Sri SAI RAM ENGINEERING COLLEGE
INSTITUTE OF TECHNOLOGY
West Tambaram, Chennai - 44

SAIRAM
DIGITAL RESOURCES

UNIT NO 5

EVENT DRIVEN PROGRAMMING

5.9 Windows – Menus-Dialog Boxes

YEAR	SEM
II	III

CS8392

OBJECT ORIENTED PROGRAMMING
(Common to CSE, EEE, EIE, ICE, IT)

COMPUTER SCIENCE & ENGINEERING

Java Swing class Hierarchy-Windows
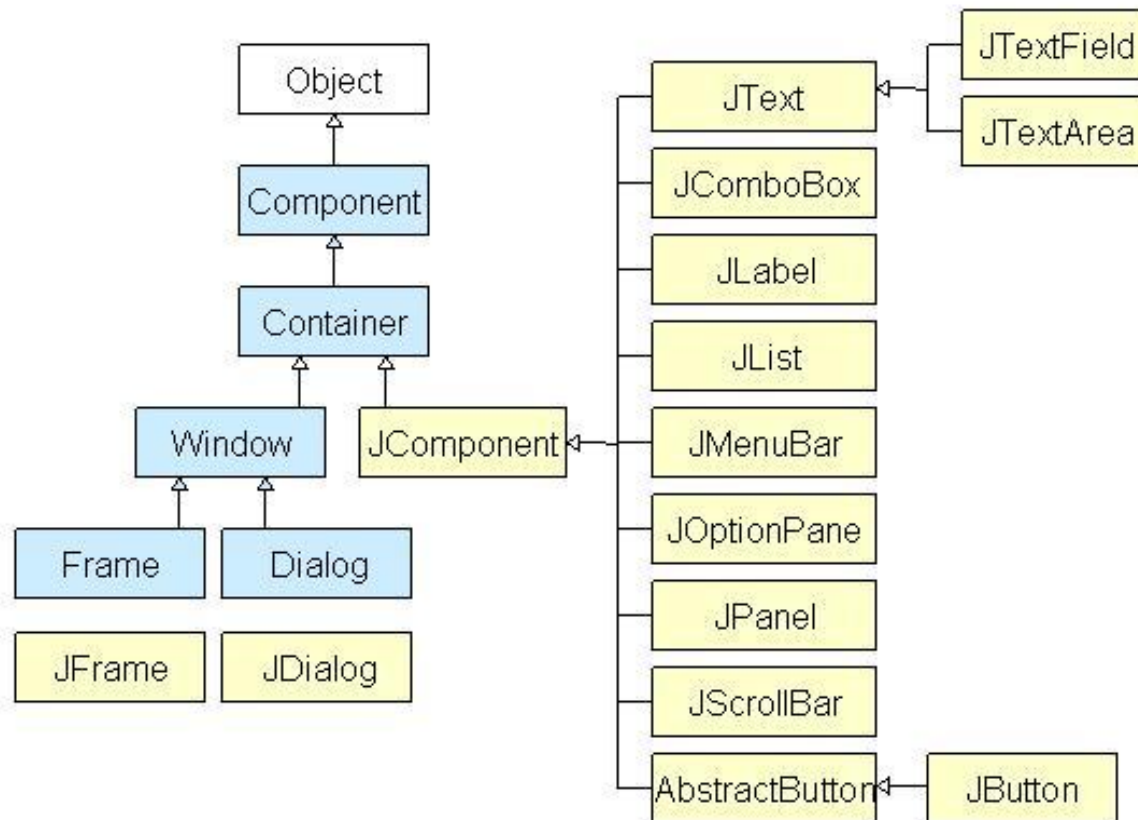
Windows

Java JFrame

JFrame is a class of javax.swing package extended by java.awt.frame, it adds support for JFC/SWING component architecture. It is the top level window, with border and a title bar. **JFrame** class has many methods which can be used to customize it. JFrame works like the main window where components like labels, buttons, textfields are added to create a GUI. Unlike Frame, **JFrame** has the option to hide or close the window with the help of setDefaultCloseOperation(int) method.

