

## ណែនាំអោយស្គាល់ GUI

### Graphic User Interface

#### I. ដូចម្តេចទៅដែលហៅ ថា GUI?

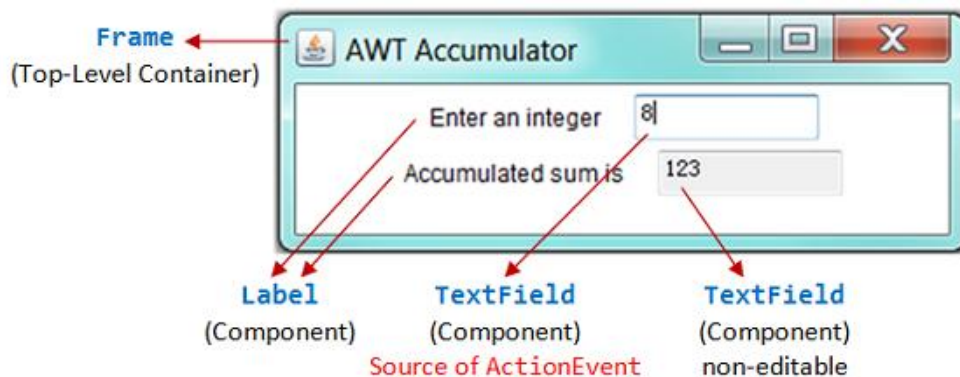
ពាក្យថា GUI សំដៅលើការបញ្ចូល និងបង្ហាញទិន្នន័យលក្ខណៈជា Form Interface ដែលអ្នកប្រើប្រាស់អាចវាយបញ្ចូលព័ត៌មាន និងបង្ហាញមានបានដោយភាពងាយស្រួល។ Java ផ្តល់លទ្ធភាពអោយអ្នក Design នូវ GUI ដោយមានភាពងាយស្រួលពេលអ្នកអាចមានជំរើសពីការសរសេរកូដ បង្កើត ឬ អាចតាមចាប់ទាញនូវ ToolBox មកដាក់ចំទីតាំងណាមួយដែលអ្នកត្រូវការ។

នៅក្នុង Java API សំរាប់ graphic form វាមានរហូតដល់ ៣ Version ដូចជា៖

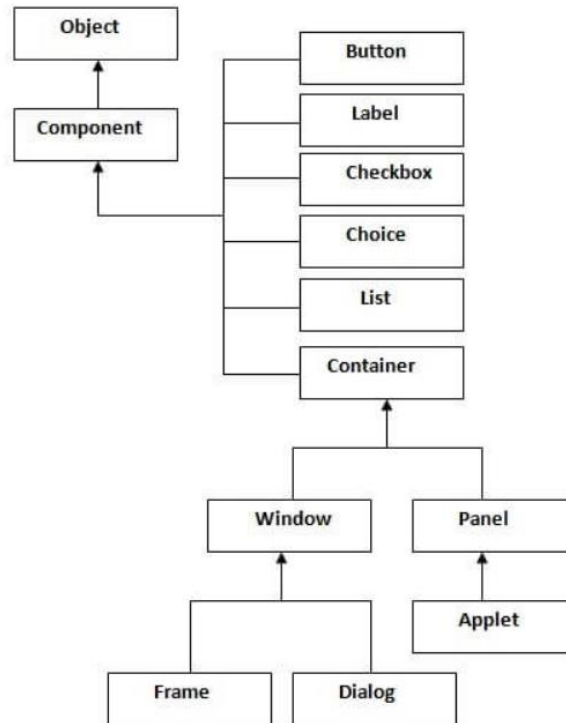
១) AWT(Abstract Windowing Toolkit): គឺជាប្រភេទជំនាន់ដំបូងគេនៃភាសា Java តាំងពី JDK ជំនាន់ទី ១ មក។

២) Swing: ជាជំនាន់បន្ទាប់ពី AWT ដែលវាបានបន្ថែមនូវសមាសធាតុ ToolBox ផ្សេងៗជាច្រើនដើម្បីបំពេញការងារប្រចាំថ្ងៃនេះ។ Swing មានភាពស្រស់ស្អាតនិងទាក់ទាញពីសំណាក់អ្នកប្រើប្រាស់ និង អ្នកបង្កើត។ វាត្រូវបានបង្កើតឡើងក្នុង JDK ជំនាន់ទី ១.២។

៣) ជំនាន់ចុងក្រោយនៃ Java គឺ JavaFx ដែលបានកំពុងប្រើប្រាស់ជាមួយនិង JDK 8 សព្វថ្ងៃនេះ។



- II. AWT(Abstract Window Toolkit) គឺជាប្រភេទ API ប្រើប្រាស់សំរាប់ develop GUI ឬ window-based applications ក្នុង java។

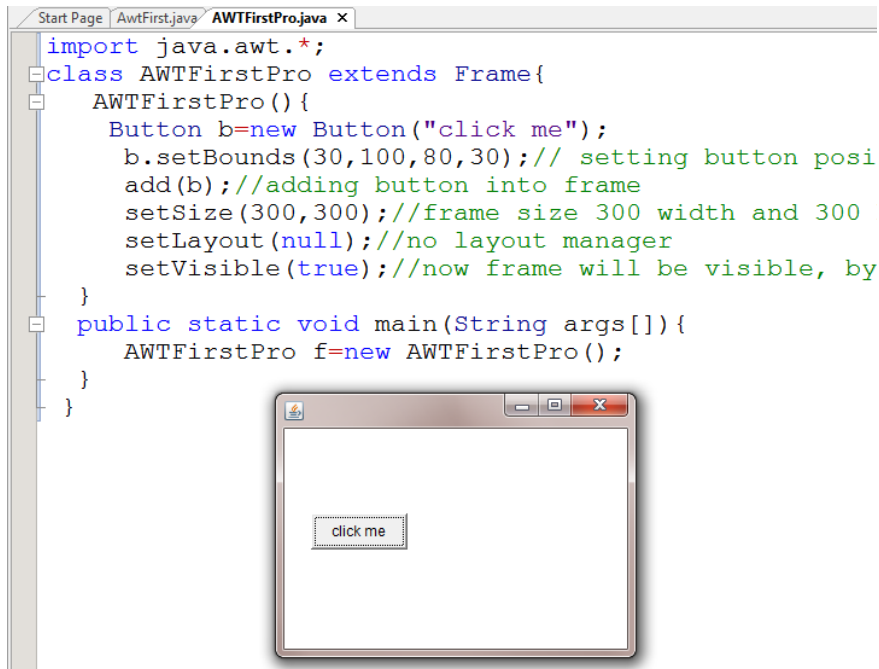



Container: គឺជាកន្លែងដែលសំរាប់ដាក់ ឬ រៀបចំ Component របស់ Form ដូចជា ៖  
TextField, Button, ដែល Extends ពី Class Container។

