Java Swing
Short Notes

Java GUI Application

AONG CHO MARMA

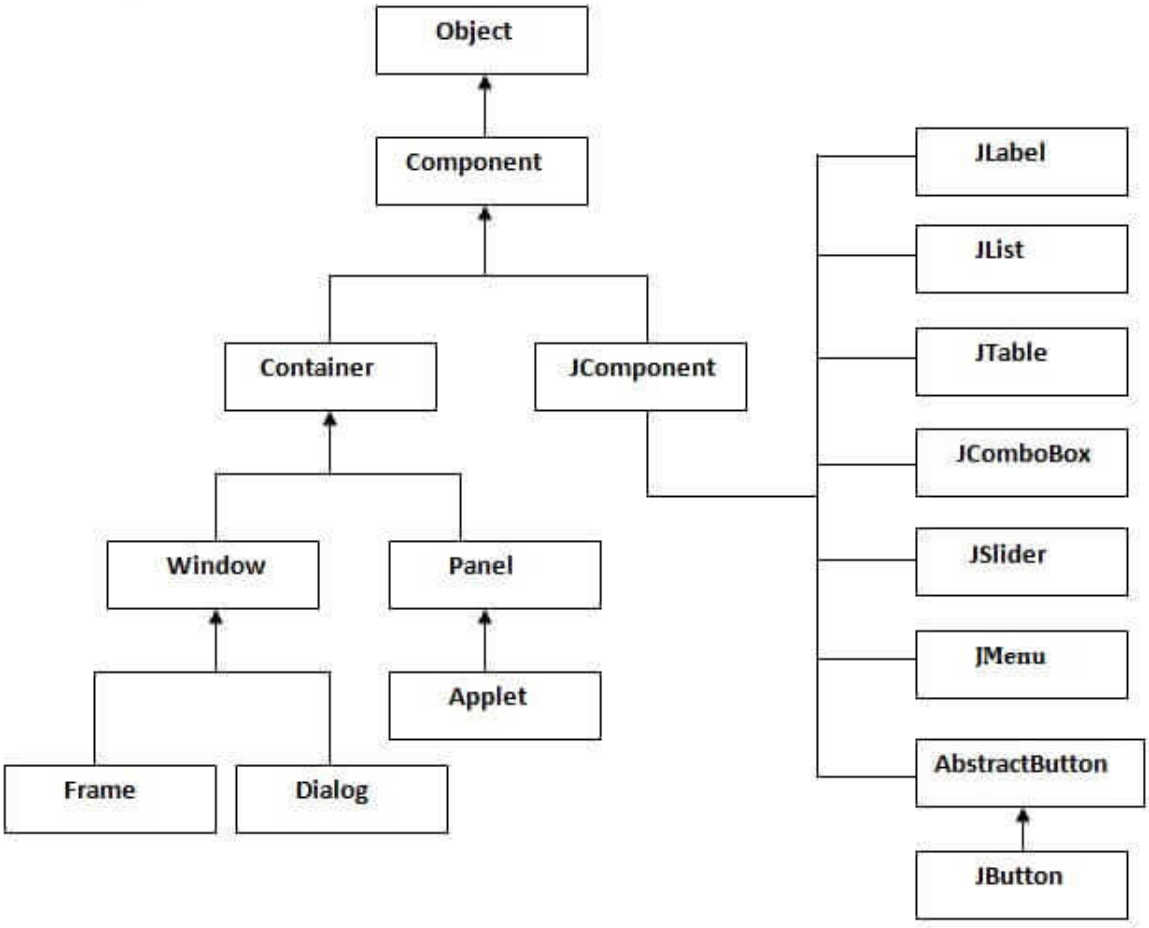
The Publisher $\boxed{\mathcal{PL}}$

Contents

Chapter 1

Swing Basics

1.1 Hierarchy of Java Swing Classes



1.2 Some Important Methods Related to `JOptionPane` Class

Methods Name	Description
showMessageDialog()	Tell the user about something that has happened.
showInputDialog()	Prompt for some input.
showOptionDialog()	The Grand Unification of the above three.
showConfirmDialog()	Asks a confirming question, like yes/no/cancel.

1.2.1 `showMessageDialog()` Method

Syntax:

```
1  JOptionPane.showMessageDialog(parameters);
```

- `component, object`
- `component, object, String, int`
- `component, object, String, int, Icon`

Parameters Descriptions -

- `Component` – The first parameter, which determines the Frame in which the dialog is displayed; if null, or if the parentComponent has no Frame, a default Frame is used.
- `Object` – The second parameter can be any objects.
- `String` – The third parameter is a String placed as the title of the message dialog window.
- `int` – The int that follows the String is the MessageType. The different MessageTypes for JOptionPane, are:
  - `ERROR_MESSAGE`
  - `INFORMATION_MESSAGE`
  - `WARNING_MESSAGE`
  - `QUESTION_MESSAGE`
  - `PLAIN_MESSAGE`
- `Icon` – The last parameter is an Icon that is displayed inside the dialog and overrides the default MessageType icon.

