



# Lecture 43

# Swing Programming - I





IIT KHARAGPUR



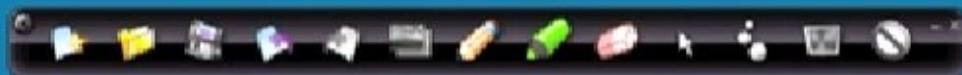
NPTEL ONLINE  
CERTIFICATION COURSES

# OBJECT ORIENTED PROGRAMMING WITH JAVA

## Java Swing Programming – I

**Debasis Samanta**

Department of Computer Science & Engineering  
Indian Institute of Technology Kharagpur

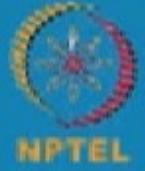




# Java Swing Basics



IIT KHARAGPUR



NPTEL  
NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR



# What is Java Swing?

- **Java Swing** is a part of **Java Foundation Classes** (JFC) that is used to create window-based applications. It is built on the top of **AWT** (Abstract Windowing Toolkit) API and entirely written in **Java**.
- Java Swing provides better **lightweight** components than AWT.
- The **javax.swing** package provides classes for Java Swing components such as JButton, JTextField, JTextArea, JRadioButton, JCheckbox, JMenu, JColorChooser etc.



IIT KHARAGPUR



NPTEL  
NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR



# AWT versus Swing

## Java AWT

AWT components are platform-dependent.

AWT components are heavyweight.

AWT doesn't support pluggable look and feel.

AWT provides less components than Swing.

AWT doesn't follows MVC(Model View Controller) where model represents data, view represents presentation and controller acts as an interface between model and view.

## Java Swing

Java swing components are platform-independent.

Swing components are lightweight.

Swing supports pluggable look and feel.

Swing provides more powerful components such as tables, lists, scrollpanes, colorchooser, tabbedpane, etc.

Swing follows MVC.



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR



# What is JFC?

The **Java Foundation Classes** (JFC) are a set of GUI components which simplify the development of desktop applications.



IIT KHARAGPUR



NPTEL  
NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



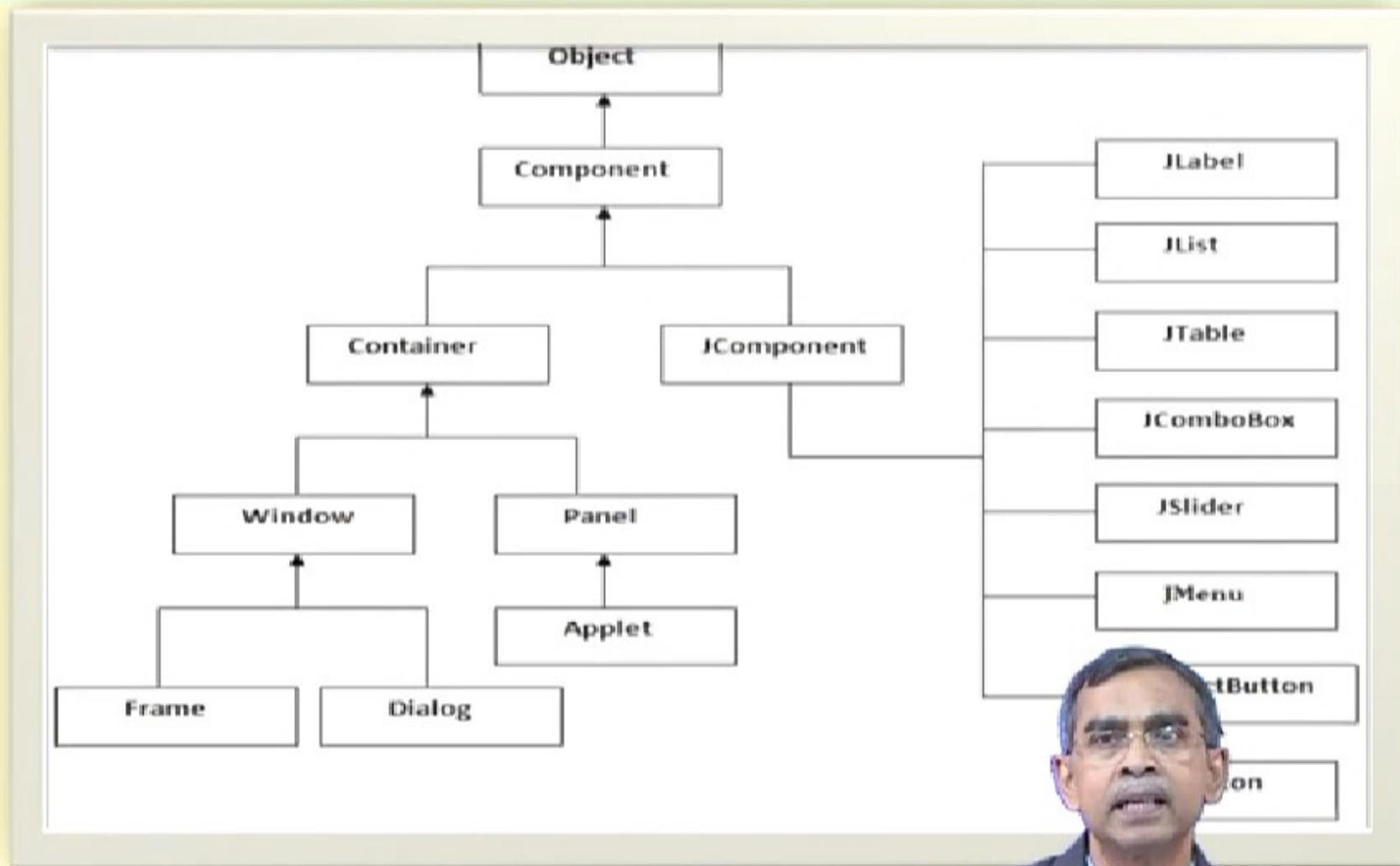
IIT KHARAGPUR





# Hierarchy of Java Swing classes

The hierarchy of Java Swing APIs

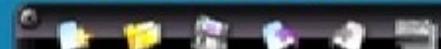


IIT KHARAGPUR

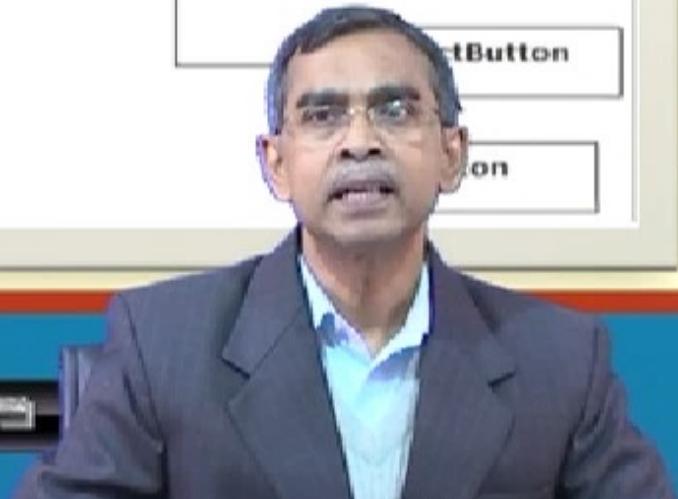


NPTEL  
ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



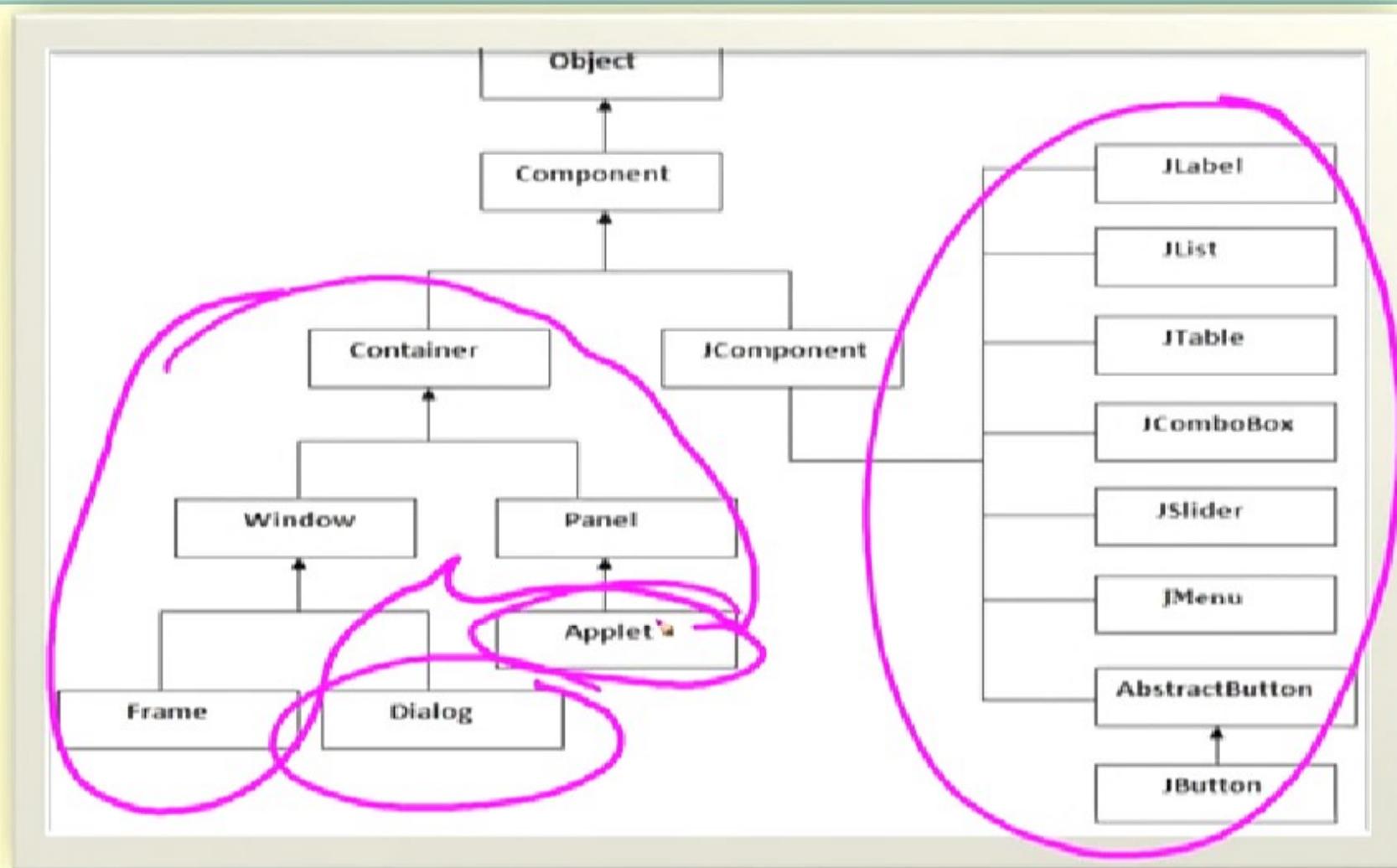
IIT KHARAGPUR





# Hierarchy of Java Swing classes

The hierarchy of Java Swing APIs



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA





# Commonly used methods in Component class

The methods of **Component** class are widely used in java swing that are given below.

<i>Method</i>	<i>Description</i>
public void add(Component c)	Add a component on another component.
public void setSize(int width,int height)	Sets size of the component.
public void setLayout(LayoutManager m)	Sets the layout manager for the component.
public void setVisible(boolean b)	Sets the visibility of the component. It is by default false.





# Java Swing examples

There are two ways to create a frame:

1. By creating the object of `Frame` class (**Association**)
2. By extending `Frame` class (**Inheritance**)

We can write the code of Swing inside the `main()`, constructor or any other method.



# Simple Java Swing : An example

Let's see a simple swing example where we are creating **one button** and adding it on the **JFrame** object inside the **main ()** method.

```
import javax.swing.*;  
  
public class FirstSwingExample {  
    public static void main(String[] args) {  
        //creating instance of JFrame  
        JFrame f = new JFrame();  
        //creating Instance of JButton  
        JButton b = new JButton("click");  
        //x axis, y axis, width, height  
        b.setBounds(130,100,100, 40);  
        f.add(b);//adding button in JFrame  
        f.setSize(400,500);//400 width and 500 height  
        f.setLayout(null);//using no layout managers  
        f.setVisible(true);//making the frame visible  
    }  
}
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



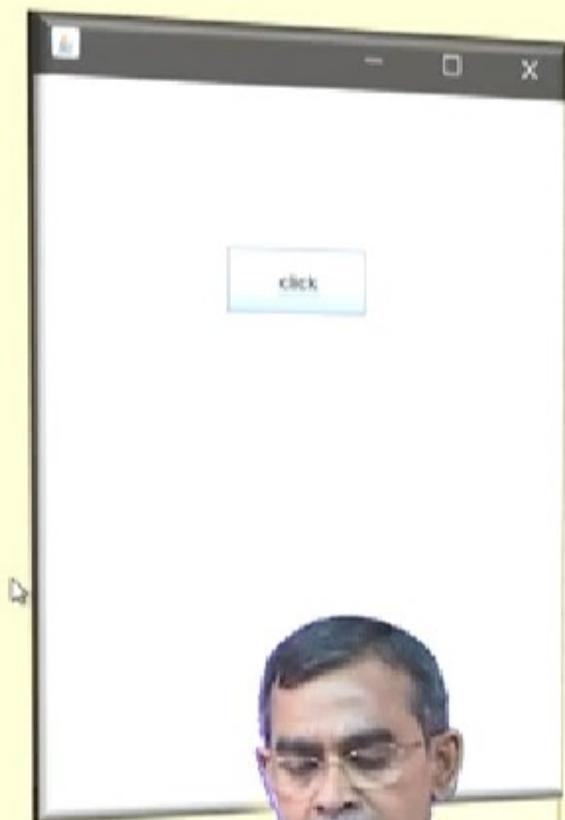
IIT KHARAGPUR



# Swing by association inside constructor

We can also write all the codes of creating `JFrame`, `JButton` and method call inside the `Java` constructor.

```
import javax.swing.*;
public class Simple {
    JFrame f;
    Simple() {
        f = new JFrame();
        JButton b = new JButton("click");
        b.setBounds(130, 100, 100, 40);
        f.add(b);
        f.setSize(400, 500);
        f.setLayout(null);
        f.setVisible(true);
    }
    public static void main(String[] args) {
        new Simple();
    }
}
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



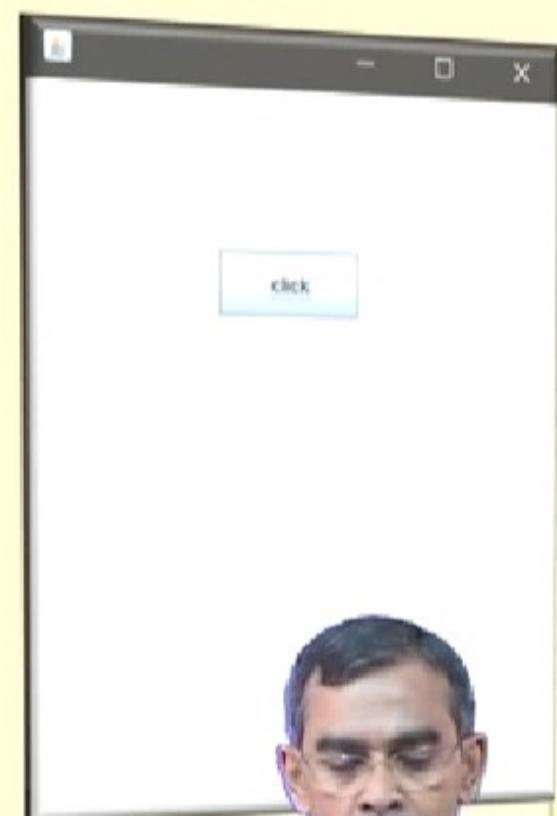
IIT KHARAGPUR



# Swing by inheritance

We can also inherit the `JFrame` class, so there is no need to create the instance of `JFrame` class explicitly.

```
import javax.swing.*;
//inheriting JFrame
public class Simple2 extends JFrame{
    Simple2(){
        JButton b = new JButton("click");
        b.setBounds(130,100,100, 40);
        add(b);
        setSize(400,500);
        setLayout(null);
        setVisible(true);
    }
    public static void main(String[] args) {
        new Simple2();
    }
}
```

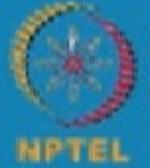




# Java Swing Components



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



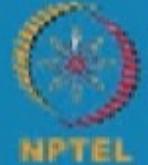
IIT KHARAGPUR



# Java Swing JButton



IIT KHARAGPUR



NPTEL  
NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA

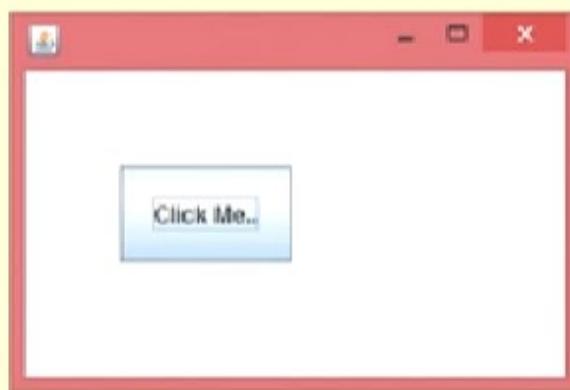


IIT KHARAGPUR



# Class JButton

The **JButton** class is used to create a labeled button that has platform independent implementation. The application result in some action when the button is pushed. It inherits **AbstractButton** class.



Below is the declaration for `javax.swing.JButton` class.

```
public class JButton extends AbstractButton implements Accessible
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR



# Class JButton : Constructors

<i>Constructor</i>	<i>Description</i>
JButton()	It creates a button with no text and icon.
JButton(String s)	It creates a button with the specified text.
JButton(Icon i)	It creates a button with the specified icon object.



IIT KHARAGPUR



NPTEL  
NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR





# Class JButton : Methods

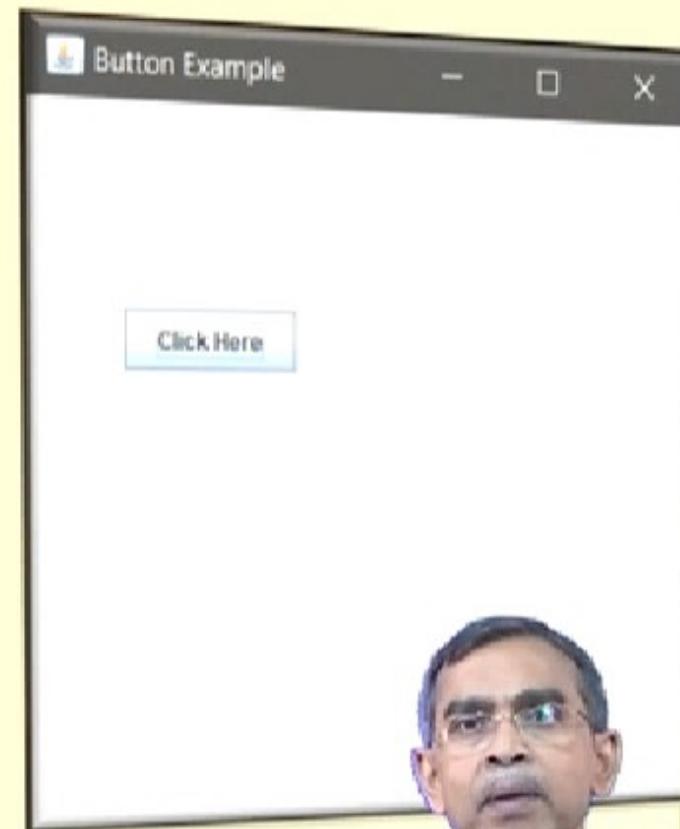
Methods	Description
void setText(String s)	It is used to set specified text on button
String getText()	It is used to return the text of the button.
void setEnabled(boolean b)	It is used to enable or disable the button.
void setIcon(Icon b)	It is used to set the specified Icon on the button.
Icon getIcon()	It is used to get the Icon of the button.
void setMnemonic(int a)	It is used to set the mnemonic on the button.
void addActionListener(ActionListener a)	It is used to add the <u>action listener</u> to this object



# Creating a JButton : An example

Let's see a simple swing example where we are creating **one button** and adding it on the **JFrame** object inside the **main()** method.

```
import javax.swing.*;
public class ButtonExample {
public static void main(String[] args) {
    JFrame f=new JFrame("Button Example");
    JButton b=new JButton("Click Here");
    b.setBounds(50,100,95,30);
    f.add(b);
    f.setSize(400,400);
    f.setLayout(null);
    f.setVisible(true);
}
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA

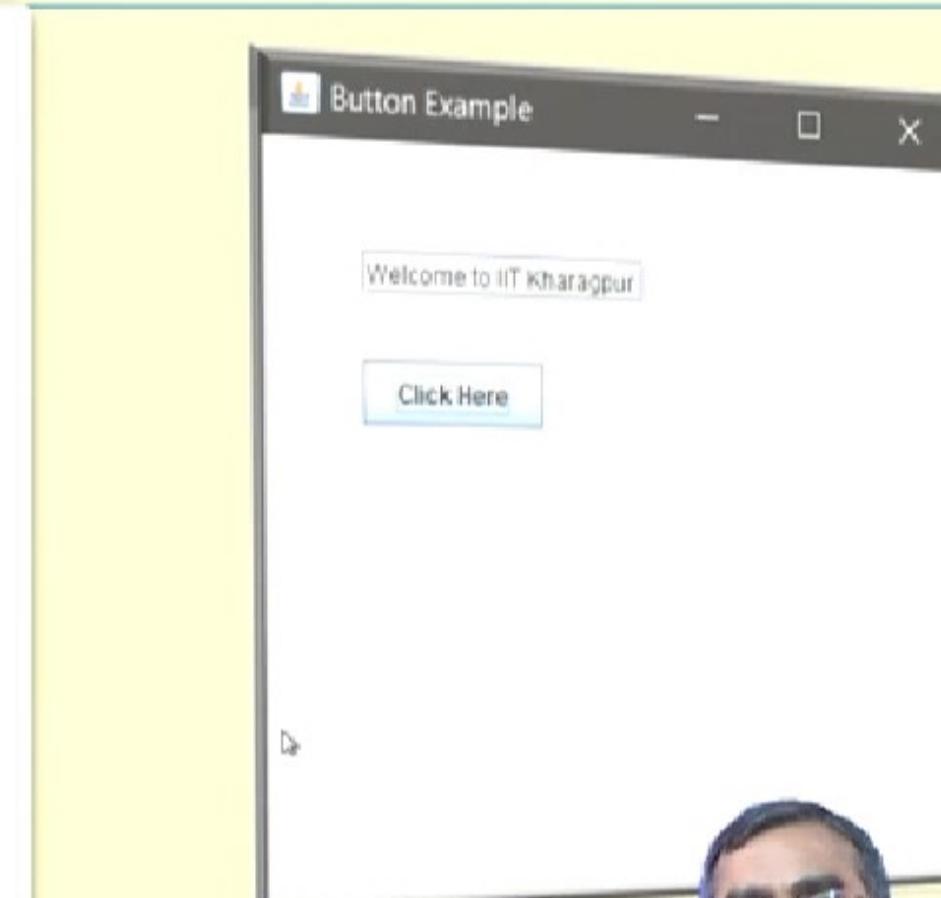


IIT KHARAGPUR



# Java JButton : An example with ActionListener

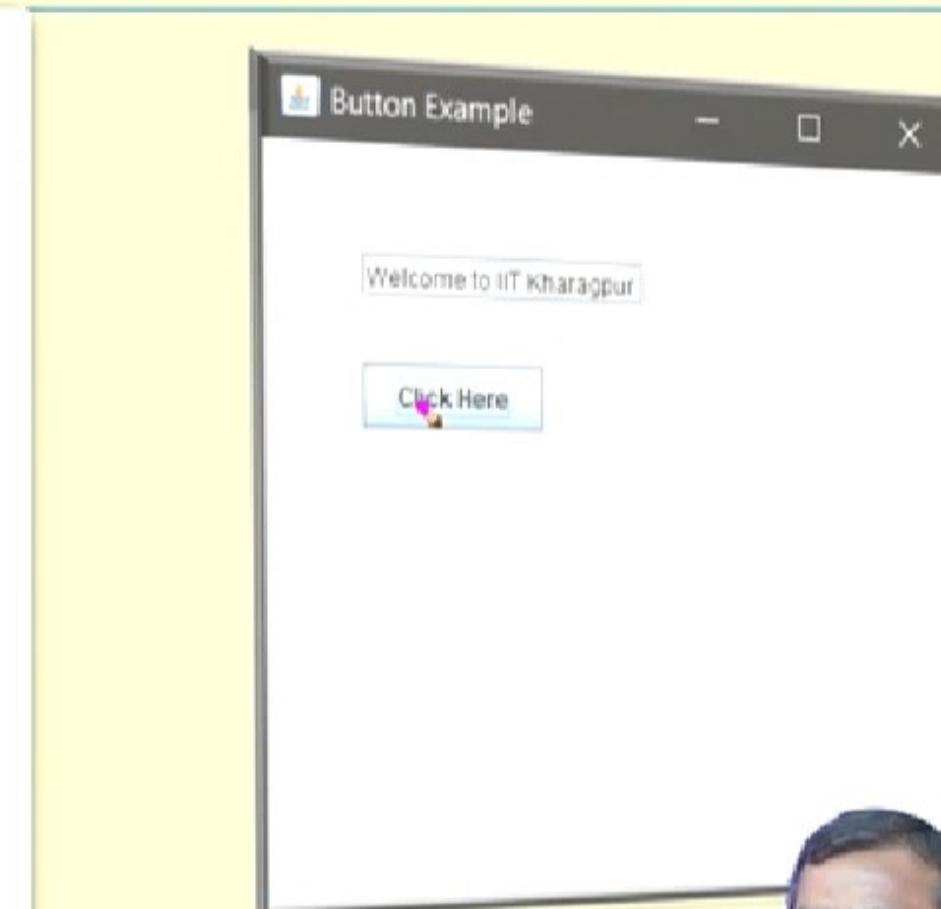
```
import java.awt.event.*;
import javax.swing.*;
public class ButtonExample {
public static void main(String[] args) {
    JFrame f=new JFrame("Button Example");
    final JTextField tf=new JTextField();
    tf.setBounds(50,50, 150,20);
    JButton b=new JButton("Click Here");
    b.setBounds(50,100,95,30);
    b.addActionListener(new ActionListener(){
        public void actionPerformed(ActionEvent e){
            tf.setText("Welcome to IIT Kharagpur.");
        }
    });
    f.add(b);f.add(tf);
    f.setSize(400,400);
    f.setLayout(null);
    f.setVisible(true);
}
```





# Java JButton : An example with ActionListener

```
import java.awt.event.*;
import javax.swing.*;
public class ButtonExample {
public static void main(String[] args) {
    JFrame f=new JFrame("Button Example");
    final JTextField tf=new JTextField();
    tf.setBounds(50,50, 150,20);
    JButton b=new JButton("Click Here");
    b.setBounds(50,100,95,30);
    b.addActionListener(new ActionListener(){
        public void actionPerformed(ActionEvent e){
            tf.setText("Welcome to IIT Kharagpur.");
        }
    });
    f.add(b);f.add(tf);
    f.setSize(400,400);
    f.setLayout(null);
    f.setVisible(true);
}
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

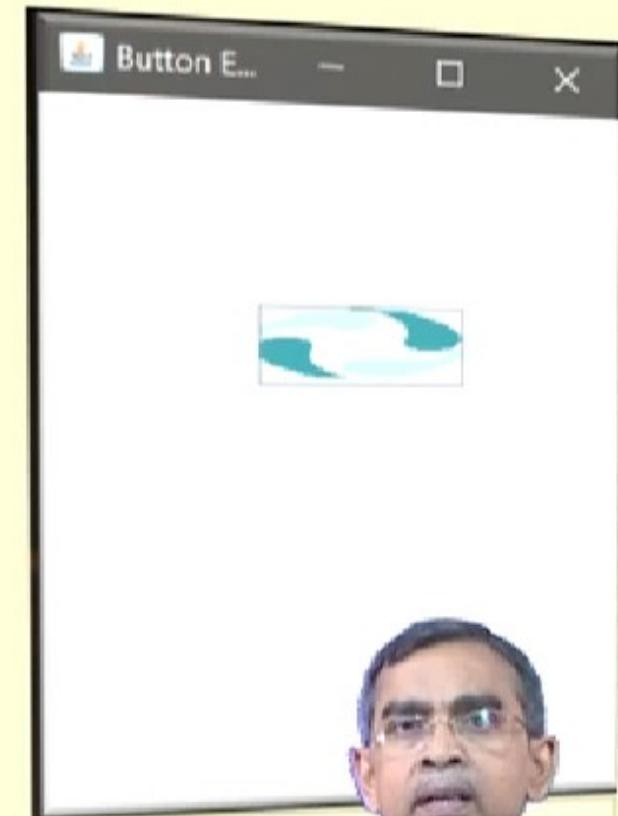
DEBASIS SAMANTA





# Java JButton : Displaying image on the button

```
import javax.swing.*;
public class ButtonExample{
ButtonExample(){
    JFrame f=new JFrame("Button Example");
    JButton b=new JButton(new ImageIcon("D:\\icon.png"));
    b.setBounds(100,100,100, 40);
    f.add(b);
    f.setSize(300,400);
    f.setLayout(null);
    f.setVisible(true);
    f.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
}
public static void main(String[] args) {
    new ButtonExample();
}
}
```



IIT KHARAGPUR



NPTEL  
NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR

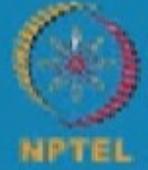




# Java Swing JLabel



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR





# Class JLabel

The object of **JLabel** class is a component for placing text in a container. It is used to display a single line of read only text. The text can be changed by an application but a user cannot edit it directly. It inherits **JComponent** class.

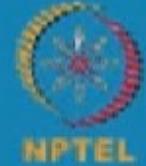


Below is the declaration for **javax.swing.JLabel** class.

