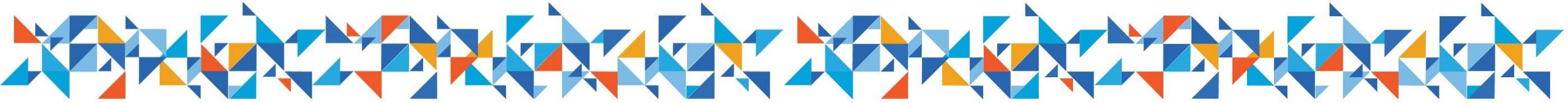
پوهنتون کاردان
KARDAN UNIVERSITY

Object Oriented Programming (Java)

Graphical User Interface (GUI)

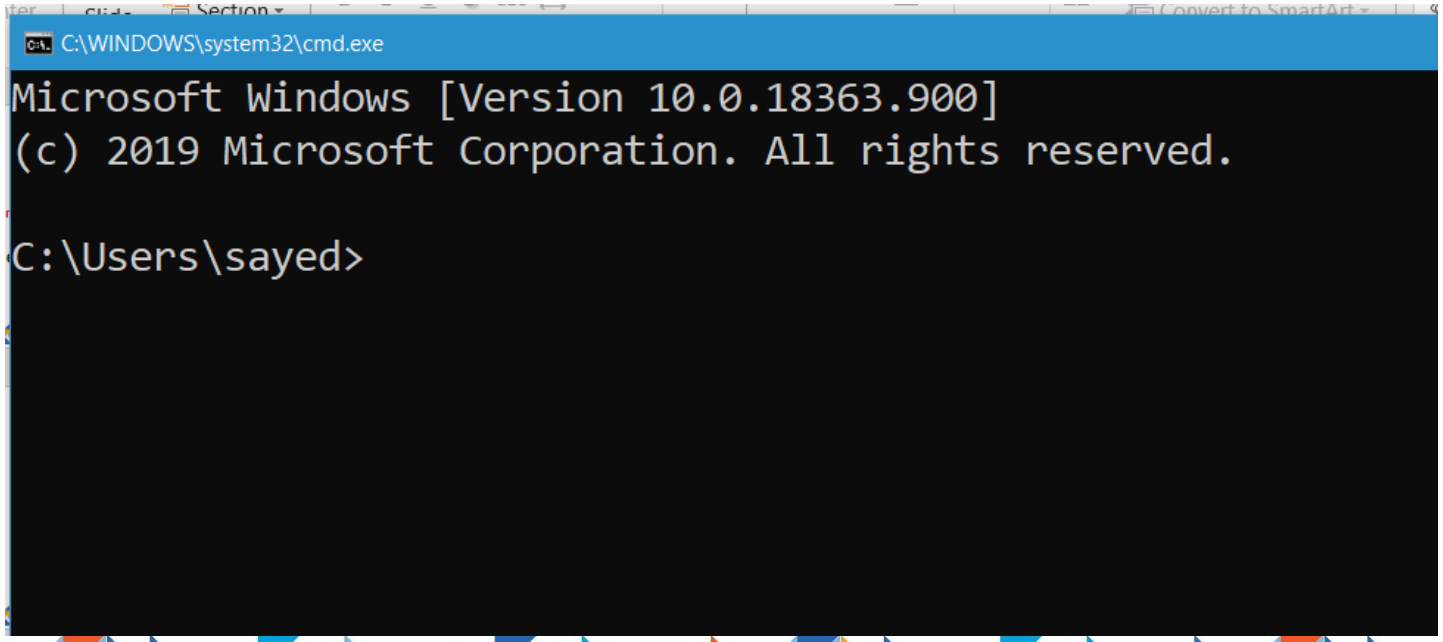
Learning Outcomes

- Graphical User Interface
- AWT vs Swing
- Java GUI
- Swing Components
- Frame
- Label
- Button
- Text Field
- Password Field
- Text Area