Example Code:

```
package SwingOne;
import javax.swing.ImageIcon;
import javax.swing.JOptionPane;
import java.awt.Image;

class MessageDialog{

  public static void main(String[] args) {
    ImageIcon icon = new ImageIcon("D:\\Programming\\Java\\PracticeProject\\Swing\\Icons\\
    done.png");
    Image image = icon.getImage();
    image = image.getScaledInstance(50, 50, Image.SCALE_SMOOTH);
    icon = new ImageIcon(image);
    JOptionPane.showMessageDialog(null, "Done","Success",JOptionPane.DEFAULT_OPTION, icon);
  }
}
```

1.2.2 JOptionPane Icons

Icon	Code	IDE Value
No icon	JOptionPane.PLAIN_MESSAGE	-1
	JOptionPane.ERROR_MESSAGE	0
	JOptionPane.INFORMATION_MESSAGE	1
	JOptionPane.WARNING_MESSAGE	2
	JOptionPane.QUESTION_MESSAGE	3

1.2.3 `showInputDialog()` Method

Example Code:

```
1  package SwingOne;
2
3  import java.awt.Image;
4  import javax.swing.ImageIcon;
5  import javax.swing.JOptionPane;
6
7  class InputDialog{
8
9    public static void main(String[] args) {
10     ImageIcon icon = new ImageIcon("D:\\Programming\\Java\\PracticeProject\\Swing\\Icons\\
       user_Icon.png");
11     Image img = icon.getImage();
12     img = img.getScaledInstance(50, 50, Image.SCALE_SMOOTH);
13     icon = new ImageIcon(img);
14
15     String name = JOptionPane.showInputDialog(null,"User Name","User Info",JOptionPane.
       INFORMATION_MESSAGE);
16     int age = 0;
17
18     try {
19       int birthY = Integer.parseInt(JOptionPane.showInputDialog(null,"Birth Year: ","User
       Info",JOptionPane.INFORMATION_MESSAGE));
20       age = 2023-birthY;
21       JOptionPane.showMessageDialog(null,"User Name: "+name+"\nAge: "+age,"User Info",
       JOptionPane.DEFAULT_OPTION,icon);
22     } catch(NumberFormatException e) {
23       JOptionPane.showMessageDialog(null,"Invalid Bith Year !!!","Warning",JOptionPane.
       WARNING_MESSAGE);
24     }
25
26
27   }
28 }
```

1.2.4 `showConfirmDialog()` Method

Example Code:

```
1  package SwingOne;
2
3  import java.awt.Image;
4  import javax.swing.ImageIcon;
5  import javax.swing.JOptionPane;
6
7  class ConfirmDialog{
8
9    public static void main(String[] args) {
10     int choice = JOptionPane.showConfirmDialog(null, "Do you want to Execute this Program ?"
       ,"Confirmation",JOptionPane.YES_NO_OPTION,JOptionPane.QUESTION_MESSAGE);
11
12     if(choice == JOptionPane.YES_OPTION) {
13       JOptionPane.showMessageDialog(null, "Congratulations !!!","Execute",JOptionPane.
       INFORMATION_MESSAGE);
14     } else {
15       JOptionPane.showMessageDialog(null, "Exit program","Exit",JOptionPane.WARNING_MESSAGE)
       ;
16     }
17
18   }
19 }
```

1.3 JFrame

1.3.1 Basic JFrame

Example Code 01:

```
package SwingOne;

import javax.swing.JFrame;

class JFrameExample{

  public static void main(String[] args) {
    JFrame frame = new JFrame();
    frame.setVisible(true);
    frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    frame.setTitle("Frame Demo");
    frame.setResizable(false);

    //----- Set frame size
//    frame.setSize(500,300);

    //------ Set frame at center
//    frame.setLocationRelativeTo(null);
    //----- Custom location
//    frame.setLocation(150,70);

    //----- Size and Location combine
    //frame.setBounds(left, top, with, height);

    frame.setBounds(150,70,500,300);
  }
}
```

Example Code 02: Using Constructor.

```
package SwingOne;

import javax.swing.JFrame;

class JFrameExample extends JFrame{

  JFrameExample() {
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    setTitle("Frame Demo");
    setBounds(150,100,350,500);
  }

  public static void main(String[] args) {
    JFrameExample frame = new JFrameExample();
    frame.setVisible(true);
  }
}
```