```
public class JLabel extends JComponent implements SwingConstants, Accessible
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR

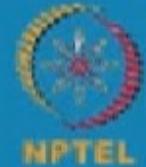


# Class JLabel : Constructors

<i>Constructor</i>	<i>Description</i>
JLabel()	Creates a JLabel instance with no image and with an empty string for the title.
JLabel(String s)	Creates a JLabel instance with the specified text.
JLabel(Icon i)	Creates a JLabel instance with the specified image.
JLabel(String s, Icon i, int horizontalAlignment)	Creates a JLabel instance with the specified text, image, and horizontal alignment.



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR





# Java JLabel methods

Methods	Description
<code>String getText()</code>	It returns the text string that a label displays.
<code>void setText(String text)</code>	It defines the single line of text this component will display.
<code>void setHorizontalAlignment(int alignment)</code>	It sets the alignment of the label's contents along the X axis.
<code>Icon getIcon()</code>	It returns the graphic image that the label displays.
<code>int getHorizontalAlignment()</code>	It returns the alignment of the label's contents along the X axis.



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA

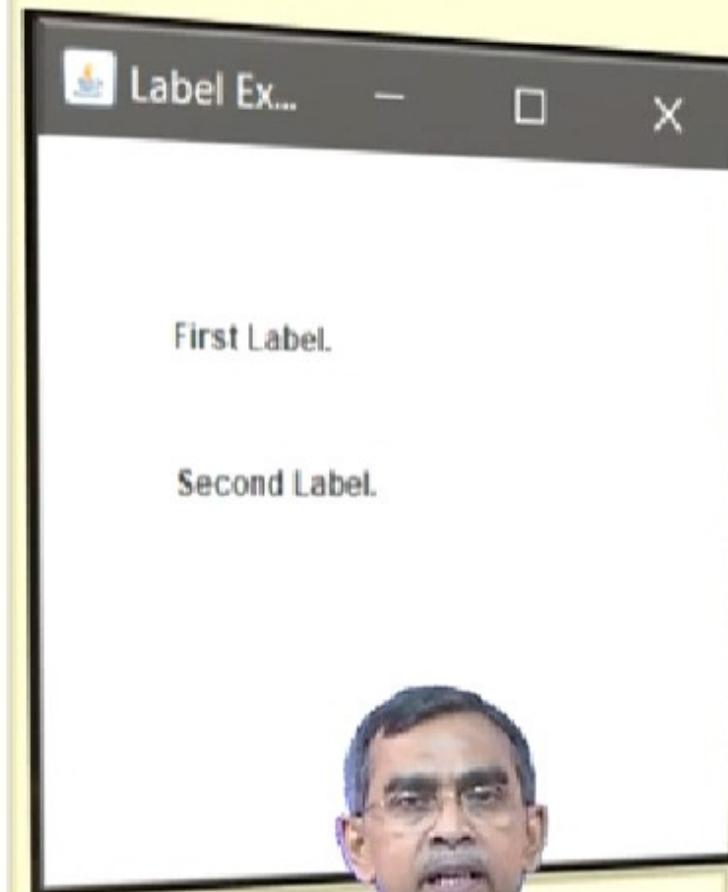


IIT KHARAGPUR



# Creating a JLabel : An example

```
import javax.swing.*;  
  
class LabelExample  
{  
public static void main(String args[]){  
    JFrame f= new JFrame("Label Example");  
    JLabel l1,l2;  
    l1=new JLabel("First Label.");  
    l1.setBounds(50,50, 100,30);  
    l2=new JLabel("Second Label.");  
    l2.setBounds(50,100, 100,30);  
    f.add(l1); f.add(l2);  
    f.setSize(300,300);  
    f.setLayout(null);  
    f.setVisible(true);  
}  
}
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR

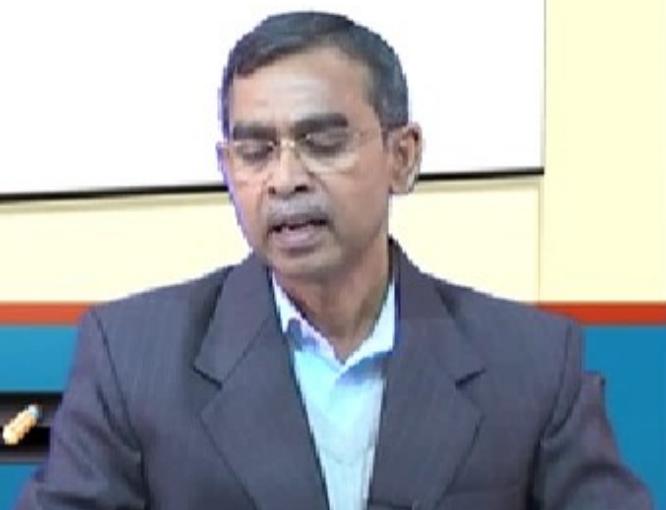
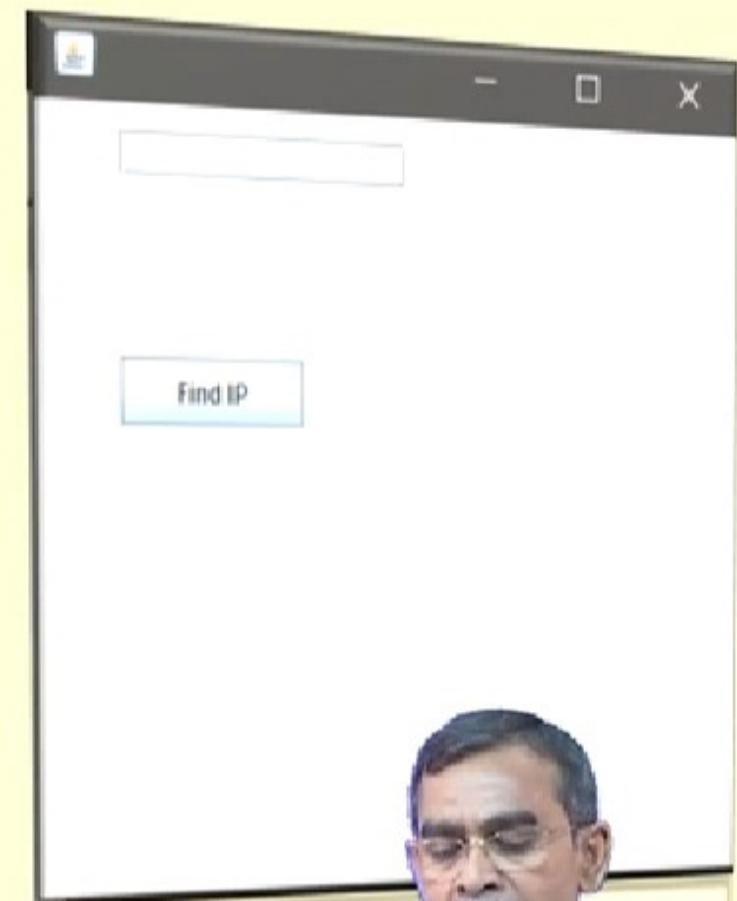




# Java JLabel : An example with ActionListener

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

public class LabelExample extends Frame implements ActionListener{
    JTextField tf; JLabel l; JButton b;
    LabelExample(){
        tf=new JTextField();
        tf.setBounds(50,50, 150,20);
        l=new JLabel();
        l.setBounds(50,100, 250,20);
        b=new JButton("Find IP");
        b.setBounds(50,150,95,30);
        b.addActionListener(this);
        add(b);add(tf);add(l);
        setSize(400,400);
        setLayout(null);
        setVisible(true);
    }
}
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR



# Java Swing JTextField



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA

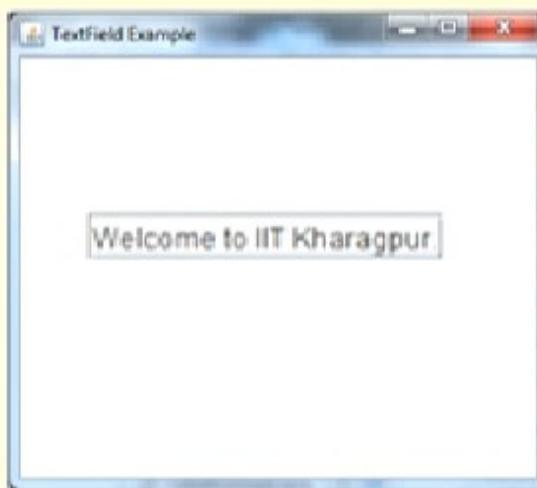


IIT KHARAGPUR



# Java JTextField

The object of a **JTextField** class is a text component that allows the editing of a single line text. It inherits **JTextComponent** class.



Below is the declaration for **javax.swing.JTextField** class.

```
public class JTextField extends JTextComponent implements SwingConstants
```



# Class JTextField : Constructors

<i>Constructor</i>	<i>Description</i>
<code>JTextField()</code>	Creates a new TextField
<code>JTextField(String text)</code>	Creates a new TextField initialized with the specified text.
<code>JTextField(String text, int columns)</code>	Creates a new TextField initialized with the specified text and columns.
<code>JTextField(int columns)</code>	Creates a new empty TextField with the specified number of columns.



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR



# Class JTextField : Methods

<i>Methods</i>	<i>Description</i>
void addActionListener(ActionListener l)	It is used to add the specified action listener to receive action events from this textfield.
Action getAction()	It returns the currently set Action for this ActionEvent source, or null if no Action is set.
void setFont(Font f)	It is used to set the current font.
void removeActionListener(ActionListener l)	It is used to remove the specified action listener so that it no longer receives action events from this textfield.



IIT KHARAGPUR



NPTEL  
NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA

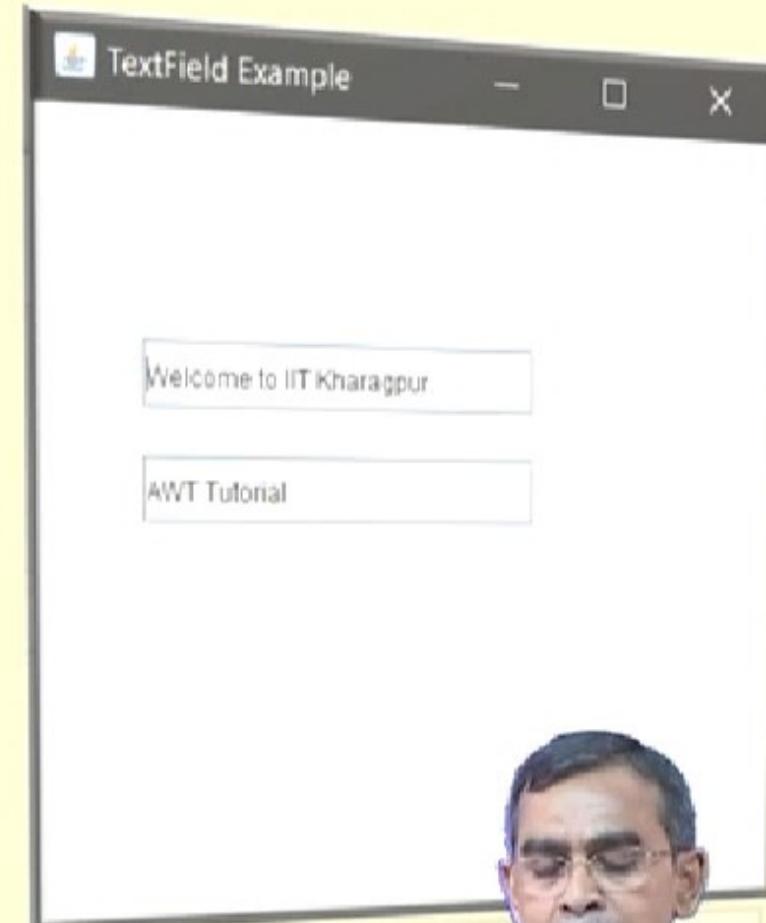


IIT KHARAGPUR



# Java JTextField : An example

```
import javax.swing.*;
class TextFieldExample
{
public static void main(String args[])
{
    JFrame f= new JFrame("TextField Example");
    JTextField t1,t2;
    t1=new JTextField("Welcome to IIT Kharagpur.");
    t1.setBounds(50,100, 200,30);
    t2=new JTextField("AWT Tutorial");
    t2.setBounds(50,150, 200,30);
    f.add(t1); f.add(t2);
    f.setSize(400,400);
    f.setLayout(null);
    f.setVisible(true);
}
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



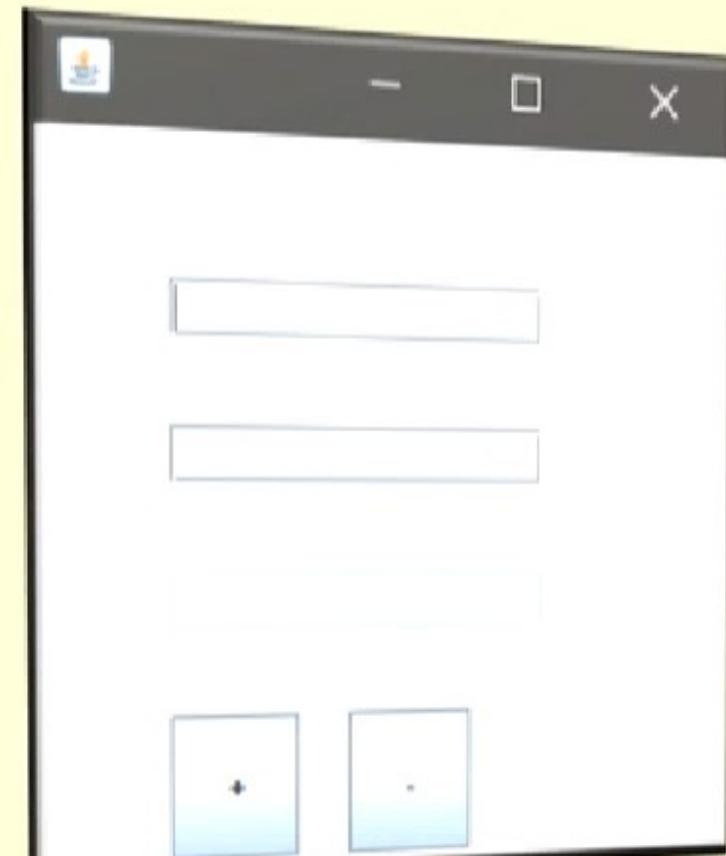
IIT KHARAGPUR





# Java JTextField : An example with ActionListener

```
import javax.swing.*;
import java.awt.event.*;
public class TextFieldExample implements ActionListener{
    JTextField tf1,tf2,tf3;
    JButton b1,b2;
    TextFieldExample(){
        JFrame f = new JFrame();
        tf1=new JTextField();
        tf1.setBounds(50,50,150,20);
        tf2=new JTextField();
        tf2.setBounds(50,100,150,20);
        tf3=new JTextField();
        tf3.setBounds(50,150,150,20);
        tf3.setEditable(false);
        b1=new JButton("+");
        b1.setBounds(50,200,50,50);
        b2=new JButton("-");
        b2.setBounds(120,200,50,50);
        b1.addActionListener(this);
        b2.addActionListener(this);
        f.add(tf1);f.add(tf2);f.add(tf3);f.add(b1);f.add(b2);
    }
}
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR



# Java JTextField : An example with ActionListener

```
f.setSize(300,300);
    f.setLayout(null);
    f.setVisible(true);
}
public void actionPerformed(ActionEvent e) {
    String s1=tf1.getText();
    String s2=tf2.getText();
    int a=Integer.parseInt(s1);
    int b=Integer.parseInt(s2);
    int c=0;
    if(e.getSource()==b1){
        c=a+b;
    }else if(e.getSource()==b2){
        c=a-b;
    }
    String result=String.valueOf(c);
    tf3.setText(result);
}
public static void main(String[] args) {
    new TextFieldExample();
}
}
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



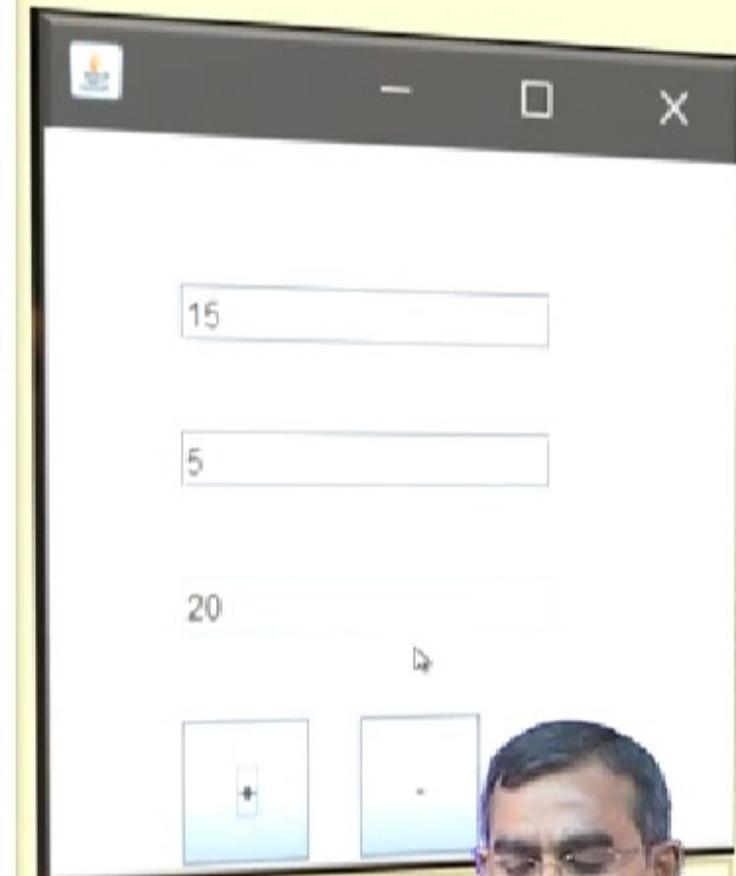
IIT KHARAGPUR





# Java JTextField : An example with ActionListener

```
f.setSize(300,300);
f.setLayout(null);
f.setVisible(true);
}
public void actionPerformed(ActionEvent e) {
    String s1=tf1.getText();
    String s2=tf2.getText();
    int a=Integer.parseInt(s1);
    int b=Integer.parseInt(s2);
    int c=0;
    if(e.getSource()==b1){
        c=a+b;
    }else if(e.getSource()==b2){
        c=a-b;
    }
    String result=String.valueOf(c);
    tf3.setText(result);
}
public static void main(String[] args) {
    new TextFieldExample();
}
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR



# Java Swing JTextArea

4

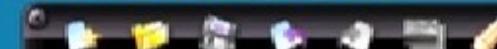


IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA

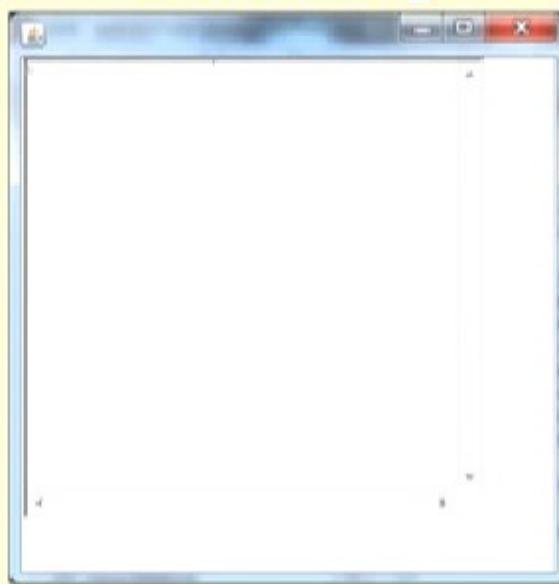


IIT KHARAGPUR



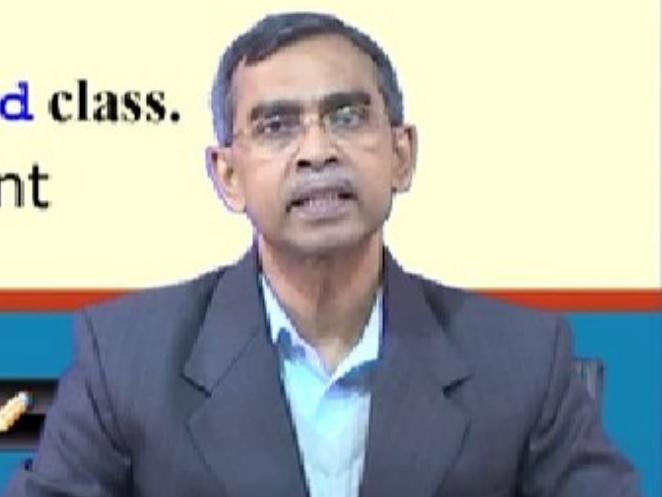
# Class JTextArea

The object of a **JTextArea** class is a multi line region that displays text. It allows the editing of multiple line text. It inherits **JTextComponent** class



Below is the declaration for **javax.swing.JTextArea** class.

**public class JTextArea extends JTextComponent**



IIT KHARAGPUR



NPTEL  
ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR



# Class JTextArea : Constructors

<i>Constructor</i>	<i>Description</i>
<code>JTextArea ()</code>	Creates a text area that displays no text initially.
<code>JTextArea (String s)</code>	Creates a text area that displays specified text initially.
<code>JTextArea (int row, int column)</code>	Creates a text area with the specified number of rows and columns that displays no text initially.
<code>JTextArea (String s, int row, int column)</code>	Creates a text area with the specified number of rows and columns that displays specified text initially.



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR





# Class JTextArea : Methods

Methods	Description
<code>void setRows(int rows)</code>	It is used to set specified number of rows.
<code>void setColumns(int cols)</code>	It is used to set specified number of columns.
<code>void setFont(Font f)</code>	It is used to set the specified font.
<code>void insert(String s, int position)</code>	It is used to insert the specified text on the specified position.
<code>void append(String s)</code>	It is used to append the given text to the end of the document.



IIT KHARAGPUR



NPTEL  
NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA

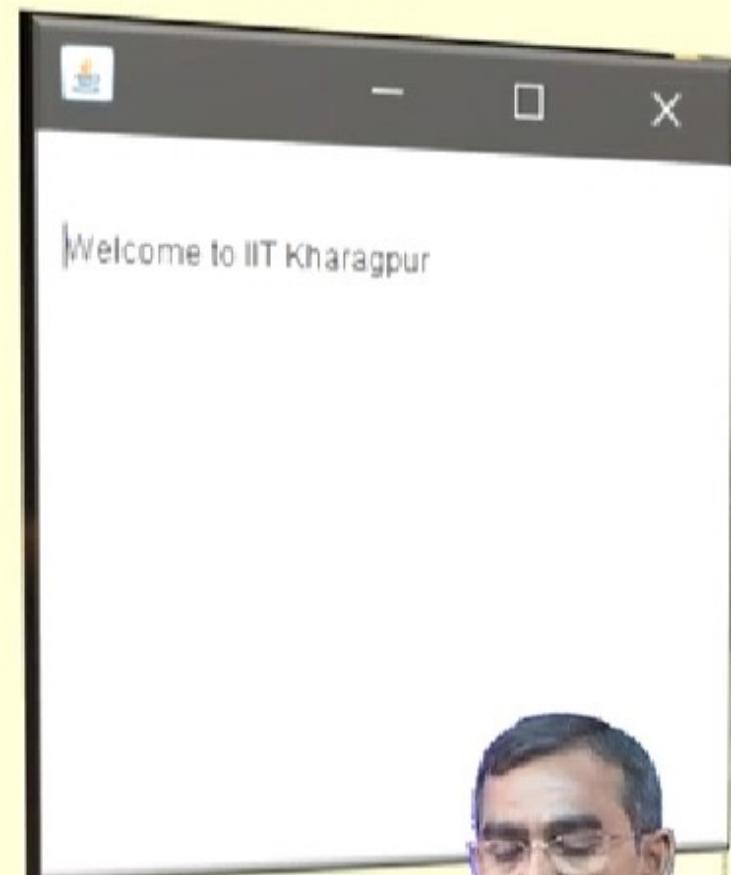


IIT KHARAGPUR



# Java JTextArea : An example

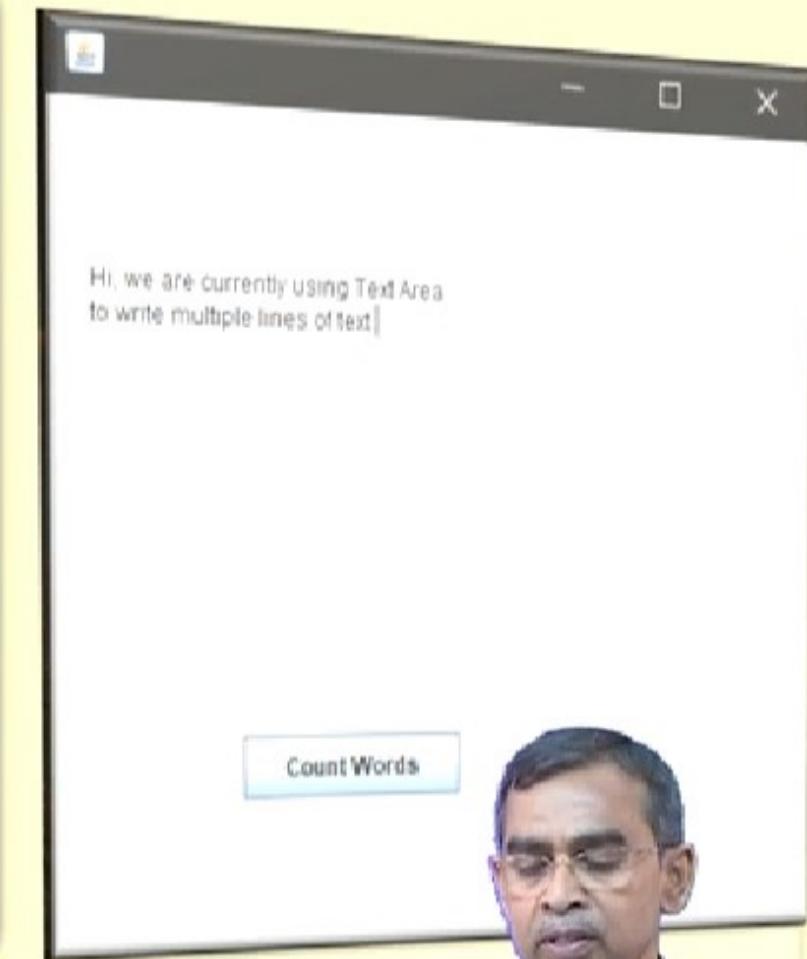
```
import javax.swing.*;
public class TextAreaExample{
    TextAreaExample(){
        JFrame f= new JFrame();
        JTextArea area=new JTextArea("Welcome to IIT
Kharagpur");
        area.setBounds(10,30, 200,200);
        f.add(area);
        f.setSize(300,300);
        f.setLayout(null);
        f.setVisible(true);
    }
    public static void main(String args[])
    {
        new TextAreaExample();
    }
}
```





# Java JTextArea : An example with ActionListener