Creating Window using JFrame

JFrame class has many constructors used to create a JFrame. Following is the description.

JFrame(): creates a frame which is invisible

JFrame(GraphicsConfiguration gc): creates a frame with a blank title and graphics configuration of screen device.

JFrame(String title): creates a JFrame with a title.

JFrame(String title, GraphicsConfiguration gc): creates a JFrame with specific Graphics configuration and specified title.

Windows-Example

JFrame Example

```java
import java.awt.FlowLayout;
import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.Jpanel;
public class JFrameExample {
    public static void main(String s[]) {
        JFrame frame = new JFrame("JFrame Example");
        JPanel panel = new JPanel();
        panel.setLayout(new FlowLayout());
        JLabel label = new JLabel("JFrame By Example");
        JButton button = new JButton();
        button.setText("Button");
        panel.add(label);
        panel.add(button);
        frame.add(panel);
        frame.setSize(200, 300);
        frame.setLocationRelativeTo(null);
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.setVisible(true);    } }
```
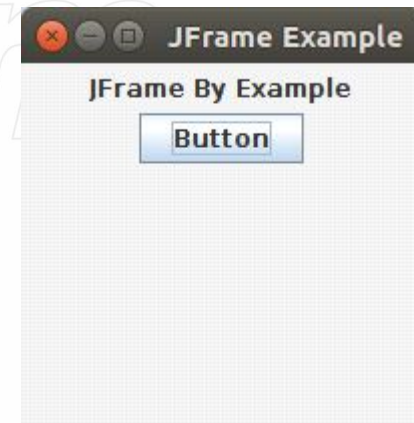
Menus

The **JMenuBar** class is used to display menubar on the window or frame. It may have several menus.

The object of **JMenu class** is a pull down menu component which is displayed from the menu bar. It inherits the JMenuItem class.

The object of **JMenuItem class** adds a simple labeled menu item. The items used in a menu must belong to the JMenuItem or any of its subclass.

JMenuBar class declaration

public class JMenuBar **extends** JComponent **implements** MenuElement, Accessible

JMenu class declaration

public class JMenu **extends** JMenuItem **implements** MenuElement, Accessible

JMenuItem class declaration

public class JMenuItem **extends** AbstractButton **implements** Accessible, MenuElement

Java JMenuItem and JMenu Example

```java
import javax.swing.*;
class MenuExample
{       JMenu menu, submenu;
        JMenuItem i1, i2, i3, i4, i5;
        MenuExample(){
        JFrame f= new JFrame("Menu and MenuItem Example");
        JMenuBar mb=new JMenuBar();
        menu=new JMenu("Menu");
        submenu=new JMenu("Sub Menu");
        i1=new JMenuItem("Item 1");
        i2=new JMenuItem("Item 2");
        i3=new JMenuItem("Item 3");
        i4=new JMenuItem("Item 4");
        i5=new JMenuItem("Item 5");
        menu.add(i1); menu.add(i2); menu.add(i3);
        submenu.add(i4); submenu.add(i5);  menu.add(submenu);
        mb.add(menu);  f.setJMenuBar(mb);
        f.setSize(400,400);  f.setLayout(null);  f.setVisible(true);
}
public static void main(String args[])
{   new MenuExample();  }
}
```
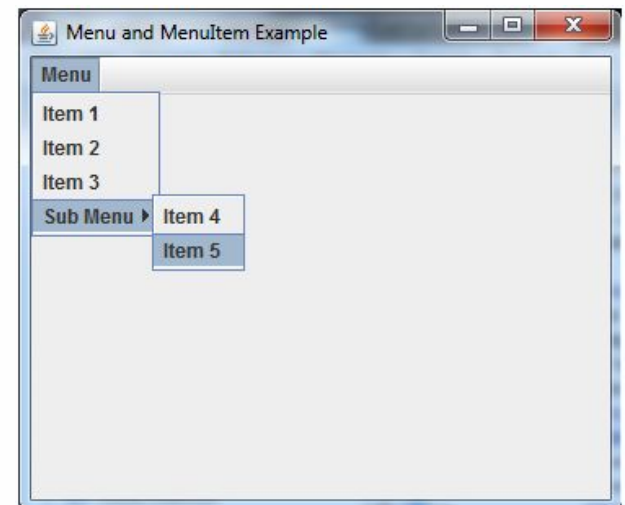
Menu Bar

```java
import javax.swing.*;
import java.awt.event.*;
public class MenuExample implements ActionListener{
JFrame f;
JMenuBar mb;
JMenu file,edit,help;
JMenuItem cut,copy,paste,selectAll;
JTextArea ta;
MenuExample(){
f=new JFrame();
cut=new JMenuItem("cut");
copy=new JMenuItem("copy");
paste=new JMenuItem("paste");
selectAll=new JMenuItem("selectAll");
cut.addActionListener(this);
copy.addActionListener(this);
paste.addActionListener(this);
selectAll.addActionListener(this);
mb=new JMenuBar();
file=new JMenu("File");
edit=new JMenu("Edit");
help=new JMenu("Help");
```