1.3.2 Frame Decoration (Icon & Background Color)

Example Code :

```
package SwingOne;

import java.awt.Color;
import java.awt.Container;
import javax.swing.ImageIcon;
import javax.swing.JFrame;

class MakeFrame extends JFrame {

  private static final long serialVersionUID = 1L;

  private ImageIcon icon = null;
  private Container container = null;

  //------- Constructor
  MakeFrame() {
    setVisible(true);
    setTitle("Frame Demo");
    setDefaultCloseOperation(EXIT_ON_CLOSE);
    setLocationRelativeTo(null);
    setSize(350,500);
    decorateFrame();        // Call decorateFrame() method
  }
  public void decorateFrame() {

    //---------- Set Frame Icon ---------------------------
    icon = new ImageIcon(getClass().getResource("done.png"));
    this.setIconImage(icon.getImage());

    //----------- Set Frame Background Color ------
    container = this.getContentPane();
//    container.setBackground(Color.ORANGE);

    //--------- Set Custom Background Color ------
    Color bgColor = Color.decode("#e6e6fa");
    container.setBackground(bgColor);
  }
}

class JFrameExample{

  public static void main(String[] args) {
    MakeFrame frame = new MakeFrame();
    frame.setLocationRelativeTo(null);
  }
}
```

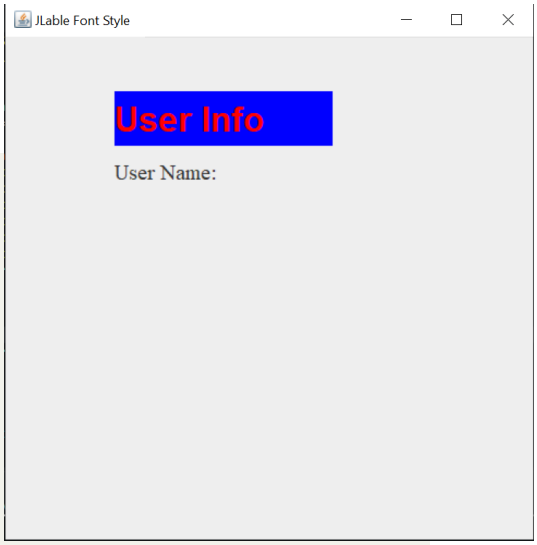
1.4 JLabel

1.4.1 Basic JLabel

```
1  package SwingTwo;
2
3  import java.awt.Color;
4  import java.awt.Container;
5  import javax.swing.ImageIcon;
6  import javax.swing.JFrame;
7  import javax.swing.JLabel;
8
9  public class JLabelDemo extends JFrame {
10
11   private static final long serialVersionUID = 1L;
12
13   private ImageIcon icon = null;
14   private Container container = null;
15   private JLabel userLabel = null, passLabel = null;
16
17   JLabelDemo() {
18     decorateFrame();
19   }
20
21   public void decorateFrame() {
22
23     container = this.getContentPane();
24     container.setLayout(null);
25
26     icon = new ImageIcon(getClass().getResource("user_Icon.png"));
27     this.setIconImage(icon.getImage());
28
29     Color bgColor = Color.decode("#e2fbfa");
30     container.setBackground(bgColor);
31
32     this.setVisible(true);
33     this.setTitle("JLabel Demo");
34     this.setDefaultCloseOperation(EXIT_ON_CLOSE);
35     this.setSize(500,350);
36     this.setLocationRelativeTo(null);
37
38     userLabel = new JLabel();
39     userLabel.setBounds(50,30,100,30);
40     userLabel.setText("User Name: ");
41     container.add(userLabel);
42
43     passLabel = new JLabel("Password: ");
44     passLabel.setBounds(50,60,100,30);
45     container.add(passLabel);
46
47   }
48
```

```
  public static void main(String[] args) {
    JLabelDemo frame = new JLabelDemo();
  }

}
```