```
import javax.swing.*;
import java.awt.event.*;
public class TextAreaExample implements ActionListener{
JLabel l1,l2;
JTextArea area;
JButton b;
TextAreaExample() {
    JFrame f = new JFrame();
    l1=new JLabel();
    l1.setBounds(50,25,100,30);
    l2=new JLabel();
    l2.setBounds(160,25,100,30);
    area=new JTextArea();
    area.setBounds(20,75,250,200);
    b=new JButton("Count Words");
    b.setBounds(100,300,120,30);
    b.addActionListener(this);
    f.add(l1);f.add(l2);f.add(area);f.add(b);
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR



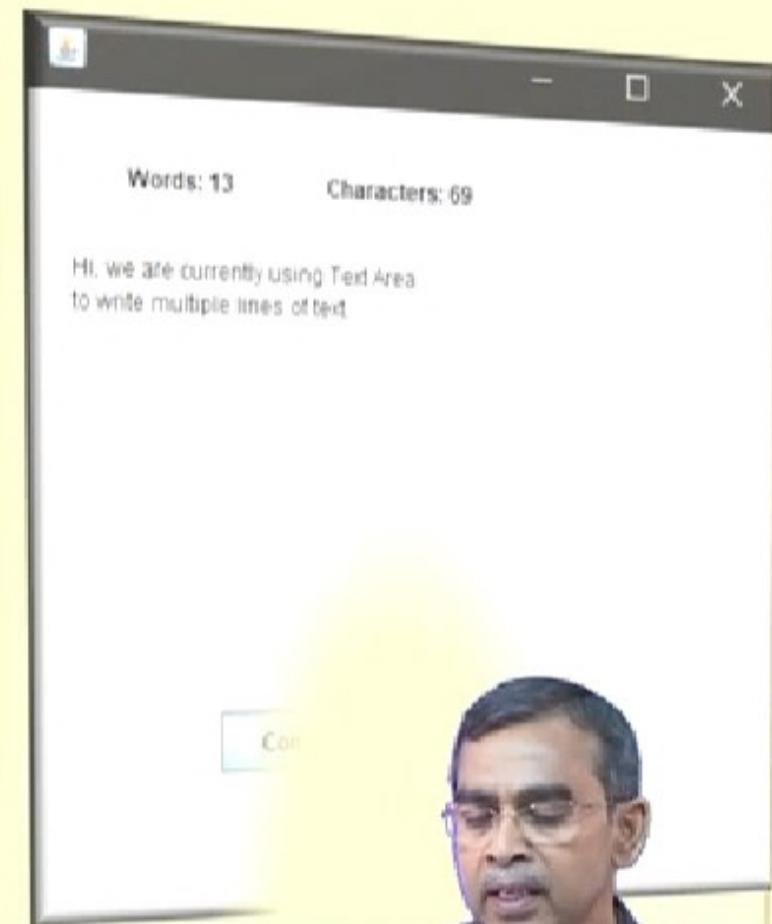


# Java JTextArea : An example with ActionListener

```
f.setSize(450,450);
f.setLayout(null);
f.setVisible(true);
}

public void actionPerformed(ActionEvent e) {
    String text=area.getText();
    String words[]=text.split("\\s");
    l1.setText("Words: "+words.length);
    l2.setText("Characters: "+text.length());
}

public static void main(String[] args) {
    new TextAreaExample();
}
}
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR



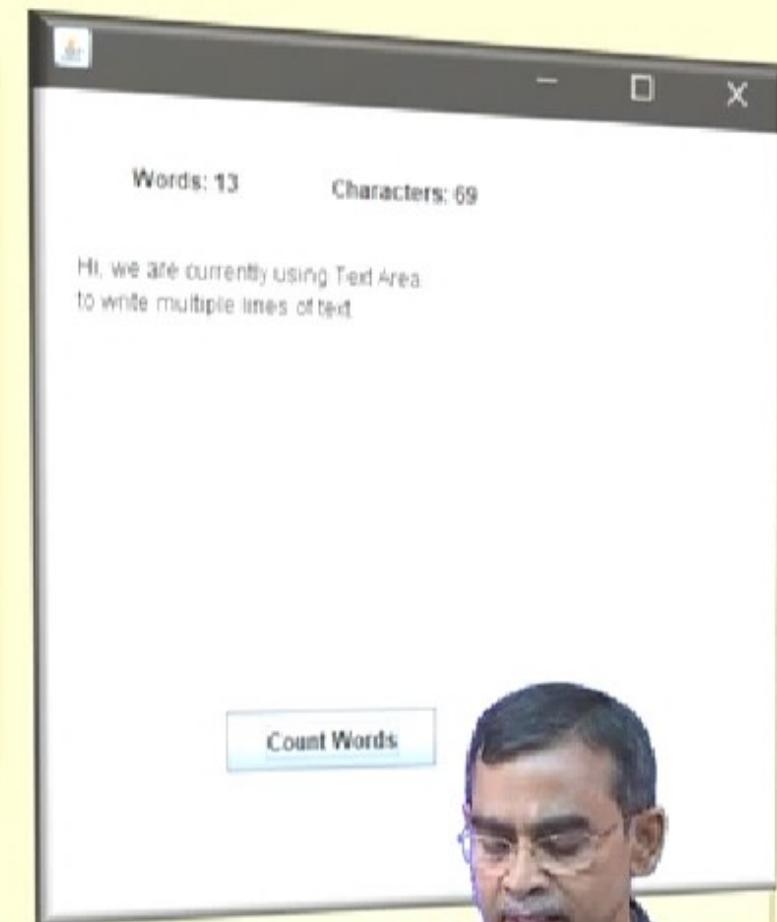


# Java JTextArea : An example with ActionListener

```
f.setSize(450,450);
f.setLayout(null);
f.setVisible(true);
}

public void actionPerformed(ActionEvent e) {
    String text=area.getText();
    String words[]=text.split("\\s");
    l1.setText("Words: "+words.length);
    l2.setText("Characters: "+text.length());
}

public static void main(String[] args) {
    new TextAreaExample();
}
}
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR





# Java Swing JPasswordField

4



IIT KHARAGPUR



NPTEL  
NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR





# Java JPasswordField

The object of a **JPasswordField** class is a text component specialized for password entry. It allows the editing of a single line of text. It inherits **JTextField** class.

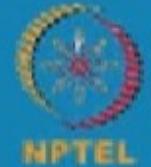
Enter the password:  ······

Below is the declaration for `javax.swing.JPasswordField` class.

**public class JPasswordField extends JTextField**



IIT KHARAGPUR



NPTEL  
ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR



# Class JPasswordField : Constructors

<i>Constructor</i>	<i>Description</i>
<code>JPasswordField()</code>	Constructs a new JPasswordField, with a default document, null starting text string, and 0 column width.
<code>JPasswordField(int columns)</code>	Constructs a new empty JPasswordField with the specified number of columns.
<code>JPasswordField(String text)</code>	Constructs a new JPasswordField initialized with the specified text.
<code>JPasswordField(String text, int columns)</code>	Construct a new JPasswordField initialized with the specified text and columns.



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA

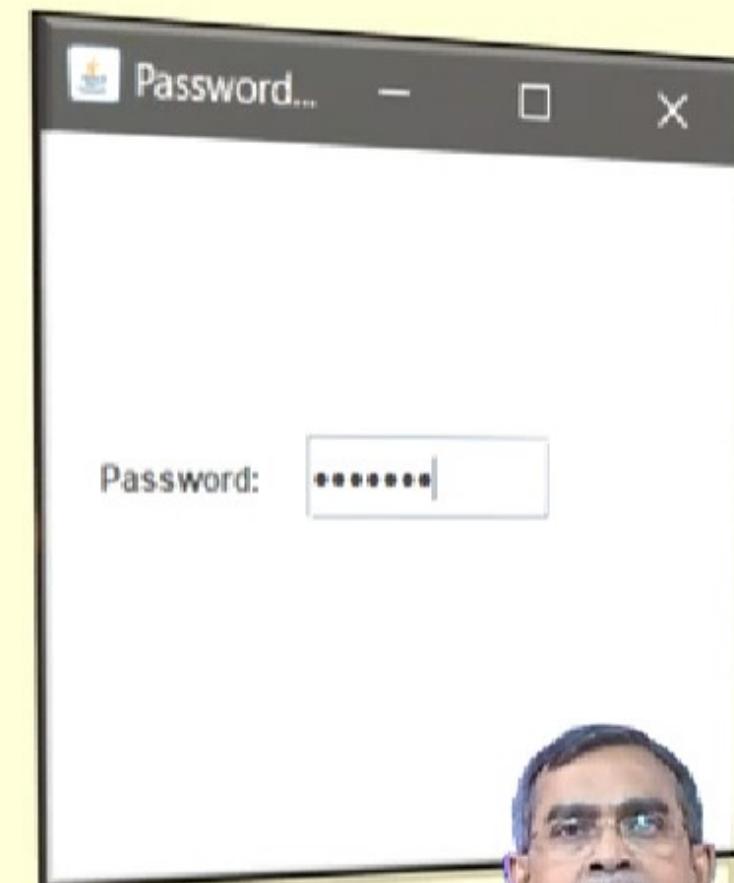


IIT KHARAGPUR



# Java JPasswordField : An example

```
import javax.swing.*;  
  
public class PasswordFieldExample {  
    public static void main(String[] args) {  
        JFrame f=new JFrame("Password Field Example");  
        JPasswordField value = new JPasswordField();  
        JLabel l1=new JLabel("Password:");  
        l1.setBounds(20,100, 80,30);  
        value.setBounds(100,100,100,30);  
        f.add(value); f.add(l1);  
        f.setSize(300,300);  
        f.setLayout(null);  
        f.setVisible(true);  
    }  
}
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



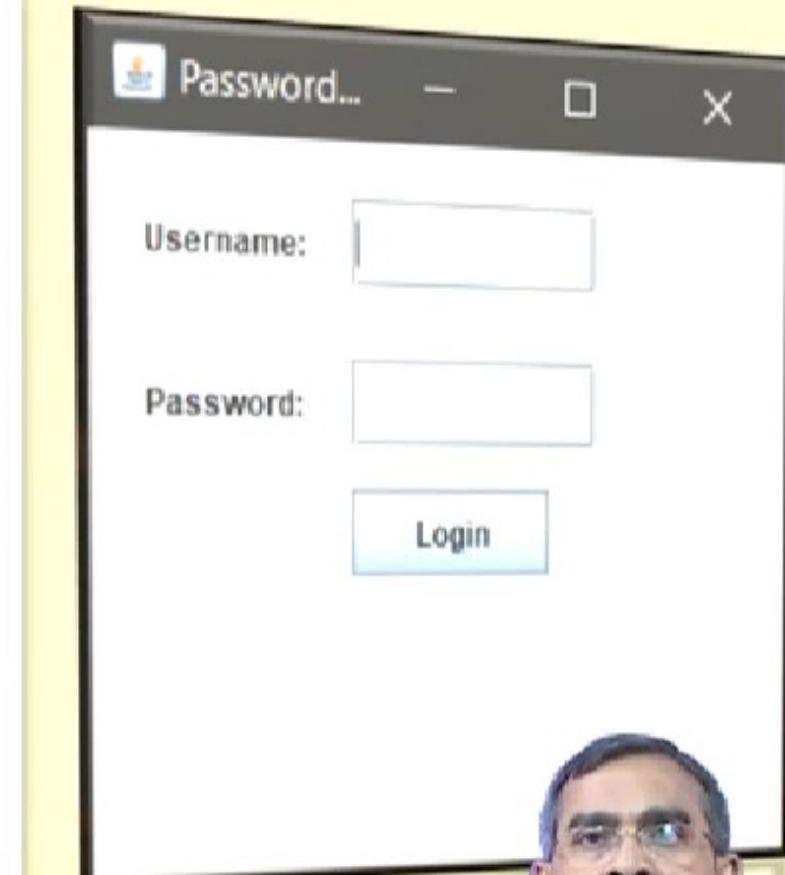
IIT KHARAGPUR





# Java JPasswordField : An example with ActionListener

```
import javax.swing.*;
import java.awt.event.*;
public class PasswordFieldExample {
    public static void main(String[] args) {
        JFrame f=new JFrame("Password Field Example");
        final JLabel label = new JLabel();
        label.setBounds(20,150, 200,50);
        final JPasswordField value = new JPasswordField();
        value.setBounds(100,75,100,30);
        JLabel l1=new JLabel("Username:");
        l1.setBounds(20,20, 80,30);
        JLabel l2=new JLabel("Password:");
        l2.setBounds(20,75, 80,30);
        JButton b = new JButton("Login");
        b.setBounds(100,120, 80,30);
        final JTextField text = new JTextField();
        text.setBounds(100,20, 100,30);
        f.add(value); f.add(l1); f.add(label); f.add(l2); f.add(b);
        f.add(text);
    }
}
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA

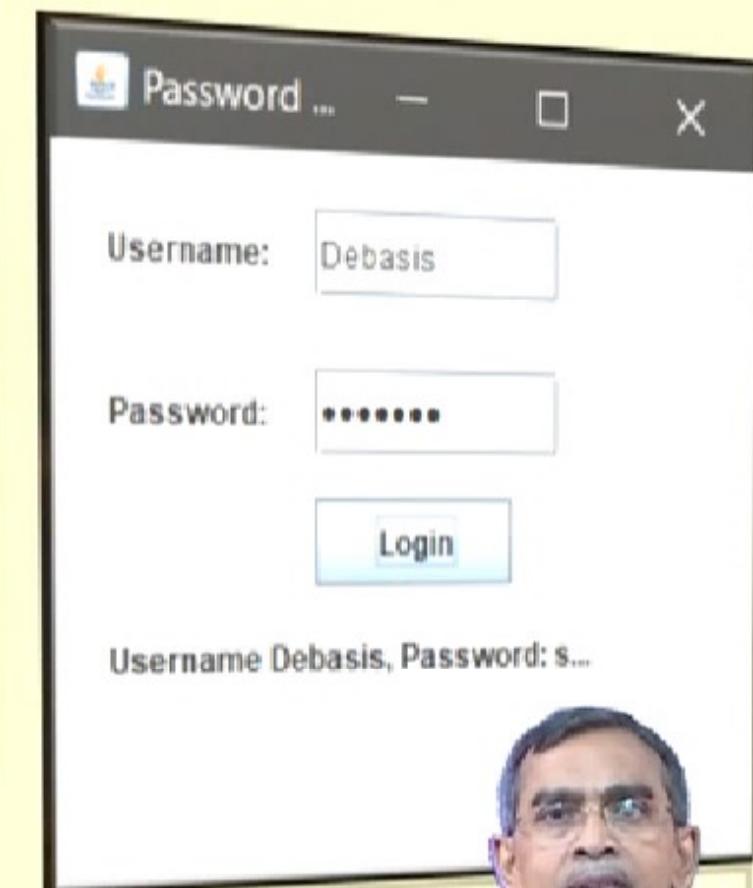


IIT KHARAGPUR



# Java JPasswordField : An example with ActionListener

```
f.setSize(300,300);
f.setLayout(null);
f.setVisible(true);
b.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        String data = "Username " + text.getText();
        data += ", Password: " + new String(value.getPassword());
        label.setText(data);
    }
})
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



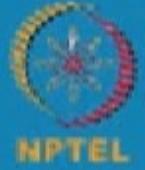
IIT KHARAGPUR



# Java Swing JCheckBox



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR





# Java JCheckBox

The **JCheckBox** class is used to create a checkbox. It is used to turn an option on (true) or off (false). Clicking on a **JCheckBox** changes its state from "on" to "off" or from "off" to "on ". It inherits **JToggleButton** class.

E-mail pubneil@gmail.com

Option 1

Option 2

Option 3

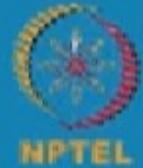
Submit Form

Below is the declaration for **javax.swing.JCheckBox** class.

```
public class JCheckBox extends JToggleButton implements Accessible
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA

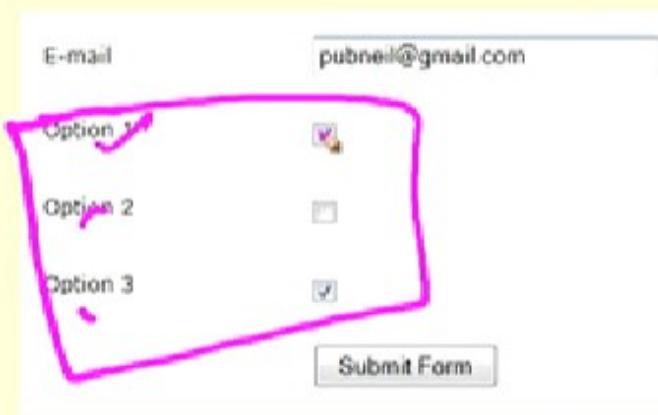


IIT KHARAGPUR



# Java JCheckBox

The **JCheckBox** class is used to create a checkbox. It is used to turn an option on (true) or off (false). Clicking on a **JCheckBox** changes its state from "on" to "off" or from "off" to "on ". It inherits **JToggleButton** class.



Below is the declaration for **javax.swing.JCheckBox** class.

**public class** JCheckBox **extends** JToggleButton **implements** Access

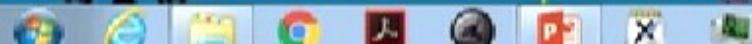
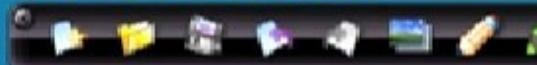


IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA





# Class JCheckBox : Constructors

<i>Constructor</i>	<i>Description</i>
JCheckBox()	Creates an initially unselected check box button with no text, no icon.
JCheckBox(String s)	Creates an initially unselected check box with text.
JCheckBox(String text, boolean selected)	Creates a check box with text and specifies whether or not it is initially selected.
JCheckBox(Action a)	Creates a check box where properties are taken from the Action supplied.





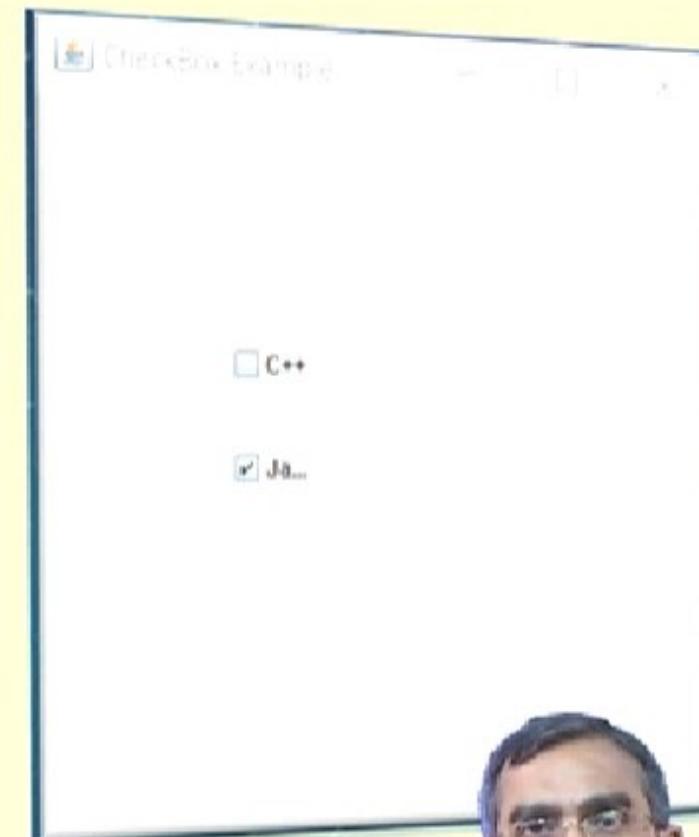
# Java JCheckBox methods

<i>Methods</i>	<i>Description</i>
AccessibleContext getAccessibleContext ()	It is used to get the AccessibleContext associated with this JCheckBox.
protected String paramString ()	It returns a <u>string</u> representation of this JCheckBox.



# Java JCheckBox : An example

```
import javax.swing.*;
public class CheckBoxExample
{
    CheckBoxExample() {
        JFrame f= new JFrame("CheckBox Example");
        JCheckBox checkBox1 = new JCheckBox("C++");
        checkBox1.setBounds(100,100, 50,50);
        JCheckBox checkBox2 = new JCheckBox("Java", true);
        checkBox2.setBounds(100,150, 50,50);
        f.add(checkBox1);
        f.add(checkBox2);
        f.setSize(400,400);
        f.setLayout(null);
        f.setVisible(true);
    }
    public static void main(String args[])
    {
        new CheckBoxExample();
    }
}
```





# Java Swing JRadioButton



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR





# Class JRadioButton

The **JRadioButton** class is used to create a radio button. It is used to choose one option from multiple options. It is widely used in exam systems or quiz.

It should be added in **ButtonGroup** to select one radio button only.

- One
- Two
- Three

Below is the declaration for `javax.swing.JRadioButton` class.

**public class** JRadioButton **extends** JToggleButton **implements** Accessible



IIT KHARAGPUR



NPTEL  
NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR



# Class JRadioButton : Constructors

<i>Constructor</i>	<i>Description</i>
JRadioButton()	Creates an unselected radio button with no text.
JRadioButton(String s)	Creates an unselected radio button with specified text.
JRadioButton(String s, boolean selected)	Creates a radio button with the specified text and selected status.





# Class JRadioButton : Methods

<i>Methods</i>	<i>Description</i>
void setText(String s)	It is used to set specified text on button.
String getText()	It is used to return the text of the button.
void setEnabled(boolean b)	It is used to enable or disable the button.
void setIcon(Icon b)	It is used to set the specified Icon on the button.
Icon getIcon()	It is used to get the Icon of the button.
void setMnemonic(int a)	It is used to set the mnemonic on the button.
void addActionListener(ActionListener a)	It is used to add the action listener to this object.



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



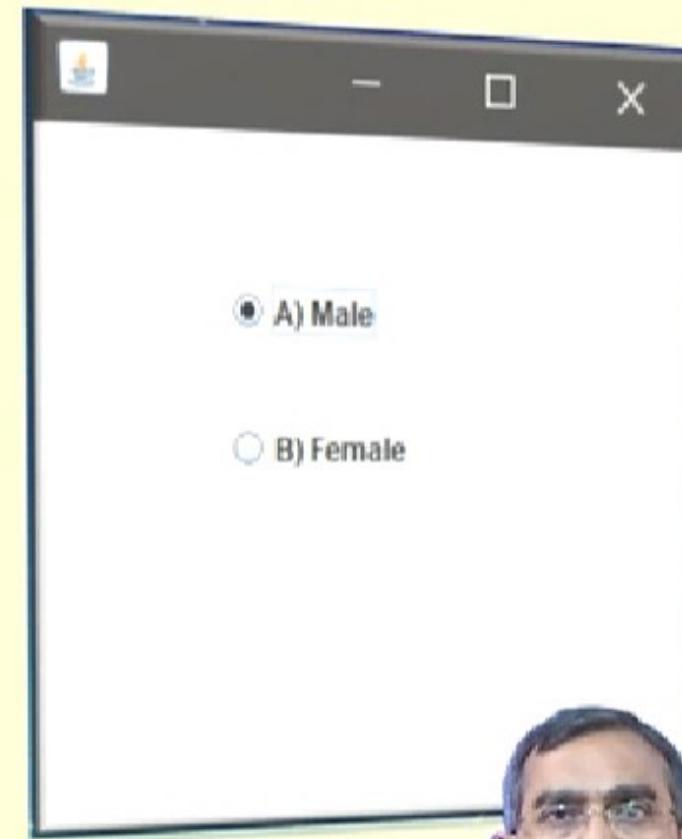
IIT KHARAGPUR





# Java JRadioButton : An example

```
import javax.swing.*;
public class RadioButtonExample {
    JFrame f;
    RadioButtonExample() {
        f=new JFrame();
        JRadioButton r1=new JRadioButton("A) Male");
        JRadioButton r2=new JRadioButton("B) Female");
        r1.setBounds(75,50,100,30);
        r2.setBounds(75,100,100,30);
        ButtonGroup bg=new ButtonGroup();
        bg.add(r1);bg.add(r2);
        f.add(r1);f.add(r2);
        f.setSize(300,300);
        f.setLayout(null);
        f.setVisible(true);
    }
    public static void main(String[] args) {
        new RadioButtonExample();
    }
}
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR

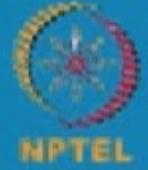




# Java Swing JComboBox



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



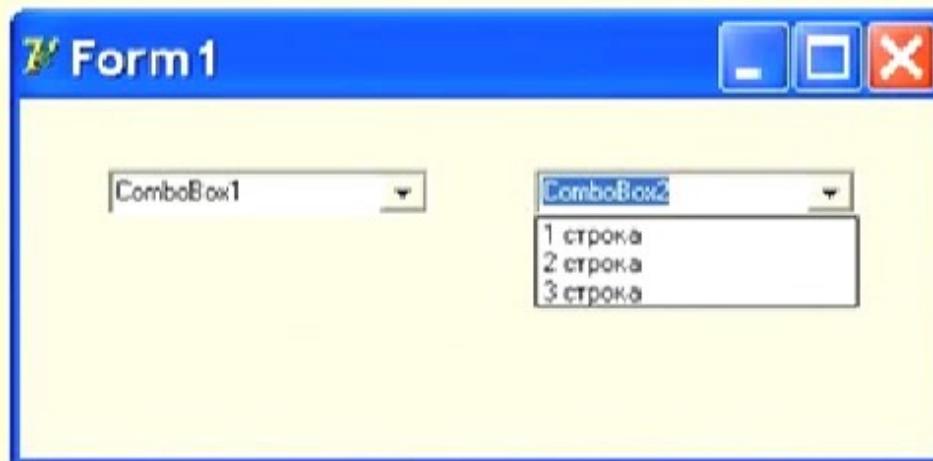
IIT KHARAGPUR





# Class JComboBox

The object of **Choice** class is used to show popup menu of choices. Choice selected by user is shown on the top of a menu. It inherits **JComponent** class.

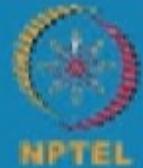


Below is the declaration for `javax.swing.JComboBox` class.

```
public class JComboBox extends JComponent implements ItemSelectable, ListData  
Listener, ActionListener, Accessible
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR



# Class JComboBox : Constructors

<i>Constructor</i>	<i>Description</i>
JComboBox ()	Creates a JComboBox with a default data model.
JComboBox (Object[] items)	Creates a JComboBox that contains the elements in the specified <a href="#">array</a> .
JComboBox (Vector<?> items)	Creates a JComboBox that contains the elements in the specified <a href="#">Vector</a> .



