Programs Chapter 1

A Simple Framed Window

```
import java.awt.*;

import javax.swing.*;

public class SwingTest {s

   public static void main(String[] args) {

      JFrame frame = new JFrame("Test Frame");

      frame.setSize(new Dimension(300,200));

      frame.setLocation(100,100);

      frame.setVisible(true);

   }

}
```

Changing Background Color

```
import java.awt.*;

import javax.swing.*;

public class SwingTest {

   public static void main(String[] args) {

      JFrame frame = new JFrame("Test Frame");

      frame.setSize(new Dimension(300,200));

      frame.setLocation(100,100);

      Container contentPane = frame.getContentPane();

      contentPane.setBackground(Color.red);

      frame.setVisible(true);

   }

}
```

Adding Check Box and Slider

```
public class SwingTest {
   public static void main(String[] args) {
      JFrame frame = new JFrame("Test Frame");
      frame.setSize(new Dimension(400,200));
      frame.setLocation(100,100);
      Container contentPane = frame.getContentPane();
      JLabel label = new JLabel("HERE IS A LABEL");
      contentPane.add(label, BorderLayout.NORTH);
      JButton button = new JButton("BUTTON");
      contentPane.add(button, BorderLayout.SOUTH);
      String[] options = {"Option 1", "Option 2", "Option 3"};
      JList list = new JList(options);
      contentPane.add(list, BorderLayout.CENTER);
      JCheckBox cbox = new JCheckBox("Check");
      contentPane.add(cbox, BorderLayout.WEST);
      JSlider slider = new JSlider();
      contentPane.add(slider, BorderLayout.EAST);
      frame.setVisible(true);
   }
}
```

Displaying a String:

```
import javax.swing.*;
import java.awt.*;
class MyComponent extends
      JComponent{ public void
      paint(Graphics g) {
        //setting font color
        g.setColor(Color.RED);
        //settting font
        g.setFont(new Font("Verdana",Font.BOLD,22));
        //displaying string
        g.drawString("Hello World!", 100, 50);
      }
}

public class ShapesEx {
        public static void main(String[] args) {
              JFrame jf=new JFrame("My Frame");
              jf.setSize(400, 300);
              jf.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
              jf.setLocationRelativeTo(null);
              MyComponent comp=new MyComponent();
              jf.add(comp);
              jf.setVisible(true);
        }
}
```

Working with 2D Shapes:

Drawing a line

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

                  class MyComponent extends
  JComponent{ public void paint(Graphics
                                              g) {
              //casting graphics to graphics 2d object
              //needed for more effects...on shapes
              Graphics2D g2=(Graphics2D)g; //setting color
              g2.setColor(Color.RED);
              //changing width
              g2.setStroke(new BasicStroke(10));
              //drawing a line
```

```
                g2.drawLine(120, 30, 50, 140); //x1,y1,x2,y2
        }
}
```

```
public class MyTest {
	public static void main(String[] args) {
		JFrame jf=new JFrame("Drawing Line");
		jf.setSize(300, 200);		//width,height
		jf.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
		/*MyComponent draw=new MyComponent();
		jf.add(draw);*/
		jf.add(new MyComponent());
		jf.setVisible(true);
	}
}
```

Drawing a Rectangle

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

		class MyComponent extends
  JComponent{ public void paint(Graphics
						g) {
		//casting graphics to graphics 2d object
		//needed for more effects...on shapes
		Graphics2D g2=(Graphics2D)g; //changing
		width
		g2.setStroke(new BasicStroke(10));
		//setting background color
		g2.setColor(Color.GREEN);
		g2.fillRect(30, 40, 120, 60);
		//drawing a rectangle
		g2.setColor(Color.RED);
		g2.drawRect(30, 40, 120, 60); //x,y,width,height
	}
}

public class MyTest {
	public static void main(String[] args) {
		JFrame jf=new JFrame("Drawing Rectangle");
		jf.setSize(300, 200);		//width,height
		jf.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
		/*MyComponent draw=new MyComponent();
		jf.add(draw);*/
		jf.add(new MyComponent());
		jf.setVisible(true);
```

```
        }
}
```

Drawing a Square:

```
g2.drawRect(30, 40, 120, 120); //x,y,width,height
//height & width Same
```

Drawing a Circle:

```
g2.drawOval(30, 40, 120, 120); //x,y,width,height
//height & width Same
```

Drawing a Ellipse:

```
g2.drawOval(30, 40, 150, 80); //x,y,width,height
//height & width Different
```