1.4.2 JLabel Font Style



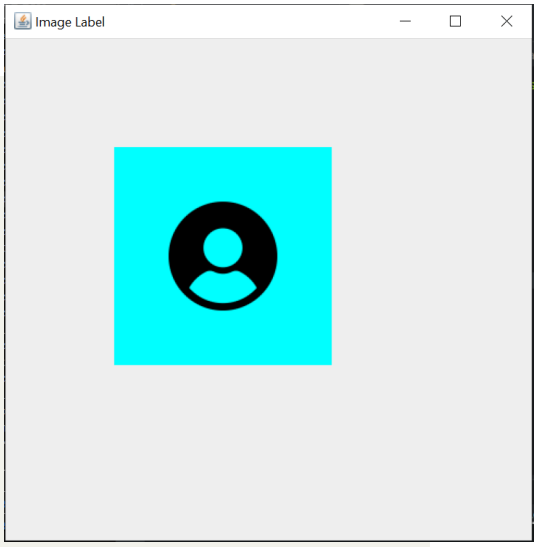
.

```
class MakeFrame extends JFrame {
  private static final long serialVersionUID = 1L;

  private Container container = null;
  private Font robotoFont = null, tnrFont=null;
  private JLabel title = null,userName = null;

  MakeFrame() {
    frameDecoration();
  }

  public void frameDecoration() {

    container = this.getContentPane();
    container.setLayout(null);

    this.setVisible(true);
    this.setTitle("JLable Font Style");
    this.setDefaultCloseOperation(EXIT_ON_CLOSE);
    this.setSize(500,500);
    this.setLocationRelativeTo(null);

    //------- Define Font Style
    robotoFont = new Font("Roboto",Font.BOLD,32);
    tnrFont = new Font("Times New Roman",Font.PLAIN,20);

    title = new JLabel("User Info");
    title.setBounds(200, 50, 100, 30);  //  x,y,with,height
    title.setFont(robotoFont);      // Set Font Style
    title.setForeground(Color.RED);   // Change Foreground Color
    title.setOpaque(true);        // Enable to change Background Color
    title.setBackground(Color.BLUE);    // Change Background Color
    container.add(title);       // Add Jlabel to Container

    userName = new JLabel("User Name: ");
    userName.setFont(tnrFont);
    userName.setBounds(100,100,100,30);
    container.add(userName);

  }
}
```

1.4.3 Set Tip Text

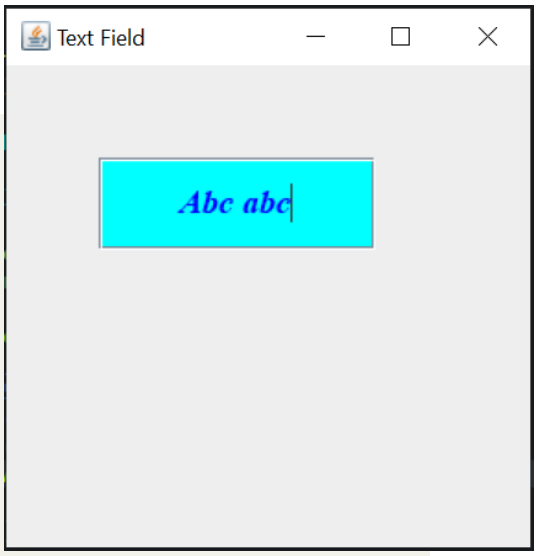
```
class MakeFrame extends JFrame {
  private static final long serialVersionUID = 1L;

  private Container container = null;
  private Font robotoFont = null;
  private JLabel title = null;

  MakeFrame() {
    frameDecoration();
  }

  public void frameDecoration() {

    container = this.getContentPane();
    container.setLayout(null);

    this.setVisible(true);
    this.setTitle("JLable Font Style");
    this.setDefaultCloseOperation(EXIT_ON_CLOSE);
    this.setSize(500,500);
    this.setLocationRelativeTo(null);

    //------- Define Font Style
    robotoFont = new Font("Roboto",Font.BOLD,20);

    title = new JLabel("User Info");
    title.setToolTipText("Title");    // Tool tip
    title.setBounds(200, 50, 100, 30);
    title.setFont(robotoFont);
    title.setForeground(Color.RED);
    title.setOpaque(true);
    title.setBackground(Color.BLUE);
    container.add(title);

  }
}
```

1.4.4 Image Label




```
class MakeFramee extends JFrame {
  private static final long serialVersionUID = 1L;

  private Container container = null;
  public JLabel imgLabel = null;
  private ImageIcon icon = null;
  private Image img = null;

  MakeFramee(){
    decorateFrame();
  }

  public void decorateFrame() {

    container = this.getContentPane();
    container.setLayout(null);

    this.setDefaultCloseOperation(EXIT_ON_CLOSE);

    //----- Set Image Label
    icon = new ImageIcon(getClass().getResource("user_Icon.png"));
    img = icon.getImage();
    img = img.getScaledInstance(100, 100, Image.SCALE_SMOOTH); // Resize Image
    icon = new ImageIcon(img);        // Re-assign image to icon
    imgLabel = new JLabel(icon);      // Instantiate image Label
    imgLabel.setBounds(100,100,200,200);
    imgLabel.setOpaque(true);
    imgLabel.setBackground(Color.cyan);
    container.add(imgLabel);        // Add image to container
  }
}

public class ImageLabel {

  public static void main(String[] args) {
    MakeFramee frame = new MakeFramee();
    frame.setVisible(true);
    frame.setTitle("Image Label");
    frame.setBounds(100,50,500,500);
    frame.setLocationRelativeTo(null);
  }

}
```