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



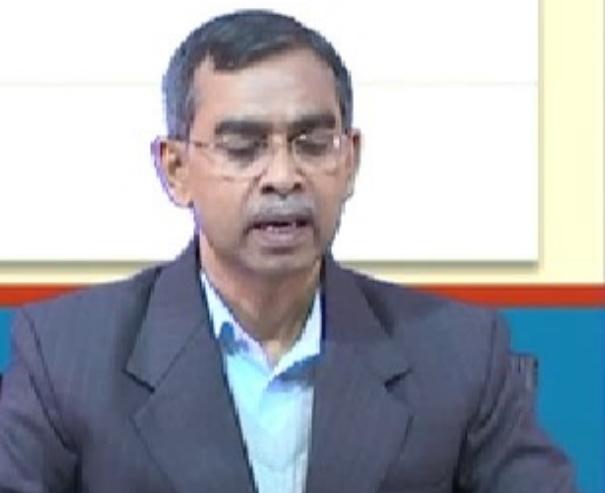
IIT KHARAGPUR





# Java JJComboBox methods

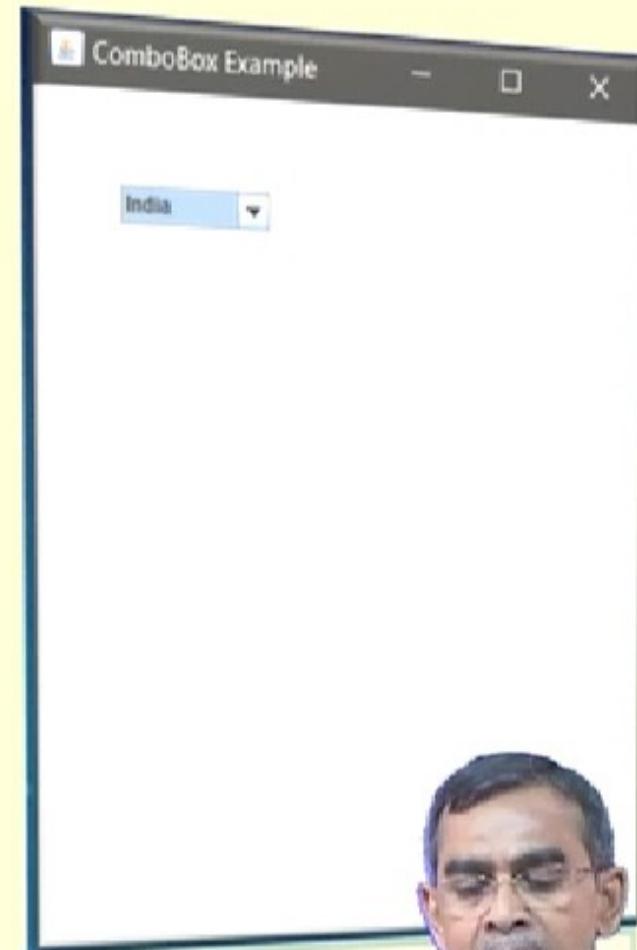
Methods	Description
void addItem(Object anObject)	It is used to add an item to the item list.
void removeItem(Object anObject)	It is used to delete an item to the item list.
void removeAllItems()	It is used to remove all the items from the list.
void setEditable(boolean b)	It is used to determine whether the JComboBox is editable.
void addActionListener(ActionListener a)	It is used to add the <a href="#">ActionListener</a> .
void addItemSelectedListener(ItemListener i)	It is used to add the <a href="#">ItemListener</a> .





# Java JComboBox : An example

```
import javax.swing.*;
public class ComboBoxExample {
JFrame f;
ComboBoxExample() {
    f=new JFrame("ComboBox Example");
    String
country[]{"India","Aus","U.S.A","England","Newzealand"};
    JComboBox cb=new JComboBox(country);
    cb.setBounds(50, 50,90,20);
    f.add(cb);
    f.setLayout(null);
    f.setSize(400,500);
    f.setVisible(true);
}
public static void main(String[] args) {
    new ComboBoxExample();
}
}
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR

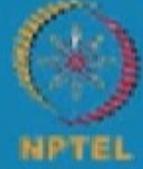




# Java Swing JTable



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR





# Class JTable

The **JTable** class is used to display data in tabular form. It is composed of rows and columns.

Message

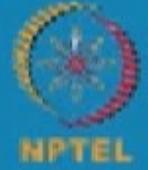
i

First Name	Last Name	Sport	# of Years	Vegetarian
Kathy	Smith	Snowboarding	5	false
John	Doe	Rowing	3	true
Sue	Black	Knitting	2	false
Jane	White	Speed reading	20	true
Joe	Brown	Pool	10	false

OK

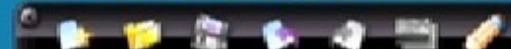


IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR



# Class JTable : Constructors

<i>Constructor</i>	<i>Description</i>
<code>JTable()</code>	Creates a table with empty cells.
<code>JTable(Object[][] rows, Object[] columns)</code>	Creates a table with the specified data.





# Java JTable : An example

```
import javax.swing.*;
public class TableExample {
    JFrame f;
    TableExample() {
        f=new JFrame();
        String data[][]={{ "101","Amit","670000"},
                        {"102","Jai","780000"},
                        {"101","Sachin","700000"}};
        String column[]={ "ID", "NAME", "SALARY"};
        JTable jt=new JTable(data,column);
        jt.setBounds(30,40,200,300);
        // JScrollPane sp=new JScrollPane(jt);
        f.add(sp);
        f.setSize(300,400);
        f.setVisible(true);
    }
    public static void main(String[] args) {
        new TableExample();
    }
}
```

ID	NAME	SALARY
101	Amit	670000
102	Jai	780000
101	Sachin	700000



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



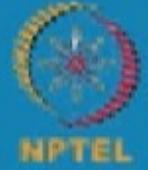
IIT KHARAGPUR



# Java Swing JList



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR





# Class JList

The object of **JList** class represents a list of text items. The list of text items can be set up so that the user can choose either one item or multiple items. It inherits **JComponent** class.

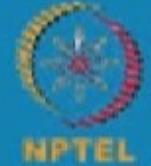


Below is the declaration for `javax.swing.JList` class.

**public class** JList **extends** JComponent **implements** Scrollable, Accessible



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR



# Class JList : Constructors

<i>Constructor</i>	<i>Description</i>
<code>JList()</code>	Creates a JList with an empty, read-only, model.
<code>JList(ary[] listData)</code>	Creates a JList that displays the elements in the specified array.
<code>JList(ListModel&lt;ary&gt; dataModel)</code>	Creates a JList that displays elements from the specified, non-null, model.



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR



# Class JList : Methods

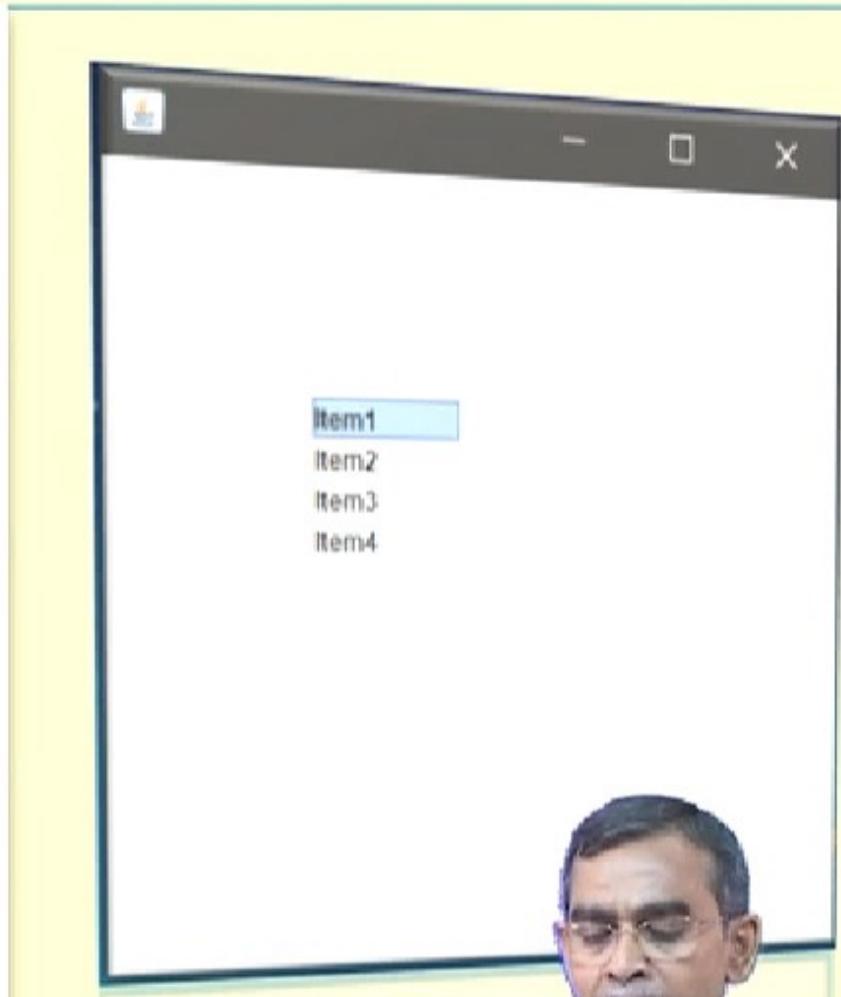
Methods	Description
<code>Void addListSelectionListener(ListSelectionListener listener)</code>	It is used to add a listener to the list, to be notified each time a change to the selection occurs.
<code>int selectedIndex()</code>	It is used to return the smallest selected cell index.
<code>ListModel getModel()</code>	It is used to return the data model that holds a list of items displayed by the JList component.
<code>void setListData(Object[] listData)</code>	It is used to create a ready-to-use ListModel from an array of objects.





# Java JList : An example

```
import javax.swing.*;
public class ListExample
{
    ListExample() {
        JFrame f = new JFrame();
        DefaultListModel<String> ll = new DefaultListModel<>();
        ll.addElement("Item1");
        ll.addElement("Item2");
        ll.addElement("Item3");
        ll.addElement("Item4");
        JList<String> list = new JList<>(ll);
        list.setBounds(100,100, 75,75);
        f.add(list);
        f.setSize(400,400);
        f.setLayout(null);
        f.setVisible(true);
    }
    public static void main(String args[])
    {
        new ListExample();
    }
}
```





# Java Swing JOptionPane



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR





# Class JOptionPane

The **JOptionPane** class is used to provide standard dialog boxes such as message dialog box, confirm dialog box and input dialog box. These dialog boxes are used to display information or get input from the user. The **JOptionPane** class inherits **JComponent** class.



Below is the declaration for `javax.swing.JOptionPane` class.

**public class** JOptionPane **extends** JComponent **implements** Accessible



IIT KHARAGPUR



NPTEL  
NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR

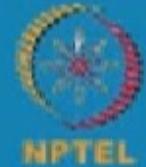


# Class JOptionPane : Constructors

<i>Constructor</i>	<i>Description</i>
JOptionPane()	It is used to create a JOptionPane with a test message.
JOptionPane(Object message)	It is used to create an instance of JOptionPane to display a message.
JOptionPane(Object message, int messageType)	It is used to create an instance of JOptionPane to display a message with specified message type and default options.



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR





# Class JOptionPane : Methods

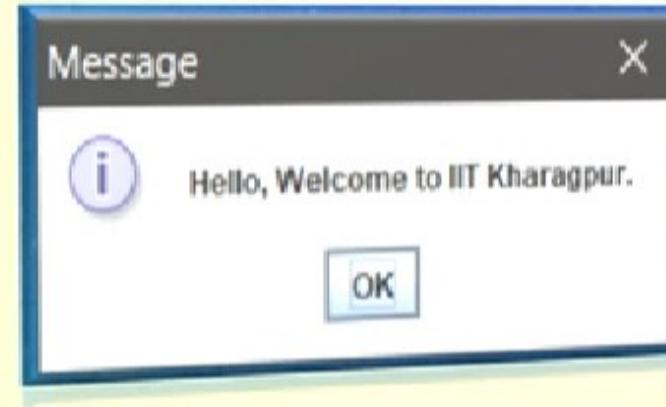
Methods	Description
JDialog createDialog(String title)	It is used to create and return a new parentless JDialog with the specified title.
static void showMessageDialog(Component parentComponent, Object message)	It is used to create an information-message dialog titled "Message".
static void showMessageDialog(Component parentComponent, Object message, String title, int messageType)	It is used to create a message dialog with given title and messageType.
static int showConfirmDialog(Component parentComponent, Object message)	It is used to create a dialog with the options Yes, No and Cancel; with the title, Select an Option.
static String showInputDialog(Component parentComponent, Object message)	It is used to show a question-message dialog requesting input from the user parented to parentComponent.
void setInputValue(Object newValue)	It is used to set the input value that was selected or input





# Java JOptionPane : An example

```
import javax.swing.*;
public class OptionPaneExample {
    JFrame f;
    OptionPaneExample() {
        f=new JFrame();
        JOptionPane.showMessageDialog(f,"Hello, Welcome to IIT
Kharagpur.");
    }
    public static void main(String[] args) {
        new OptionPaneExample();
    }
}
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR

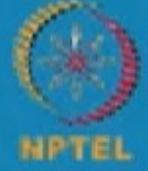




# Java Swing JScrollPane



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



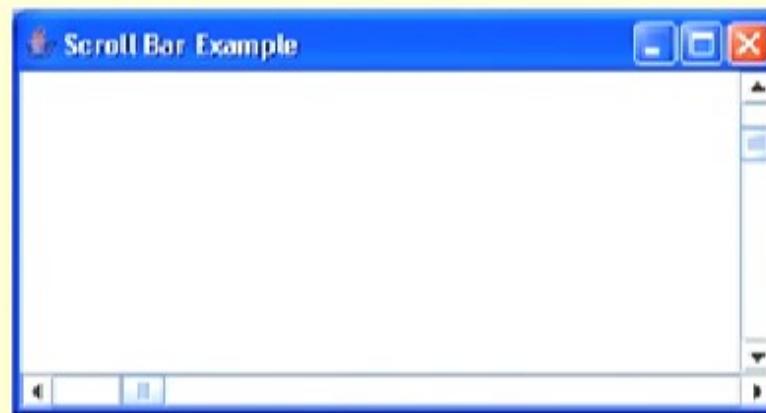
IIT KHARAGPUR





# Java JScrollBar

The object of **JScrollBar** class is used to add horizontal and vertical scrollbar. It is an implementation of a scrollbar. It inherits **JComponent** class.



Below is the declaration for `javax.swing.JScrollBar` class.

**public class** JScrollBar **extends** JComponent **implements** Adjustable, Accessible



# Java JScrollPane constructors

<i>Constructor</i>	<i>Description</i>
<code>JScrollBar()</code>	Creates a vertical scrollbar with the initial values.
<code>JScrollBar(int orientation)</code>	Creates a scrollbar with the specified orientation and the initial values.
<code>JScrollBar(int orientation, int value, int extent, int min, int max)</code>	Creates a scrollbar with the specified orientation, value, extent, minimum, and maximum.



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA

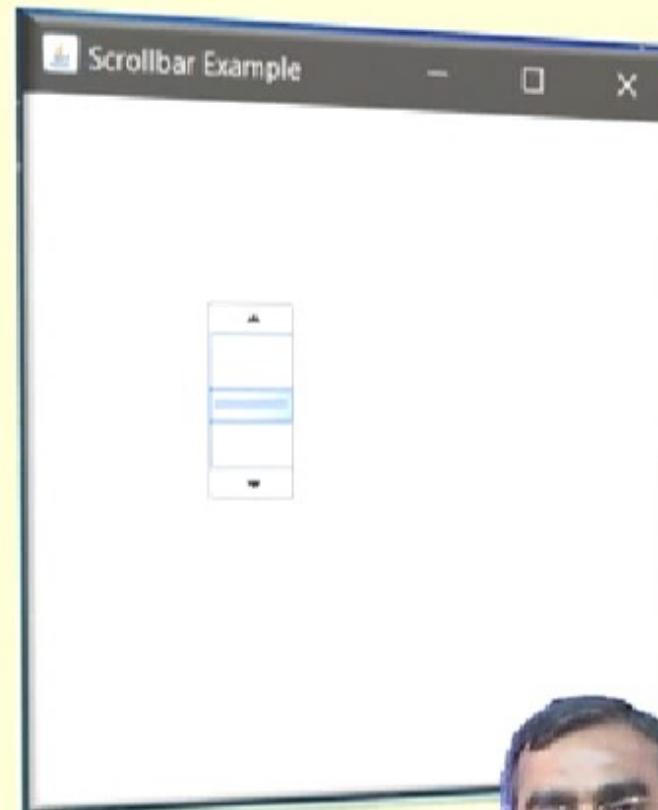


IIT KHARAGPUR



# Java JScrollPane : An example

```
import javax.swing.*;
class ScrollBarExample
{
    ScrollBarExample()
    {
        JFrame f= new JFrame("Scrollbar Example");
        JScrollPane s=new JScrollPane();
        s.setBounds(100,100, 50,100);
        f.add(s);
        f.setSize(400,400);
        f.setLayout(null);
        f.setVisible(true);
    }
    public static void main(String args[])
    {
        new ScrollBarExample();
    }
}
```





# Java Swing JMenuBar



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR





## Class MenuBar

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

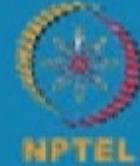


Below is the declaration for `javax.swing.JMenuBar` class.

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



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



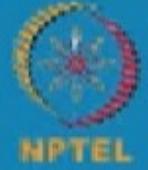
IIT KHARAGPUR



# Java Swing JMenu

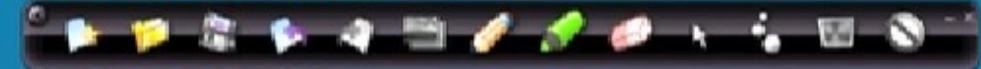


IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA

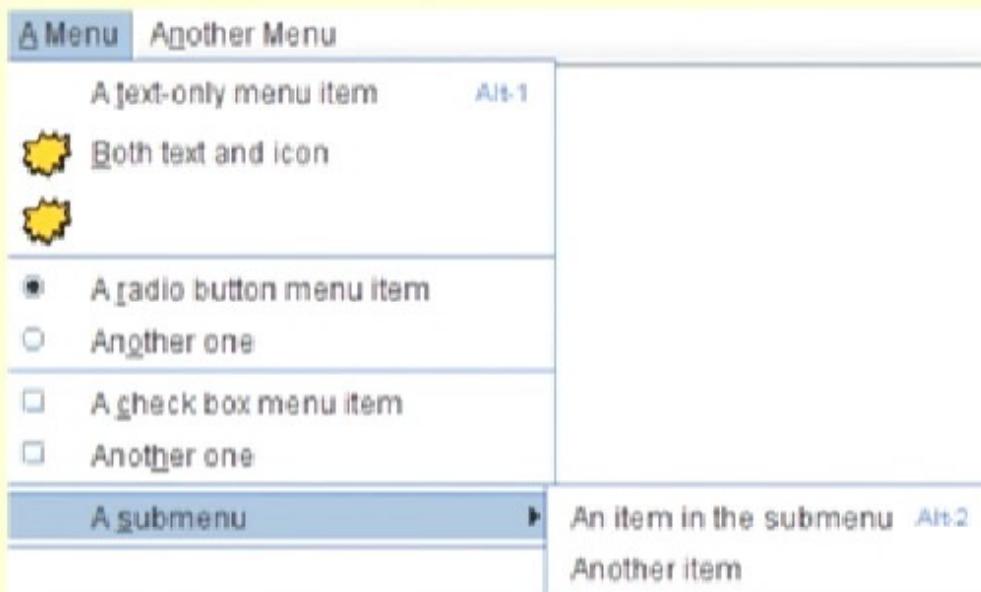


IIT KHARAGPUR



# Class JMenu

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



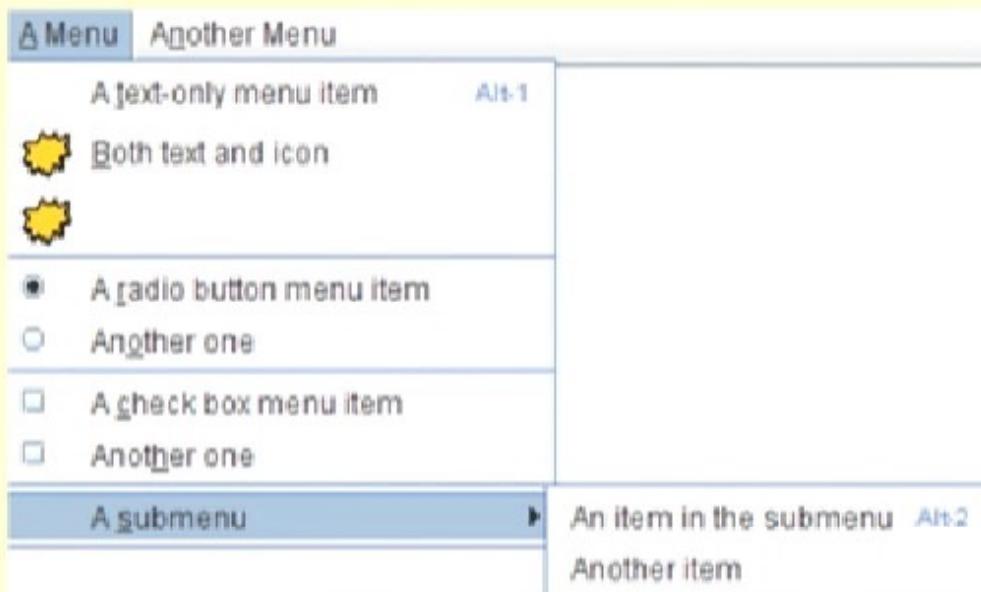
Below is the declaration for `javax.swing.JMenuItem` class.

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



# Class JMenuItem

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.



Below is the declaration for `javax.swing.JMenuItem` class.

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



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA

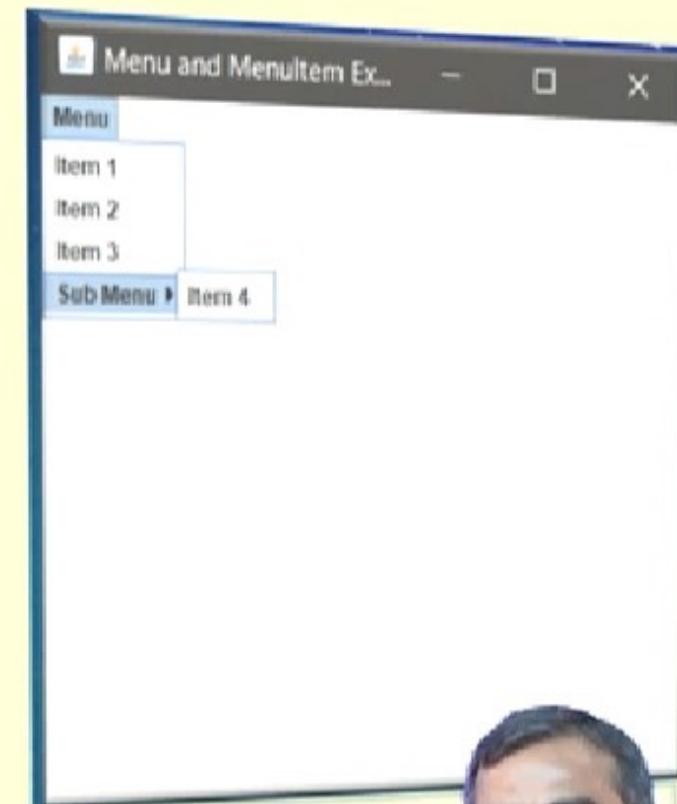


IIT KHARAGPUR



# Java JMenuItem & JMenu : An example

```
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");
        menu.add(i1); menu.add(i2); menu.add(i3);
        submenu.add(i4);
        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();
    }
}
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR





# Java Swing JPopupMenu



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



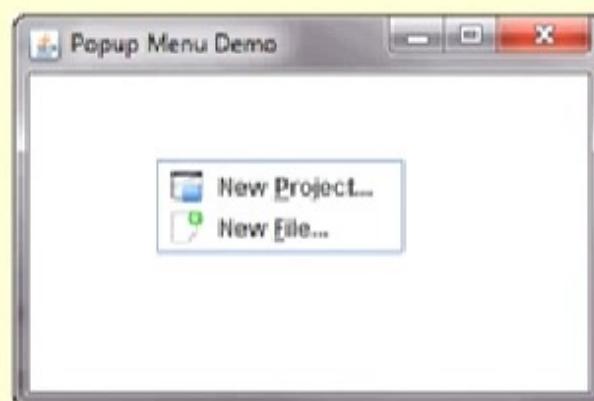
IIT KHARAGPUR





# Class JPopupMenu

**JPopupMenu** can be dynamically popped up at specific position within a component. It inherits the **JComponent** class.



Below is the declaration for `javax.swing.JPopupMenu` class.

```
public class JPopupMenu extends JComponent implements Accessible, MenuElement
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR



# Class JPopupMenu : Constructors

<i>Constructor</i>	<i>Description</i>
JPopupMenu()	Constructs a JPopupMenu without an "invoker".
JPopupMenu(String label)	Constructs a JPopupMenu with the specified title.