Drawing an Arc:

```
public void paint(Graphics g) {
        Graphics2D g2=(Graphics2D)g;
        g2.setColor(Color.RED);
        g2.setStroke(new BasicStroke(10));
        //g2.drawArc(x, y, width, height, startAngle, arcAngle);
        g2.drawArc(30, 50, 100, 140, 130, 150);
      }
```

Drawing a Polygon

```
public void paint(Graphics g) {
        Graphics2D g2=(Graphics2D)g;
        g2.setColor(Color.RED);
        g2.setStroke(new BasicStroke(5));
        //drawing a polygon
        int xPoly[] = {100, 150, 200, 175, 250}; int yPoly[] =
       {150, 100, 125, 225, 100}; g2.drawPolygon(xPoly,
       yPoly, 5); //g2.drawPolyline(xPoints, yPoints,
       nPoints);
 }
```

Using Special Font and Color for Text:

```
JLabel lbl=new JLabel();
lbl.setText("This is a text!");
lbl.setSize(100, 50);
//setting color
lbl.setForeground(Color.RED);
//setting font
lbl.setFont(new Font("Verdana",Font.BOLD,22));
```

Displaying Images:

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

class SwingEx{
        SwingEx(){
                JFrame jf=new JFrame("My Frame");
                jf.setSize(350, 350);
                jf.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
                jf.setLayout(null);
                jf.setLocationRelativeTo(null);

                ImageIcon image=new ImageIcon
                                ("/Users/Desktop/test.jpg");
                Image newimg=image.getImage().getScaledInstance(200, 200,
                                Image.SCALE_SMOOTH);
                ImageIcon newImg=new
                ImageIcon(newimg); JLabel lbl=new
                JLabel(newImg); lbl.setLocation(60, 50);
                lbl.setSize(200, 200); jf.add(lbl);

                jf.setVisible(true);
        }
}

public class Example {
        public static void main(String[] args) { new
              SwingEx();
        }
}
```

Mouse Events:

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

class SwingEx{
        SwingEx(){
                JFrame jf=new JFrame("My Frame");
                jf.setSize(350, 350);
                jf.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
                jf.setLayout(null);
                jf.setLocationRelativeTo(null);

                JButton btn=new JButton("Click Me!");
                btn.setSize(100, 50); btn.setLocation(100,
                50); jf.add(btn);

                JLabel lbl=new JLabel("Result");
                lbl.setSize(100, 50);
                lbl.setLocation(100, 100);
                jf.add(lbl);

                //adding mouse click event
                btn.addMouseListener(new MouseListener() {
                        public void mouseClicked(MouseEvent e) {
                                lbl.setText("Mouse Clicked!");
                        }
                        public void mousePressed(MouseEvent e) {
                                lbl.setText("Mouse Pressed!");
                        }

                        public void mouseReleased(MouseEvent e) {
                                lbl.setText("Mouse Released!");
                        }

                        public void mouseEntered(MouseEvent e) {
                                lbl.setText("Mouse Entered!");
                        }

                        public void mouseExited(MouseEvent e) {
                                lbl.setText("Mouse Exited!");
                        }
                });

                jf.setVisible(true);
```

```
        }
}

public class Example {
        public static void main(String[] args) { new
                SwingEx();
        }
}
```

Key Events:

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

class SwingEx{
        SwingEx(){
                JFrame jf=new JFrame("My Frame");
                jf.setSize(350, 350);
                jf.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
                jf.setLayout(null);
                jf.setLocationRelativeTo(null);

                JTextField txt=new JTextField();
                txt.setSize(200, 50);
                txt.setLocation(60, 100);
                jf.add(txt);

                JLabel lbl=new JLabel("Event Result");
                lbl.setSize(100, 50); lbl.setLocation(100,
                150); jf.add(lbl);

                //adding Key Event
                txt.addKeyListener(new KeyListener() {
                        public void keyTyped(KeyEvent e) {
                                lbl.setText("Key Typed!");
                        }

                        public void keyPressed(KeyEvent e) {
                                lbl.setText("Key Pressed!");
                        }

                        public void keyReleased(KeyEvent e) {
                                lbl.setText("Key Released!");
                        }
```

```
            });

            jf.setVisible(true);
        }
}

public class Example {
        public static void main(String[] args) {
            new SwingEx();
        }
}
```

Example of ItemListener:

Event triggered on Combo Box item selection.