Panel: ក៏ជាកន្លែងរៀបចំ Component របស់ Form ដែរ ដូចជា Button និង Label។

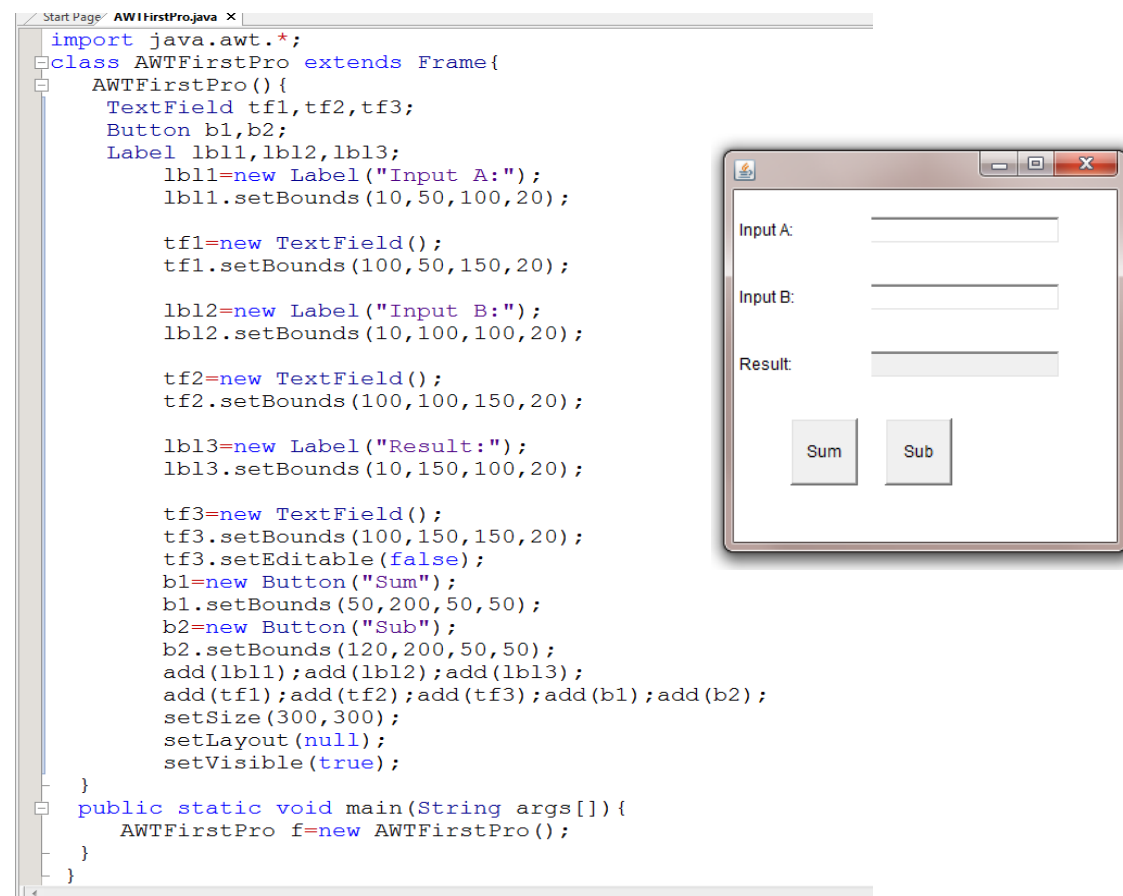
Frame: គឺជាកន្លែងរៀបចំ Component ដែលអាចប្រើប្រាស់ជាមួយនិង Menu, Title Bar.

## ឧទាហរណ៍ ១ ៖ ការរៀបចំ Layout តាមរយៈ SetBound Layout



- ➔ AWT Basics
- ➔ Event Handling
- ➔ AWT Button
- ➔ AWT Label
- ➔ AWT TextField
- ➔ **AWT TextArea** 
- ➔ AWT Checkbox
- ➔ AWT CheckboxGroup
- ➔ AWT Choice
- ➔ AWT List
- ➔ AWT Canvas
- ➔ AWT Scrollbar
- ➔ AWT MenuItem & Menu
- ➔ AWT PopupMenu
- ➔ AWT Panel
- ➔ AWT Dialog
- ➔ AWT Toolkit

## ឧទាហរណ៍ ២ ៖ ការរៀបចំ Layout តាមរយៈ SetBound Layout

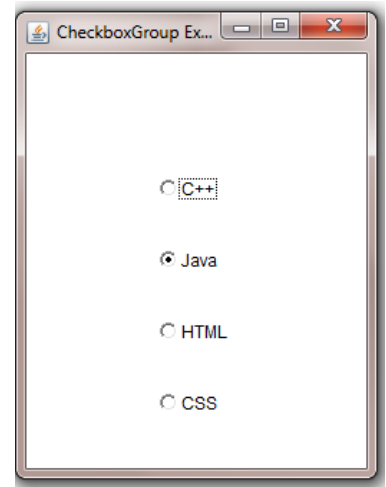


## ឧទាហរណ៍ ៣ ៖ ការរៀបចំ Layout តាមរយៈ SetBound Layout

```
import java.awt.*;
class AWTFirstPro extends Frame{
    AWTFirstPro(){
        Frame f= new Frame("CheckboxGroup Example");
        CheckboxGroup cbg = new CheckboxGroup();
        Checkbox checkBox1 = new Checkbox("C++", cbg, false);
        checkBox1.setBounds(100,100, 50,50);
        Checkbox checkBox2 = new Checkbox("Java", cbg, true);
        checkBox2.setBounds(100,150, 50,50);

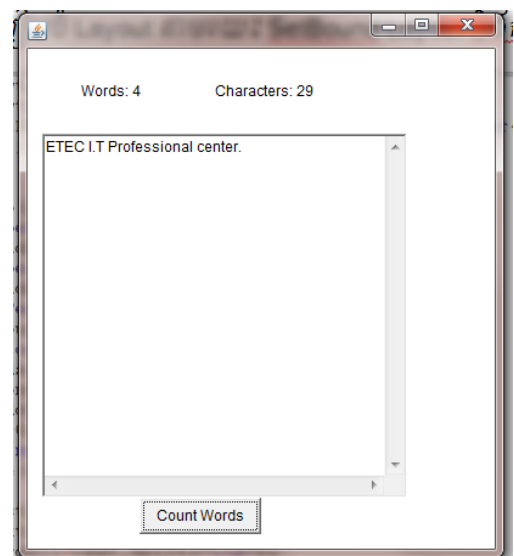
        Checkbox checkBox3 = new Checkbox("HTML", cbg, false);
        checkBox3.setBounds(100,200, 50,50);

        Checkbox checkBox4 = new Checkbox("CSS", cbg, false);
        checkBox4.setBounds(100,250, 50,50);
        f.add(checkBox1);
        f.add(checkBox2);
        f.add(checkBox3);
        f.add(checkBox4);
        f.setSize(400,400);
        f.setLayout(null);
        f.setVisible(true);
    }
    public static void main(String args[]){
        AWTFirstPro f=new AWTFirstPro();
    }
}
```