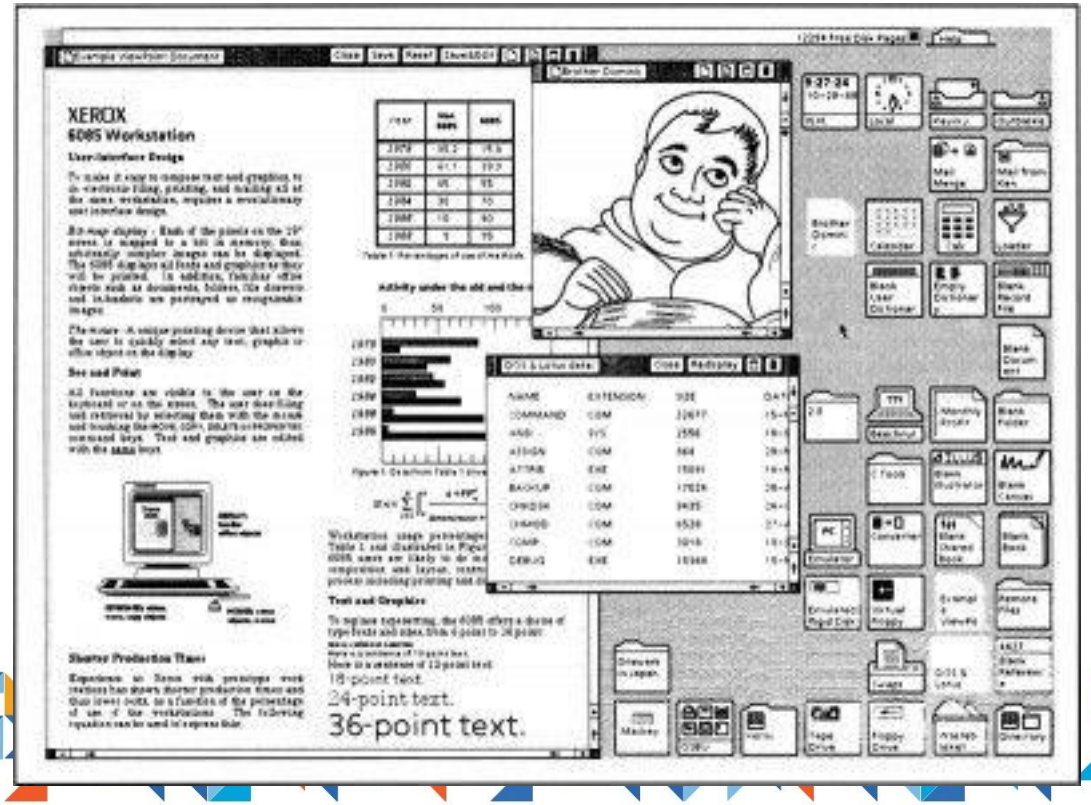
What is Graphical User Interface

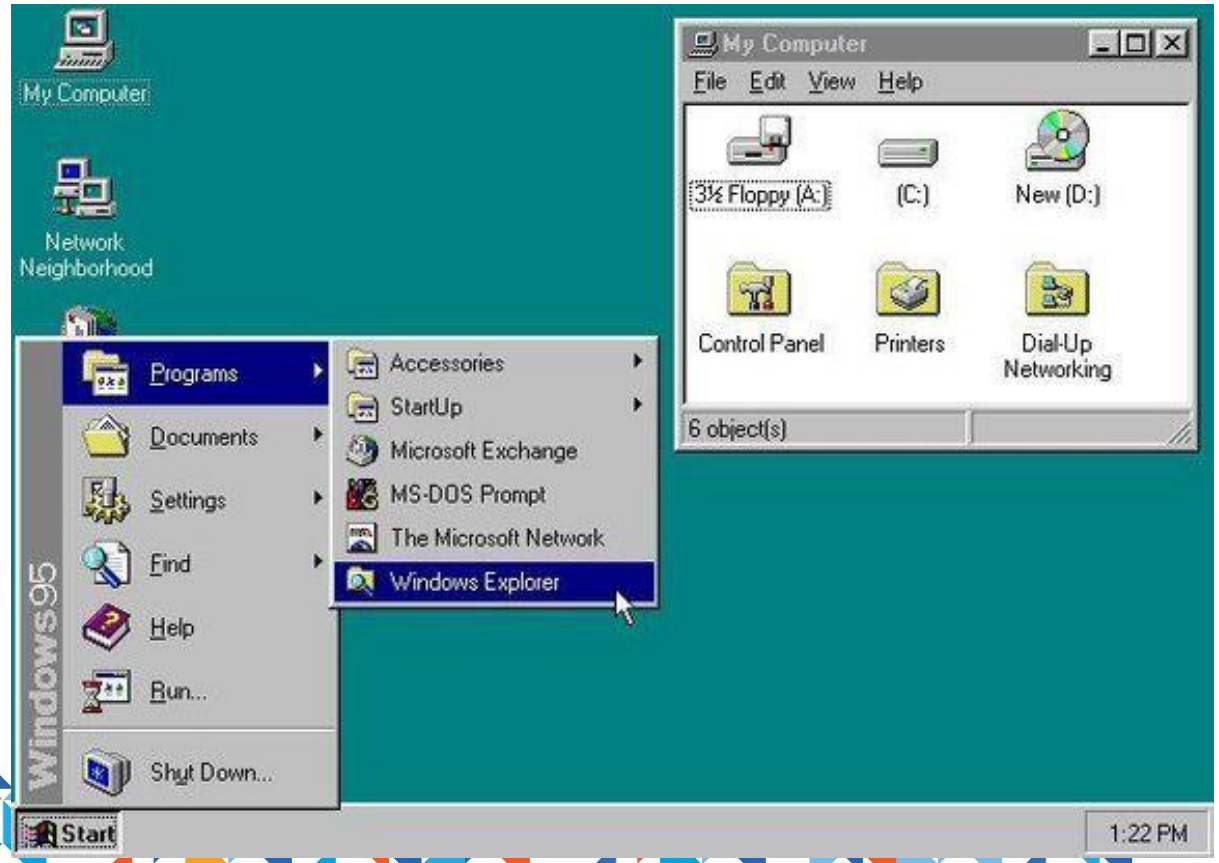
- The previous command line systems were awkward to use
- It was not user friendly
- Difficult for end users to interact with