```
import java.awt.*;
import javax.swing.*;
import java.awt.event.*;
class SwingEx{
        SwingEx(){
            JFrame jf=new JFrame("My Frame");
            jf.setSize(400,350);
            jf.setLocationRelativeTo(null);
            jf.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
            jf.setLayout(null);
            JComboBox cmb=new JComboBox();
            cmb.addItem("BCA");
            cmb.addItem("BBA");
            cmb.addItem("MCA");
            cmb.addItem("MBA");
            cmb.setSize(120, 60);
            cmb.setLocation(100,60);
            cmb.setFocusable(false);
            jf.add(cmb);

            cmb.addItemListener(new ItemListener() {
                    public void itemStateChanged(ItemEvent e) {
                        if(e.getStateChange()==ItemEvent.SELECTED) {
                            //getting selected Item
                            String item=cmb.getSelectedItem()
                                                .toString();

                      //displaying
                    System.out.println("Selected Item: "+item);
```

```
                }
            }
        });

        jf.setVisible(true);
    }
}

public class Example {
    public static void main(String[] args) { new
        SwingEx();

    }
}
```

Using Mouse Adapter Class

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

class SwingEx{
    SwingEx(){
        JFrame jf=new JFrame("My Frame");
        jf.setSize(400,350);
        jf.setLocationRelativeTo(null);
        jf.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        jf.setLayout(null);

        JButton btn=new JButton("Click Me!");
        btn.setSize(120, 50); btn.setLocation(100,
        50); jf.add(btn);

        //Using mouse adapter class
        btn.addMouseListener(new MouseAdapter() {
        //we don't need to implement all methods of MouseListener
            //we can use method as per the requirement
            public void mouseEntered(MouseEvent e) {
                btn.setText("Mouse Entered!");
            }
        });

        jf.setVisible(true);
    }
}
```

```
public class ExampleA {
	public static void main(String[] args) { new
		SwingEx();

	}
}
```

Using Key Adapter Class

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

class SwingEx{
	SwingEx(){
	    JFrame jf=new JFrame("My Frame");
	    jf.setSize(400,350);
	    jf.setLocationRelativeTo(null);
	    jf.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
	    jf.setLayout(null);

	    JTextField txt=new JTextField();

	    txt.setSize(120, 50);
	    txt.setLocation(100, 50);
	    jf.add(txt);

	    txt.addKeyListener(new KeyAdapter() {
	        public void keyTyped(KeyEvent e) {
	                JOptionPane.showMessageDialog(null,"Key Typed!");
	        }
	    });


	    jf.setVisible(true);
	}
}

public class Example {
	public static void main(String[] args) { new
		SwingEx();

	}
}
```

Using Window Adapter Class

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

class SwingEx{
        SwingEx(){
            JFrame jf=new JFrame("My Frame");
            jf.setSize(300,250);
            jf.setLocationRelativeTo(null);
            jf.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
            jf.setLayout(null);

            //adding window event to jframe
            jf.addWindowListener(new WindowAdapter() {
                public void windowOpened(WindowEvent e) {
                        System.out.println("Frame Opened!");
                }
            });

            jf.setVisible(true);
        }
}

public class Example {
        public static void main(String[] args) { new
                SwingEx();

        }
}
```

MVC design Pattern

3 classes for model, view and controller required + 1 class for startup.
So, in total 4 classes are required.

Source Code

StudentView.java

```
import javax.swing.*;
public class StudentView {
        public JLabel lbl1,lbl2,lbl3;
        public JTextField txt1,txt2;
        public JButton btn1,btn2;
```

```
public StudentView() {
  JFrame jf=new JFrame("Student Form");
  jf.setSize(400, 300);
  jf.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
  jf.setLayout(null);
  jf.setLocationRelativeTo(null);

  lbl1=new JLabel("Enter Sid:");
  lbl1.setSize(100, 30);
  lbl1.setLocation(20, 20);
  jf.add(lbl1);

  txt1=new JTextField();
  txt1.setSize(120, 30);
  txt1.setLocation(100, 20);
  jf.add(txt1);

  lbl2=new JLabel("Enter Name:");
  lbl2.setSize(100, 30);
  lbl2.setLocation(20, 60);
  jf.add(lbl2);

  txt2=new JTextField();
  txt2.setSize(120, 30);
  txt2.setLocation(100, 60);
  jf.add(txt2);

  btn1=new JButton("Save");
  btn1.setSize(100, 20);
  btn1.setLocation(50, 110);
  jf.add(btn1);

  btn2=new JButton("Display");
  btn2.setSize(100, 20);
  btn2.setLocation(160, 110);
  jf.add(btn2);

  lbl3=new JLabel("Result:");

  lbl3.setSize(200, 30);
  lbl3.setLocation(20, 140);
  jf.add(lbl3);

  jf.setVisible(true);
}
```

```
}
```