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



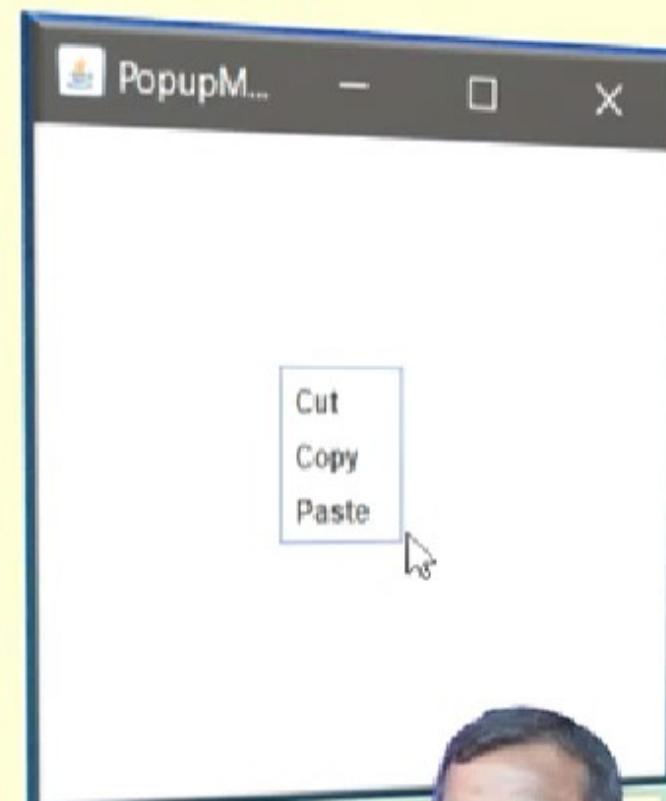
IIT KHARAGPUR





# Java JPopupMenu : An example

```
import javax.swing.*;
import java.awt.event.*;
class PopupMenuExample {
    PopupMenuExample() {
        final JFrame f = new JFrame("PopupMenu Example");
        final JPopupMenu popupmenu = new JPopupMenu("Edit");
        JMenuItem cut = new JMenuItem("Cut");
        JMenuItem copy = new JMenuItem("Copy");
        JMenuItem paste = new JMenuItem("Paste");
        popupmenu.add(cut); popupmenu.add(copy); popupmenu.add(paste);
        f.addMouseListener(new MouseAdapter() {
            public void mouseClicked(MouseEvent e) {
                popupmenu.show(f, e.getX(), e.getY());
            }
        });
        f.add(popupmenu);
        f.setSize(300, 300);
        f.setLayout(null);
        f.setVisible(true);
    }
    public static void main(String args[]) {
        new PopupMenuExample();
    }
}
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR



# Java Swing JCheckBoxMenuItem



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR





# Java JCheckBoxMenuItem

**JCheckBoxMenuItem** class represents checkbox which can be included on a menu. A **JCheckBoxMenuItem** can have text or a graphic icon or both, associated with it. **MenuItem** can be selected or deselected. **MenuItems** can be configured and controlled by actions.

## Nested class

<i>Modifier and Type</i>	<i>Class</i>	<i>Description</i>
protected class	JCheckBoxMenuItem.AccessibleJCheckBoxMenuItem	This <a href="#">class</a> implements accessibility support for the JcheckboxMenuItem class.



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR



# Class JCheckBoxMenuItem : Constructors

<i>Constructor</i>	<i>Description</i>
JCheckBoxMenuItem()	It creates an initially unselected check box menu item with no set text or icon.
JCheckBoxMenuItem(Action a)	It creates a menu item whose properties are taken from the Action supplied.
JCheckBoxMenuItem(Icon icon)	It creates an initially unselected check box menu item with an icon.
JCheckBoxMenuItem(String text)	It creates an initially unselected check box menu item with text.
JCheckBoxMenuItem(String text, boolean b)	It creates a check box menu item with the specified text and selection state.
JCheckBoxMenuItem(String text, Icon icon)	It creates an initially unselected check box menu item with the specified text and icon.
JCheckBoxMenuItem(String text, Icon icon, boolean b)	It creates a check box menu item with the specified text, icon, and selection state.



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR



# Class JCheckBoxMenuItem : Methods

Modifier	Method	Description
AccessibleContext	getAccessibleContext ()	It gets the AccessibleContext associated with this JCheckBoxMenuItem.
Object[]	getSelectedObjects ()	It returns an <a href="#">array</a> (length 1) containing the check box menu item <a href="#">label</a> or null if the check box is not selected.
boolean	getState ()	It returns the selected-state of the item.
<a href="#">String</a>	getUIClassID ()	It returns the name of the L&F class that renders this component.
protected String	paramString ()	It returns a string representation of this JCheckBoxMenuItem.
void	setState (boolean b)	It sets the selected-state of the item.



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA

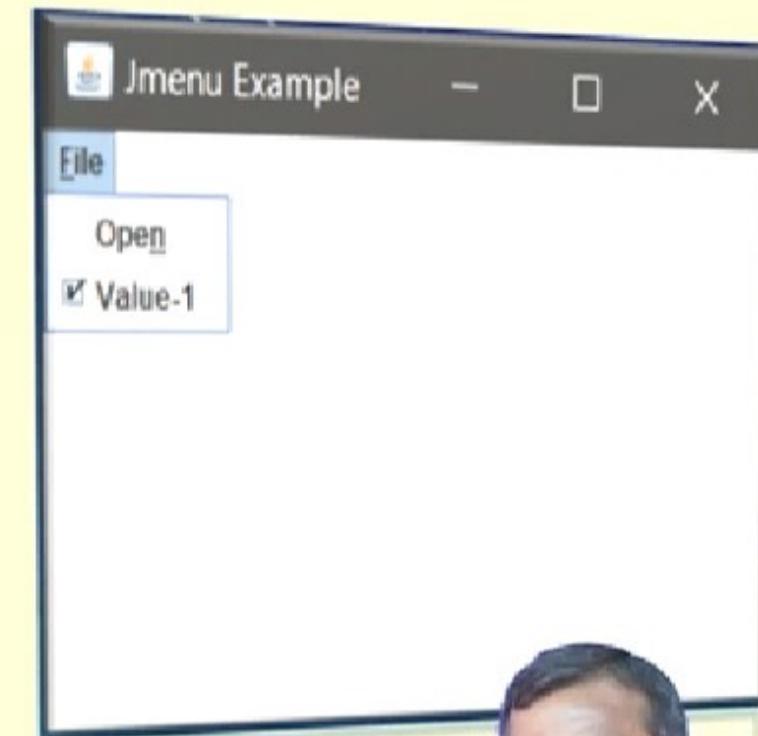


IIT KHARAGPUR



# Java JCheckBoxMenuItem : An example

```
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.KeyEvent;
import javax.swing.AbstractButton;
import javax.swing.Icon;
import javax.swing.JCheckBoxMenuItem;
import javax.swing.JFrame;
import javax.swing.JMenu;
import javax.swing.JMenuBar;
import javax.swing.JMenuItem;
public class JavaCheckBoxMenuItemExample {
    public static void main(final String args[]) {
        JFrame frame = new JFrame("Jmenu Example");
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        JMenuBar menuBar = new JMenuBar();
        JMenu fileMenu = new JMenu("File");
        fileMenu.setMnemonic(KeyEvent.VK_F);
        menuBar.add(fileMenu);
        JMenuItem menuItem1 = new JMenuItem("Open", KeyEvent.VK_N);
        fileMenu.add(menuItem1);
        JCheckBoxMenuItem caseMenuItem = new JCheckBoxMenuItem("Option_1");
        caseMenuItem.setMnemonic(KeyEvent.VK_C);
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



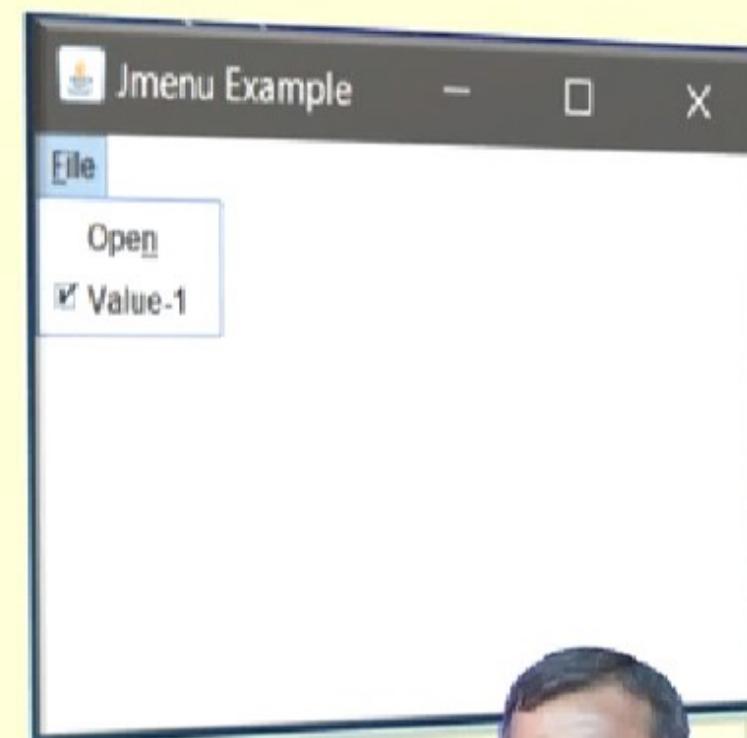
IIT KHARAGPUR





# Java JCheckBoxMenuItem : An example

```
fileMenu.add(caseMenuItem);
    ActionListener aListener = new ActionListener() {
        public void actionPerformed(ActionEvent event) {
            AbstractButton aButton = (AbstractButton) event.getSource();
            boolean selected = aButton.getModel().isSelected();
            String newLabel;
            Icon newIcon;
            if (selected) {
                newLabel = "Value-1";
            } else {
                newLabel = "Value-2";
            }
            aButton.setText(newLabel);
        }
    };
    caseMenuItem.addActionListener(aListener);
frame.setJMenuBar(menuBar);
frame.setSize(350, 250);
frame.setVisible(true);
}
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR

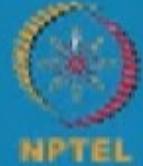


# Questions to think...

- Why Swing components are called lightweights, whereas AWT components are called heavyweight?
- How Swing is platform independent whereas AWT is not so?



IIT KHARAGPUR



NPTEL  
NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR



# Lecture 44

# Swing Programming - II



IIT KHARAGPUR



NPTEL

NPTEL ONLINE  
CERTIFICATION COURSES

# OBJECT ORIENTED PROGRAMMING WITH JAVA

## Java Swing Programming – II

**Debasis Samanta**

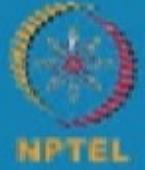
Department of Computer Science & Engineering  
Indian Institute of Technology Kharagpur



## More on Java Swing



IIT KHARAGPUR



NPTEL  
NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



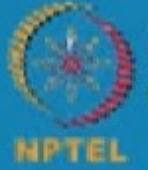
IIT KHARAGPUR



# Java Swing JSeparator



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR



# Class JSeparator

The object of **JSeparator** class is used to provide a general purpose component for implementing divider lines. It is used to draw a line to separate widgets in a Layout. It inherits **JComponent** class.



Below is the declaration for `javax.swing.JSeparator` class.

```
public class JSeparator extends JComponent implements SwingConstants, Accessible
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR



# Class JSeparator : Constructors

<i>Constructor</i>	<i>Description</i>
<code>JSeparator()</code>	Creates a new horizontal separator.
<code>JSeparator(int orientation)</code>	Creates a new separator with the specified horizontal or vertical orientation.



IIT KHARAGPUR



NPTEL  
NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR

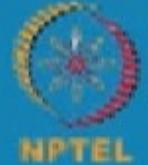


# Class JSeparator : Methods

<i>Methods</i>	<i>Description</i>
<code>void setOrientation(int orientation)</code>	It is used to set the orientation of the separator.
<code>int getOrientation()</code>	It is used to return the orientation of the separator.



IIT KHARAGPUR



NPTEL  
NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



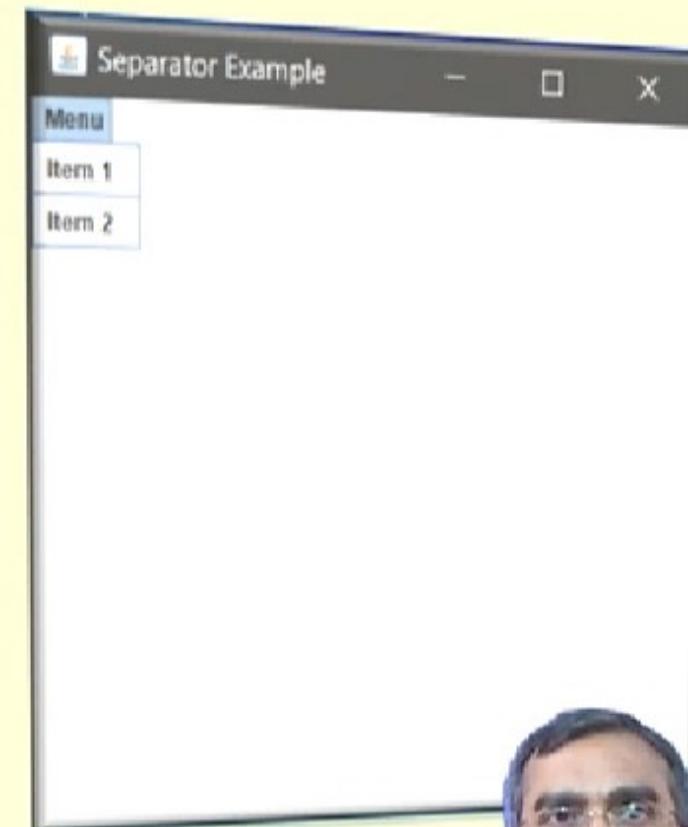
IIT KHARAGPUR





# Java JSeparator : An example

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



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



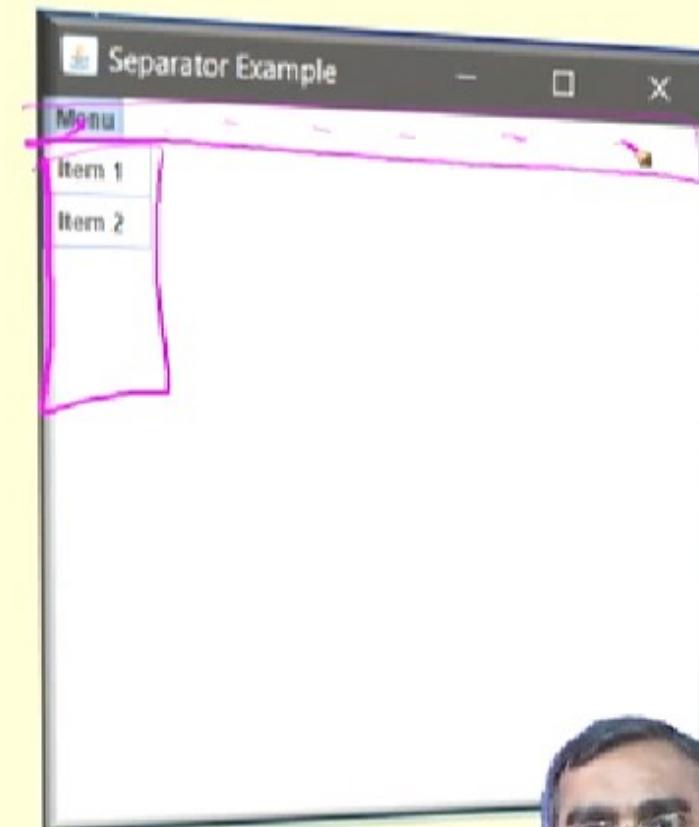
IIT KHARAGPUR





# Java JSeparator : An example

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

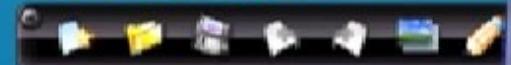


IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA





# Java Swing JProgressBar



IIT KHARAGPUR



NPTEL  
NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR



# Class JProgressBar

The **JProgressBar** class is used to display the progress of the task. It inherits **JComponent** class.



Below is the declaration for `javax.swing.JProgressBar` class.

```
public class JProgressBar extends JComponent implements SwingConstants, Accessible
```



IIT KHARAGPUR



NPTEL  
NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR



# Class JProgressBar : Constructors

<i>Constructor</i>	<i>Description</i>
JProgressBar()	It is used to create a horizontal progress bar but no string text.
JProgressBar(int min, int max)	It is used to create a horizontal progress bar with the specified minimum and maximum value.
JProgressBar(int orient)	It is used to create a progress bar with the specified orientation, it can be either Vertical or Horizontal by using SwingConstants.VERTICAL and SwingConstants.HORIZONTAL constants.
JProgressBar(int orient, int min, int max)	It is used to create a progress bar with the specified orientation, minimum and maximum value.



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR



# Class JProgressBar : Methods

<i>Method</i>	<i>Description</i>
<code>void setStringPainted(boolean b)</code>	It is used to determine whether string should be displayed.
<code>void setString(String s)</code>	It is used to set value to the progress string.
<code>void setOrientation(int orientation)</code>	It is used to set the orientation, it may be either vertical or horizontal by using SwingConstants.VERTICAL and SwingConstants.HORIZONTAL constants.
<code>void setValue(int value)</code>	It is used to set the current value on the progress bar.



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA

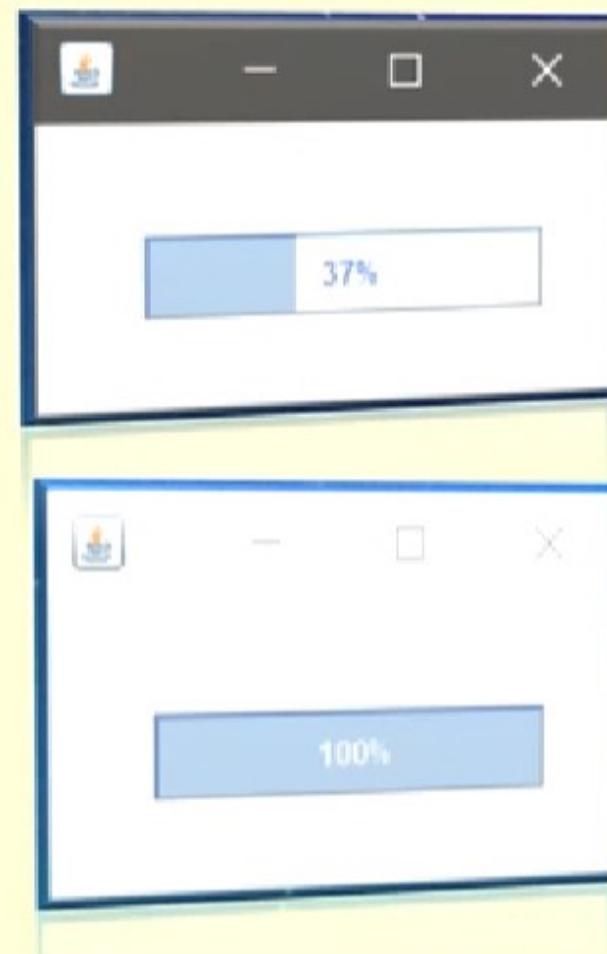


IIT KHARAGPUR



# Java JProgressBar : An example

```
import javax.swing.*;
public class ProgressBarExample extends JFrame{
    JProgressBar jb; int i=0,num=0;
    ProgressBarExample(){
        jb=new JProgressBar(0,2000);
        jb.setBounds(40,40,160,30);
        jb.setValue(0);
        jb.setStringPainted(true);
        add(jb);
        setSize(250,150); setLayout(null);
    }
    public void iterate(){
        while(i<=2000){
            jb.setValue(i);
            i=i+20;
            try{Thread.sleep(150);}catch(Exception e){}
        }
    }
    public static void main(String[] args) {
        ProgressBarExample m=new ProgressBarExample();
        m.setVisible(true);
        m.iterate();
    }
}
```



IIT KHARAGPUR



NPTEL  
ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



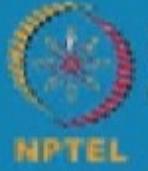
IIT KHARAGPUR



# Java Swing JTree



IIT KHARAGPUR



NPTEL  
NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR





# Class JTree

The **JTree** class is used to display the tree structured data or hierarchical data. **JTree** is a complex component. It has a 'root node' at the top most which is a parent for all nodes in the tree. It inherits **JComponent** class.

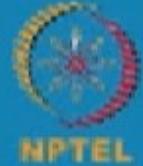


Below is the declaration for `javax.swing.JTree` class.

```
public class JTree extends JComponent implements Scrollable, Accessible
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR

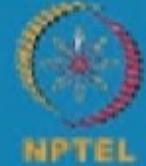


# Class JTree : Constructors

<i>Constructor</i>	<i>Description</i>
<code>JTree()</code>	Creates a JTree with a sample model.
<code>JTree(Object[] value)</code>	Creates a JTree with every element of the specified array as the child of a new root node.
<code>JTree(TreeNode root)</code>	Creates a JTree with the specified TreeNode as its root, which displays the root node.

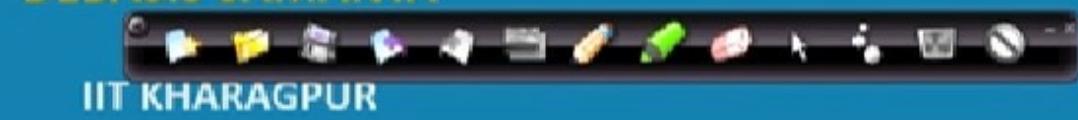


IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

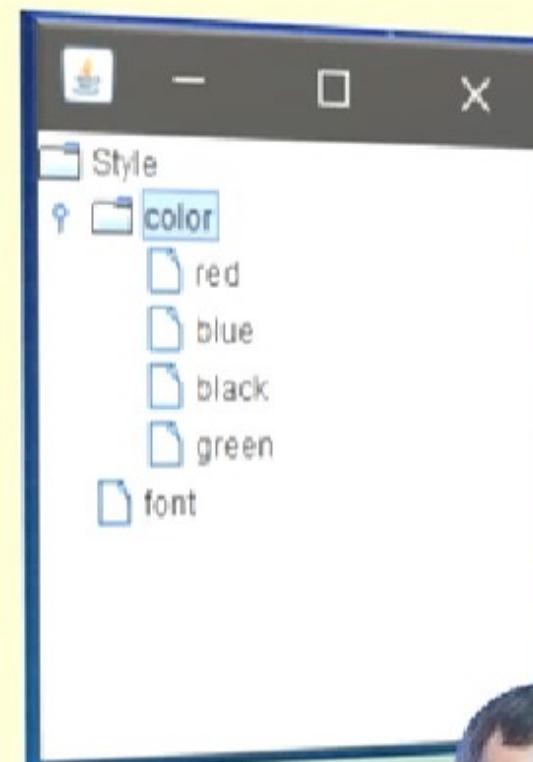
DEBASIS SAMANTA





# Java JTree : An example

```
import javax.swing.*;
import javax.swing.tree.DefaultMutableTreeNode;
public class TreeExample {
JFrame f;
TreeExample(){
    f=new JFrame();
    DefaultMutableTreeNode style = new DefaultMutableTreeNode("Style");
    DefaultMutableTreeNode color = new DefaultMutableTreeNode("color");
    DefaultMutableTreeNode font = new DefaultMutableTreeNode("font");
    style.add(color);    style.add(font);
    DefaultMutableTreeNode red=new DefaultMutableTreeNode("red");
    DefaultMutableTreeNode blue=new DefaultMutableTreeNode("blue");
    DefaultMutableTreeNode black=new DefaultMutableTreeNode("black");
    DefaultMutableTreeNode green=new DefaultMutableTreeNode("green");
    color.add(red); color.add(blue); color.add(black); color.add(green);
    JTree jt=new JTree(style);    f.add(jt);
    f.setSize(200,200);
    f.setVisible(true);
}
public static void main(String[] args) {
    new TreeExample();
}
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA

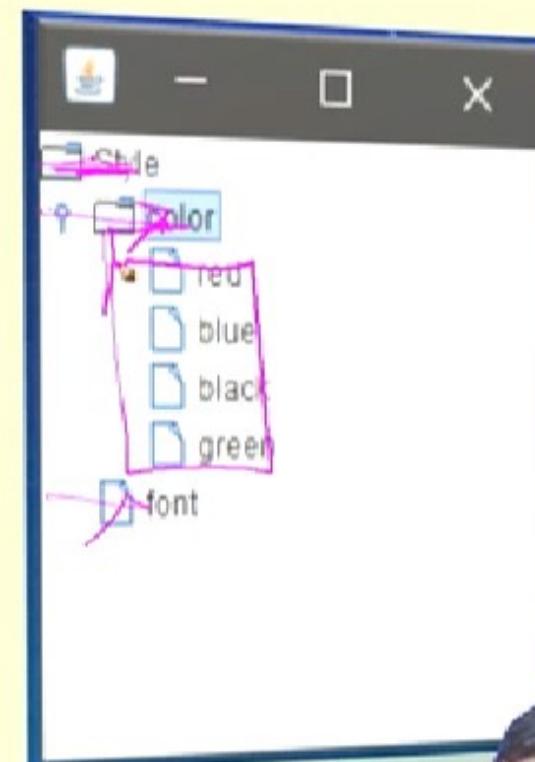


IIT KHARAGPUR



# Java JTree : An example

```
import javax.swing.*;
import javax.swing.tree.DefaultMutableTreeNode;
public class TreeExample {
JFrame f;
TreeExample(){
    f=new JFrame();
    DefaultMutableTreeNode style = new DefaultMutableTreeNode("Style");
    DefaultMutableTreeNode color = new DefaultMutableTreeNode("color");
    DefaultMutableTreeNode font = new DefaultMutableTreeNode("font");
    style.add(color);    style.add(font);
    DefaultMutableTreeNode red=new DefaultMutableTreeNode("red");
    DefaultMutableTreeNode blue=new DefaultMutableTreeNode("blue");
    DefaultMutableTreeNode black=new DefaultMutableTreeNode("black");
    DefaultMutableTreeNode green=new DefaultMutableTreeNode("green");
    color.add(red); color.add(blue); color.add(black); color.add(green);
    JTree jt=new JTree(style);    f.add(jt);
    f.setSize(200,200);
    f.setVisible(true);
}
public static void main(String[] args) {
    new TreeExample();
}
```

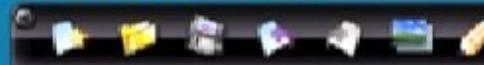


IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

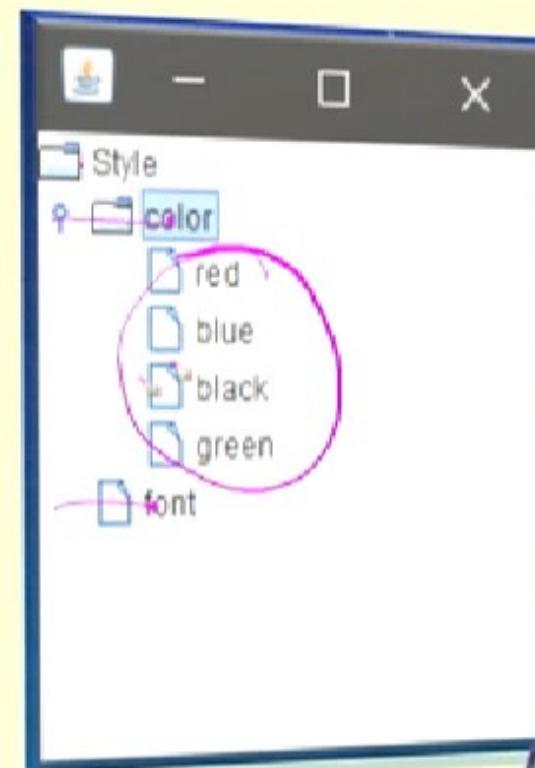
DEBASIS SAMANTA





# Java JTree : An example

```
import javax.swing.*;
import javax.swing.tree.DefaultMutableTreeNode;
public class TreeExample {
JFrame f;
TreeExample(){
    f=new JFrame();
    DefaultMutableTreeNode style = new DefaultMutableTreeNode("Style");
    DefaultMutableTreeNode color = new DefaultMutableTreeNode("color");
    DefaultMutableTreeNode font = new DefaultMutableTreeNode("font");
    style.add(color);    style.add(font);
    DefaultMutableTreeNode red=new DefaultMutableTreeNode("red");
    DefaultMutableTreeNode blue=new DefaultMutableTreeNode("blue");
    DefaultMutableTreeNode black=new DefaultMutableTreeNode("black");
    DefaultMutableTreeNode green=new DefaultMutableTreeNode("green");
    color.add(red); color.add(blue); color.add(black); color.add(green);
    JTree jt=new JTree(style);    f.add(jt);
    f.setSize(200,200);
    f.setVisible(true);
}
public static void main(String[] args) {
    new TreeExample();
}
```





# Java Swing JColorChooser

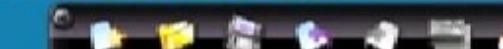


IIT KHARAGPUR



NPTEL  
NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA

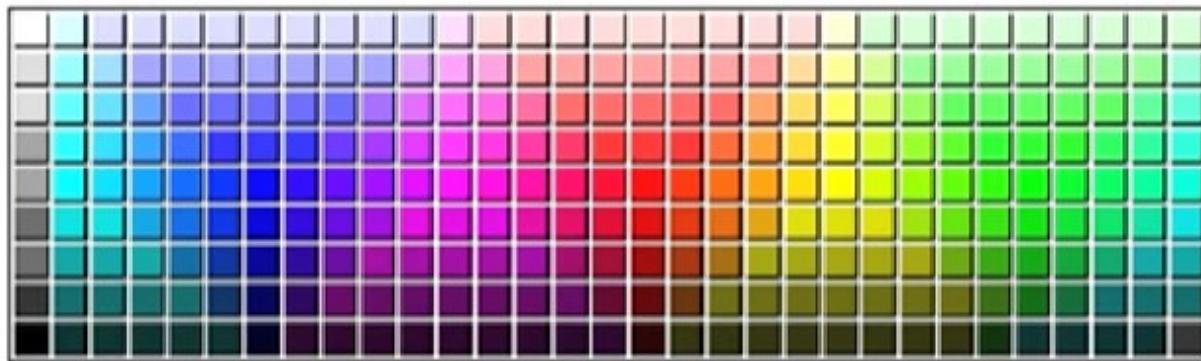


IIT KHARAGPUR



# Class JColorChooser

The **JColorChooser** class is used to create a color chooser dialog box so that user can select any color. It inherits **JComponent** class.



Below is the declaration for `javax.swing.JColorChooser` class

**public class** JColorChooser **extends** JToggleButton **implements** Acc



IIT KHARAGPUR



NPTEL  
NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



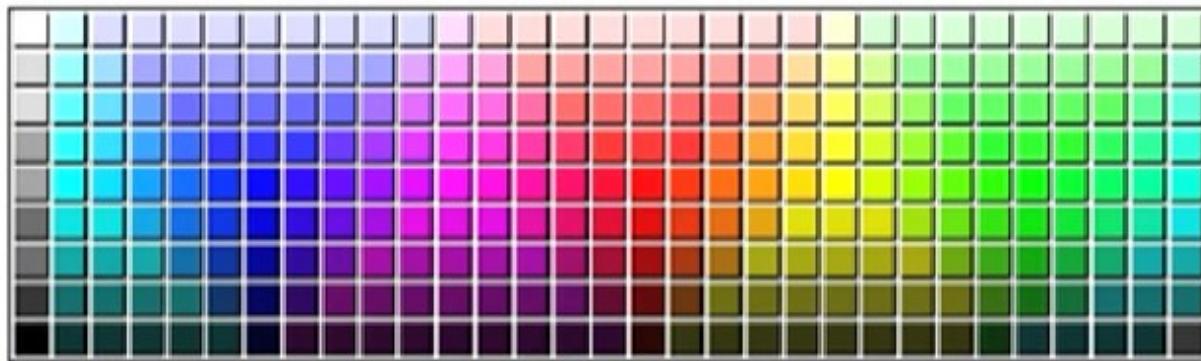
IIT KHARAGPUR





# Class JColorChooser

The **JColorChooser** class is used to create a color chooser dialog box so that user can select any color. It inherits **JComponent** class.