Contd.

- With the development of Home PCs the need for GUI has increased
- XEROX PARC (1973)
- Windows 95 (1995) etc.



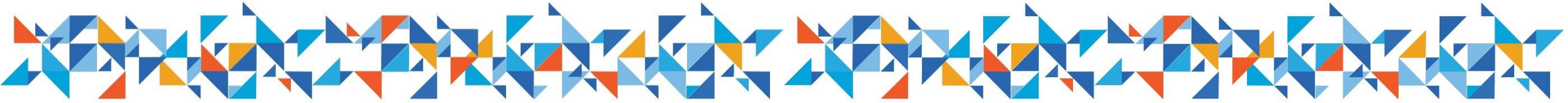


JAVA Swing

پوهنتون کاردان
KARDAN UNIVERSITY

- **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.

- Unlike AWT, Java Swing provides platform-independent and lightweight components.

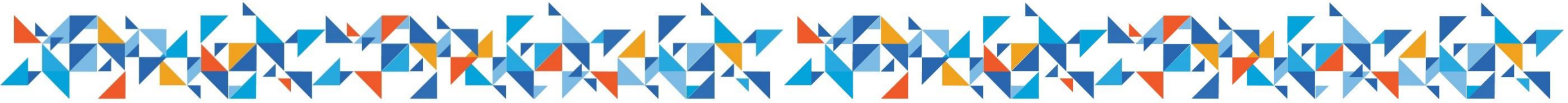


Difference between AWT and Swing

No.	Java AWT	Java Swing
1)	AWT components are **platform-dependent**.	Java swing components are **platform-independent**.
2)	AWT components are **heavyweight**.	Swing components are **lightweight**.
3)	AWT **doesn't support pluggable look and feel**.	Swing **supports pluggable look and feel**.
4)	AWT provides **less components** than Swing.	Swing provides **more powerful components** such as tables, lists, scrollpanes, colorchooser, tabbedpane etc.
5)	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.	Swing **follows MVC**.