1.4.5 Text Field



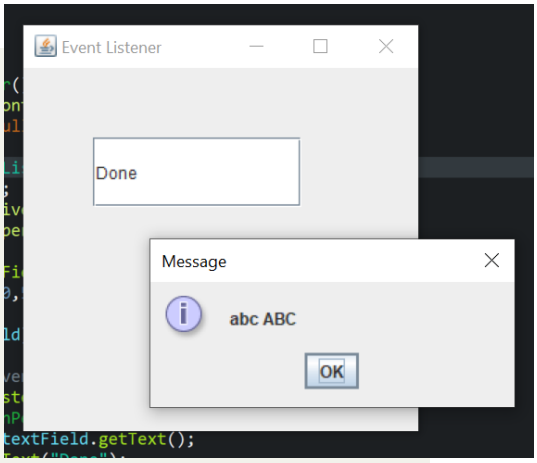
.

```
class MKFrame extends JFrame {
  private static final long serialVersionUID = 1L;

  private Container container = null;
  private JTextField textField = null;

  MKFrame() {
    decorateFrame();
  }

  public void decorateFrame() {
    container = this.getContentPane();
    container.setLayout(null);

    this.setTitle("Text Field");
    this.setSize(300,300);
    this.setLocationRelativeTo(null);
    this.setDefaultCloseOperation(EXIT_ON_CLOSE);

    font = new Font("Times New Roman",Font.BOLD + Font.ITALIC,18);

    textField = new JTextField();
    textField.setFont(font);
    textField.setBounds(50,50,150,50);
    textField.setBackground(Color.cyan);
    textField.setForeground(Color.blue);
    textField.setHorizontalAlignment(JTextField.CENTER); // Writing start form center

    container.add(textField);
  }
}

public class TextField {

  public static void main(String[] args) {

    MKFrame frame = new MKFrame();
    frame.setVisible(true);

  }

}
```

1.5 Event Listener

1.5.1 Read text form text field and show in messageDialoug()

.



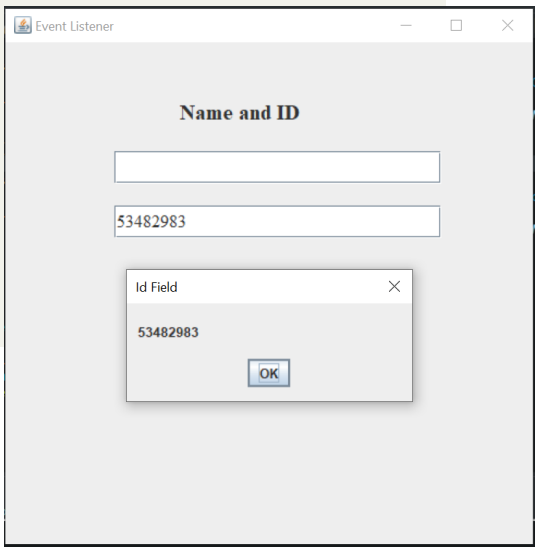

```
class MKFrame extends JFrame {
  private static final long serialVersionUID = 1L;

  private Container container = null;
  private JTextField textField = null;

  MKFrame() {
    actionListener();
  }

  public void actionListener() {
    container = this.getContentPane();
    container.setLayout(null);

    this.setTitle("Event Listener");
    this.setSize(300,300);
    this.setLocationRelativeTo(null);
    this.setDefaultCloseOperation(EXIT_ON_CLOSE);

    textField = new JTextField();
    textField.setBounds(50,50,150,50);

    container.add(textField);

    //----------------- Event Listener
    textField.addActionListener(new ActionListener() {
      public void actionPerformed(ActionEvent e) {
        String msg = textField.getText(); // Get message from textField
        textField.setText("Done");      // Replace text with "Done" msg
        if(!(msg.isEmpty())) {
          JOptionPane.showMessageDialog(null, msg);
        } else {
          JOptionPane.showMessageDialog(null, "Empty Message !");
        }
      }
    });
  }
}

public class TextField {
  public static void main(String[] args) {

    MKFrame frame = new MKFrame();
    frame.setVisible(true);
  }
}
```

1.5.2 Perform EventListener for JTextField and Show Message Using showMessageDialog

.

```
class MakeFrame extends JFrame {
  private static final long serialVersionUID = 1L;

  private Container container = null;
  private JLabel heading = null;
  private JTextField nameField = null, IdField = null;
  private Font font = null;

  MakeFrame(int width, int height) {
    this.setTitle("Event Listener");
    this.setSize(width,height);
    this.setLocationRelativeTo(null);
    this.setDefaultCloseOperation(EXIT_ON_CLOSE);
    frameActions();
  }

  public void frameActions() {
    container = this.getContentPane();
    container.setLayout(null);

    font = new Font("Times New Roman",Font.BOLD,20);

    // Crate and add JLabel to the container
    heading = new JLabel("Name and ID");
    heading.setFont(font);
    heading.setBounds(160,50,150,30);
    container.add(heading);

    font = new Font("Times New Roman",Font.PLAIN,16);

    // Configure and add JTextField to the Container
    nameField = new JTextField();
    nameField.setBounds(100,100,300,30);
    nameField.setFont(font);
    container.add(nameField);

    IdField = new JTextField();
    IdField.setBounds(100,150,300,30);
    IdField.setFont(font);
    container.add(IdField);

    // Instantiate EventHandler Class and pass the JTextFields as parameter
    EventHandler eventHandler = new EventHandler(nameField, IdField);

    nameField.addActionListener(eventHandler);
    IdField.addActionListener(eventHandler);
  }
}