Below is the declaration for `javax.swing.JColorChooser` class.

```
public class JColorChooser extends JToggleButton implements Accessible
```

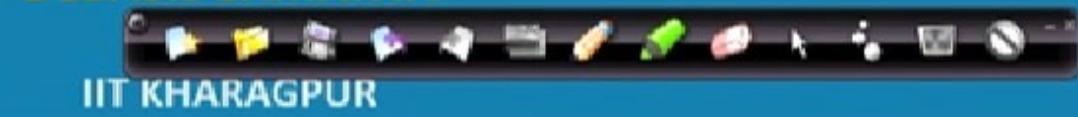


IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA





# Class JColorChooser : Constructors

<i>Constructor</i>	<i>Description</i>
JColorChooser()	It is used to create a color chooser panel with white color initially.
JColorChooser(color initialcolor)	It is used to create a color chooser panel with the specified color initially.



IIT KHARAGPUR



NPTEL  
ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR





# Class JColorChooser : Methods

<i>Method</i>	<i>Description</i>
<code>void addChooserPanel(AbstractColorChooserPanel panel)</code>	It is used to add a color chooser panel to the color chooser.
<code>static Color showDialog(Component c, String title, Color initialColor)</code>	It is used to show the color chooser dialog box.

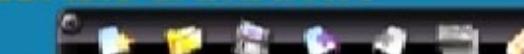


IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA

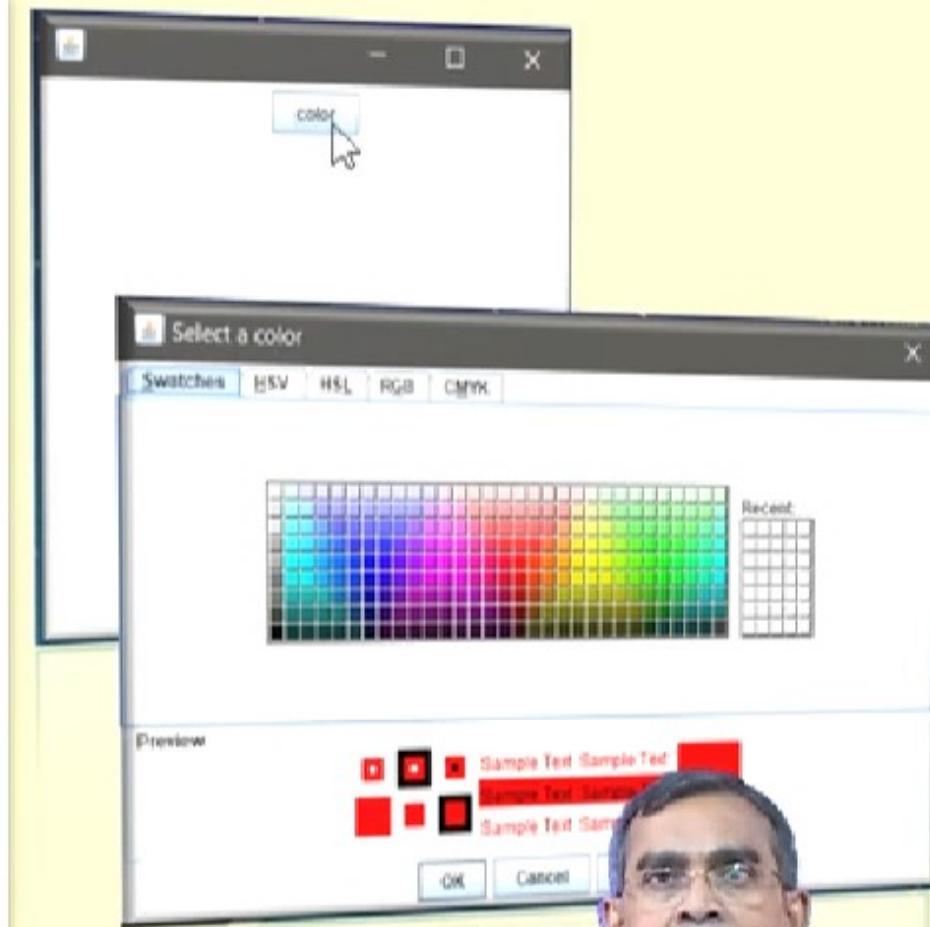


IIT KHARAGPUR



# Java JColorChooser : An example

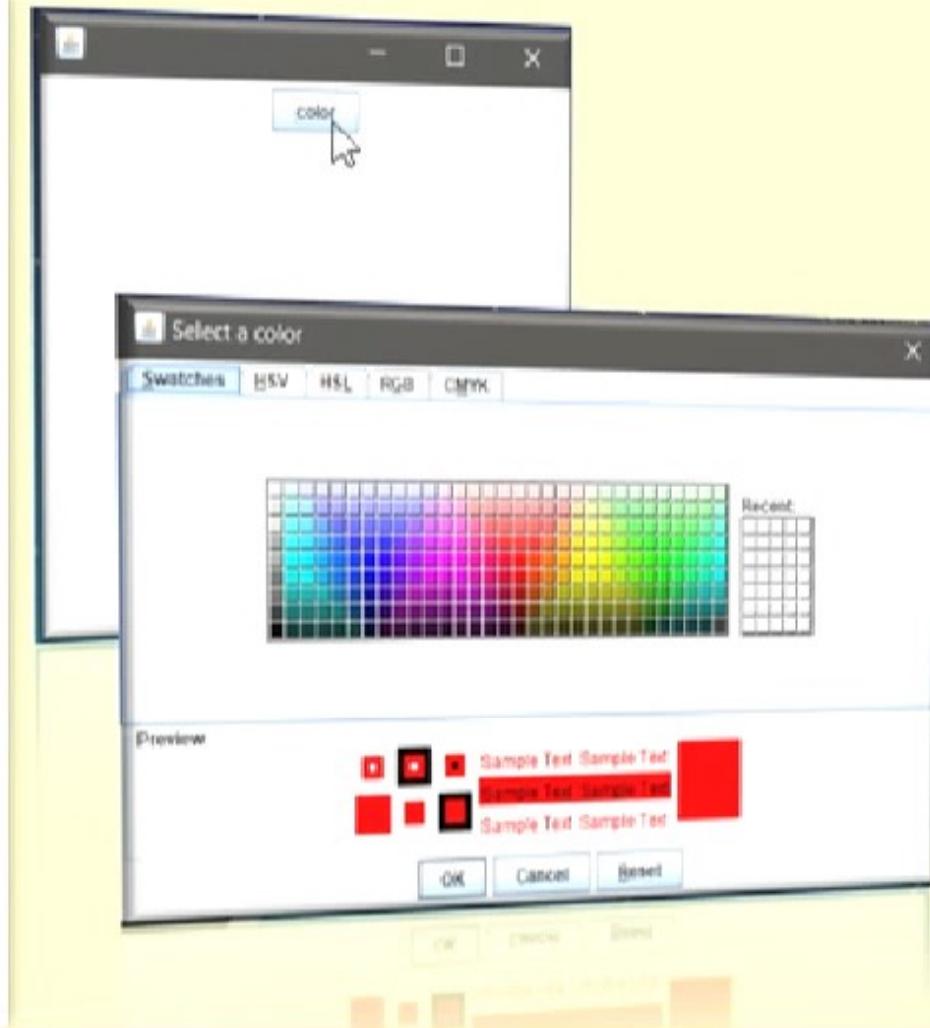
```
import java.awt.event.*;
import java.awt.*;
import javax.swing.*;
public class ColorChooserExample extends JFrame implements ActionListener {
JButton b;Container c;
ColorChooserExample(){
    c=getContentPane();
    c.setLayout(new FlowLayout());
    b=new JButton("color");
    b.addActionListener(this);
    c.add(b);  }
public void actionPerformed(ActionEvent e) {
Color initialcolor=Color.RED;
Color color=JColorChooser.showDialog(this,"Select a color",initialcolor);
c.setBackground(color);
}
public static void main(String[] args) {
    ColorChooserExample ch=new ColorChooserExample();
    ch.setSize(400,400);
    ch.setVisible(true);
    ch.setDefaultCloseOperation(EXIT_ON_CLOSE);
}
```



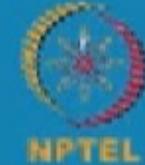


# Java JColorChooser : An example

```
import java.awt.event.*;
import java.awt.*;
import javax.swing.*;
public class ColorChooserExample extends JFrame implements ActionListener {
JButton b;Container c;
ColorChooserExample(){
    c=getContentPane();
    c.setLayout(new FlowLayout());
    b=new JButton("color");
    b.addActionListener(this);
    c.add(b);  }
public void actionPerformed(ActionEvent e) {
Color initialcolor=Color.RED;
Color color=JColorChooser.showDialog(this,"Select a color",initialcolor);
c.setBackground(color);
}
public static void main(String[] args) {
    ColorChooserExample ch=new ColorChooserExample();
    ch.setSize(400,400);
    ch.setVisible(true);
    ch.setDefaultCloseOperation(EXIT_ON_CLOSE);
}
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



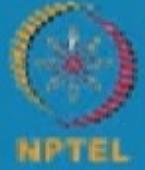
IIT KHARAGPUR



# Java Swing JTabbedPane

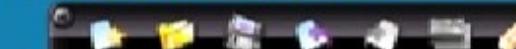


IIT KHARAGPUR



NPTEL  
NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



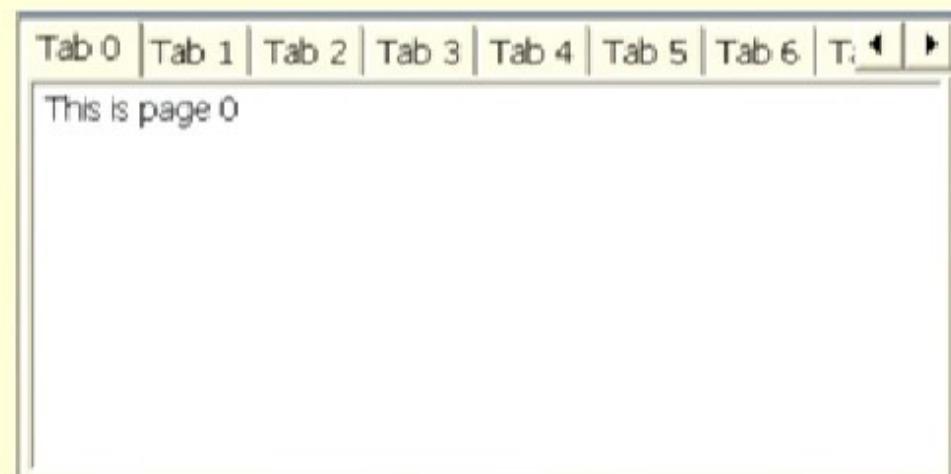
IIT KHARAGPUR





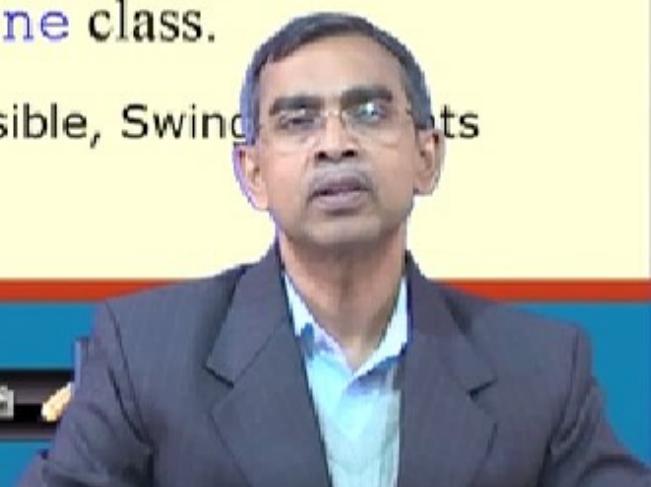
# Class JTabbedPane

The **JTabbedPane** class is used to switch between a group of components by clicking on a tab with a given title or icon. It inherits **JComponent** class



Below is the declaration for `javax.swing.JTabbedPane` class.

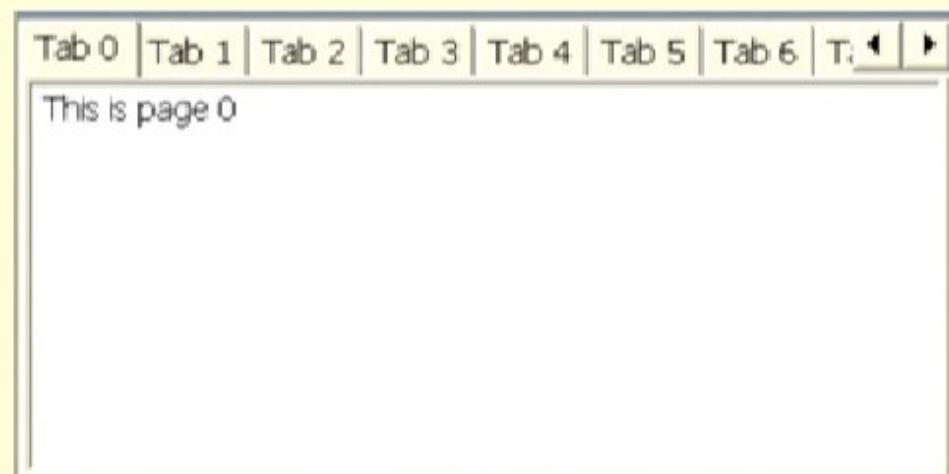
```
public class JTabbedPane extends JComponent implements Serializable, Accessible, SwingConstants
```





# Class JTabbedPane

The **JTabbedPane** class is used to switch between a group of components by clicking on a tab with a given title or icon. It inherits **JComponent** class



Below is the declaration for `javax.swing.JTabbedPane` class.

```
public class JTabbedPane extends JComponent implements Serializable, Accessible, SwingConstants
```



# Class JTabbedPane : Constructors

<i>Constructor</i>	<i>Description</i>
<code>JTabbedPane()</code>	Creates an empty TabbedPane with a default tab placement of <code>JTabbedPane.Top</code> .
<code>JTabbedPane(int tabPlacement)</code>	Creates an empty TabbedPane with a specified tab placement.
<code>JTabbedPane(int tabPlacement, int tabLayoutPolicy)</code>	Creates an empty TabbedPane with a specified tab placement and tab layout policy.



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

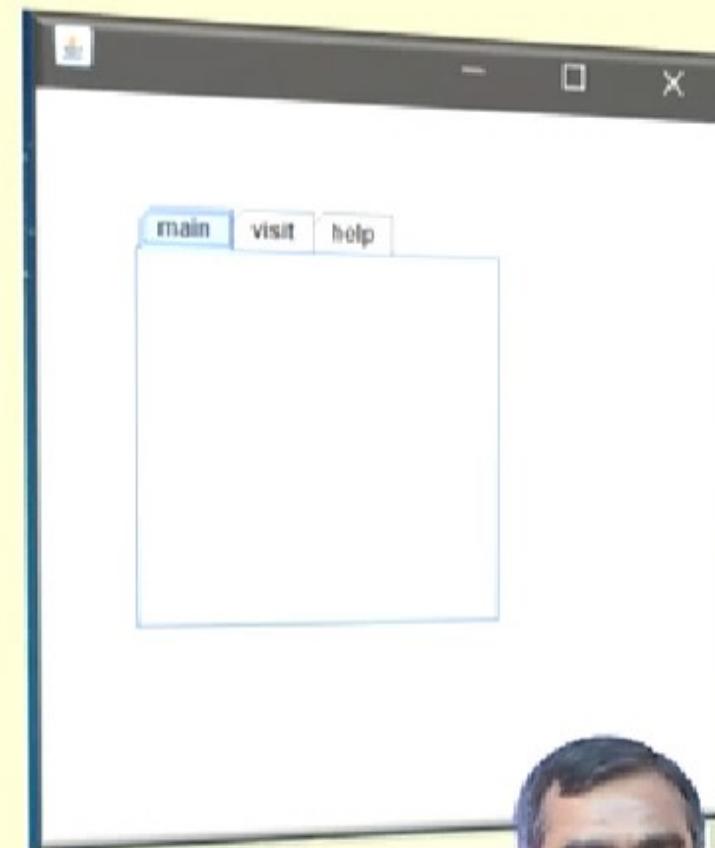
DEBASIS SAMANTA



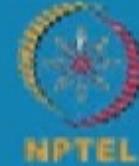


# Java JTabbedPane : An example

```
import javax.swing.*;
public class TabbedPaneExample {
JFrame f;
TabbedPaneExample(){
    f=new JFrame();
    JTextArea ta=new JTextArea(200,200);
    JPanel p1=new JPanel();
    p1.add(ta);
    JPanel p2=new JPanel();
    JPanel p3=new JPanel();
    JTabbedPane tp=new JTabbedPane();
    tp.setBounds(50,50,200,200);
    tp.add("main",p1);
    tp.add("visit",p2);
    tp.add("help",p3);
    f.add(tp);
    f.setSize(400,400);
    f.setLayout(null);
    f.setVisible(true); }
public static void main(String[] args) {
    new TabbedPaneExample();
}
}
```

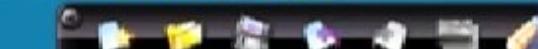


IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR





# Java Swing JSlider



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



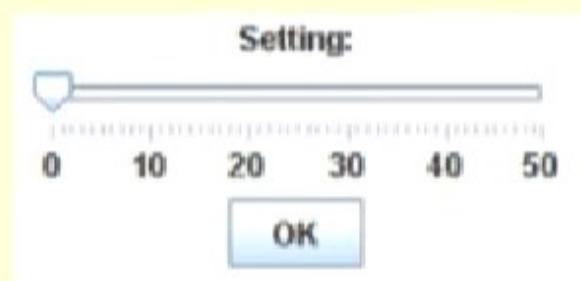
IIT KHARAGPUR





# Class JSlider

The Java **JSlider** class is used to create the slider. By using **JSlider**, a user can select a value from a specific range.



IIT KHARAGPUR



NPTEL  
ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR





# Class JSlider : Constructors

<i>Constructor</i>	<i>Description</i>
<code>JSlider()</code>	creates a slider with the initial value of 50 and range of 0 to 100.
<code>JSlider(int orientation)</code>	creates a slider with the specified orientation set by either <code>JSlider.HORIZONTAL</code> or <code>JSlider.VERTICAL</code> with the range 0 to 100 and initial value 50.
<code>JSlider(int min, int max)</code>	creates a horizontal slider using the given min and max.
<code>JSlider(int min, int max, int value)</code>	creates a horizontal slider using the given min, max and value.
<code>JSlider(int orientation, int min, int max, int value)</code>	creates a slider using the given orientation, min, max and value.





# Class JSlider : Methods

<i>Method</i>	<i>Description</i>
public void setMinorTickSpacing(int n)	is used to set the minor tick spacing to the slider.
public void setMajorTickSpacing(int n)	is used to set the major tick spacing to the slider.
public void setPaintTicks(boolean b)	is used to determine whether tick marks are painted.
public void setPaintLabels(boolean b)	is used to determine whether labels are painted.
public void setPaintTracks(boolean b)	is used to determine whether track is painted.



IIT KHARAGPUR



NPTEL  
NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



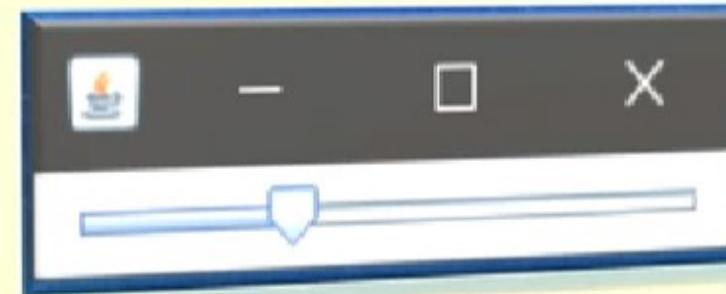
IIT KHARAGPUR



# Java JSlider : An example

```
import javax.swing.*;
public class SliderExample1 extends JFrame{
public SliderExample1() {
JSlider slider = new JSlider(JSlider.HORIZONTAL, 0, 50, 25);
JPanel panel=new JPanel();
panel.add(slider);
add(panel);
}

public static void main(String s[]) {
SliderExample1 frame=new SliderExample1();
frame.pack();
frame.setVisible(true);
}
}
```

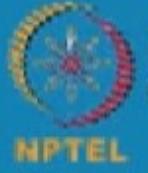




# Java Swing JSpinner



IIT KHARAGPUR



NPTEL  
NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR



# Class JSpinner

The object of **JSpinner** class is a single line input field that allows the user to select a number or an object value from an ordered sequence.



Below is the declaration for `javax.swing.JSpinner` class.

```
public class JSpinner extends JComponent implements Accessible
```



IIT KHARAGPUR



NPTEL  
NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR



# Class JSpinner : Constructors

<i>Constructor</i>	<i>Description</i>
<code>JSpinner()</code>	It is used to construct a spinner with an Integer SpinnerNumberModel with initial value 0 and no minimum or maximum limits.
<code>JSpinner(SpinnerModel model)</code>	It is used to construct a spinner for a given model.



IIT KHARAGPUR



NPTEL  
ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR



# Class JSpinner : Methods

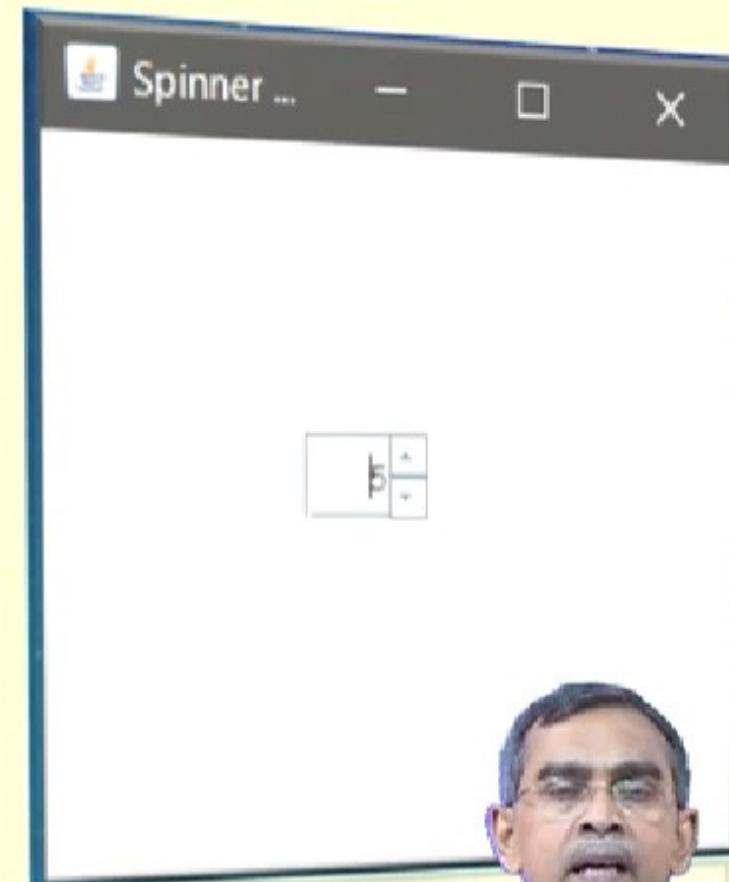
<i>Method</i>	<i>Description</i>
<code>void addChangeListener(ChangeListener listener)</code>	It is used to add a listener to the list that is notified each time a change to the model occurs.
<code>Object getValue()</code>	It is used to return the current value of the model.



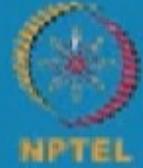


# Java JSpinner : An example

```
import javax.swing.*;  
public class SpinnerExample {  
    public static void main(String[] args) {  
        JFrame f=new JFrame("Spinner Example");  
        SpinnerModel value =  
            new SpinnerNumberModel(5, //initial value  
                0, //minimum value  
                10, //maximum value  
                1); //step  
        JSpinner spinner = new JSpinner(value);  
        spinner.setBounds(100,100,50,30);  
        f.add(spinner);  
        f.setSize(300,300);  
        f.setLayout(null);  
        f.setVisible(true);  
    }  
}
```

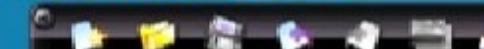


IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA

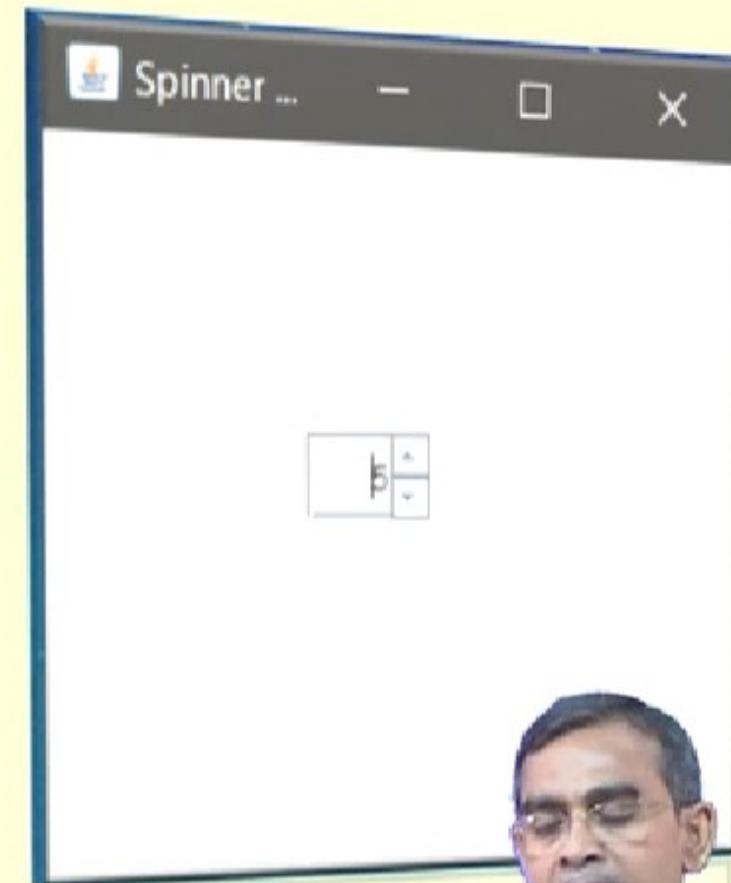


IIT KHARAGPUR



# Java JSpinner : An example

```
import javax.swing.*;  
public class SpinnerExample {  
    public static void main(String[] args) {  
        JFrame f=new JFrame("Spinner Example");  
        SpinnerModel value =  
            new SpinnerNumberModel(5, //initial value  
                0, //minimum value  
                10, //maximum value  
                1); //step  
        JSpinner spinner = new JSpinner(value);  
        spinner.setBounds(100,100,50,30);  
        f.add(spinner);  
        f.setSize(300,300);  
        f.setLayout(null);  
        f.setVisible(true);  
    }  
}
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA

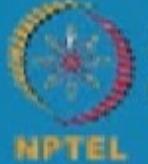




# Java Swing JFileChooser

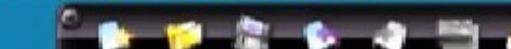


IIT KHARAGPUR



NPTEL  
NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA

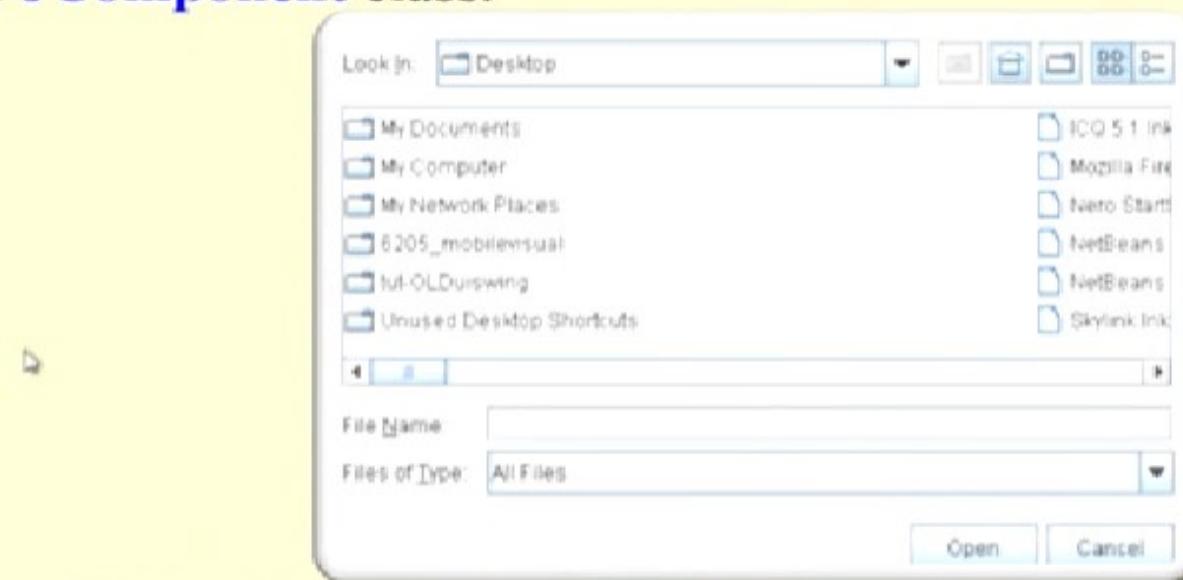


IIT KHARAGPUR



# Class JFileChooser

The object of **JFileChooser** class represents a dialog window from which the user can select file. It inherits **JComponent** class.