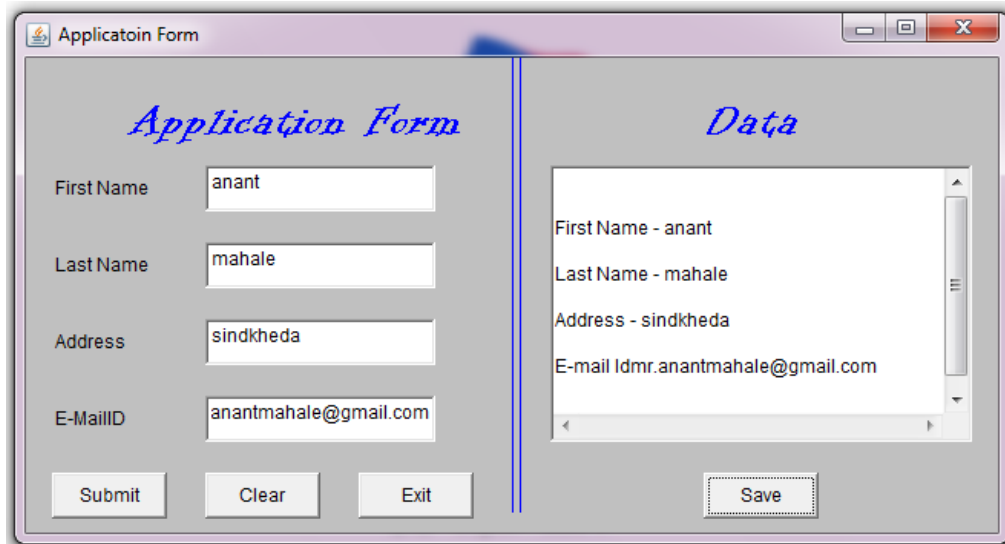
## ឧទាហរណ៍ ៤ ៖ ការរៀបចំ Layout តាមរយៈ SetBound Layout និងការដាក់ Events?

```
import java.awt.*;
import java.awt.event.*;
class AWTFirstPro extends Frame implements ActionListener{
    Label l1,l2;
    TextArea area;
    Button b;
    AWTFirstPro(){
        l1=new Label();
        l1.setBounds(50,50,100,30);
        l2=new Label();
        l2.setBounds(160,50,100,30);
        area=new TextArea();
        area.setBounds(20,100,300,300);
        b=new Button("Count Words");
        b.setBounds(100,400,100,30);
        b.addActionListener(this);
        add(l1);add(l2);add(area);add(b);
        setSize(400,450);
        setLayout(null);
        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[]){
        AWTFirstPro f=new AWTFirstPro();
    }
}
```



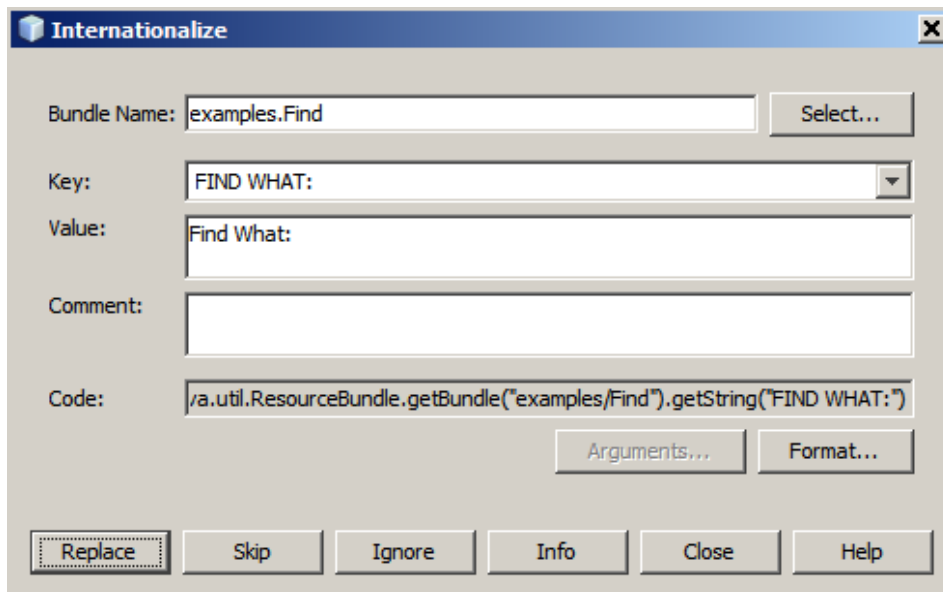
## លំហាត់អនុវត្តន៍

១)



ចូរធ្វើការបញ្ចូលព័ត៌មានតាម Text នីមួយៗ ហើយចុចលើ Button Submit បញ្ជូនទិន្នន័យ ចូលទៅក្នុង TextArea ដូចក្នុងរូប? ហើយចុច Button Save យកទិន្នន័យទាំងនោះទៅទុកក្នុង File មួយឈ្មោះ application.txt?

២)



ចូរធ្វើការ Design នូវ GUI ដូចខាងលើ?

សូមទៅមើលបន្ថែម៖ <https://www.javatpoint.com/java-awt-textarea>



III. Java Swing: គឺជាប្រភេទ Version មួយទៀតដែលបង្កើតចេញពី AWT ដែលវាបានបន្ថែមទាំងរូបរាង និង ToolBox ថ្មីៗ ជាច្រើនសំរាប់បំរើក្នុងការងារបច្ចុប្បន្ន។ Java Swing គឺជាប្រភេទ Version មួយមានការនិយមប្រើប្រាស់ខ្លាំងពីសំណាក់អ្នកប្រើប្រាស់ផ្សេងៗ ដោយសារវាអាច Support ជាមួយនិង MVC, Plugin ហើយ Plat form របស់វាមិនអាស្រ័យ គ្នានោះទេ។