```

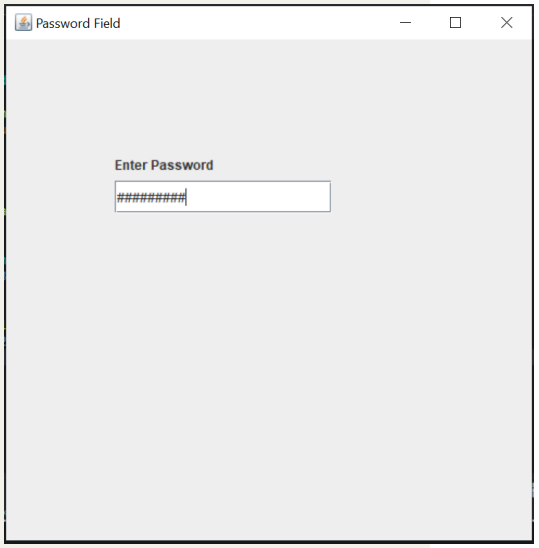
```
50 // Implements the ActionListner interface
51 class EventHandler implements ActionListener {
52
53   private JTextField nameField = null, idField = null;
54
55   EventHandler(JTextField name, JTextField Id) {
56     nameField = name;
57     idField = Id;
58   }
59
60   public void actionPerformed(ActionEvent e) {
61     String str = null;
62     if(e.getSource() == nameField) {
63       str = nameField.getText();
64       if(str.isEmpty()) {
65         JOptionPane.showMessageDialog(null, "Empty Name Field","Name Field",JOptionPane.
     WARNING_MESSAGE);
66       } else {
67         JOptionPane.showMessageDialog(null, str,"Name Field",JOptionPane.PLAIN_MESSAGE);
68       }
69     } else {
70       str = idField.getText();
71       if(str.isEmpty()) {
72         JOptionPane.showMessageDialog(null, "Empty Id Field","Id Field",JOptionPane.
     WARNING_MESSAGE);
73       } else {
74         JOptionPane.showMessageDialog(null, str, "Id Field", JOptionPane.PLAIN_MESSAGE);
75       }
76     }
77   }
78 }
79
80 public class EventListener1 {
81
82   public static void main(String[] args) {
83     MakeFrame frame = new MakeFrame(500,500);
84     frame.setVisible(true);
85   }
86
87 }
```



1.5.3 Password Field

.

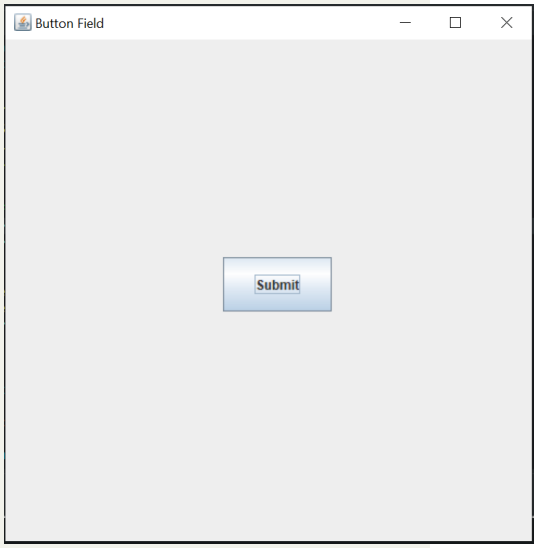
```
class CreatFrame extends JFrame{
  private static final long serialVersionUID = 1L;

  private Container container = null;
  private JPasswordField passField = null;

  CreatFrame() {
    this.setTitle("Password Field");
    this.setSize(500,500);
    this.setDefaultCloseOperation(EXIT_ON_CLOSE);
    this.setLocationRelativeTo(null);
    frameActions();
  }

  public void frameActions() {
    container = this.getContentPane();
    container.setLayout(null);

    JLabel label = new JLabel("Enter Password");
    label.setBounds(100,100,200,30);
    container.add(label);

    passField = new JPasswordField();
    passField.setBounds(100,130,200,30);
    passField.setEchoChar('#');
    container.add(passField);

  }
}

public class PasswordField {
  public static void main(String[] args) {
    CreatFrame frame = new CreatFrame();
    frame.setVisible(true);
  }
}
```



1.5.4 Button Field

.