Swing Components

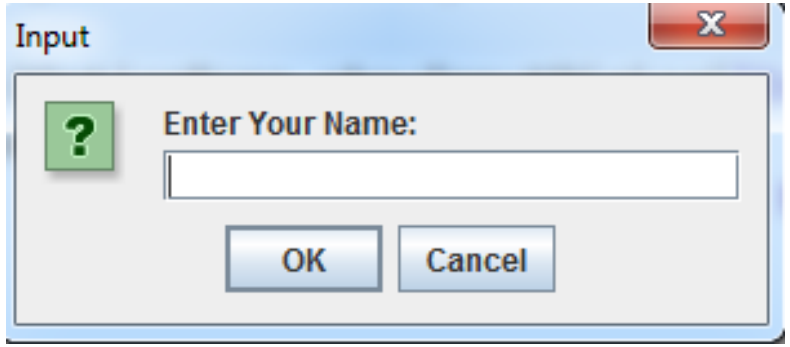
- Some of the Swing Components are as follows
- JOptionPane
- JFrame
- JButton
- JPanel
- JCheckBox
- JTextField
- JPasswordField
- JRadioButton in ButtonGroup
- JComboBox
- JLabel
- JMenuBar
- JMenuItem
- JCheckBox



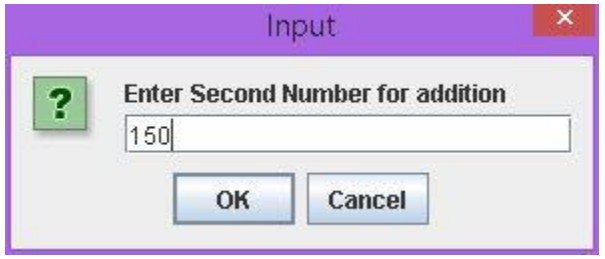
JOptionPane

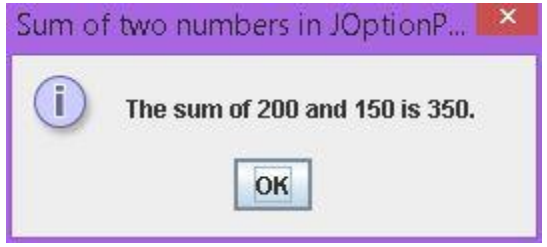
- The class **JOptionPane** is a component which provides standard methods to pop up a standard dialog box for a value or informs the user of something.

```
4  public class Option {
5
6      public static void main(String[] args) {
7          String name;
8          String fName;
9
0          name=JOptionPane.showInputDialog("Enter Your Name:");
```



Arithmetic Operations using JOptionPane

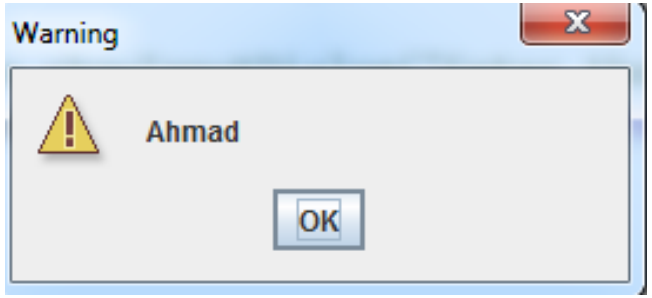




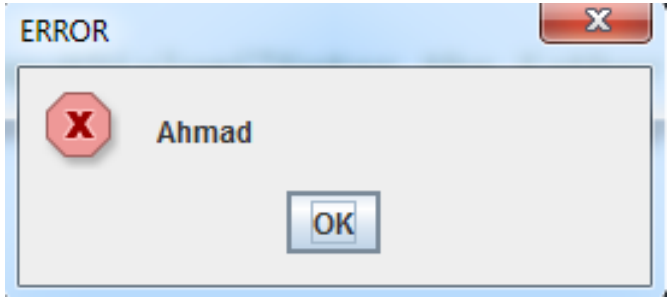
Arithmetic Operations using JOptionPane

```
import javax.swing.JOptionPane;
public class optionapanr {
    public static void main(String[] args) {

        String a=JOptionPane.showInputDialog("Enter First Number for addtion");
        String b=JOptionPane.showInputDialog("Enter Second Number for addition");

        int num1=Integer.parseInt(a);
        int  num2=Integer.parseInt(b);
        int  add=num1+num2;
        JOptionPane.showMessageDialog(null,"The sum of "+num1+ " and "+ num2+ " is "+add+".",
                "Sum of two numbers in JOptionPane",JOptionPane.INFORMATION_MESSAGE);
        String c=JOptionPane.showInputDialog("Enter First Number for subtraction");
        String d=JOptionPane.showInputDialog("Enter Second Number for subtraction");

        int num3=Integer.parseInt(c);
        int  num4=Integer.parseInt(d);

        int sub=num3-num4;
JOptionPane.showMessageDialog(null, "The subtraction of "+num3+" and "+num4+" is "+sub+".",
        "subtraction of 2 numbers",JOptionPane.INFORMATION_MESSAGE);
    }
}
```

Contd.