ឧទាហរណ៍ ១៖ ការបង្កើតនូវ JLabel, JTextField & JButton

```

Start Page AWTFirstPro.java x
import java.awt.event.*;
import javax.swing.*;
class AWTFirstPro{

    public static void main(String args[]){
        JFrame f= new JFrame("TextField Example");
        JTextField t1,t2,t3;
        JLabel lbl1, lbl2, lbl3;
        JButton btn1, btn2, btn3;
        lbl1=new JLabel("Input A:");
        lbl1.setBounds(10, 50, 90, 30);
        t1=new JTextField("Input A");
        t1.setBounds(100, 50, 200, 30);

        lbl2=new JLabel("Input B:");
        lbl2.setBounds(10, 100, 90, 30);
        t2=new JTextField("Input B");
        t2.setBounds(100, 100, 200, 30);

        lbl3=new JLabel("Result:");
        lbl3.setBounds(10, 150, 90, 30);
        t3=new JTextField("Result");
        t3.setBounds(100, 150, 200, 30);

        btn1=new JButton("Cal");
        btn1.setBounds(20, 200, 100, 30);

        btn2=new JButton("Clear");
        btn2.setBounds(140, 200, 100, 30);

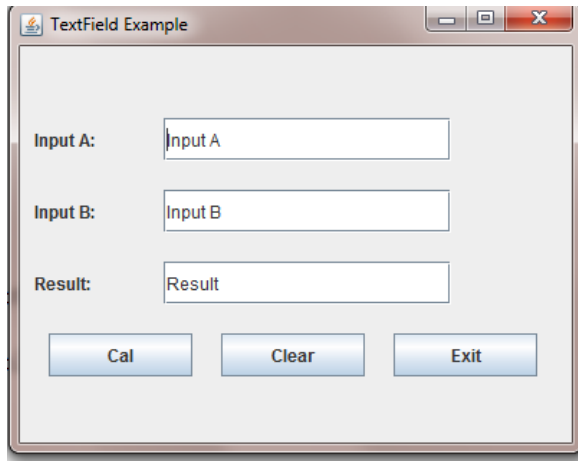
        btn3=new JButton("Exit");
        btn3.setBounds(260, 200, 100, 30);
        f.add(btn1); f.add(btn2); f.add(btn3);

        f.add(lbl1); f.add(t1); f.add(lbl2); f.add(t2);
        f.add(lbl3); f.add(t3);

        f.setSize(400, 400);
        f.setLayout(null);
        f.setVisible(true);
    }
}

```

លទ្ធផលទទួលបាន៖



ឧទាហរណ៍ ២៖ ការបង្កើតនូវ JTextArea(សំរាប់បង្ហាញText ឬ បញ្ចូល Text បានច្រើន

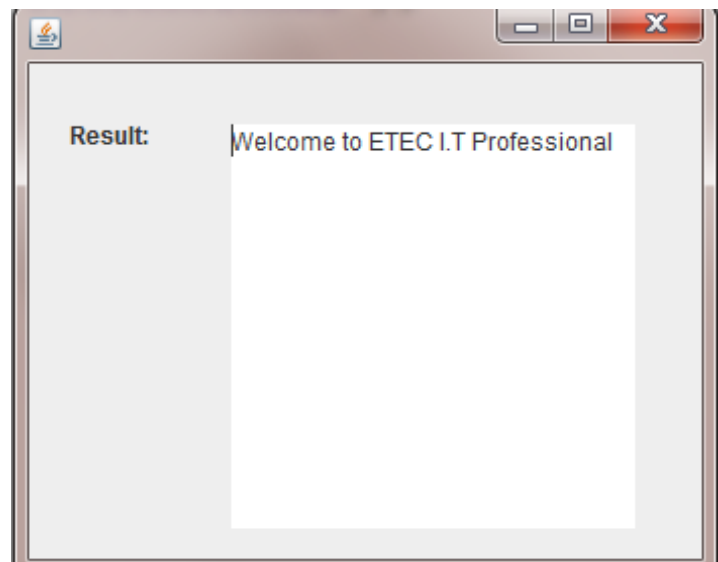
Record។

```

Start Page AWTFirstPro.java x
import java.awt.event.*;
import javax.swing.*;
class AWTFirstPro{
    public static void main(String args[]){
        JFrame f= new JFrame();
        JTextArea area=new JTextArea("Welcome to ETEC I.T Professional");
        JLabel lbl=new JLabel("Result:");
        lbl.setBounds(20,20,100,30);
        area.setBounds(100,30, 200,200);
        f.add(area);
        f.add(lbl);
        f.setSize(300,300);
        f.setLayout(null);
        f.setVisible(true);
    }
}

```

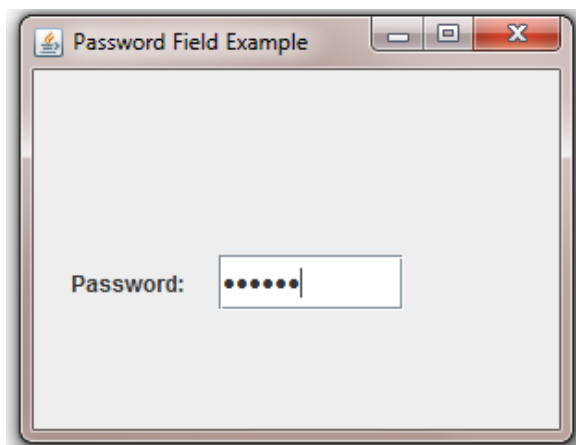
លទ្ធផលទទួលបាន៖



ឧទាហរណ៍ ៣៖ ការបង្កើតនូវ JPasswordField (សំរាប់បញ្ចូលទិន្នន័យជា Password) ។

```
Start Page AWTFirstPro.java X
import java.awt.event.*;
import javax.swing.*;
class AWTFirstPro{
    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);
    }
}
```

លទ្ធផលទទួលបាន៖



ឧទាហរណ៍ ៤៖ ការបង្កើតនូវ JCheckBox (សំរាប់ការជ្រើសរើសទិន្នន័យ)

```
Start Page AWTFirstPro.java X
import java.awt.event.*;
import javax.swing.*;
class AWTFirstPro extends JFrame implements ActionListener{
    JLabel l;
    JCheckBox cb1,cb2,cb3;
    JButton b;
    AWTFirstPro(){
        l=new JLabel("Food Ordering System");
        l.setBounds(50,50,300,20);
        cb1=new JCheckBox("Pizza @ 100");
        cb1.setBounds(100,100,150,20);
        cb2=new JCheckBox("Burger @ 30");
        cb2.setBounds(100,150,150,20);
        cb3=new JCheckBox("Tea @ 10");
        cb3.setBounds(100,200,150,20);
    }
}
```

