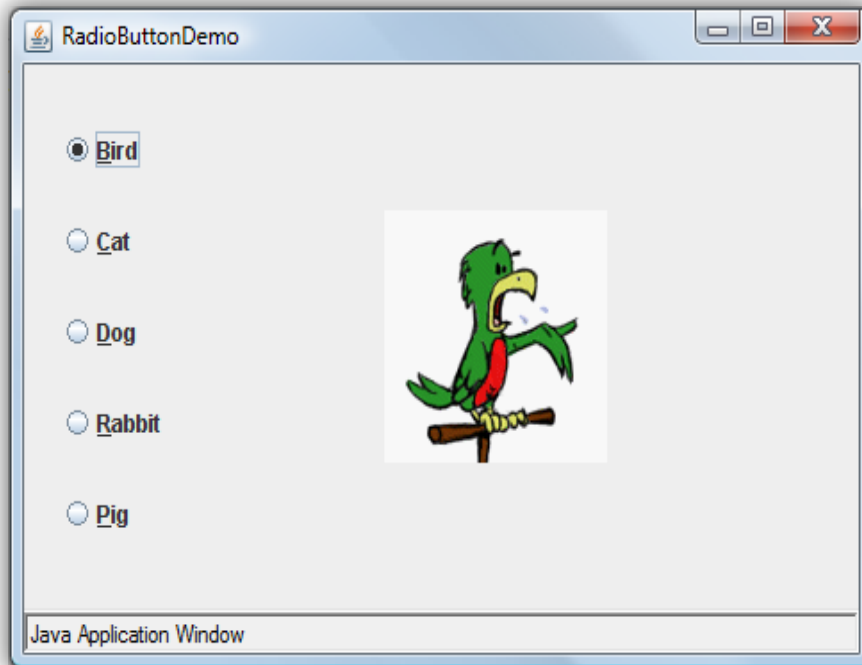
# JCheckBox



# JCheckBox

- Methods:
  - `setSelected (boolean b)`
  - `isSelected ()`
  - `setText (String text)`
  - `getText ()`
- Events:
  - `actionPerformed`

# JRadioButton

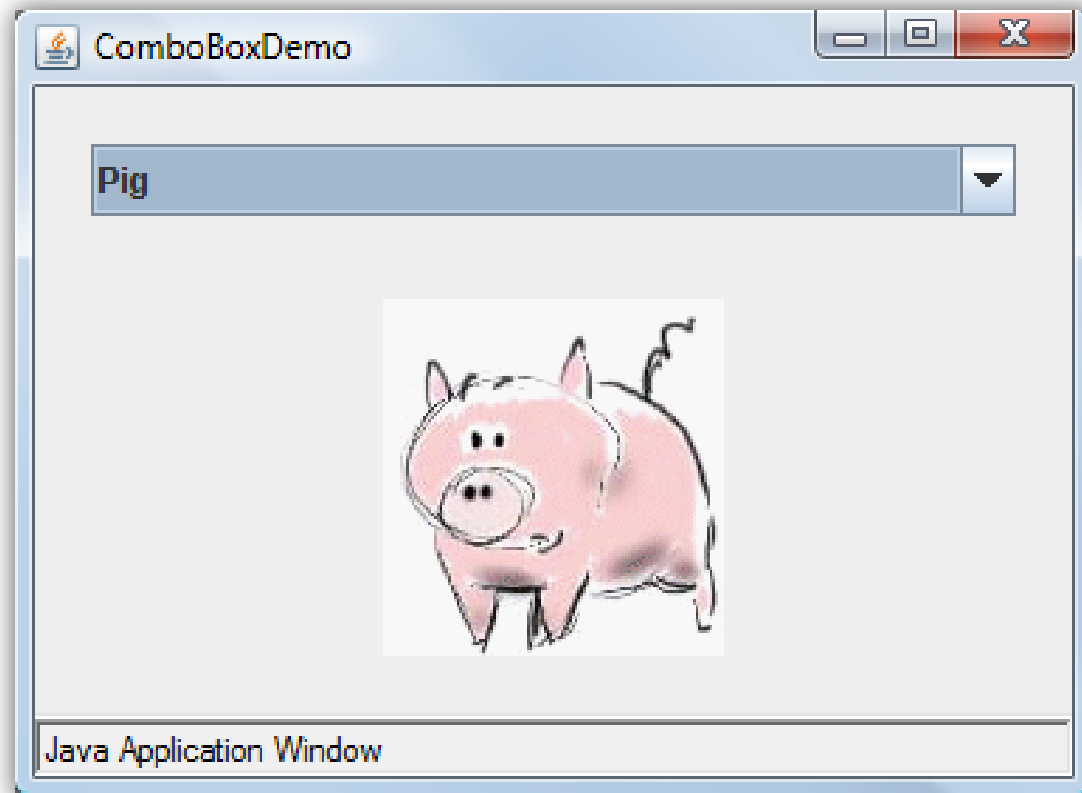


```
ButtonGroup group = new ButtonGroup();  
group.add (birdButton);  
group.add (catButton);  
group.add (dogButton);  
group.add (rabbitButton);  
group.add (pigButton);
```