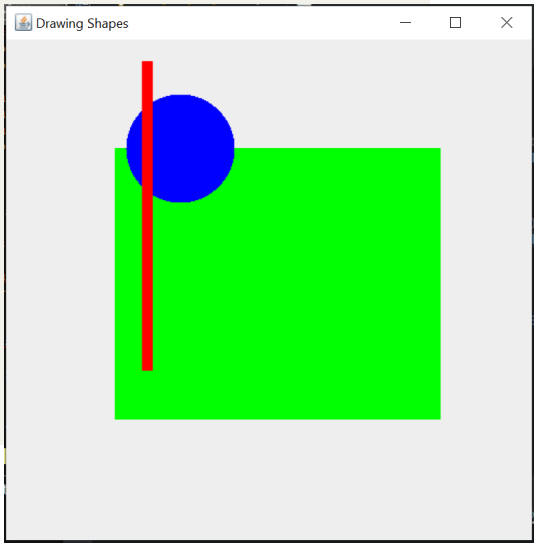
```
class BuildFrame extends JFrame {
  private static final long serialVersionUID = 1L;

  private Container container = null;
  private JButton button = null;

  BuildFrame () {
    this.setTitle("Button Field");
    this.setSize(500,500);
    this.setLocationRelativeTo(null);
    frameActions();
  }

  public void frameActions() {
    container = this.getContentPane();
    container.setLayout(null);

    button = new JButton("Submit");
    button.setSize(100,50);
    button.setLocation(200,200);
    container.add(button);
  }
}

public class ButtonField {

  public static void main(String[] args) {

    BuildFrame frame = new BuildFrame();
    frame.setVisible(true);
  }
}
```



1.6 Drawing Shapes

.

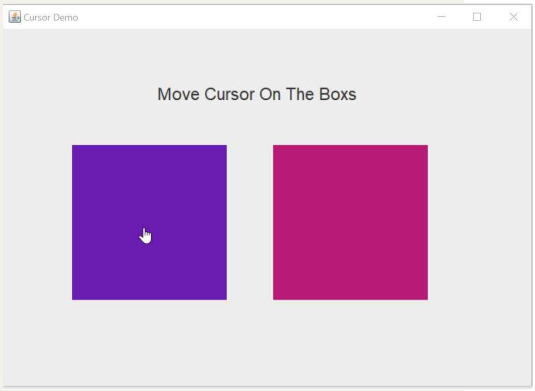
```
DrawingPanel() {

    this.setTitle("Drawing Shapes");
    this.setSize(500,500);
    this.setLocationRelativeTo(null);
    this.setDefaultCloseOperation(EXIT_ON_CLOSE);

    JPanel drawintPanel = new JPanel() {
      public void paintComponent(Graphics g) {
        Graphics2D g2 = (Graphics2D) g;

        //Rectangle2D.Double(x,y,width,height);
        Shape rect = new Rectangle2D.Double(100,100,300,250);
        g2.setColor(Color.GREEN);
        g2.fill(rect);

        // Ellipse2D.Double(x,y,width,height);
        Shape circle = new Ellipse2D.Double(110,50,100,100);
        g2.setColor(Color.blue);
        g2.fill(circle);

        // Draw line form (x1,y2) to (x2,y2)
        Shape line = new Line2D.Double(130,25,130,300);
        g2.setStroke(new BasicStroke(10));
        g2.setColor(Color.red);
        g2.draw(line);
      }
    };

    this.getContentPane().add(drawintPanel);
    this.setVisible(true);
  }
}

public class DrawingApp {
  public static void main(String[] args) {
    DrawingPanel frame = new DrawingPanel();
  }
}
```