StudentModel.java

```
public class StudentModel {
        private int sid;
        private String name;

        public void setId(int sid) {
          this.sid=sid;
        }

        public int getId() {
          return sid;
        }

        public void setName(String name) {
          this.name=name;
        }

        public String getName() {
          return name;
        }
}
```

StudentController.java

```
import javax.swing.*;
public class StudentController {
    StudentView v;
    StudentModel m;
        public void initController() {
            //initializing view
            v=new StudentView();
            //initializing model
            m=new StudentModel();
            //registering events
            v.btn1.addActionListener(e->saveClicked());
            v.btn2.addActionListener(e->displayClicked());
    }
    public void saveClicked() {
            int sid=Integer.parseInt(v.txt1.getText());
            String name=v.txt2.getText();
            m.setId(sid);
```

```
            m.setName(name);
            JOptionPane.showMessageDialog(null, "Saved Successfully!");
    }

    public void displayClicked() {
            v.lbl3.setText("Sid: "+m.getId()+" Name: "+m.getName());
    }
}
```

And finally,
Driver Class…

Example.java

```
//Driver Class
public class Example {
        public static void main(String[] args) { StudentController
                cont=new StudentController(); cont.initController();
        }
}
```

(Displaying Multiple Rows using JList)

StudentView.java

```
import javax.swing.*;
public class StudentView {
        public JLabel lbl1,lbl2,lbl3;
        public JTextField txt1,txt2;
        public JButton btn1,btn2;
        //required for creating a empty list
        DefaultListModel lmodel;

         public StudentView() {
           JFrame jf=new JFrame("Student Form");
           jf.setSize(400, 300);
           jf.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
           jf.setLayout(null);
           jf.setLocationRelativeTo(null);

           lbl1=new JLabel("Enter Sid:");
           lbl1.setSize(100, 30);
           lbl1.setLocation(20, 20);
```

```
            jf.add(lbl1);

            txt1=new JTextField();
            txt1.setSize(120, 30);
            txt1.setLocation(100, 20);
            jf.add(txt1);

            lbl2=new JLabel("Enter Name:");
            lbl2.setSize(100, 30);
            lbl2.setLocation(20, 60);
            jf.add(lbl2);

            txt2=new JTextField();
            txt2.setSize(120, 30);
            txt2.setLocation(100, 60);
            jf.add(txt2);

            btn1=new JButton("Save");
            btn1.setSize(100, 20);
            btn1.setLocation(50, 110);
            jf.add(btn1);

            btn2=new JButton("Display");
            btn2.setSize(100, 20);
            btn2.setLocation(160, 110);
            jf.add(btn2);

            //creating a JList
            lmodel=new DefaultListModel();

            JList jl=new JList(lmodel);
            jl.setSize(200, 100);
            jl.setLocation(20, 140);
            jf.add(jl);

            jf.setVisible(true);
        }
}
```

StudentModel.java

```
public class StudentModel {
    private int sid;
    private String name;
```

```
    public void setId(int sid) {
        this.sid=sid;
    }

    public int getId() {
        return sid;
    }

    public void setName(String name) {
        this.name=name;
    }

    public String getName() {
        return name;
    }
}
```

StudentController.java

```
import java.util.*;
import javax.swing.*;
public class StudentController {
    StudentView v;
    //for storing multiple data
    ArrayList<StudentModel> data;

    public void initController() {
                        //model initialization not required & will be initialized
                    in add button clicked
            //initializing view
            v=new StudentView();
            //initializing ArrayList
            data=new ArrayList<>();

            //registering events
            v.btn1.addActionListener(e->saveClicked());
            v.btn2.addActionListener(e->displayClicked());
    }

    public void saveClicked() {
            int sid=Integer.parseInt(v.txt1.getText()); String
            name=v.txt2.getText(); //transferring to model
            StudentModel m=new StudentModel();
```

```
            m.setId(sid);
            m.setName(name);
            //adding model to ArrayList
            data.add(m);
                JOptionPane.showMessageDialog(null, "Saved Successfully!");
    }

    public void displayClicked() {
            //clearing previous data'
            v.lmodel.clear();
            //getting data from ArrayList
            for(StudentModel st:data) {
                //System.out.println(st.getId()+" "+st.getName());
                v.lmodel.addElement(st.getId()+" "+st.getName());
            }

    }
}
```