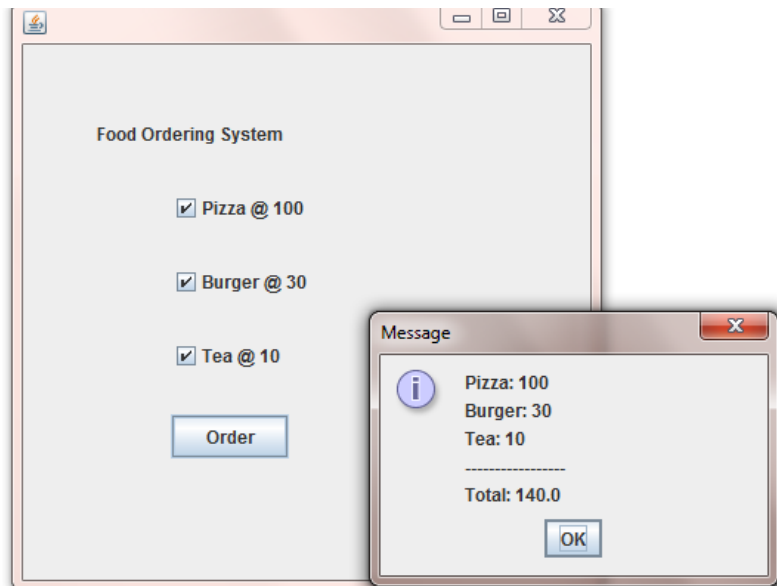


```

        b=new JButton("Order");
        b.setBounds(100,250,80,30);
        b.addActionListener(this);
        add(l);add(cb1);add(cb2);add(cb3);add(b);
        setSize(400,400);
        setLayout(null);
        setVisible(true);
        setDefaultCloseOperation(EXIT_ON_CLOSE);
    }
    public void actionPerformed(ActionEvent e){
        float amount=0;
        String msg="";
        if(cb1.isSelected()){
            amount+=100;
            msg+="Pizza: 100\n";
        }
        if(cb2.isSelected()){
            amount+=30;
            msg+="Burger: 30\n";
        }
        if(cb3.isSelected()){
            amount+=10;
            msg+="Tea: 10\n";
        }
        msg+="-----\n";
        JOptionPane.showMessageDialog(this,msg+"Total: "+amount);
    }
    public static void main(String args[]){
        new AWTFirstPro();
    }
}

```

លទ្ធផលទទួលបាន៖



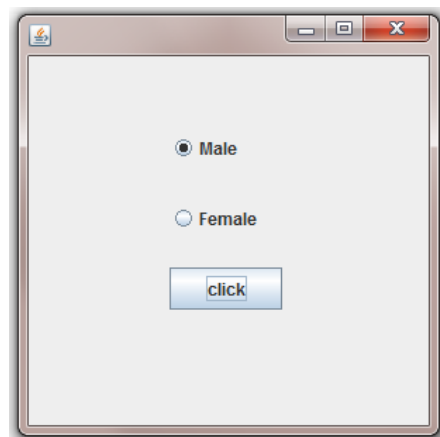
## ឧទាហរណ៍ ៥៖ ការប្រើប្រាស់ជាមួយនិង JRadioButton

```

Start Page AWTFirstPro.java x
import java.awt.event.*;
import javax.swing.*;
class AWTFirstPro extends JFrame implements ActionListener{
    JRadioButton rb1,rb2;
    JButton b;
    AWTFirstPro(){
        rb1=new JRadioButton("Male");
        rb1.setBounds(100,50,100,30);
        rb2=new JRadioButton("Female");
        rb2.setBounds(100,100,100,30);
        ButtonGroup bg=new ButtonGroup();
        bg.add(rb1);bg.add(rb2);
        b=new JButton("click");
        b.setBounds(100,150,80,30);
        b.addActionListener(this);
        add(rb1);add(rb2);add(b);
        setSize(300,300);
        setLayout(null);
        setVisible(true);
    }
    public void actionPerformed(ActionEvent e){
        if(rb1.isSelected()){
            JOptionPane.showMessageDialog(this,"You are Male.");
        }
        if(rb2.isSelected()){
            JOptionPane.showMessageDialog(this,"You are Female.");
        }
    }
    public static void main(String args[]){
        new AWTFirstPro();
    }
}

```

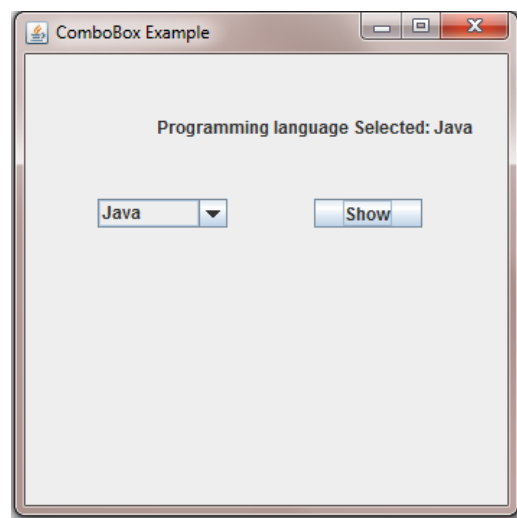
លទ្ធផលទទួលបាន៖



## ឧទាហរណ៍ ៦៖ ការប្រើប្រាស់ជាមួយនិង JComboBox

```
Start Page DemoSwing.java X
import java.awt.event.*;
import javax.swing.*;
class DemoSwing {
    JFrame f;
    DemoSwing() {
        f=new JFrame("ComboBox Example");
        final JLabel label = new JLabel();
        label.setHorizontalAlignment(JLabel.CENTER);
        label.setSize(400,100);
        JButton b=new JButton("Show");
        b.setBounds(200,100,75,20);
        String languages[]={"C", "C++", "C#", "Java", "PHP"};
        final JComboBox cb=new JComboBox(languages);
        cb.setBounds(50, 100,90,20);
        f.add(cb); f.add(label); f.add(b);
        f.setLayout(null);
        f.setSize(350,350);
        f.setVisible(true);
        b.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                String data = "Programming language Selected: "
                    + cb.getItemAt(cb.getSelectedIndex());
                label.setText(data);
            }
        });
    }
    public static void main(String args[]){
        new DemoSwing();
    }
}
```

លទ្ធផលទទួលបាន៖



## លំហាត់អនុវត្តន៍

១)



Registration Form in Java

**Registration Form in Windows Form:**

Name: Sandeep Sharma

Email-ID: sandy05.1991@gmail.com

Create Password: .....

Confirm Password: .....

Country: India

State: U.P

Phone No: 9717789441

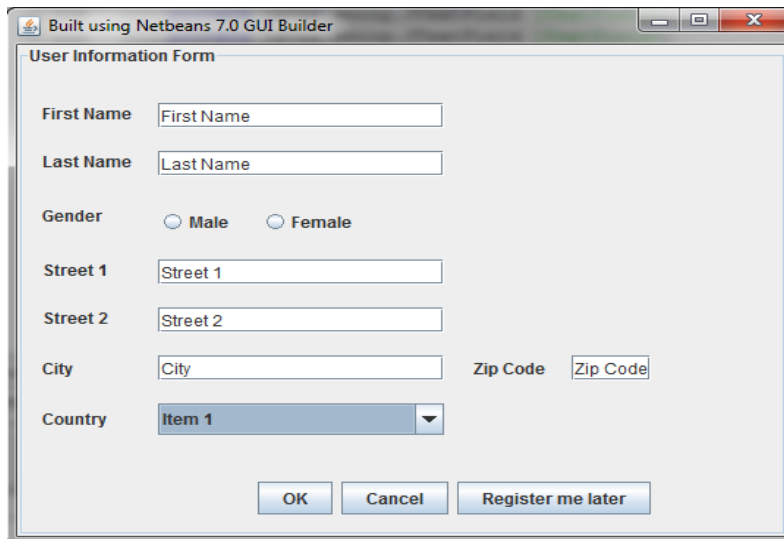
Buttons: Submit, Clear

Message: Data Saved Successfully

OK

ចូរចុច Button Submit ដោយបោះពាក្យថា Data Saved Successfully?