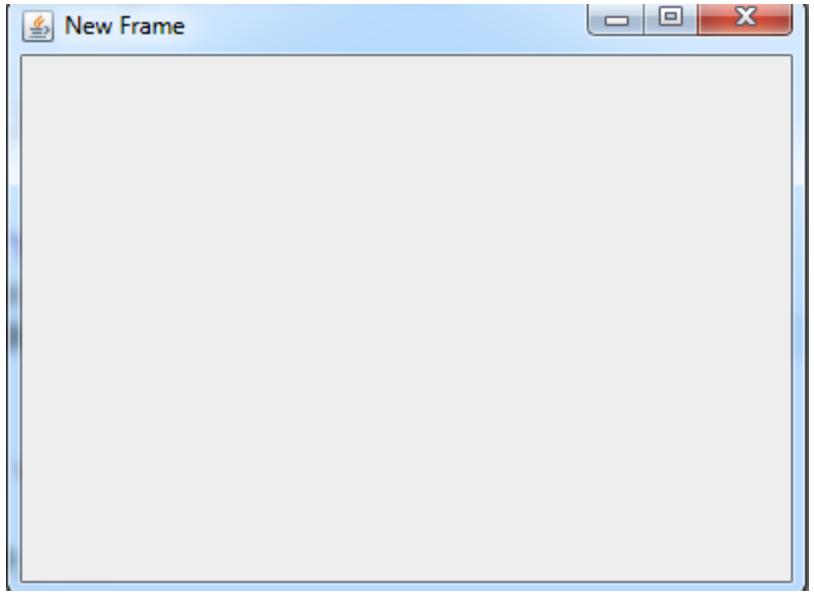


- There are different output messages:
- Normal message:
  - JOptionPane.*showMessageDialog(**null,***name);
- Warning Message:
  - JOptionPane.*showMessageDialog(**null,*** name,” Warning”, JOptionPane.WARNING_MESSAGE);
- Error Message:
  - JOptionPane.*showMessageDialog(**null,*** name,” Error”, JOptionPane.ERROR_MESSAGE);



How to create a Frame?

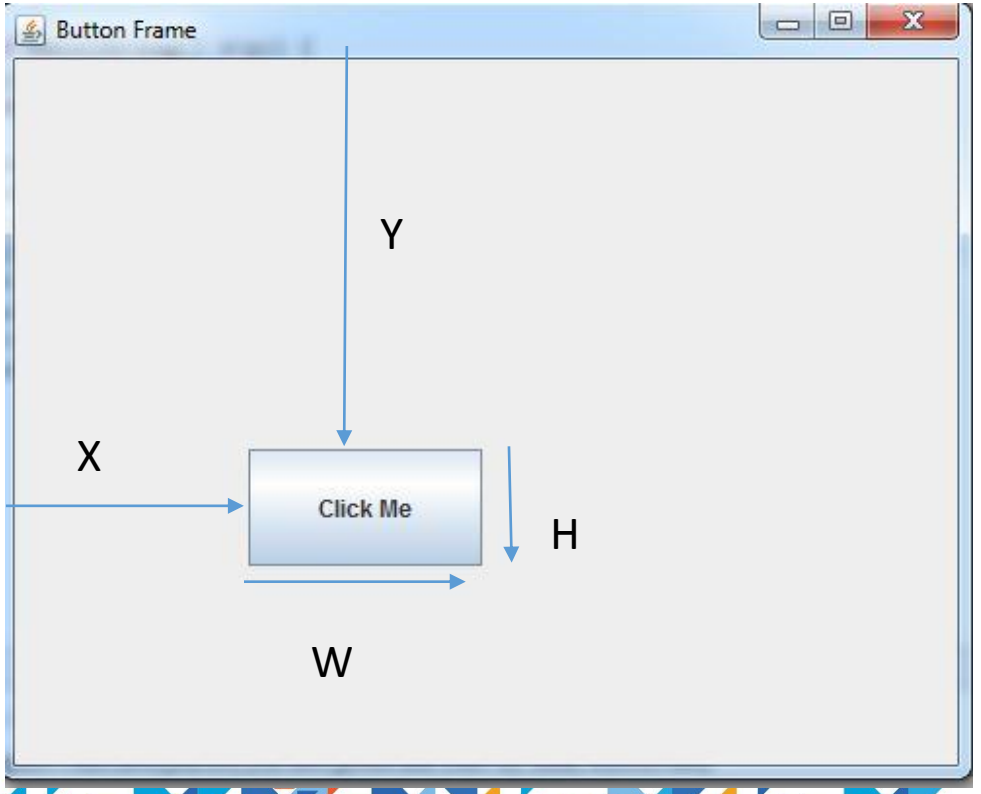
- Frame can contain different components in it,
- It has a top panel, with the minimize, maximize and close buttons
- JFrame works like the main window where components like labels, buttons, textfields are added to create a GUI.