Below is the declaration for `javax.swing.JFileChooser` class.

**public class** JFileChooser **extends** JComponent **implements** Accessible



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR

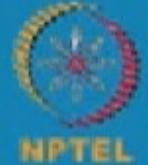


# Class JFileChooser : Constructors

<i>Constructor</i>	<i>Description</i>
JFileChooser()	Constructs a JFileChooser pointing to the user's default directory.
JFileChooser(File currentDirectory)	Constructs a JFileChooser using the given File as the path.
JFileChooser(String currentDirectoryPath)	Constructs a JFileChooser using the given path.



IIT KHARAGPUR



NPTEL  
ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA

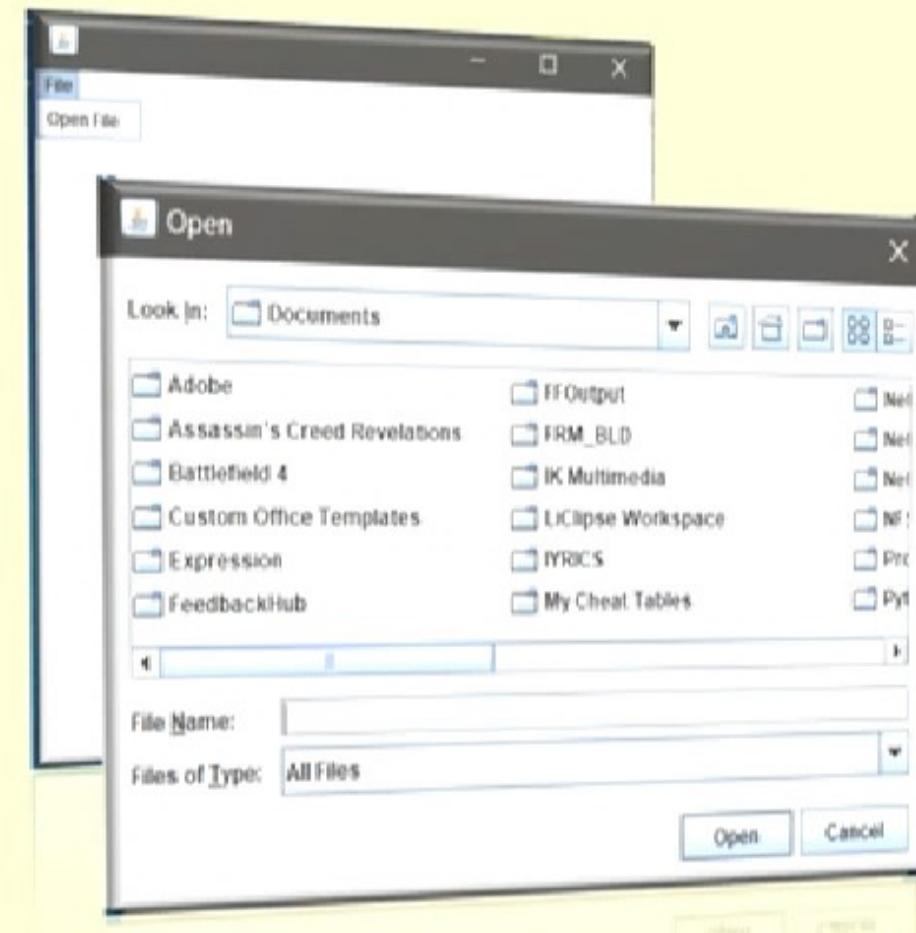


IIT KHARAGPUR

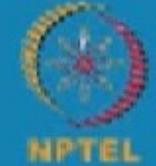


# Java JFileChooser : An example

```
import javax.swing.*;
import java.awt.event.*;
import java.io.*;
public class FileChooserExample extends JFrame implements ActionListener{
JMenuBar mb;
JMenu file;
JMenuItem open;
JTextArea ta;
FileChooserExample(){
open=new JMenuItem("Open File");
open.addActionListener(this);
file=new JMenu("File");
file.add(open);
mb=new JMenuBar();
mb.setBounds(0,0,800,20);
mb.add(file);
ta=new JTextArea(800,800);
ta.setBounds(0,20,800,800);
add(mb);
add(ta);}
public void actionPerformed(ActionEvent e) {
if(e.getSource()==open){
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA

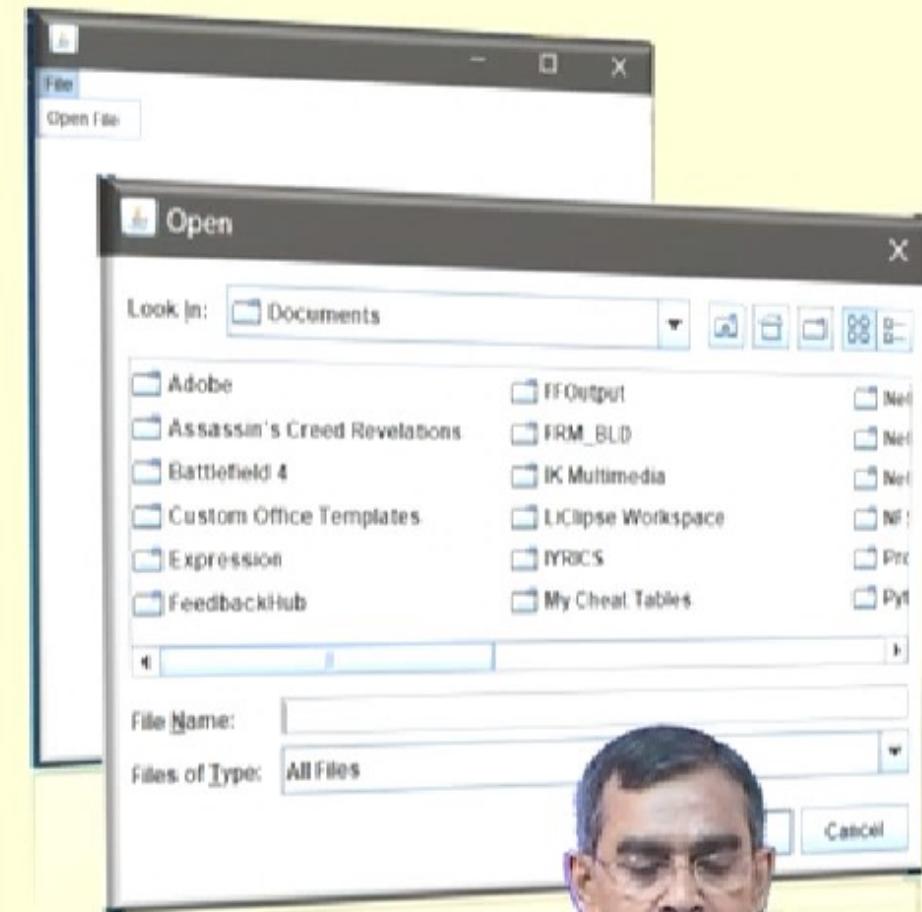


IIT KHARAGPUR



# Java JFileChooser : An example

```
JFileChooser fc=new JFileChooser();
int i=fc.showOpenDialog(this);
if(i==JFileChooser.APPROVE_OPTION){
    File f=fc.getSelectedFile();
    String filepath=f.getPath();
    try{
        BufferedReader br=new BufferedReader(new FileReader(filepath));
        String s1="",s2="";
        while((s1=br.readLine())!=null){
            s2+=s1+"\n";
        }
        ta.setText(s2);
        br.close();
    }catch (Exception ex) {ex.printStackTrace();}
}
public static void main(String[] args) {
    FileChooserExample om=new FileChooserExample();
    om.setSize(500,500);
    om.setLayout(null);
    om.setVisible(true);
    om.setDefaultCloseOperation(EXIT_ON_CLOSE);}
}
```

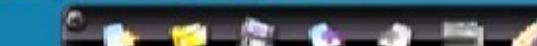


IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR



# Java Swing JToggleButton

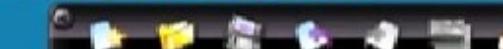


IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR

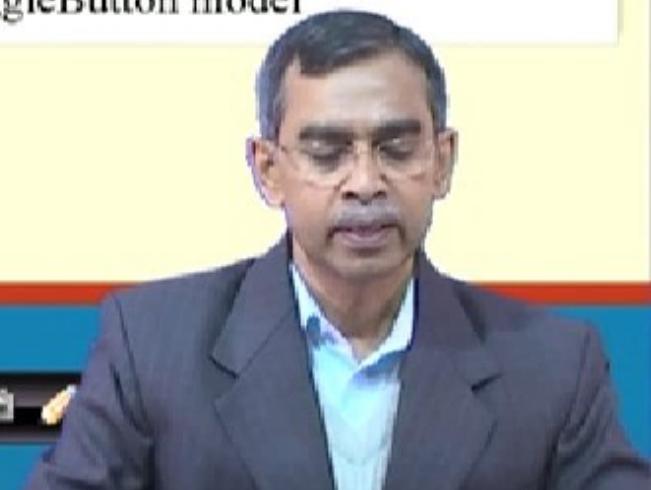


# Class JToggleButton

**JToggleButton** is used to create toggle button, it is two-states button to switch on or off.

## Nested Classes

<i>Modifier and Type</i>	<i>Class</i>	<i>Description</i>
protected class	JToggleButton.AccessibleJToggleButton	This class implements accessibility support for the JToggleButton class.
static class	JToggleButton.ToggleButtonModel	The ToggleButton model





# Class JToggleButton : Constructor

<i>Constructor</i>	<i>Description</i>
JToggleButton ()	It creates an initially unselected toggle button without setting the text or image.
JToggleButton (Action a)	It creates a toggle button where properties are taken from the Action supplied.
JToggleButton (Icon icon)	It creates an initially unselected toggle button with the specified image but no text.
JToggleButton (Icon icon, boolean selected)	It creates a toggle button with the specified image and selection state, but no text.
JToggleButton (String text)	It creates an unselected toggle button with the specified text.
JToggleButton (String text, boolean selected)	It creates a toggle button with the specified text and selection state.
JToggleButton (String text, Icon icon)	It creates a toggle button that has the specified text and image, and that is initially unselected.
JToggleButton (String text, Icon icon, boolean selected)	It creates a toggle button with the specified text, image, and selection state.



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR



# Class JToggleButton : Methods

<i>Modifier and Type</i>	<i>Method</i>	<i>Description</i>
AccessibleContext	getAccessibleContext ()	It gets the AccessibleContext associated with this JToggleButton.
String	getUIClassID ()	It returns a string that specifies the name of the l&f class that renders this component.
protected String	paramString ()	It returns a string representation of this JToggleButton.
void	updateUI ()	It resets the UI property to a value from the current look and feel.



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA

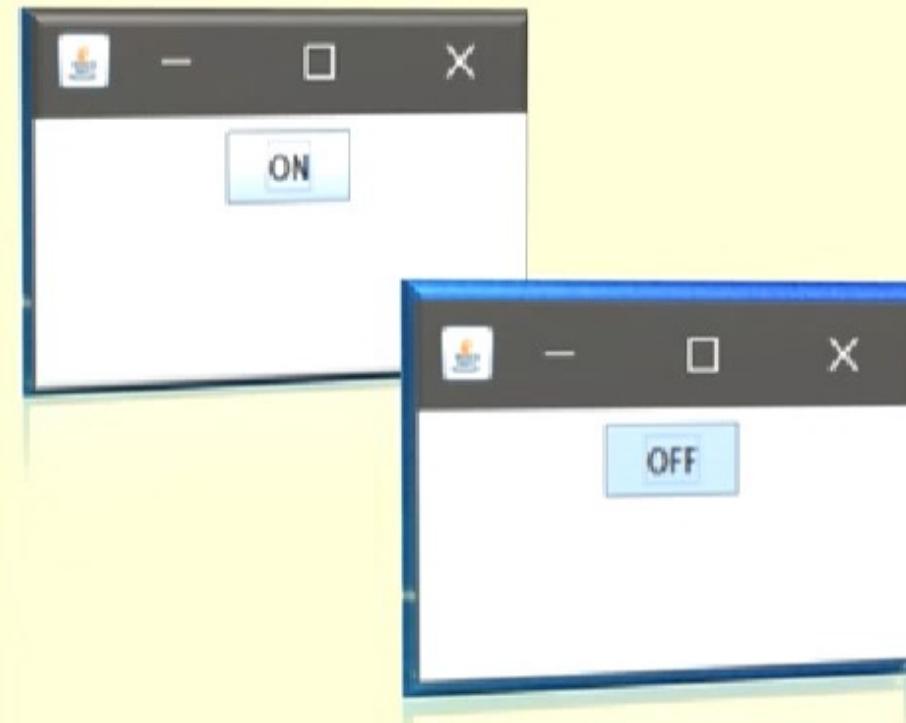


IIT KHARAGPUR



# Java JToggleButton : An example

```
import java.awt.FlowLayout;
import java.awt.event.ItemEvent;
import java.awt.event.ItemListener;
import javax.swing.JFrame;
import javax.swing.JToggleButton;
public class ToggleButtonExample extends JFrame implements
ItemListener {
    public static void main(String[] args) {
        new ToggleButtonExample();
    }
    private JToggleButton button;
    ToggleButtonExample() {
        setTitle("JToggleButton with ItemListener Example");
        setLayout(new FlowLayout());
        setJToggleButton();
        setAction();
        setSize(200, 200);
        setVisible(true);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    }
}
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA

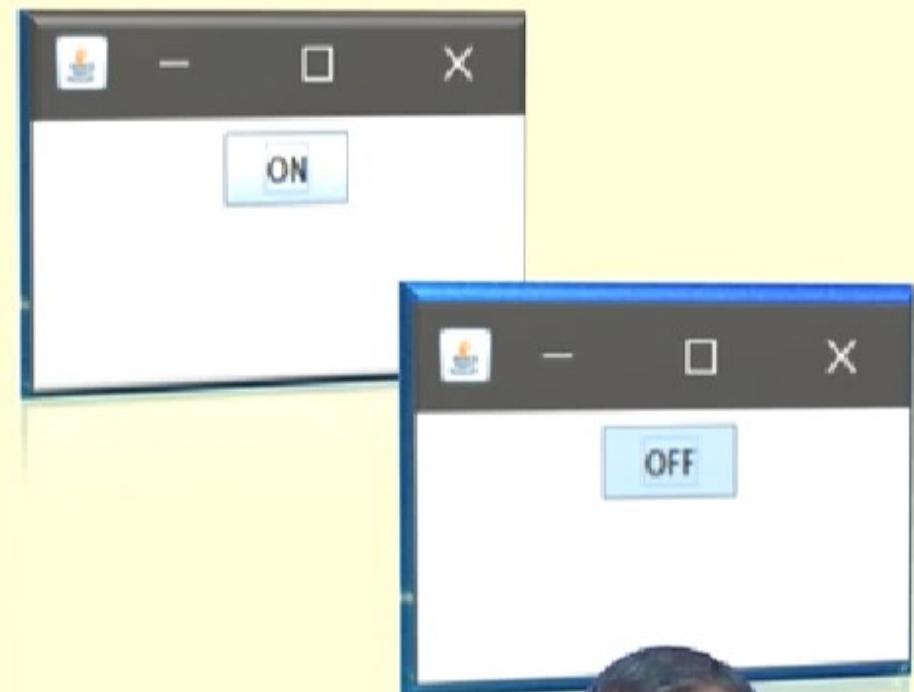


IIT KHARAGPUR



# Java JToggleButton : An example

```
private void setJToggleButton() {  
    button = new JToggleButton("ON");  
    add(button);  
}  
private void setAction() {  
    button.addItemListener(this);  
}  
public void itemStateChanged(ItemEvent eve) {  
    if (button.isSelected())  
        button.setText("OFF");  
    else  
        button.setText("ON");  
}
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR





# Java Swing JToolBar

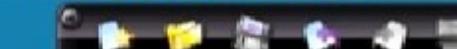


IIT KHARAGPUR



NPTEL  
NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR



# Class JToolBar

**JToolBar** container allows us to group other components, usually buttons with icons in a row or column. **JToolBar** provides a component which is useful for displaying commonly used actions or controls.

## Nested Classes

<i>Modifier and Type</i>	<i>Class</i>	<i>Description</i>
protected class	JToolBar.AccessibleJToolBar ↳	This class implements accessibility support for the JToolBar class.
static class	JToolBar.Separator	A toolbar-specific separator.

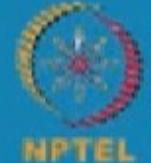


# Class JToolBar : Constructors

<i>Constructor</i>	<i>Description</i>
JToolBar()	It creates a new tool bar; orientation defaults to HORIZONTAL.
JToolBar(int orientation)	It creates a new tool bar with the specified orientation.
JToolBar(String name)	It creates a new tool bar with the specified name.
JToolBar(String name, int orientation)	It creates a new tool bar with a specified name and orientation.



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR

# Class JToolBar : Methods

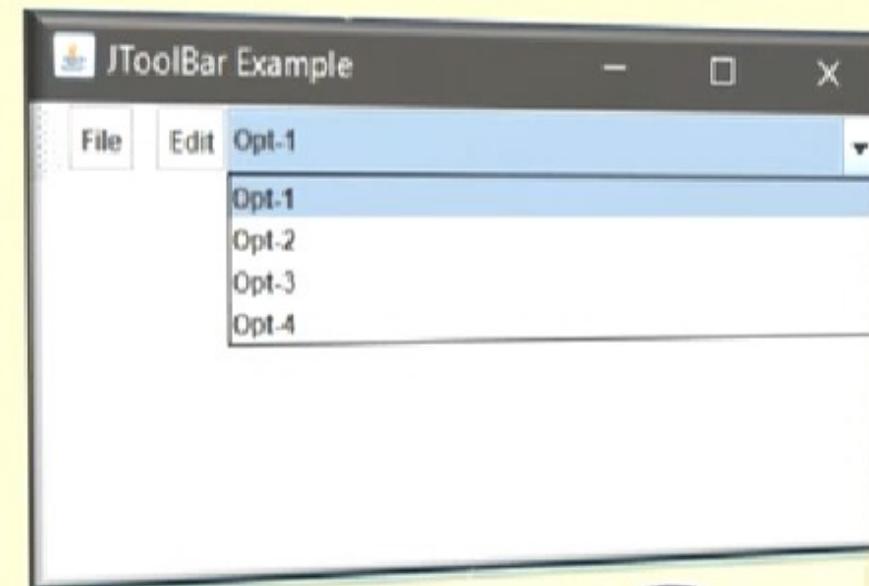
<i>Modifier and Type</i>	<i>Method</i>	<i>Description</i>
JButton	add (Action a)	It adds a new JButton which dispatches the action.
protected void	addImpl (Component comp, Object constraints, int index)	If a JButton is being added, it is initially set to be disabled.
void	addSeparator ()	It appends a separator of default size to the end of the tool bar.
protected PropertyChangeListener	createActionChangeListener (JButton b)	It returns a properly configured PropertyChangeListener which updates the control as changes to the Action occur, or null if the default property change listener for the control is desired.
protected JButton	createActionComponent (Action a)	Factory method which creates the JButton for Actions added to the JToolBar.
ToolBarUI	getUI ()	It returns the tool bar's current UI.
void	setUI (ToolBarUI ui)	It sets the L&F object that renders this component.
void	setOrientation (int o)	It sets the orientation of the tool bar.



# Java JToolBar : An example

```
import java.awt.BorderLayout;
import java.awt.Container;
import javax.swing.JButton;
import javax.swing.JComboBox;
import javax.swing.JFrame;
import javax.swing.JScrollPane;
import javax.swing.JTextArea;
import javax.swing.JToolBar;

public class ToolBarExample {
    public static void main(final String args[]) {
        JFrame myframe = new JFrame("JToolBar Example");
        myframe.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        JToolBar toolbar = new JToolBar();
        toolbar.setRollover(true);
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



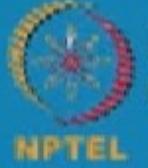
IIT KHARAGPUR



# Java Swing JViewport



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR





# Class JViewport

The **JViewport** class is used to implement scrolling. **JViewport** is designed to support both logical scrolling and pixel-based scrolling. The viewport's child, called the view, is scrolled by calling the **JViewport.setViewPosition()** method.

## Nested Classes

<i>Modifier and Type</i>	<i>Class</i>	<i>Description</i>
protected class	JViewport.AccessibleJViewport	This class implements accessibility support for the Jviewport class.
protected class	JViewport.ViewListener	A listener for the view.



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR



# Class JViewport : Fields

<i>Modifier and Type</i>	<i>Field</i>	<i>Description</i>
static int	BACKINGSTORE_SCROLL_MODE	It draws viewport contents into an offscreen image.
protected Image	backingStoreImage	The view image used for a backing store.
static int	BLIT_SCROLL_MODE	It uses graphics.copyArea to implement scrolling.
protected boolean	isViewSizeSet	True when the viewport dimensions have been determined.
protected Point	lastPaintPosition	The last viewPosition that we've painted, so we know how much of the backing store image is valid.
protected boolean	scrollUnderway	The scrollUnderway flag is used for components like JList.
static int	SIMPLE_SCROLL_MODE	This mode uses the very simple method of redrawing the entire contents of the scrollpane each time it is scrolled.



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR



# Class JViewport : Constructors

<i>Constructor</i>	<i>Description</i>
JViewport()	Creates a JViewport.



IIT KHARAGPUR



NPTEL  
NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR

# Class JViewport : Methods

Modifier and Type	Method	Description
	void addChangeListener(ChangeListener listener)	It adds a ChangeListener to the list that is notified each time the view's size, position, or the viewport's extent size has changed.
protected LayoutManager	createLayoutManager()	Subclasses can override this to install a different layout manager (or null) in the constructor.
protected Jviewport.ViewListener	createViewListener()	It creates a listener for the view.
int	getScrollMode()	It returns the current scrolling mode.
Component	getView()	It returns the JViewport's one child or null.
Point	getViewPosition()	It returns the view coordinates that appear in the upper left hand corner of the viewport, or 0,0 if there's no view.
Dimension	getViewSize()	If the view's size hasn't been explicitly set, return the preferred size, otherwise return the view's current size.
void	setExtentSize(Dimension newExtent)	It sets the size of the visible part of the view using view coordinates.
void	setScrollMode(int mode)	It used to control the method of scrolling the viewport contents.
void	setViewSize(Dimension newSize)	It sets the size of the view.