២)



Built using Netbeans 7.0 GUI Builder

**User Information Form**

First Name: First Name

Last Name: Last Name

Gender: ☐ Male ☐ Female

Street 1: Street 1

Street 2: Street 2

City: City Zip Code: Zip Code

Country: Item 1

Buttons: OK, Cancel, Register me later

៣)



**Login**

UserName: |

Password:

Login

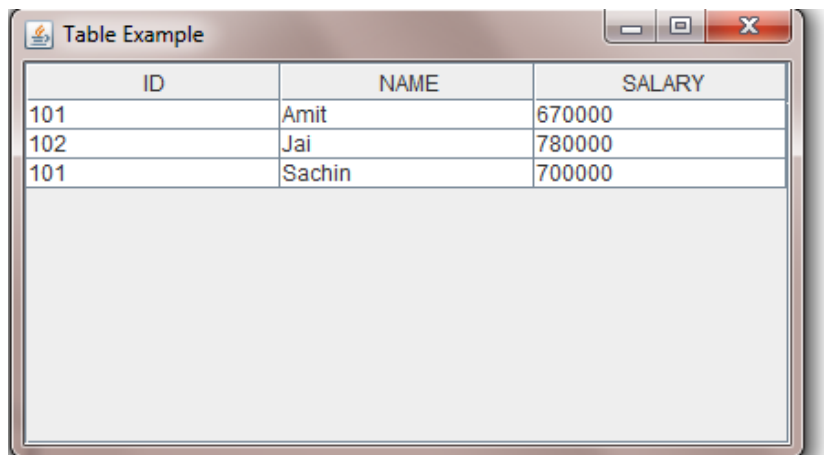
## ឧទាហរណ៍ ៧៖ ការប្រើប្រាស់ជាមួយនិង JTABLE

```

Start Page DemoSwing.java x
import javax.swing.event.*;
import javax.swing.*;
class DemoSwing {
    public static void main(String args[]){
        JFrame f = new JFrame("Table Example");
        String data[][]={ {"101","Amit","670000"},
                           {"102","Jai","780000"},
                           {"101","Sachin","700000"} };
        String column[]={"ID","NAME","SALARY"};
        final JTable jt=new JTable(data,column);
        jt.setCellSelectionEnabled(true);
        ListSelectionModel select= jt.getSelectionModel();
        select.setSelectionMode(ListSelectionModel.SINGLE_SELECTION);
        select.addListSelectionListener(new ListSelectionListener() {
            public void valueChanged(ListSelectionEvent e) {
                String Data = null;
                int[] row = jt.getSelectedRows();
                int[] columns = jt.getSelectedColumns();
                for (int i = 0; i < row.length; i++) {
                    for (int j = 0; j < columns.length; j++) {
                        Data = (String) jt.getValueAt(row[i], columns[j]);
                    }
                }
                System.out.println("Table element selected is: " + Data);
            }
        });
        JScrollPane sp=new JScrollPane(jt);
        f.add(sp);
        f.setSize(300, 200);
        f.setVisible(true);
    }
}

```

លទ្ធផលទទួលបាន៖

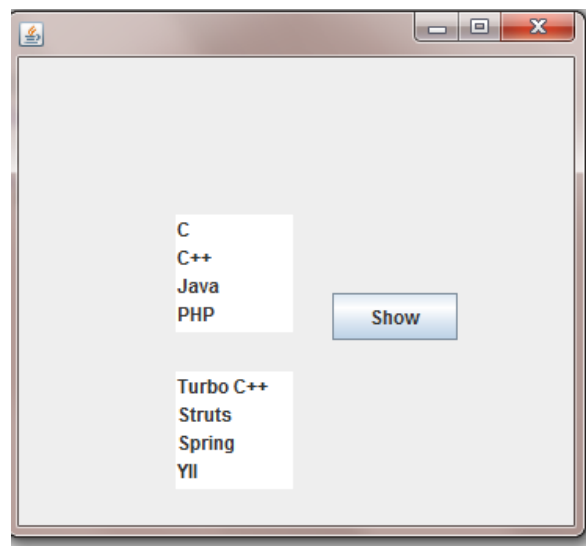


ID	NAME	SALARY
101	Amit	670000
102	Jai	780000
101	Sachin	700000

## ឧទាហរណ៍ ៨៖ ការប្រើប្រាស់ជាមួយនិង JLIST

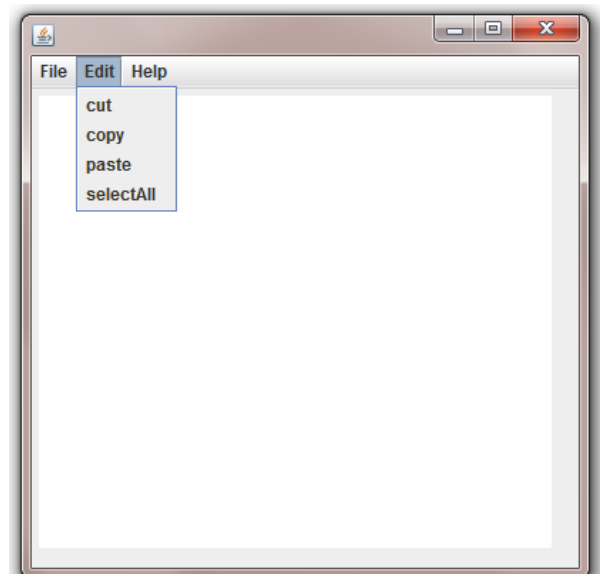
```
import java.awt.event.*;
import javax.swing.*;
class DemoSwing {
    DemoSwing() {
        JFrame f= new JFrame();
        final JLabel label = new JLabel();
        label.setSize(500,100);
        JButton b=new JButton("Show");
        b.setBounds(200,150,80,30);
        final DefaultListModel<String> l1 = new DefaultListModel<>();
        l1.addElement("C");
        l1.addElement("C++");
        l1.addElement("Java");
        l1.addElement("PHP");
        final JList<String> list1 = new JList<>(l1);
        list1.setBounds(100,100, 75,75);
        DefaultListModel<String> l2 = new DefaultListModel<>();
        l2.addElement("Turbo C++");
        l2.addElement("Struts");
        l2.addElement("Spring");
        l2.addElement("YII");
        final JList<String> list2 = new JList<>(l2);
        list2.setBounds(100,200, 75,75);
        f.add(list1); f.add(list2); f.add(b); f.add(label);
        f.setSize(450,450);
        f.setLayout(null);
        f.setVisible(true);
        b.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                String data = "";
                if (list1.getSelectedIndex() != -1) {
                    data = "Programming language Selected: " + list1.getSelectedValue();
                    label.setText(data);
                }
                if(list2.getSelectedIndex() != -1){
                    data += ", Framework Selected: ";
                    for(Object frame :list2.getSelectedValues()){
                        data += frame + " ";
                    }
                }
                label.setText(data);
            }
        });
    }
    public static void main(String args[]){
        new DemoSwing();
    }
}
```