- import javax.swing.JFrame;
- //
- JFrame frame=new JFrame(“New Frame”);

JButton

- The JButton class is used to create a labeled button
- The application result in some action when the button is pushed by using ActionListener class

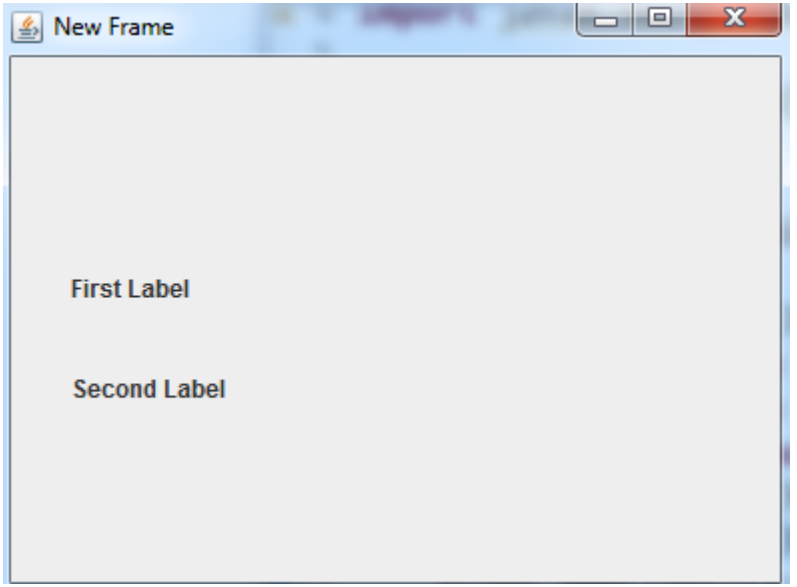


- import javax.swing.JButton;
- ///
- JButton button=new JButton(“Click Me”);
- button.setBounds(x,y,w,h)
- frame.add(button);

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.

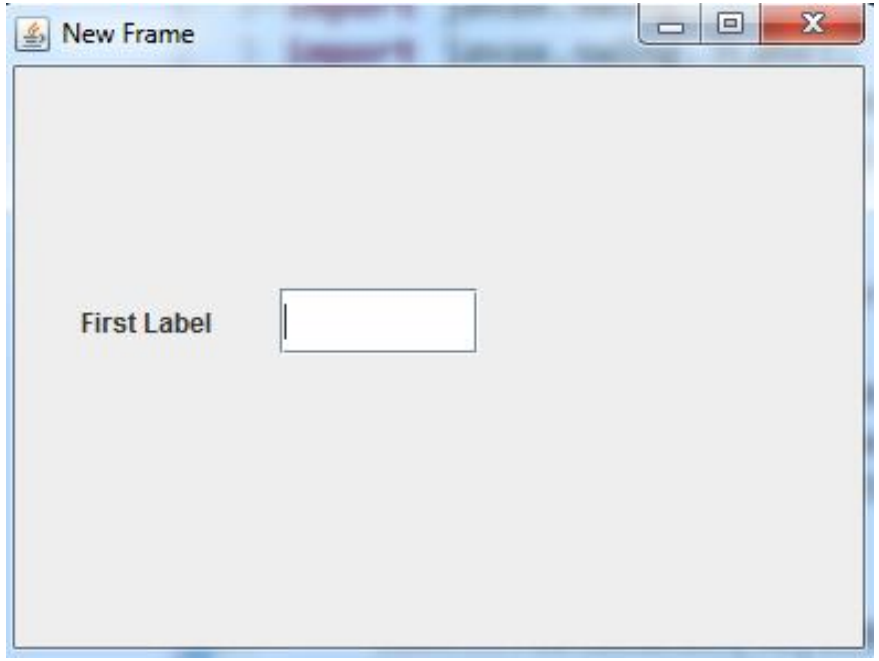
- import javax.swing.JLabel;
- //
- JLabel label=new JLabel(“First Label”);
- label.setBounds(30,100,90,30);
- frame.add(label);