# JRadioButton

- Methods:
  - `setSelected (boolean b)`
  - `isSelected ()`
  - `setText (String text)`
  - `getText ()`
- Events:
  - `actionPerformed`

# JComboBox



# JComboBox

- DefaultComboBoxModel

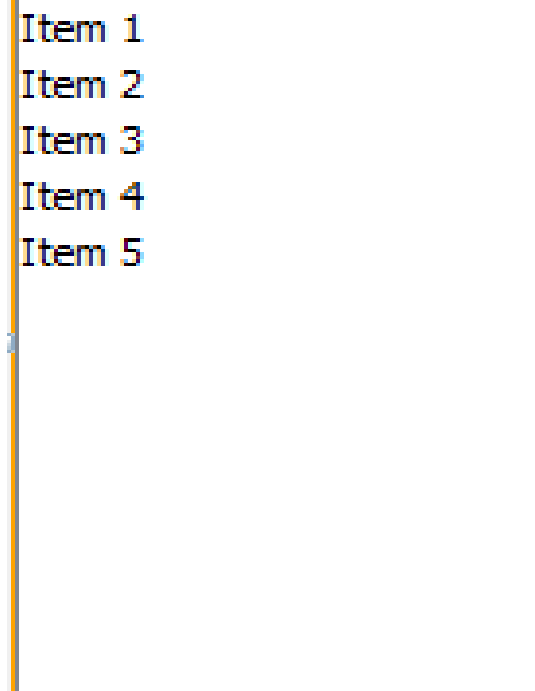
- Methods:

- `addElement (Object o)`
    - `getElementAt (int index)`
    - `getIndexOf (Object item)`
    - `getSelectedItem ()`
    - `removeAllItems ()`
    - `removeItemAt (int index)`
    - `setSelectedItem (int index)`

# JComboBox

- Methods:
  - `setModel (ComboBoxModel model)`
  - `getModel ()`
  - `setSelectedIndex (int index)`
  - `getSelectedItem ()`
  - `getSelectedIndex ()`
  - `getItemCount ()`
- Events:
  - `actionPerformed`

# JList





# JList

- DefaultListModel

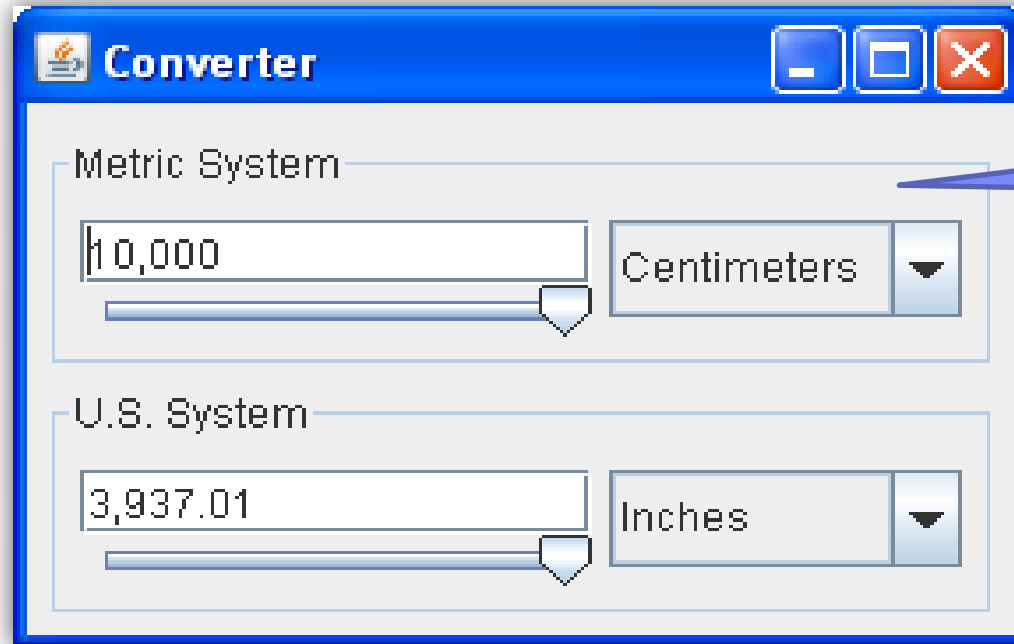
- Methods:

- **addElement** (Object e)
    - **get** (int index)
    - **getSize** ()
    - **getElementAt** (int index)
    - **remove** (int index)
    - **Elements**()
    - **removeAllElements** ()

# JList

- Methods:
  - `setModel (ListModel model), getModel ()`
  - `getMaxSelectionIndex (), getMinSelectionIndex ()`
  - `getSelectedIndex (), getSelectedIndices ()`
  - `getSelectedValue (), getSelectedValues ()`
- Events:
  - `valueChanged`

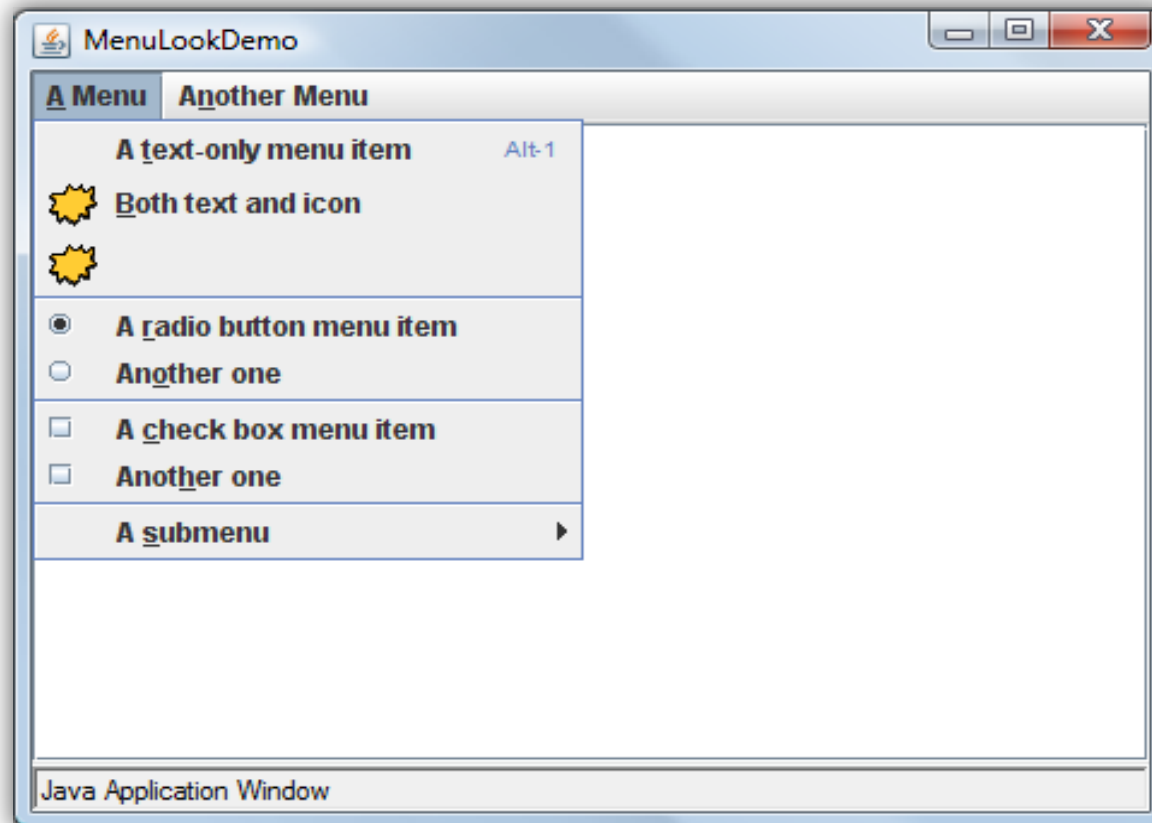
# JPanel



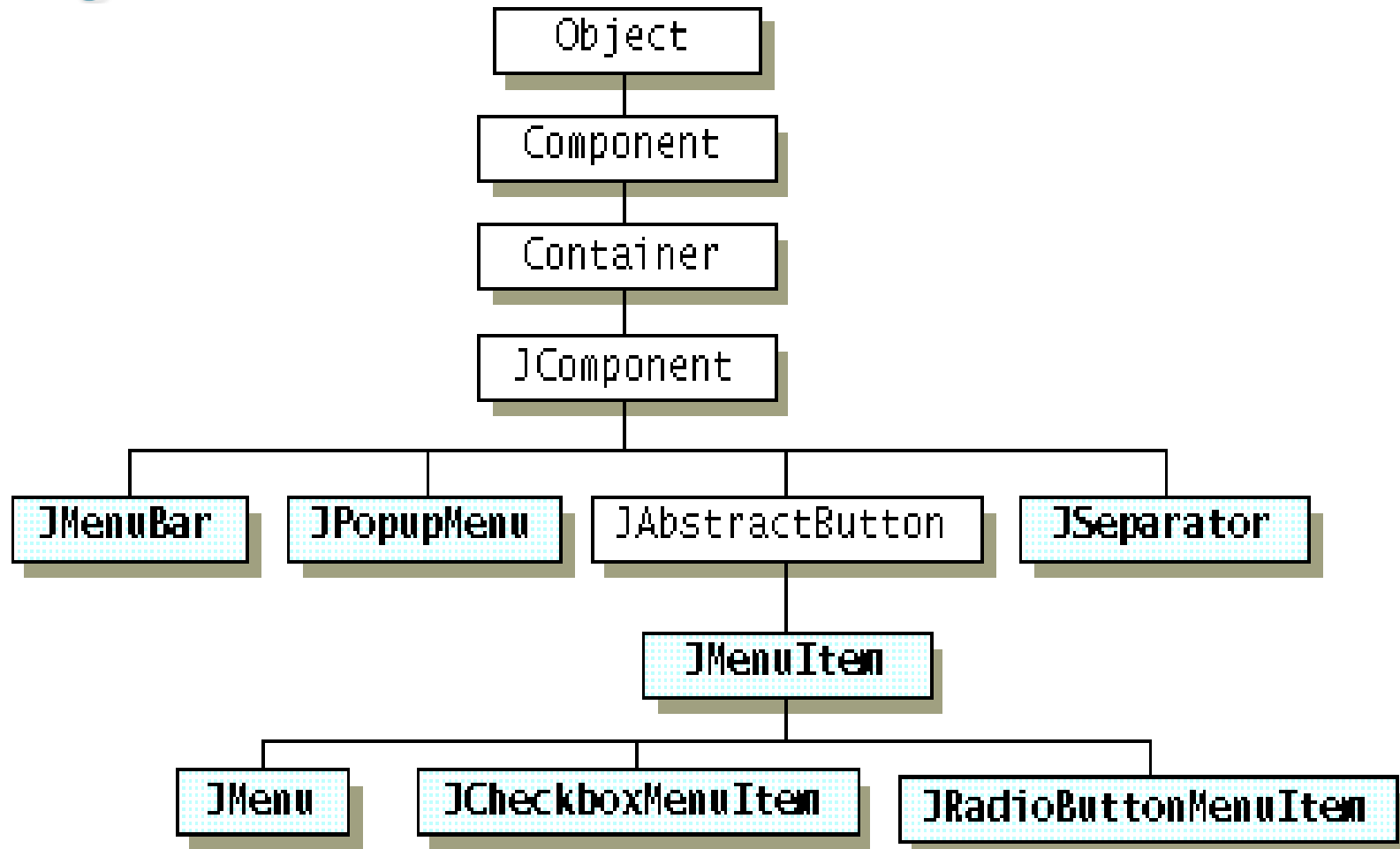
# JPanel

- Methods:
  - **setBorder** (**Border** border)
  - **add** (**Component** comp)
- Events:
  - **mouseClicked**

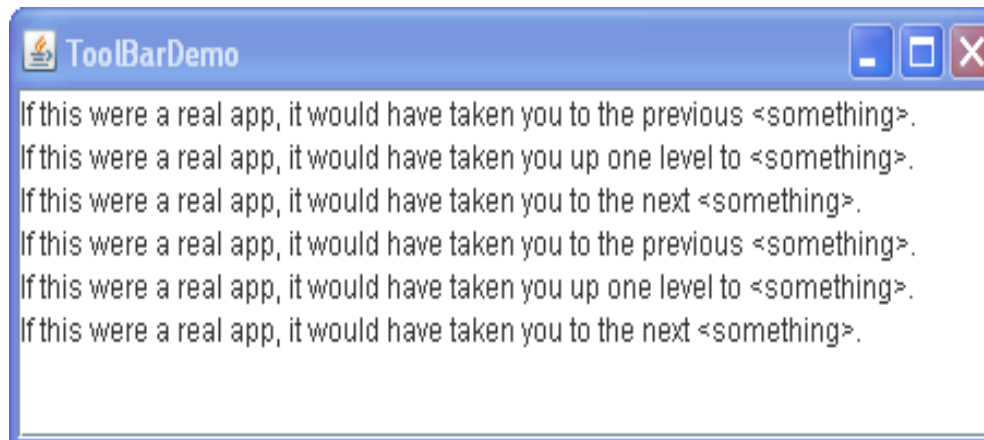
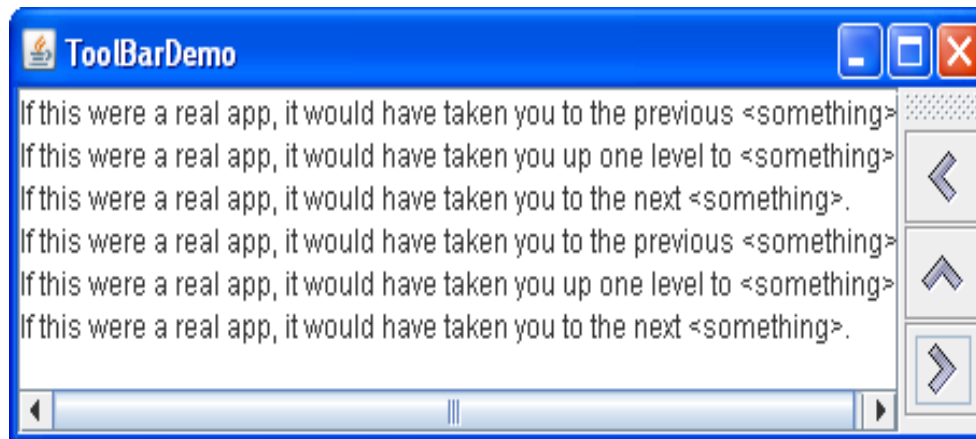
# Swing Menu