លទ្ធផលទទួលបាន៖



## ឧទាហរណ៍ ៩៖ ការប្រើប្រាស់ជាមួយនិង JMenu or Sub Menu

```
Start Page DemoSwing.java x
import java.awt.event.*;
import javax.swing.*;
public class DemoSwing implements ActionListener{
    JFrame f;
    JMenuBar mb;
    JMenu file,edit,help;
    JMenuItem cut,copy,paste,selectAll;
    JTextArea ta;
    DemoSwing(){
        f=new JFrame();
        cut=new JMenuItem("cut");
        copy=new JMenuItem("copy");
        paste=new JMenuItem("paste");
        selectAll=new JMenuItem("selectAll");
        cut.addActionListener(this);
        copy.addActionListener(this);
        paste.addActionListener(this);
        selectAll.addActionListener(this);
        mb=new JMenuBar();
        file=new JMenu("File");
        edit=new JMenu("Edit");
        help=new JMenu("Help");
        edit.add(cut);edit.add(copy);edit.add(paste);edit.add(selectAll);
        mb.add(file);mb.add(edit);mb.add(help);
        ta=new JTextArea();
        ta.setBounds(5,5,360,320);
        f.add(mb);f.add(ta);
        f.setJMenuBar(mb);
        f.setLayout(null);
        f.setSize(400,400);
        f.setVisible(true);
    }
    public void actionPerformed(ActionEvent e) {
        if(e.getSource()==cut)
            ta.cut();
        if(e.getSource()==paste)
            ta.paste();
        if(e.getSource()==copy)
            ta.copy();
        if(e.getSource()==selectAll)
            ta.selectAll();
    }
    public static void main(String[] args) {
        new DemoSwing();
    }
}
```



លទ្ធផលទទួលបាន៖

សូមមើល <https://www.javatpoint.com/java-jmenuitem-and-jmenu>

លំហាត់អនុវត្តន៍

១)

**Database Application Example**

File Help

Studen...	Stud Fi...	Stud La...	Stud Dat...	Stud ...	Stud ...	Stud Ma...	Stud Co...
78998	Satkorn	Chengmo	Jan 26, 1968	34 Upto...	MSc. IT	2	Thailand
79678	Mohammed	Fajr	Apr 20, 1975	Pearl Ap...	MSc. Ex...	2	UEA
87990	Haslina	Mahathir	Nov 12, 1970	345, Ne...	MSc. ICT	2	Malaysia
88799	Mustar	Mohd Dali	Jun 24, 1979	345, Sid...	MSc. Mul...	1	Indonesia
88889	Albukori	Zaman Khan	Jul 8, 1969	4-5, De...	MSc. IT	1	Malaysia

Student Id:

Stud First Name:

Stud Last Name:

Stud Date Of Birth:

Stud Address:

Stud Program:

Stud Marital Status:

Stud Country:

២)

**Employee Records Editor**

employee_id	full_name	gender	department	position	salary
1	John Doe	Male	Admin	CEO	6,500
2	Richard Castle	Male	Investigation	Writer	2,600
3	Sheldan ...	Male	Lab	Scientist	1,300
4	Penny	Female	Kitchen	Chef	1,200

Employee ID:

Full Name:

Gender:  Department:

Position:  Salary:



## IV. Layout Manager

គឺជាប្រភេទ Layout ដែលមានស្រាប់សំរាប់រៀបចំ Toolbox របស់អ្នកទៅតាម Layout មានស្រាប់របស់ Java។ Layout ដែលមានស្រាប់ដូចជា BorderLayout, FlowLayout, GridLayout, CardLayout, GridBagLayout, BoxLayout, GroupLayout។

១) Border Layout: គឺជាប្រភេទ Layout ដែលរៀបចំនូវ Control របស់អ្នកតាម ទិសតំបន់ របស់ Layout ដូចជា៖ NORTH, SOUTH, EAST, WEST និង CENTER ។

ឧទាហរណ៍ ១៖

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

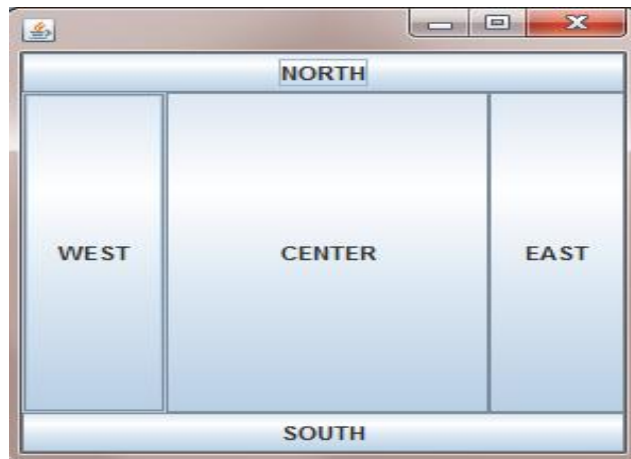
public class LayoutManager{
    JFrame f;
    public LayoutManager () {
        f=new JFrame ();

        JButton b1=new JButton ("NORTH");
        JButton b2=new JButton ("SOUTH");
        JButton b3=new JButton ("EAST");
        JButton b4=new JButton ("WEST");
        JButton b5=new JButton ("CENTER");

        f.add(b1,BorderLayout.NORTH);
        f.add(b2,BorderLayout.SOUTH);
        f.add(b3,BorderLayout.EAST);
        f.add(b4,BorderLayout.WEST);
        f.add(b5,BorderLayout.CENTER);

        f.setSize(300,300);
        f.setVisible(true);
    }
    public static void main(String[] args) {
        new LayoutManager();
    }
}
```

លទ្ធផលទទួលបាន៖



២) Grid Layout: គឺជាការរៀបចំ តាមបែបជាតារាង ដែលមានជួរដេក និង ជួរឈរ។  
ឧទាហរណ៍ ១៖

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

public class LayoutManager{
    JFrame f;
    LayoutManager(){
        f=new JFrame();

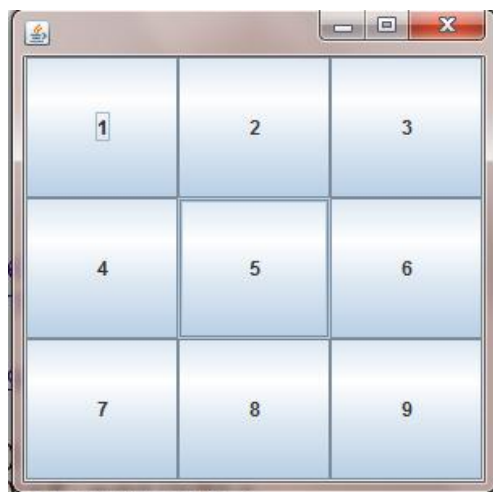
        JButton b1=new JButton("1");
        JButton b2=new JButton("2");
        JButton b3=new JButton("3");
        JButton b4=new JButton("4");
        JButton b5=new JButton("5");
        JButton b6=new JButton("6");
        JButton b7=new JButton("7");
        JButton b8=new JButton("8");
        JButton b9=new JButton("9");

        f.add(b1);f.add(b2);f.add(b3);f.add(b4);f.add(b5);
        f.add(b6);f.add(b7);f.add(b8);f.add(b9);

        f.setLayout(new GridLayout(3,3));
        //setting grid layout of 3 rows and 3 columns

        f.setSize(300,300);
        f.setVisible(true);
    }
    public static void main(String[] args) {
        new LayoutManager();
    }
}
```