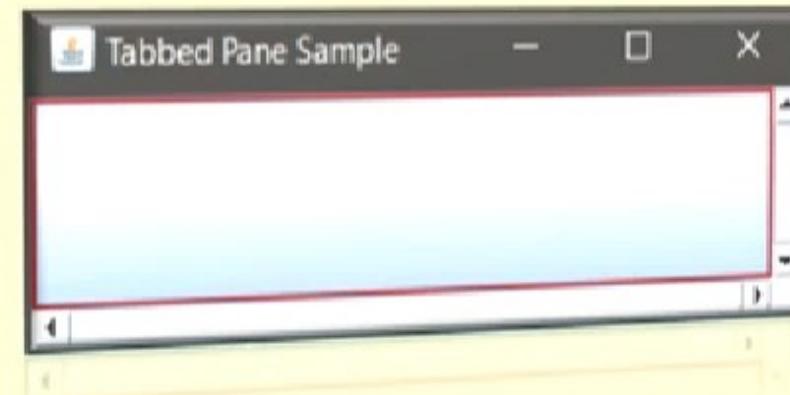




# Java JViewport : An example

```
import java.awt.BorderLayout;
import java.awt.Color;
import java.awt.Dimension;
import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JScrollPane;
import javax.swing.border.LineBorder;
public class ViewPortClass2 {
    public static void main(String[] args) {
        JFrame frame = new JFrame("Tabbed Pane Sample");
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        JLabel label = new JLabel("Label");
        label.setPreferredSize(new Dimension(1000, 1000));
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR





# Java JViewport : An example

```
JScrollPane jScrollPane = new JScrollPane(label);

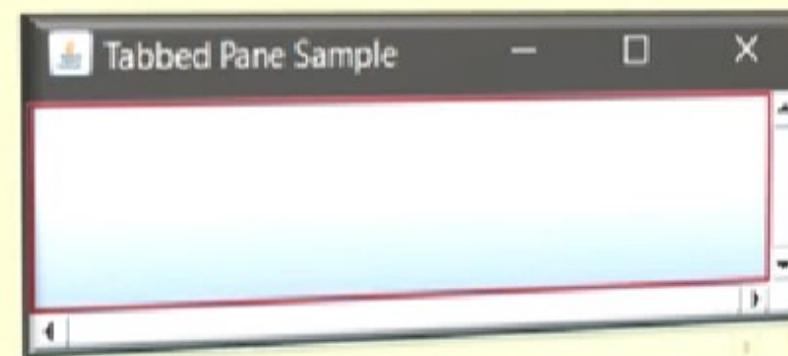
JButton jButton1 = new JButton();

jScrollPane.setHorizontalScrollBarPolicy(JScrollPane.HORIZONTAL_SCROLLBAR_ALWAYS);

jScrollPane.setVerticalScrollBarPolicy(JScrollPane.VERTICAL_SCROLLBAR_ALWAYS);
    jScrollPane.setViewportBorder(new LineBorder(Color.RED));
    jScrollPane.setViewport().add(jButton1, null);

frame.add(jScrollPane, BorderLayout.CENTER);
frame.setSize(400, 150);
frame.setVisible(true);
}

}
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR

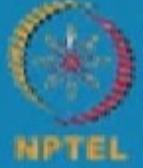




# Java Swing Container



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



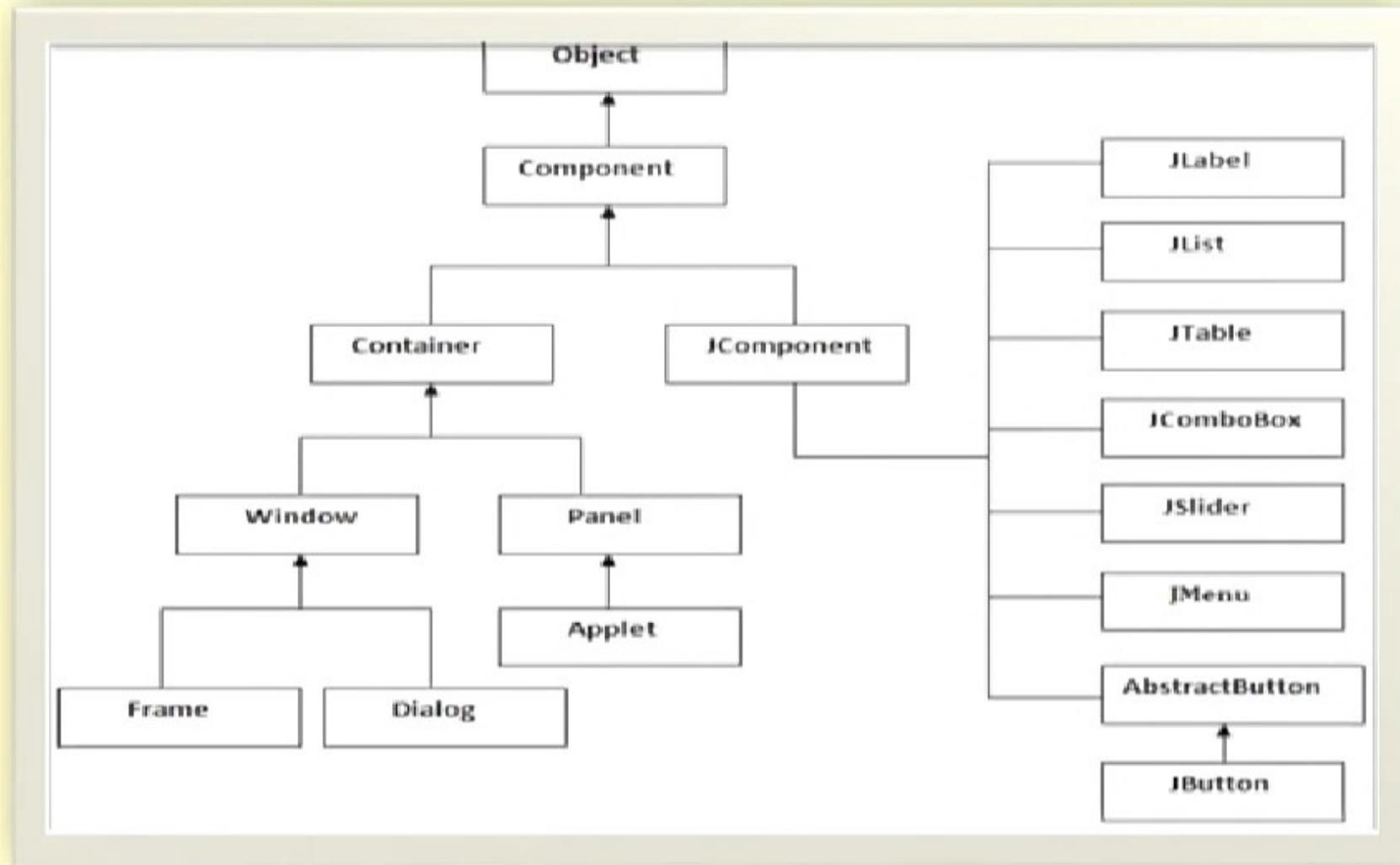
IIT KHARAGPUR



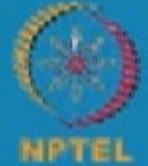


# Hierarchy of Java Swing classes

The hierarchy of Java Swing APIs



IIT KHARAGPUR



NPTEL  
ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



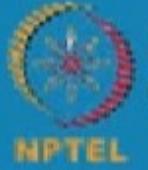
IIT KHARAGPUR



# Java Swing JPanel



IIT KHARAGPUR



NPTEL  
NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



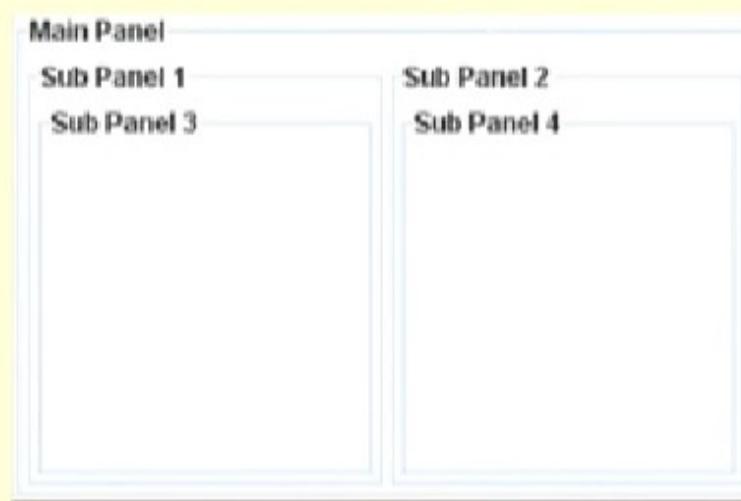
IIT KHARAGPUR



# Class JPanel

The **JPanel** is a simplest container class. It provides space in which an application can attach any other component. It inherits the **JComponents** class.

It doesn't have title bar.



Below is the declaration for `javax.swing.JPanel` class.

**public class JPanel extends JComponent implements Accessible**



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR

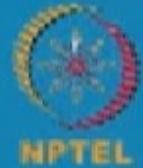


# Class JPanel : Constructors

<i>Constructor</i>	<i>Description</i>
<code>JPanel()</code>	It is used to create a new JPanel with a double buffer and a flow layout.
<code>JPanel(boolean isDoubleBuffered)</code>	It is used to create a new JPanel with FlowLayout and the specified buffering strategy.
<code>JPanel(LayoutManager layout)</code>	It is used to create a new JPanel with the specified layout manager.



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



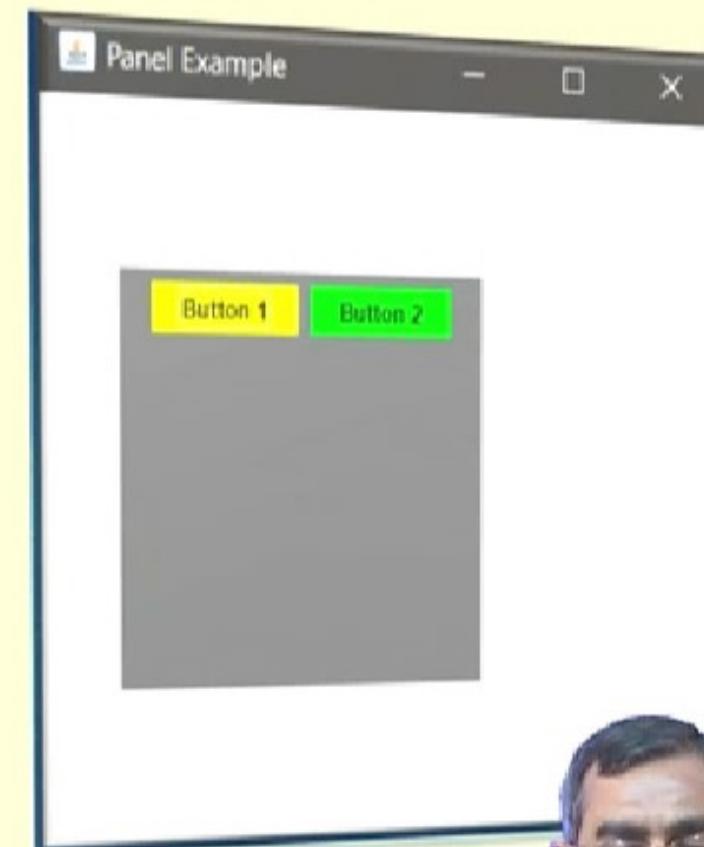
IIT KHARAGPUR





# Java JPanel : An example

```
import java.awt.*;
import javax.swing.*;
public class PanelExample {
    PanelExample() {
        JFrame f = new JFrame("Panel Example");
        JPanel panel=new JPanel();
        panel.setBounds(40,80,200,200);
        panel.setBackground(Color.gray);
        JButton b1=new JButton("Button 1");
        b1.setBounds(50,100,80,30);
        b1.setBackground(Color.yellow);
        JButton b2=new JButton("Button 2");
        b2.setBounds(100,100,80,30);
        b2.setBackground(Color.green);
        panel.add(b1); panel.add(b2);
        f.add(panel);
        f.setSize(400,400);
        f.setLayout(null);
        f.setVisible(true);
    }
    public static void main(String args[]) {
        new PanelExample();
    }
}
```

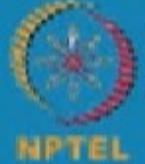




# Java Swing JFrame

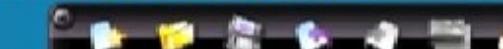


IIT KHARAGPUR



NPTEL  
NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR



# Class JFrame

The **javax.swing.JFrame** class is a type of container which inherits the **java.awt.Frame** class. **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.

## Nested Class

<i>Modifier and Type</i>	<i>Class</i>	<i>Description</i>
protected class	JFrame.AccessibleFrame	This class implements accessibility support for the JFrame class.



# Class JFrame : Fields

<i>Modifier and Type</i>	<i>Field</i>	<i>Description</i>
protected AccessibleContext	accessibleContext	The accessible context property.
static int	EXIT_ON_CLOSE	The exit application default window close operation.
protected JRootPane	rootPane	The JRootPane instance that manages the contentPane and optional menuBar for this frame, as well as the glassPane.
protected boolean	rootPaneCheckingEnabled	If true then calls to add and setLayout will be forwarded to the contentPane.



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR

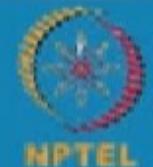


# Class JFrame : Constructors

<i>Constructor</i>	<i>Description</i>
JFrame()	It constructs a new frame that is initially invisible.
JFrame(GraphicsConfiguration gc)	It creates a Frame in the specified GraphicsConfiguration of a screen device and a blank title.
JFrame(String title)	It creates a new, initially invisible Frame with the specified title.
JFrame(String title, GraphicsConfiguration gc)	It creates a JFrame with the specified title and the specified GraphicsConfiguration of a screen device.



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR



# Class JFrame : Methods

<i>Modifier and Type</i>	<i>Method</i>	<i>Description</i>
protected void	addImpl(Component comp, Object constraints, int index)	Adds the specified child Component.
protected JRootPane	createRootPane()	Called by the constructor methods to create the default rootPane.
protected void	frameInit()	Called by the constructors to init the JFrame properly.
void	setContentPane(Container contentPane)	It sets the contentPane property
static void	setDefaultLookAndFeelDecorated(boolean defaultLookAndFeelDecorated)	Provides a hint as to whether or not newly created JFrames should have their Window decorations (such as borders, widgets to close the window, title...) provided by the current look and feel.
void	setIconImage(Image image)	It sets the image to be displayed as the icon for this window.
void	setJMenuBar(JMenuBar menuBar)	It sets the menubar for this frame.
void	setLayeredPane(JLayeredPane layeredPane)	It sets the layeredPane property.
JRootPane	getRootPane()	It returns the rootPane object for this frame.
TransferHandler	getTransferHandler()	It gets the transferHandler property.



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



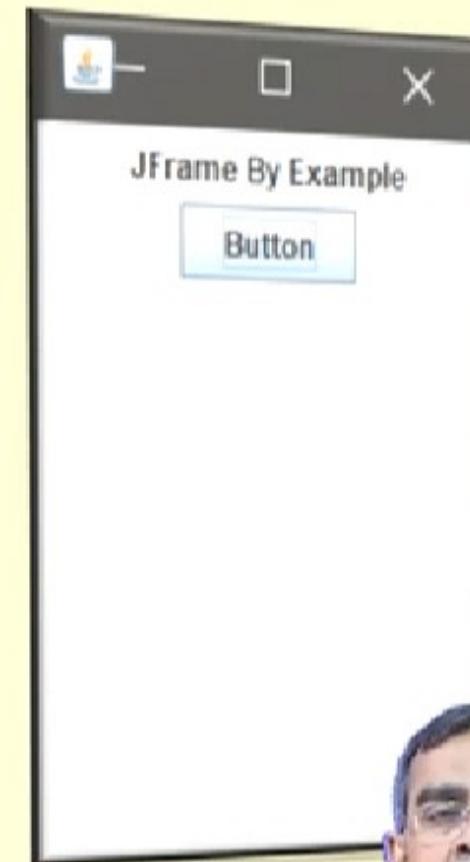
IIT KHARAGPUR





# Java JFrame : An example

```
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);
    }
}
```

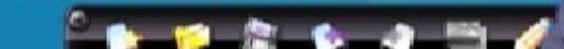


IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR





# Java Swing Calculator

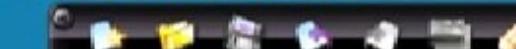


IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR

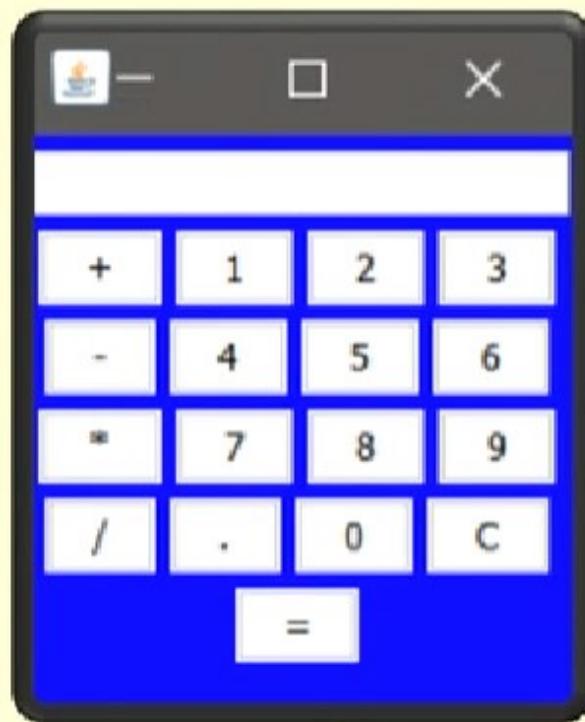




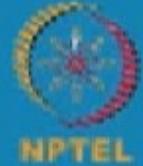
# Calculator with Swing : JCalculator

## Methods used :

- `add(Component c)` : adds component to container.
- `addActionListener(ActionListener d)` : add `actionListener` for specified component
- `setBackground(Color c)` : sets the background color of the specified container
- `setSize(int a, int b)` : sets the size of container to specified dimensions.
- `setText(String s)` : sets the text of the label to s.
- `getText()` : returns the text of the label.

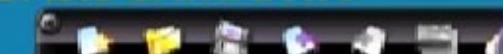


IIT KHARAGPUR



NPTEL  
NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA

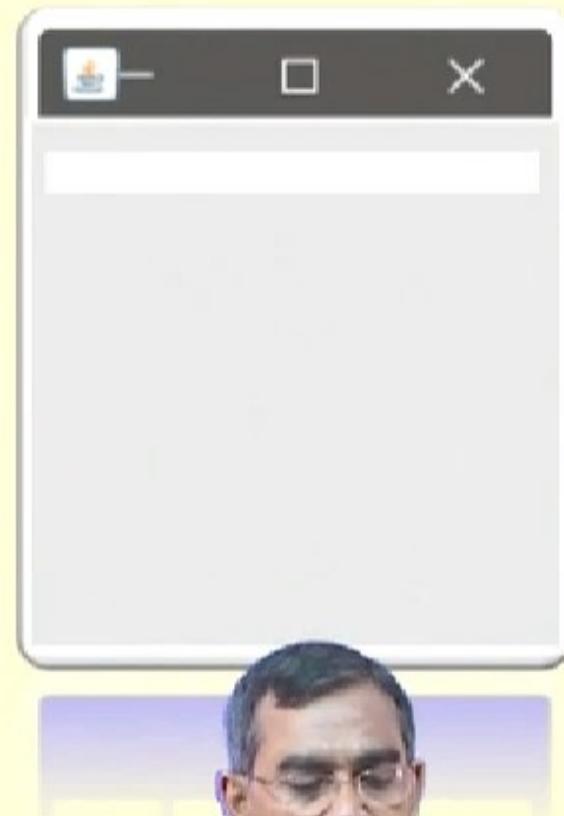


IIT KHARAGPUR



# Jcalculator design : Step-1

```
// Java program to create a simple calculator
// with basic +, -, /, * using java swing elements
import java.awt.event.*;
import javax.swing.*;
import java.awt.*;
class Calculator extends JFrame implements ActionListener {
    // create a frame
    static JFrame f;
    // create a textfield
    static JTextField l;
    // store oprerator and operands
    String s0, s1, s2;
    // default constructor
    Calculator()
    {
        s0 = s1 = s2 = "";
    }
    // main function
    public static void main(String args[])
    {
        // create a frame
        f = new JFrame("Swing Calculator");
    }
}
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA

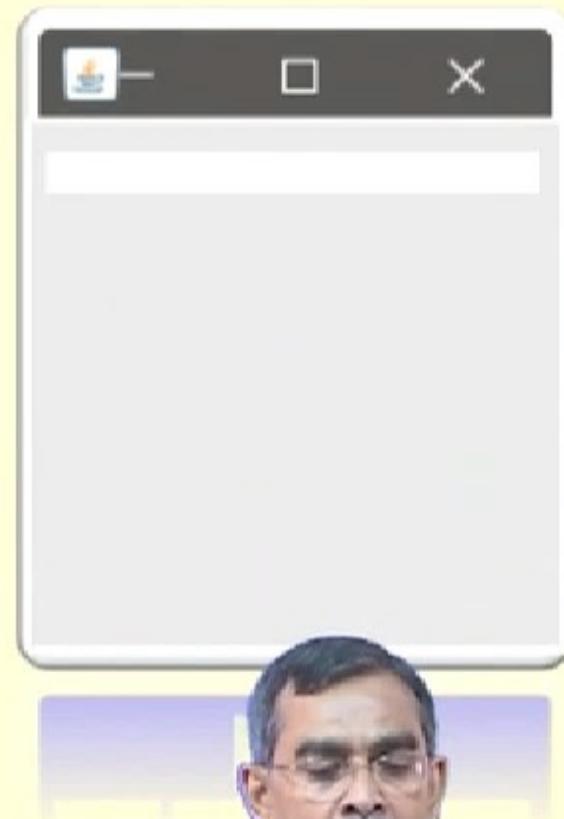


IIT KHARAGPUR



# JCalculator design : Step-2

```
try {
    // set look and feel
    UIManager.setLookAndFeel(UIManager.getSystemLookAndFeelClassName());
}
catch (Exception e) {
    System.err.println(e.getMessage());
}
// create a object of class
Calculator c = new Calculator();
// create a textfield
l = new JTextField(16);
// set the textfield to non editable
l.setEditable(false);
// create number buttons and some operators
JButton b0, b1, b2, b3, b4, b5, b6, b7, b8, b9, ba, bs, bd, bm, be, beq, beql;
// create number buttons
b0 = new JButton("0");
b1 = new JButton("1");
b2 = new JButton("2");
b3 = new JButton("3");
b4 = new JButton("4");
b5 = new JButton("5");
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA

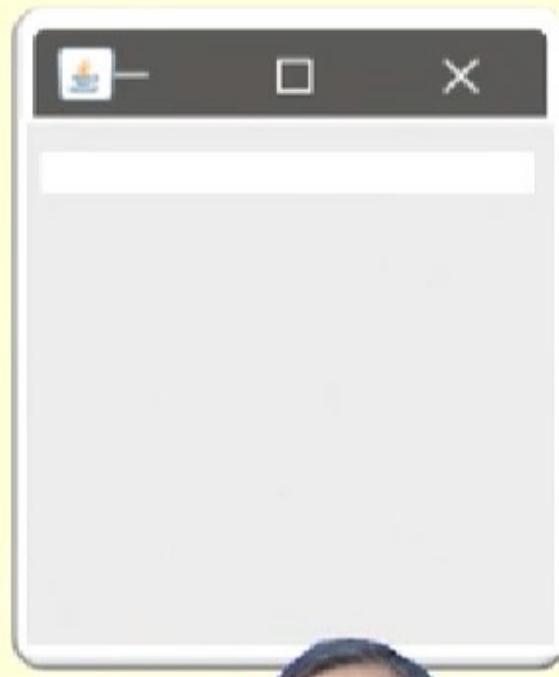


IIT KHARAGPUR



# JCalculator design : Step-2

```
try {
    // set look and feel
    UIManager.setLookAndFeel(UIManager.getSystemLookAndFeelClassName());
}
catch (Exception e) {
    System.err.println(e.getMessage());
}
// create a object of class
Calculator c = new Calculator();
// create a textfield
l = new JTextField(16);
// set the textfield to non editable
l.setEditable(false);
// create number buttons and some operators
JButton b0, b1, b2, b3, b4, b5, b6, b7, b8, b9, ba, bs, bd, bm, be, beq, beql;
// create number buttons
b0 = new JButton("0");
b1 = new JButton("1");
b2 = new JButton("2");
b3 = new JButton("3");
b4 = new JButton("4");
b5 = new JButton("5");
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

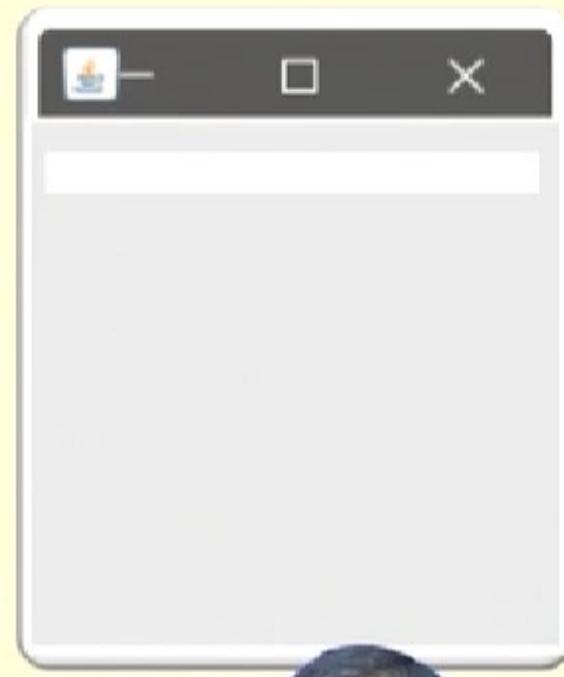
DEBASIS SAMANTA





# JCalculator design : Step-2

```
try {
    // set look and feel
    UIManager.setLookAndFeel(UIManager.getSystemLookAndFeelClassName());
}
catch (Exception e) {
    System.err.println(e.getMessage());
}
// create a object of class
Calculator c = new Calculator();
// create a textfield
l = new JTextField(16);
// set the textfield to non editable
l.setEditable(false);
// create number buttons and some operators
JButton b0, b1, b2, b3, b4, b5, b6, b7, b8, b9, ba, bs, bd, bm, be, beq, beql;
// create number buttons
b0 = new JButton("0");
b1 = new JButton("1");
b2 = new JButton("2");
b3 = new JButton("3");
b4 = new JButton("4");
b5 = new JButton("5");
```

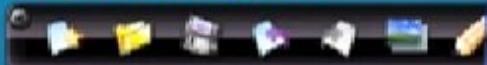


IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

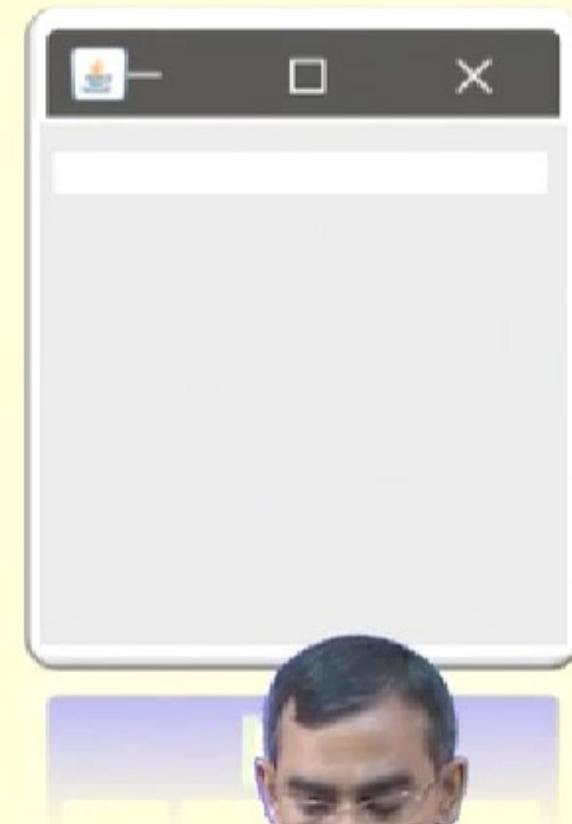
DEBASIS SAMANTA





# JCalculator design : Step-3

```
b6 = new JButton("6");
b7 = new JButton("7");
b8 = new JButton("8");
b9 = new JButton("9");
// equals button
beq1 = new JButton("=");
// create operator buttons
ba = new JButton("+");
bs = new JButton("-");
bd = new JButton("/");
bm = new JButton("*");
beq = new JButton("C");
// create . button
be = new JButton(".");
// create a panel
JPanel p = new JPanel();
// add action listeners
bm.addActionListener(c);
bd.addActionListener(c);
bs.addActionListener(c);
ba.addActionListener(c);
b9.addActionListener(c);
```

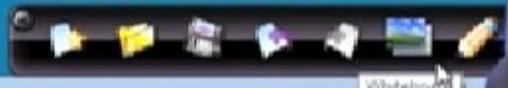


IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA





# JCalculator design : Step-4

```
b8.addActionListener(c);  
b7.addActionListener(c);  
b6.addActionListener(c);  
b5.addActionListener(c);  
b4.addActionListener(c);  
b3.addActionListener(c);  
b2.addActionListener(c);  
b1.addActionListener(c);  
b0.addActionListener(c);  
be.addActionListener(c);  
beq.addActionListener(c);  
beql.addActionListener(c);  
// add elements to panel  
p.add(l);  
p.add(ba);  
p.add(b1);  
p.add(b2);  
p.add(b3);  
p.add(bs);  
p.add(b4);  
p.add(b5);  
p.add(b6);
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA





# JCalculator design : Step-5

```
p.add(bm);
p.add(b7);
p.add(b8);
p.add(b9);
p.add(bd);
p.add(be);
p.add(b0);
p.add(beg);
p.add(beg1);
// set Background of panel
p.setBackground(Color.blue);
// add panel to frame
f.add(p);
f.setSize(200, 220);
f.show();
}

public void actionPerformed(ActionEvent e)
{
    String s = e.getActionCommand();
    // if the value is a number
    if ((s.charAt(0) >= '0' && s.charAt(0) <= '9') || s.charAt(0) == '.') {
        // if operand is present then add to second no
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR



# JCalculator design : Step-6

```
if (!s1.equals(""))
    s2 = s2 + s;
else
    s0 = s0 + s;
// set the value of text
l.setText(s0 + s1 + s2);
}
else if (s.charAt(0) == 'C') {
    // clear the one letter
    s0 = s1 = s2 = "";
    // set the value of text
    l.setText(s0 + s1 + s2);
}
else if (s.charAt(0) == '=') {
    double te;
    // store the value in 1st
    if (s1.equals("+"))
        te = (Double.parseDouble(s0) + Double.parseDouble(s2));
    else if (s1.equals("-"))
        te = (Double.parseDouble(s0) - Double.parseDouble(s2));
    else if (s1.equals("/"))
        te = (Double.parseDouble(s0) / Double.parseDouble(s2));
    else if (s1.equals("*"))
        te = (Double.parseDouble(s0) * Double.parseDouble(s2));
    l.setText(te + "");
}
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR



# JCalculator design : Step-7

```
else
    te = (Double.parseDouble(s0) * Double.parseDouble(s2));
    // convert it to string
    s0 = Double.toString(te);
    // place the operator
    s1 = s;
    // make the operand blank
    s2 = "";
}
// set the value of text
l.setText(s0 + s1 + s2);
}
}
```



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR

# Questions to think...

- Can we mix both Swing and AWT elements in the same program?
- What is the class counts in AWT and Swing?



IIT KHARAGPUR



NPTEL ONLINE  
CERTIFICATION COURSES

DEBASIS SAMANTA



IIT KHARAGPUR