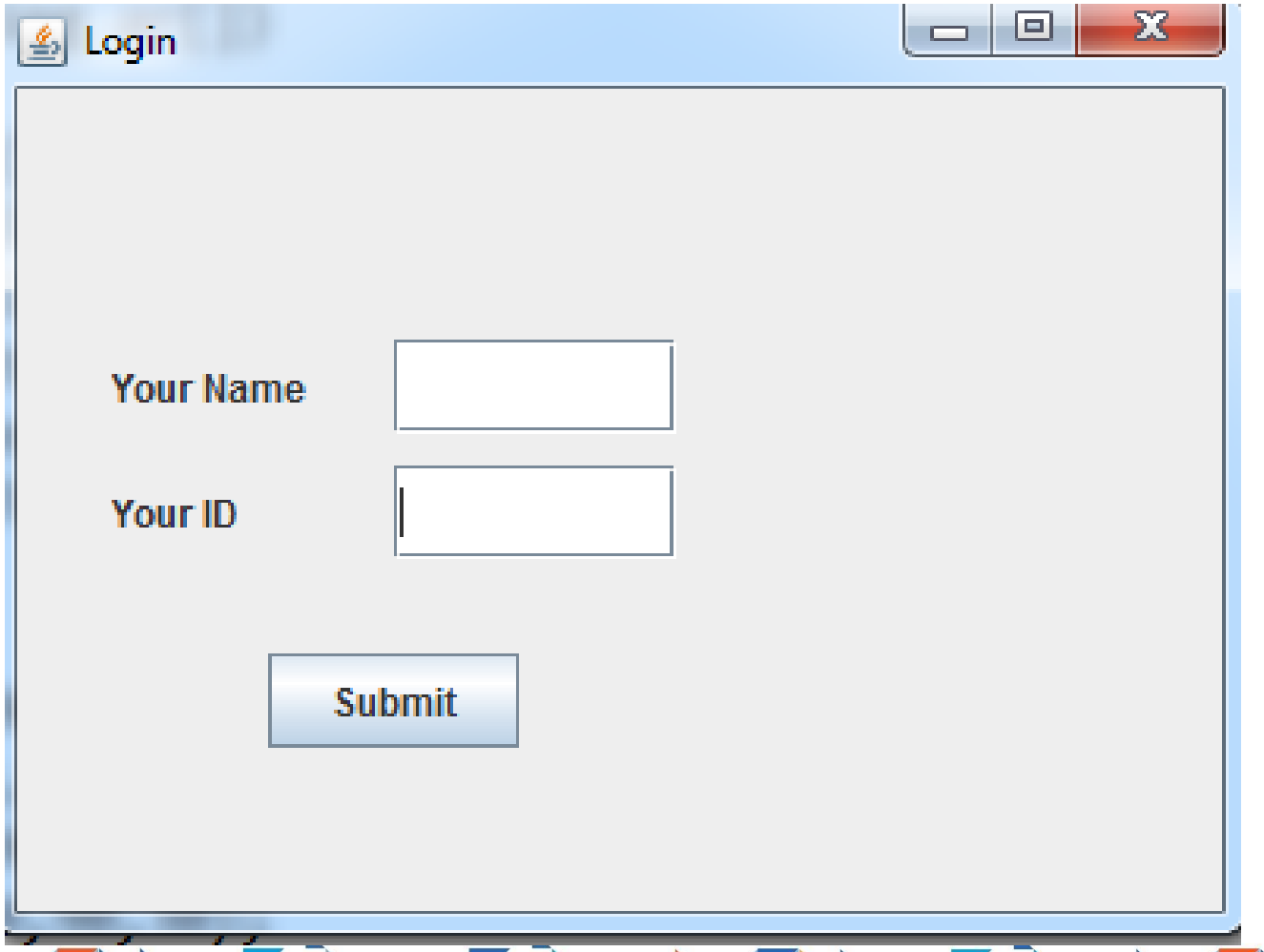
JTextField

- JTextField is used to get input values
- It is having one line input text

- import javax.swing.JTextField;
- //
- JTextField input=new JTextField();
- input.setBounds(40,100, 90,30);