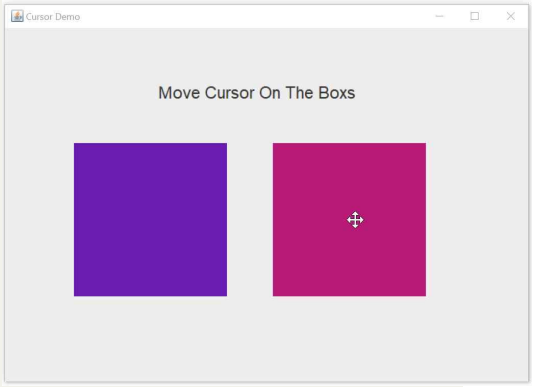


1.7 Set Cursor

.

```
1
2  package SwingTwo;
3
4  import java.awt.*;
5
6  import javax.swing.*;
7
8  public class CursorDemo extends JFrame {
9    private static final long serialVersionUID = 1L;
10
11   private Container container;
12   private JLabel heading, label1, label2;
13   private Font font;
14   private Cursor cursor1, cursor2;
15
16   CursorDemo() {
17     this.setTitle("Cursor Demo");
18     this.setSize(700,500);
19     this.setLocationRelativeTo(null);
20     this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
21     initComponents();
22   }
23
24   public void initComponents() {
25
26     font = new Font("Arial",Font.PLAIN,22);
27     cursor1 = new Cursor(Cursor.HAND_CURSOR);
28     cursor2 = new Cursor(Cursor.MOVE_CURSOR);
29
30     container = this.getContentPane();
31     container.setLayout(null);
32
33     //----- Heading Label
34     heading = new JLabel("Move Cursor On The Boxs");
35     heading.setFont(font);
36     heading.setBounds(200,70,300,30);
37     container.add(heading);
38
39     //--- Label 1
40     label1 = new JLabel();
41     label1.setFont(font);
42     label1.setBounds(90,150,200,200);
43     label1.setOpaque(true);
44     label1.setBackground(Color.decode("#6a1cb0"));
45     label1.setCursor(cursor1); // Add cursor
46     container.add(label1);
47
48
49
```

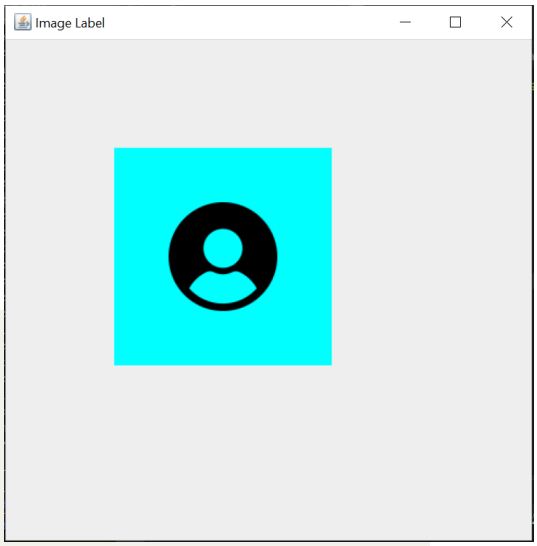




```
50
51      //--- Label 2
52      label2 = new JLabel();
53      label2.setFont(font);
54      label2.setBounds(350,150,200,200);
55      label2.setOpaque(true);
56      label2.setBackground(Color.decode("#b71b77"));
57      label2.setCursor(cursor2);  // Add cursor
58      container.add(label2);
59
60    }
61
62    public static void main(String[] args) {
63
64      CursorDemo frame = new CursorDemo();
65      frame.setVisible(true);
66    }
67
68  }
```

Chapter 2

Latex Ticks



2.1 Figure Overlap On The Plain Text

.

```
class MakeFramee extends JFrame {

  private Container container = null;
  public JLabel imgLabel = null;
  private ImageIcon icon = null;
  private Image img = null;

  MakeFramee(){
    decorateFrame();
  }

  public void decorateFrame() {

    container = this.getContentPane();
    container.setLayout(null);

    this.setDefaultCloseOperation(EXIT_ON_CLOSE);

    //----- Set Image Label
    icon = new ImageIcon(getClass().getResource("user_Icon.png"));
    img = icon.getImage();
    img = img.getScaledInstance(100, 100, Image.SCALE_SMOOTH); // Resize Image
    icon = new ImageIcon(img);        // Re-assign image to icon
    imgLabel = new JLabel(icon);      // Instantiate image Label
    imgLabel.setBounds(100,100,200,200);
    imgLabel.setOpaque(true);
    imgLabel.setBackground(Color.cyan);
    container.add(imgLabel);        // Add image to container
  }
}
```

2.2 More Control over the Figure

.

```
DrawingPanel() {

    this.setTitle("Drawing Shapes");
    this.setSize(500,500);
    this.setLocationRelativeTo(null);
    this.setDefaultCloseOperation(EXIT_ON_CLOSE);

    JPanel drawintPanel = new JPanel() {
      public void paintComponent(Graphics g) {
        Graphics2D g2 = (Graphics2D) g;

        //Rectangle2D.Double(x,y,width,height);
        Shape rect = new Rectangle2D.Double(100,100,300,250);
        g2.setColor(Color.GREEN);
        g2.fill(rect);

        // Ellipse2D.Double(x,y,width,height);
        Shape circle = new Ellipse2D.Double(110,50,100,100);
        g2.setColor(Color.blue);
        g2.fill(circle);

        // Draw line form (x1,y2) to (x2,y2)
        Shape line = new Line2D.Double(130,25,130,300);
        g2.setStroke(new BasicStroke(10));
        g2.setColor(Color.red);
        g2.draw(line);
      }
    };

    this.getContentPane().add(drawintPanel);
    this.setVisible(true);
  }
}

public class DrawingApp {
  public static void main(String[] args) {
    DrawingPanel frame = new DrawingPanel();
  }
}
```