Example.java

```
//Driver Class
public class Example {
        public static void main(String[] args) {
                StudentController cont=new StudentController();
                cont.initController();
        }
}
```

(Using JTable for Displaying Multiple Rows)

StudentView.java

```
import javax.swing.*;
import javax.swing.table.*;
public class StudentView {
        public JLabel lbl1,lbl2,lbl3;
        public JTextField txt1,txt2;
        public JButton btn1,btn2;
        //required for creating a empty list
        DefaultTableModel tmodel;

         public StudentView() {
           JFrame jf=new JFrame("Student Form");
```

```
jf.setSize(400, 300);
jf.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
jf.setLayout(null);
jf.setLocationRelativeTo(null);

lbl1=new JLabel("Enter Sid:");
lbl1.setSize(100, 30);
lbl1.setLocation(20, 20);
jf.add(lbl1);

txt1=new JTextField();
txt1.setSize(120, 30);
txt1.setLocation(100, 20);
jf.add(txt1);

lbl2=new JLabel("Enter Name:");

lbl2.setSize(100, 30);
lbl2.setLocation(20, 60);
jf.add(lbl2);

txt2=new JTextField();
txt2.setSize(120, 30);
txt2.setLocation(100, 60);
jf.add(txt2);

btn1=new JButton("Save");
btn1.setSize(100, 20);
btn1.setLocation(50, 110);
jf.add(btn1);

btn2=new JButton("Display");
btn2.setSize(100, 20);
btn2.setLocation(160, 110);
jf.add(btn2);

//creating empty table with default table model
String cols[]= {"Sid","Name"};
tmodel=new DefaultTableModel(cols,0); //0 rows
JTable jt=new JTable(tmodel);
JScrollPane jp=new JScrollPane(jt);
jp.setLocation(50, 150); jp.setSize(200,
100); jf.add(jp);

jf.setVisible(true);
```

```
        }
}
```

StudentModel.java

```
public class StudentModel {
    private int sid;
    private String name;

    public void setId(int sid) {
        this.sid=sid;
    }

    public int getId() {
        return sid;
    }

    public void setName(String name) {
        this.name=name;
    }

    public String getName() {
        return name;
    }
}
```

StudentController.java

```
import java.util.*;
import javax.swing.*;
public class StudentController {
    StudentView v;
    //for storing multiple data
    ArrayList<StudentModel> data;

    public void initController() {
            //initializing ArrayList
            data=new ArrayList<>();
            //initializing view
            v=new StudentView();
            //registering events
            v.btn1.addActionListener(e->saveClicked());
            v.btn2.addActionListener(e->displayClicked());
    }
```

```
    public void saveClicked() {
            int sid=Integer.parseInt(v.txt1.getText()); String
            name=v.txt2.getText(); //transferring to model
            StudentModel m=new StudentModel();

            m.setId(sid);
            m.setName(name);
            //adding model to ArrayList
            data.add(m);
                JOptionPane.showMessageDialog(null, "Saved Successfully!");
    }

        public void displayClicked() {
            //clearing all rows
            v.tmodel.setRowCount(0);
            //getting data from ArrayList
            for(StudentModel st:data) {
                //putting data in Object
                Object[] obj= {st.getId(),st.getName()};
                v.tmodel.addRow(obj);
            }

    }
}
```

Example.java

```
//Driver Class
public class Example {
        public static void main(String[] args) { StudentController
                cont=new StudentController(); cont.initController();
        }
}
```

Designing a JTable

```
import java.awt.*;
import javax.swing.*;
import javax.swing.table.*;
//Driver Class
public class Example {
        public static void main(String[] args) {
                JFrame jf=new JFrame("Student Form");
                jf.setSize(400, 300);
```

```
            jf.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
            jf.setLayout(null);
        jf.setLocationRelativeTo(null);

        //creating a JTable
        String cols[]= {"Sid","Name","Address"}; DefaultTableModel
        tmodel=new DefaultTableModel(cols,0); JTable jt=new
        JTable(tmodel);
        tmodel.addRow(new Object[] {101,"Ram","Btm"});
        tmodel.addRow(new Object[] {102,"Shyam","Btm"});
        tmodel.addRow(new Object[] {103,"Hari","Btm"});
        tmodel.addRow(new Object[] {104,"Sita","Ktm"});

        //designing JTable
        //designing headings
        jt.getTableHeader().setBackground(Color.DARK_GRAY);
        jt.getTableHeader().setForeground(Color.WHITE);
        jt.getTableHeader().setFont(new Font(null,Font.BOLD,16));
        //designing rows
        jt.setBackground(Color.CYAN);
        jt.setFont(new Font("Consolas",Font.ITALIC,15));
        JScrollPane jp=new JScrollPane(jt);
        jp.setSize(300, 150);
        jp.setLocation(50, 50);
        jf.add(jp);

        jf.setVisible(true);
    }
}
```