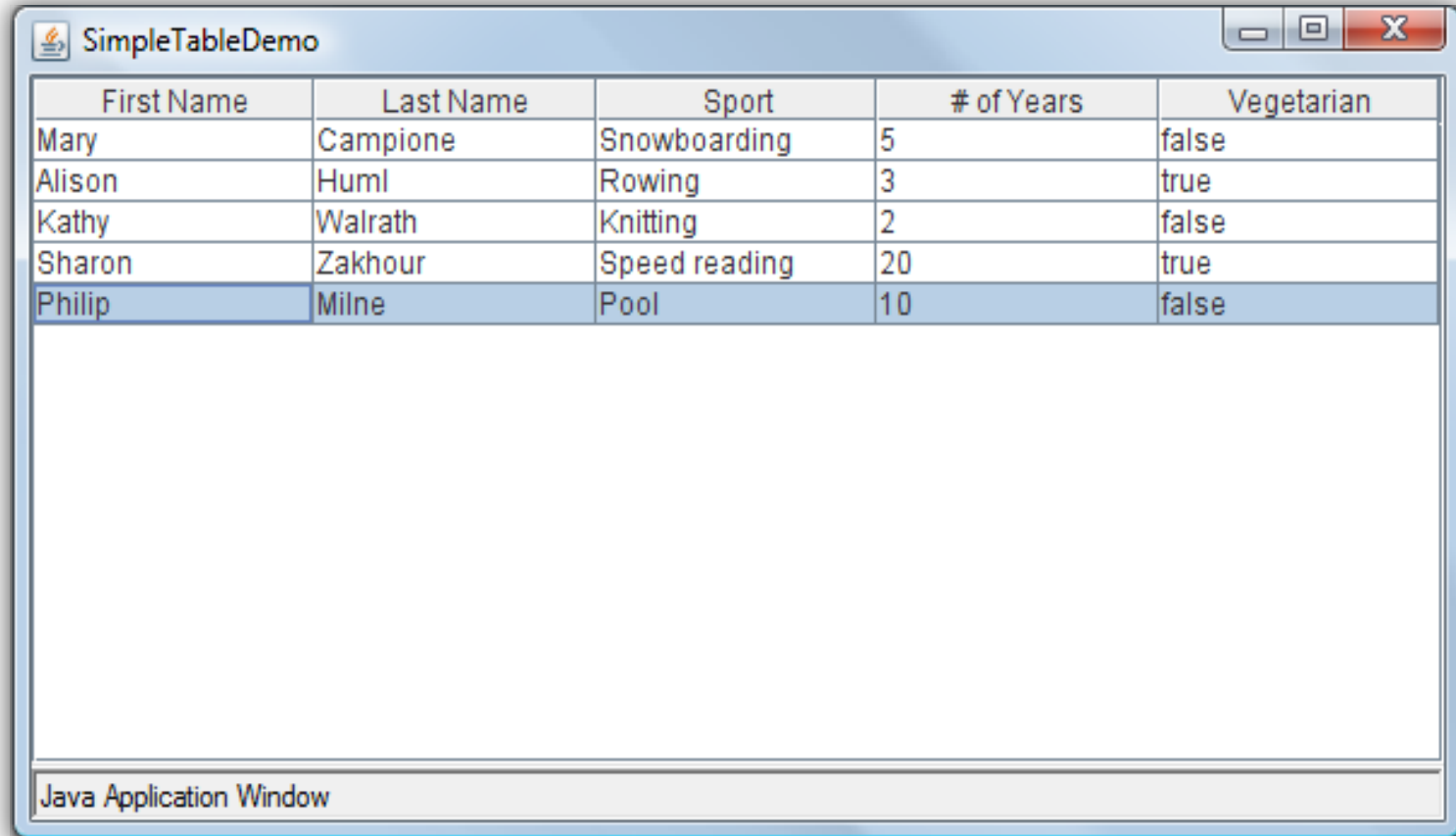
# Swing Menu



# JToolBar



# JTable



SimpleTableDemo

First Name	Last Name	Sport	# of Years	Vegetarian
Mary	Campione	Snowboarding	5	false
Alison	Huml	Rowing	3	true
Kathy	Walrath	Knitting	2	false
Sharon	Zakhour	Speed reading	20	true
Philip	Milne	Pool	10	false