លទ្ធផលទទួលបាន៖



៣) Flow Layout: គឺជាការរៀបចំ ទិសដៅ ដូចជា RIGHT, LEFT, CENTER។  
ឧទាហរណ៍ ១៖

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

public class LayoutManager{
    JFrame f;
    LayoutManager() {
        f=new JFrame();

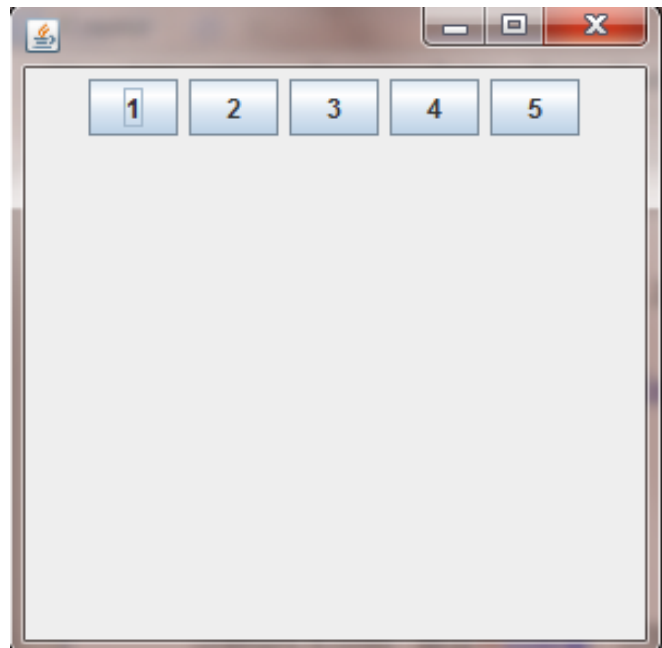
        JButton b1=new JButton("1");
        JButton b2=new JButton("2");
        JButton b3=new JButton("3");
        JButton b4=new JButton("4");
        JButton b5=new JButton("5");

        f.add(b1);f.add(b2);f.add(b3);f.add(b4);f.add(b5);

        f.setLayout(new FlowLayout(FlowLayout.CENTER));
        //setting flow layout of right alignment

        f.setSize(300,300);
        f.setVisible(true);
    }
    public static void main(String[] args) {
        new LayoutManager();
    }
}
```

លទ្ធផលទទួលបាន៖



២) Box Layout: គឺជាការរៀបចំ Components ទៅតាម Vertical ឬ Horizontal ដូចជា X\_AXIS, Y\_AXIS ។

ឧទាហរណ៍ ១៖

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

public class LayoutManager extends Frame {
    Button buttons[];

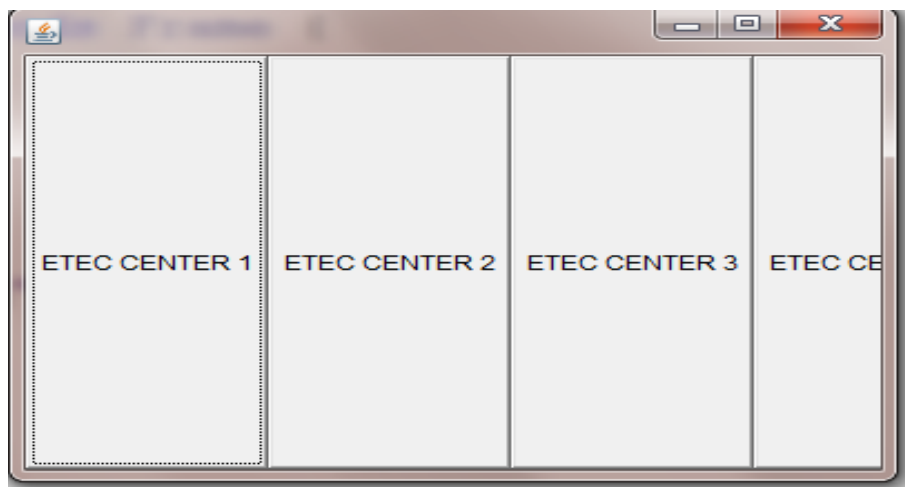
    public LayoutManager () {
        buttons = new Button [5];

        for (int i = 0;i<5;i++) { |
            buttons[i] = new Button ("ETEC CENTER " + (i + 1));
            add (buttons[i]);
        }

        setLayout (new BoxLayout (this, BoxLayout.X_AXIS));
        setSize(400,400);
        setVisible(true);
    }

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

លទ្ធផលទទួលបាន៖



២) Card Layout: សំដៅលើការរៀបចំ Layout ទៅតាមសន្លឹកបៀវ មួយហ្នឹង ដែលអ្នកអាចបញ្ជាទៅ Method មានស្រាប់របស់ Card Layout ដូចជា next, previous, first, last.

ឧទាហរណ៍ ១៖

```

LayoutManager.java * X
JButton b1,b2,b3,b4,b5;
Container c;
LayoutManager() {

    c=getContentPane();
    card=new CardLayout(40,30);
    //create CardLayout object with 40 hor space and 30 ver space
    c.setLayout(card);

    b1=new JButton("ETEC");
    b2=new JButton("I.T");
    b3=new JButton("Professional");
    b4=new JButton("Training");
    b5=new JButton("CENTER");
    b1.addActionListener(this);
    b2.addActionListener(this);
    b3.addActionListener(this);
    b4.addActionListener(this);
    b5.addActionListener(this);
    c.add("a",b1);c.add("b",b2);c.add("c",b3);
    c.add("c",b4);c.add("c",b5);

}

public void actionPerformed(ActionEvent e) {
    card.next(c);
}

public static void main(String[] args) {
    LayoutManager cl=new LayoutManager();
    cl.setSize(400,400);
    cl.setVisible(true);
    cl.setDefaultCloseOperation(EXIT_ON_CLOSE);
}
}

```

លទ្ធផលទទួលបាន៖