Getting a Selected Row From JTable

```
import java.awt.*;
import javax.swing.*;
import javax.swing.table.*;
//Driver Class
public class Example {
    public static void main(String[] args) {
            JFrame jf=new JFrame("Student Form");
            jf.setSize(400, 300);
            jf.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
            jf.setLayout(null);
        jf.setLocationRelativeTo(null);

        //creating a JTable
```

```
        String cols[]= {"Sid","Name","Address"}; DefaultTableModel
        tmodel=new DefaultTableModel(cols,0);
        JTable jt=new JTable(tmodel);
        tmodel.addRow(new Object[] {101,"Ram","Btm"});
        tmodel.addRow(new Object[] {102,"Shyam","Btm"});
        tmodel.addRow(new Object[] {103,"Hari","Btm"});
        tmodel.addRow(new Object[] {104,"Sita","Ktm"});
        //designing JTable
        //designing headings
        jt.getTableHeader().setBackground(Color.DARK_GRAY);
        jt.getTableHeader().setForeground(Color.WHITE);
        jt.getTableHeader().setFont(new Font(null,Font.BOLD,16));
        //designing rows
        jt.setBackground(Color.CYAN);
        jt.setFont(new Font("Consolas",Font.ITALIC,15));
        JScrollPane jp=new JScrollPane(jt);
        jp.setSize(300, 150);
        jp.setLocation(50, 50);
        jf.add(jp);

        JButton btn=new JButton("Click Me");
        btn.setSize(100, 30); btn.setLocation(50,
        200); jf.add(btn);

        btn.addActionListener(e->{
            //getting row number
                int row=jt.getSelectedRow();
                if(row==-1) {
                    JOptionPane.showMessageDialog(null, "Please Select
                                Row!");
                }else {
                    //getting data
                            int sid=Integer.parseInt(tmodel.getValueAt(row, 0)
                                    .toString());
                        String name=tmodel.getValueAt(row, 1).toString();
                     String address=tmodel.getValueAt(row, 2).toString();
                     JOptionPane.showMessageDialog(null, sid+" "+name+"
                                 "+address);
                }

        });

        jf.setVisible(true);
    }
```

```
}
```

Getting a Selected Row from JList

```
import java.awt.*;
import javax.swing.*;
//Driver Class
public class Example {
        public static void main(String[] args) {
                JFrame jf=new JFrame("Student Form");
                jf.setSize(400, 300);
                jf.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
                jf.setLayout(null);
               jf.setLocationRelativeTo(null);
              DefaultListModel lmodel=new DefaultListModel();
              JList jl=new JList(lmodel);
              lmodel.addElement("101 Ram");
              lmodel.addElement("102 Shyam");
              lmodel.addElement("103 Hari");
              jl.setSize(200, 100);
              jl.setLocation(50, 50);
              jf.add(jl);
              JButton btn=new JButton("Click Me");
              btn.setSize(100, 30); btn.setLocation(50,
              200); jf.add(btn);

            btn.addActionListener(e->{
                int index=jl.getSelectedIndex();
                if(index==-1) {
                   JOptionPane.showMessageDialog(null, "Please select item!");
                }else {
                        String data=jl.getSelectedValue().toString();
                        JOptionPane.showMessageDialog(null, data);
                }});
              jf.setVisible(true);
        }
}
```

demonstrate FlowLayout

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

```
public class FlowLayoutDemo
{
	JFrame f;
	FlowLayoutDemo ()
	{
		f = new JFrame ();
		JLabel l1 = new JLabel ("Enter Name");
		JTextField tf1 = new JTextField (10);
		JButton b1 = new JButton ("SUBMIT");
		f.add (l1);
		f.add (tf1);
		f.add (b1);
		    f.setLayout (new FlowLayout (FlowLayout.RIGHT));

		//setting flow layout of right alignment
		f.setSize (300, 300); f.setVisible (true);
	}
	public static void main (String[]args)
	{
		new FlowLayoutDemo ();
	}
}
```