Java Application Window



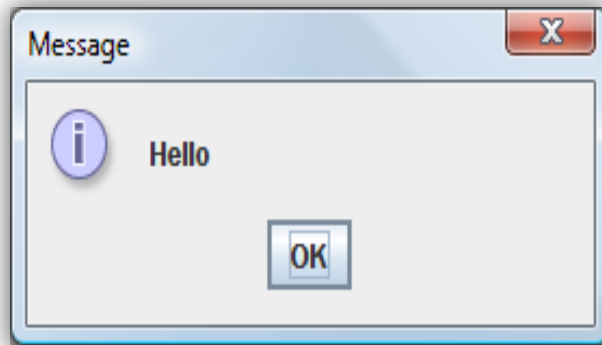
# JTable

- DefaultTableModel
  - addColumn (Object obj)
  - addRow (Object obj)
  - getColumnCount ()
  - getRowCount ()
  - getValueAt (int row, int col)
  - setValueAt (Object obj, int row, int col)

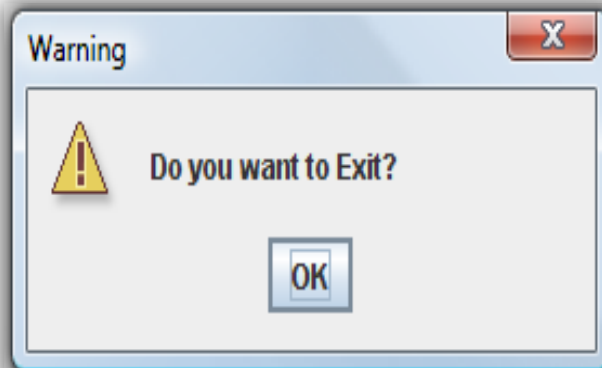
# JTable

- Methods:
  - `setModel (TableModel tm)`
  - `getModel ()`
  - `getValueAt (int row, int col)`
  - `getRowCount ()`
  - `getColumnCount ()`
- Events:
  - `MouseClicked`

# JOptionPane

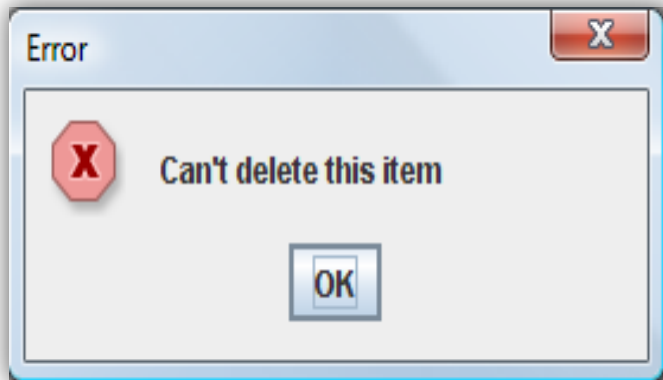


```
JOptionPane.showMessageDialog(frame,  
    "Hello",  
    "Message",  
    JOptionPane.INFORMATION_MESSAGE);
```

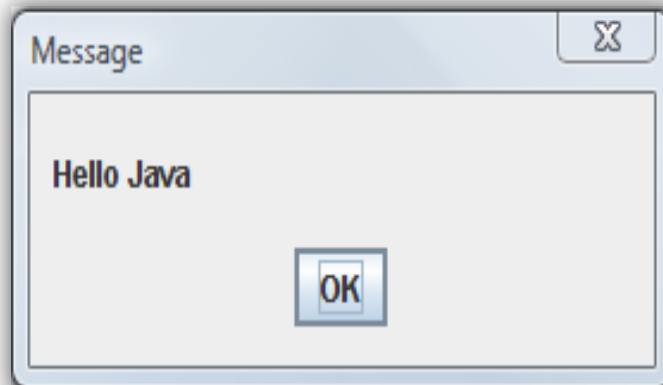


```
JOptionPane.showMessageDialog(frame,  
    "Do you want to Exit",  
    "Warning",  
    JOptionPane.WARNING_MESSAGE);
```