Menu Bar

```java
edit.add(cut);edit.add(copy);edit.add(paste);edit.add(selectAll);
mb.add(file);mb.add(edit);mb.add(help);
ta=new JTextArea();
ta.setBounds(5,5,360,320);
f.add(mb);f.add(ta);
f.setJMenuBar(mb);
f.setLayout(null);
f.setSize(400,400);
f.setVisible(true);
}
public void actionPerformed(ActionEvent e) {
if(e.getSource()==cut)
ta.cut();
if(e.getSource()==paste)
ta.paste();
if(e.getSource()==copy)
ta.copy();
if(e.getSource()==selectAll)
ta.selectAll();
}
public static void main(String[] args) {
    new MenuExample();    } }
```
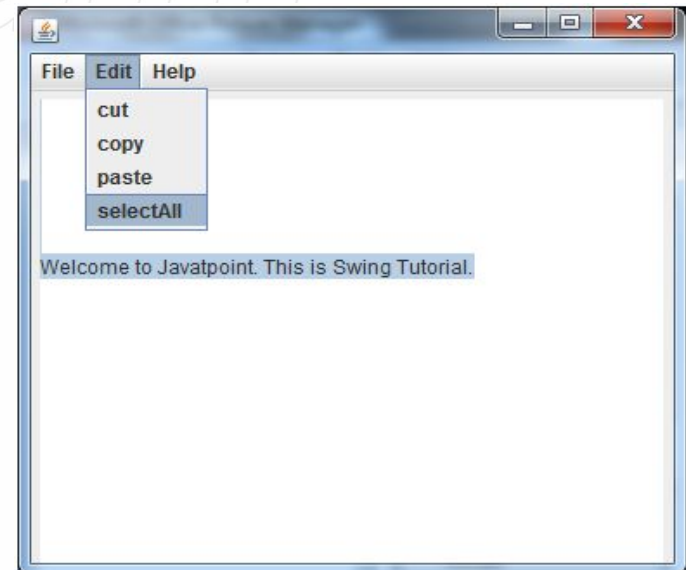
Dialog Boxes

The **JDialog** control represents a top level window with a border and a title used to take some form of input from the user. It inherits the Dialog class.
Unlike JFrame, it doesn't have maximize and minimize buttons.

Java JDialog Example
```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
public class DialogExample {
    private static JDialog d;
    DialogExample() {
        JFrame f= new JFrame();
        d = new JDialog(f , "Dialog Example", true);
        d.setLayout( new FlowLayout() );
        JButton b = new JButton ("OK");
        b.addActionListener ( new ActionListener()
        {
            public void actionPerformed( ActionEvent e )
            {
                DialogExample.d.setVisible(false);
            }
        });
```

Dialog Boxes

```
d.add( new JLabel ("Click button to continue."));
     d.add(b);
     d.setSize(300,300);
     d.setVisible(true);
   }
   public static void main(String args[])
   {
     new DialogExample();
   }
}
```

Video Links

Java Window - https://www.youtube.com/watch?v=djpTYzcZ3kY

MenuBar,Menu,MenuItem- https://www.youtube.com/watch?v=THgJ3oe7HQk

Dialog box- https://www.youtube.com/watch?v=uoh3WJF9V1o