Border Layout

```
import java.awt.*;
public class BorderLayoutDemo
{
	public static void main (String[]args)
	{
		Frame f1 = new Frame ();
		f1.setSize (250, 250);
		Button b1 = new Button ("Button1");
		Button b2 = new Button ("Button2");
		Button b3 = new Button ("Button3");
		Button b4 = new Button ("Button4");
		Button b5 = new Button ("Button5");
		f1.add (b1, BorderLayout.NORTH);
		f1.add (b2, BorderLayout.EAST);
		f1.add (b3, BorderLayout.WEST);
		f1.add (b4, BorderLayout.SOUTH);
```

```
            f1.add (b5);
            f1.setVisible (true);
      }
}
```

Grid Layout

```
import java.awt.*;
import javax.swing.*;
public class GridLayoutDemo
{
      public static void main (String[]args)
      {
            Frame f1 = new Frame ();
            f1.setSize (250, 250);
            GridLayout ob = new GridLayout (2, 2);
            f1.setLayout (ob);
            Panel p1 = new Panel ();
            Label l1 = new Label ("Enter name");
            TextField tf = new TextField (10);
            Button b1 = new Button ("Submit");
            p1.add (l1);
            p1.add (tf);
            p1.add (b1);
            f1.add (p1);
            Panel p2 = new Panel ();
            f1.add (p2);
            Panel p3 = new Panel ();
            f1.add (p3);
            Label l2 = new Label ("Welcome to Java");
            f1.add (l2);
            f1.setVisible (true);
      }
}
```

Card Layout

```
import java.awt.*;
import javax.swing.*;
//Driver Class
public class Example {
        public static void main(String[] args) {
                JFrame jf=new JFrame("My Frame");
                jf.setSize(400, 300);
```

```
    jf.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
  jf.setLocationRelativeTo(null);
  jf.setLayout(null);

  //creating jpanel to demonstrate cardlayout
  JPanel jp=new JPanel();
  jp.setSize(200, 80);
  jp.setLocation(80, 40);
  jp.setBackground(Color.CYAN);
  //adding cardlayout
  CardLayout card=new CardLayout(40,30);
jp.setLayout(card);
JLabel lbl1=new JLabel("Label 1");
JLabel lbl2=new JLabel("Label 2");
JLabel lbl3=new JLabel("Label 3");
//adding level to cards
jp.add("card1",lbl1);
jp.add("card2",lbl2);
jp.add("card3",lbl3);

//creating another JPanel for buttons
JPanel jp1=new JPanel();
  jp1.setSize(300, 100);
  jp1.setLocation(40, 120);
  jp1.setLayout(new FlowLayout());

  JButton prev=new JButton("Previous");
  JButton next=new JButton("Next");
  JButton first=new JButton("First");
  JButton last=new JButton("Last");
  JButton show=new JButton("Show Card");
  jp1.add(prev);
  jp1.add(next);
  jp1.add(first);
  jp1.add(last);
  jp1.add(show);

  //adding panels to frame
  jf.add(jp);
  jf.add(jp1);

  //adding click event to buttons
  prev.addActionListener(e->{
    //previous
    card.previous(jp);
```

```
        });

        next.addActionListener(e->{
          //next
          card.next(jp);
        });

        first.addActionListener(e->{
          card.first(jp);
        });

        last.addActionListener(e->{
          card.last(jp);
        });

        show.addActionListener(e->{
          card.show(jp, "card2");
        });

        jf.setVisible(true);
    }
}
```