# JOptionPane

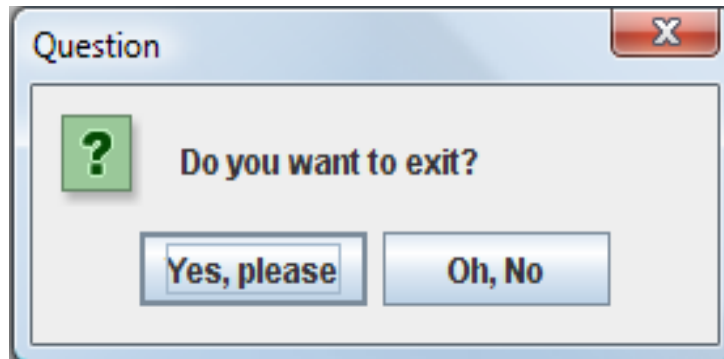


```
JOptionPane.showMessageDialog(frame,  
    "Can't delete this item",  
    "Error",  
    JOptionPane.ERROR_MESSAGE);
```



```
JOptionPane.showMessageDialog(frame,  
    "Hello Java",  
    "Message",  
    JOptionPane.PLAIN_MESSAGE);
```

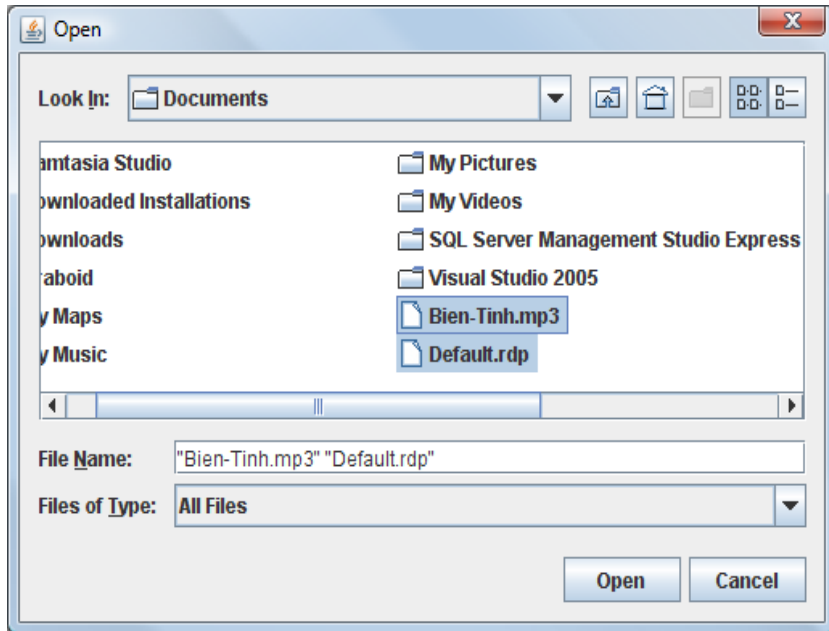
# JOptionPane



```
int n = JOptionPane.showConfirmDialog(frame,
    "Do you want to exit?",
    "Question",
    JOptionPane.YES_NO_OPTION);
if (n == JOptionPane.YES_OPTION) {
    // YES
} else {
    // NO
}
```

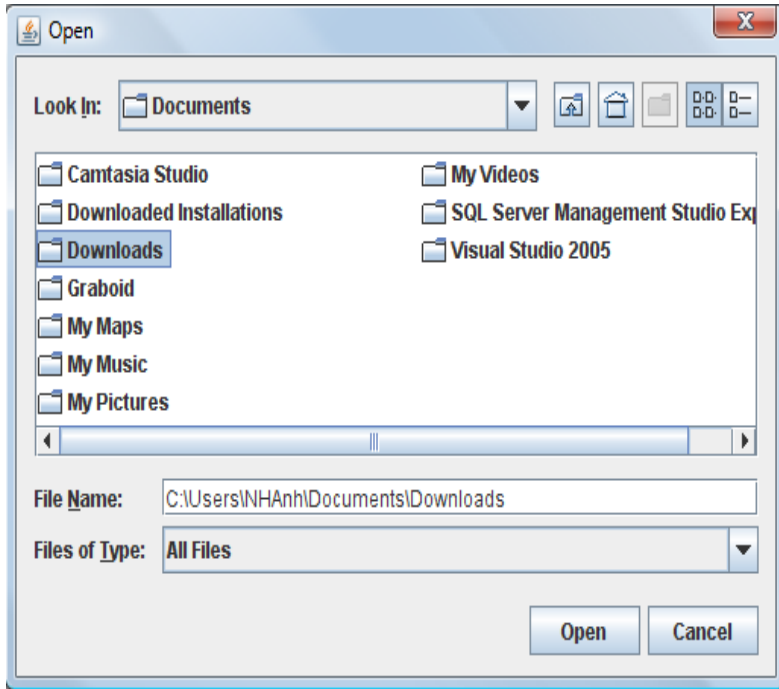
```
Object[] options = {"Yes, please",
    "Oh, No"};
int n = JOptionPane.showOptionDialog(frame,
    "Do you want to exit?",
    "Question",
    JOptionPane.YES_NO_OPTION,
    JOptionPane.QUESTION_MESSAGE,
    null, //do not use a custom Icon
    options, //the titles of buttons
    options[0]); //default button title
if (n == JOptionPane.YES_OPTION) {
    // YES
} else {
    // NO
}
```

# JFileChooser



```
JFileChooser jc = new JFileChooser();
jc.setMultiSelectionEnabled(true);
int n = jc.showOpenDialog(frame); // frame is parent
if (n == jc.OPEN_DIALOG) {
    for (File f : jc.getSelectedFiles()) {
        // To do
    }
}
```

# JFileChooser



```
JFileChooser fc = new JFileChooser();
fc.setFileSelectionMode(JFileChooser.DIRECTORIES_ONLY);

int n = fc.showOpenDialog(frame); // frame is parent
if (n == JFileChooser.OPEN_DIALOG) {
    for (File f : fc.getSelectedFiles()) {
        if (f.isDirectory()) {
            // To do
        }
    }
}
```

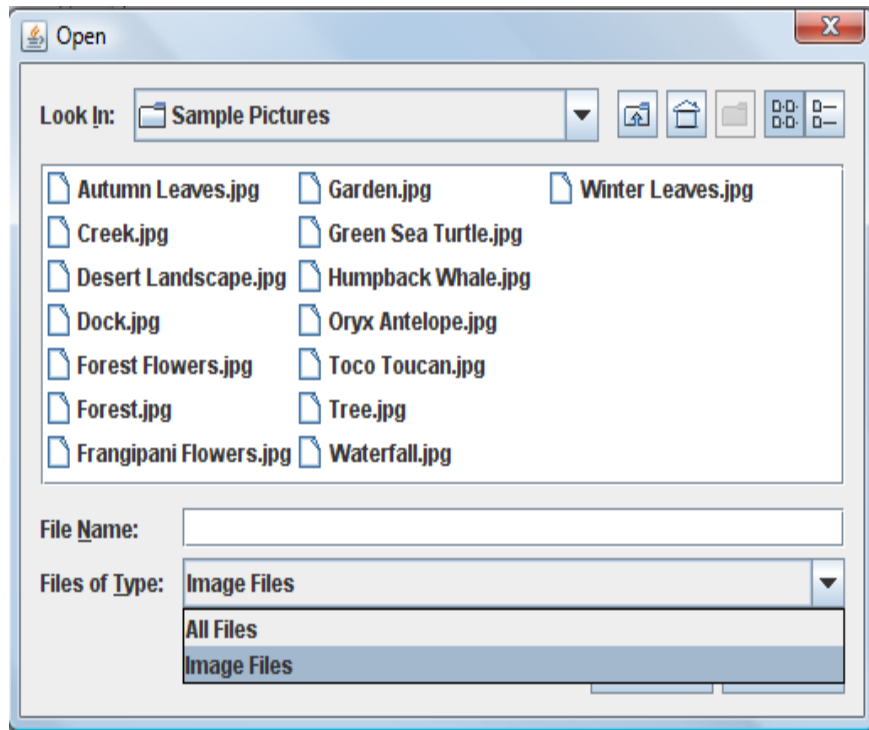
# JFileChooser

```
public class MyImageFilter extends FileFilter {
    @Override
    public boolean accept(File f) {
        if (f.isDirectory()) {
            return true;
        }
        String[] s = f.getName().split("\\.");

        if (s.length > 0) {
            String extension = s[1];
            if (extension != null) {
                if (extension.equals("gif") || extension.equals("bmp") ||
                    extension.equals("jpg") || extension.equals("png")) {
                    return true;
                } else {
                    return false;
                }
            }
        }
        return false;
    }
    @Override
    public String getDescription() {
        return "Image Files";
    }
}
```



# JFileChooser



```
JFileChooser fc = new JFileChooser();
fc.setMultiSelectionEnabled(true);
fc.addChoosableFileFilter(new MyImageFilter());
int n = fc.showOpenDialog(this); // this is a frame
if (n == JFileChooser.OPEN_DIALOG) {
    for (File f : fc.getSelectedFiles()) {
        if (f.isDirectory()) {
            // To do
        }
    }
}
```

# Tham khảo

- <http://java.sun.com/docs/books/tutorial/uiswing/components/index.html>