Spring Layout:

```
import java.awt.*;
import javax.swing.*;
//Driver Class
public class Example {
    public static void main(String[] args) {
            JFrame jf = new JFrame("Spring Layout Example");
            jf.setSize(300, 200);
            jf.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
            jf.setLocationRelativeTo(null);

            Container cp = jf.getContentPane();
            //adding spring layout
            SpringLayout spl=new SpringLayout();
            cp.setLayout(spl);

            //creating buttons

          JButton btn1=new JButton("Button 1");
          JButton btn2=new JButton("Button 2");
          JButton btn3=new JButton("Button 3");
```

```
            cp.add(btn1);
            cp.add(btn2);
            cp.add(btn3);

            //adding constraints
            //for button 1
            spl.putConstraint(SpringLayout.WEST, btn1,
                        24, SpringLayout.WEST, cp);
            spl.putConstraint(SpringLayout.NORTH, btn1,
                         9, SpringLayout.NORTH, cp);

            //for button 2
            spl.putConstraint(SpringLayout.WEST, btn2,
                        49, SpringLayout.WEST, cp);
            spl.putConstraint(SpringLayout.NORTH, btn2,
                        10, SpringLayout.SOUTH, btn1);

            //for button 3
            spl.putConstraint(SpringLayout.WEST, btn3,
                        45, SpringLayout.EAST, btn1);
            spl.putConstraint(SpringLayout.NORTH, btn3,
                         9, SpringLayout.NORTH, cp);


              jf.setVisible(true);
    }
}
```

Message Dialog

A message dialog shows information with the OK button. The method does not return any value.

```
import javax.swing.*;
public class Example {
    public static void main(String[] args) {
        JOptionPane.showMessageDialog(null, "Test Message",
                    "Title",JOptionPane.WARNING_MESSAGE);
    }
}
```

Confirm Dialog

We can display a confirmation dialog box by using the showConfirmDialog() method. The user's response is indicated by the return value.

```
import javax.swing.*;
```

```
public class Example {
	public static void main(String[] args) {
		int res = JOptionPane.showConfirmDialog(null,
				"Save the file?",
				"Title Message",
				JOptionPane.YES_NO_OPTION,
				//JOptionPane.YES_NO_OPTION
				//JOptionPane.OK_CANCEL_OPTION
				JOptionPane.QUESTION_MESSAGE);
		System.out.println(res);
		//yes/ok=0, no=1, cancel=2

	}
}
```

Input Dialog

```
import javax.swing.*;
public class Example {
	public static void main(String[] args) {
		String res=JOptionPane.showInputDialog(null,
				"Enter any value");
		System.out.println(res);

	}
}
```

Example 2:

```
import javax.swing.*;
public class Example {
	public static void main(String[] args) {
	/*Object res = JOptionPane.showInputDialog(parentComponent,
				message,
				title,
				messageType,
				icon,
				selectionValues,
				initialSelectionValue);
		*/
		Object res = JOptionPane.showInputDialog(null,
				"Select a Value:",
				"Title",
				JOptionPane.INFORMATION_MESSAGE,
				null,
				new String[] {"BCA","BBA","MCA"},
```

```
                    "BBA");
            System.out.println(res);

        }
}
```

Option Dialog

```
import javax.swing.*;
public class Example {
        public static void main(String[] args) {
                int res = JOptionPane.showOptionDialog(null,
                        "Do you want to Exit?", "Title",

                        JOptionPane.DEFAULT_OPTION,
                        JOptionPane.QUESTION_MESSAGE,
                        null,
                        new String[] {"YES","NO"},
                        "NO");

                System.out.println(res);

        }
}
```

JFileChooser Dialog

A JFileChooser lets the user select a file from the file system. We can create an object of the JFileChooser class. It allows the user to choose only files, only directories, or both.

Example:

```
import javax.swing.*;
import java.io.*;
public class Example {
        public static void main(String[] args) {
                //display file chooser
                JFileChooser fch=new JFileChooser();
                fch.setDialogTitle("Open a        picture file");
                int res = fch.showDialog(null,"Attach");

               if (res == JFileChooser.APPROVE_OPTION) { File selectedFile
                  = fch.getSelectedFile(); System.out.println("we selected:
                  " + selectedFile);
               }
```

```
    }
}
```

JColorChooser Dialog

A JColorChooser is a Swing component that lets we choose a color graphically in a JDialog.

```
import javax.swing.*;
import java.awt.*;
public class Example {
    public static void main(String[] args) { //
        Display a color chooser dialog
        Color color = JColorChooser.showDialog(null,
            "Select a color", Color.BLACK);
        //parent,message,initial color

        System.out.println("Selected Color: "+color);

    }
}
```

Custom JDialog

We can create custom JDialog as per our requirement and place different components on it as follows:

```
JDialog dialog=new JDialog();
```

Example:

```
import javax.swing.*;
public class Example {
    public static void main(String[] args) {
        JDialog jd=new JDialog();
        jd.setSize(200, 150);
        jd.setModal(true);
        jd.setTitle("Custom Dialog");

        JLabel lbl=new JLabel("I am Label");
        jd.add(lbl);

        jd.setVisible(true);
    }
}
```