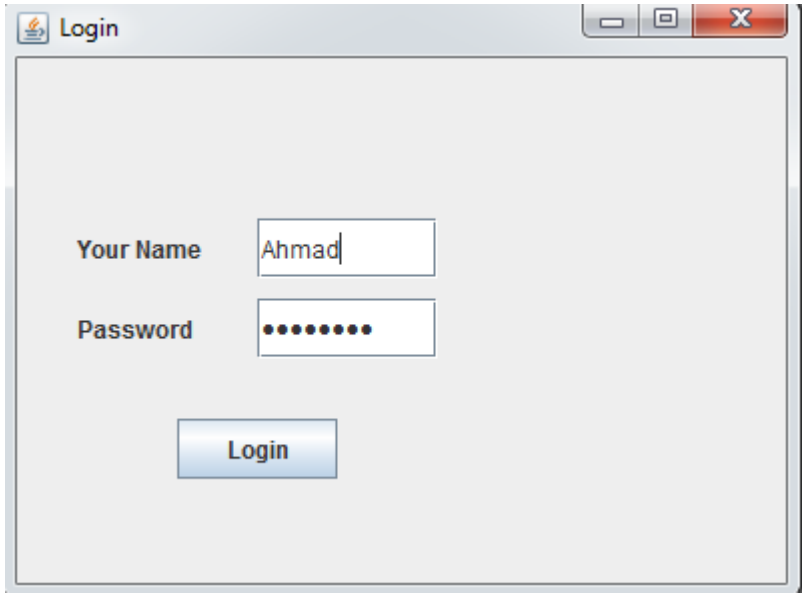
Create the following



JPasswordField

- Password Field is used to provide input for the passwords
- It encrypts the password

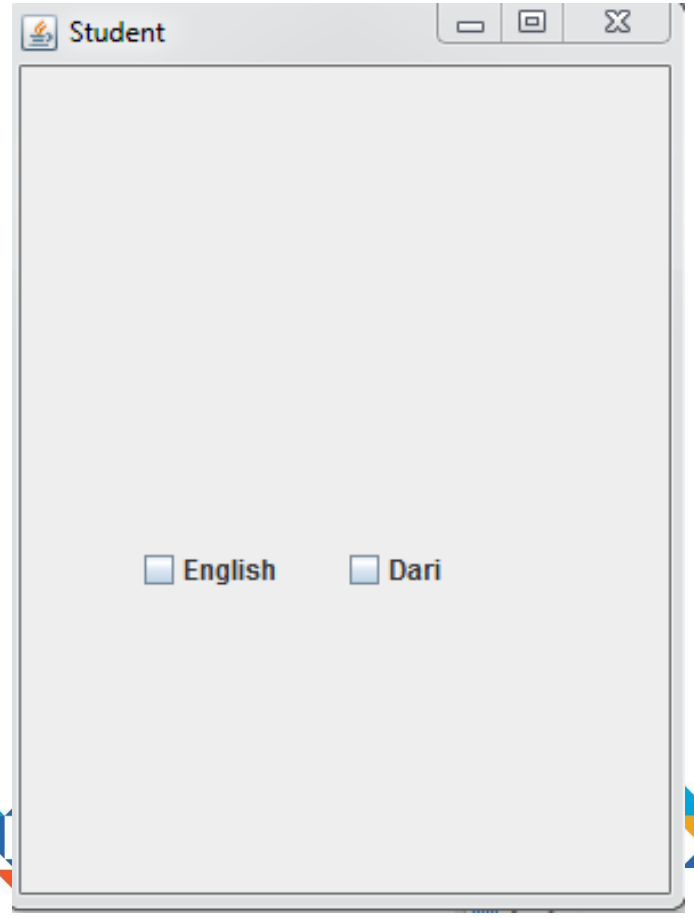
- import javax.swing.JPasswordField;
- //
- JPasswordField pass=new JPasswordField();



JCheckBox

- The JCheckBox class is used to create checkboxes.
- It is used to turn an option on (true) or off (false) by selecting and deselecting it.
- You can choose as many options as available

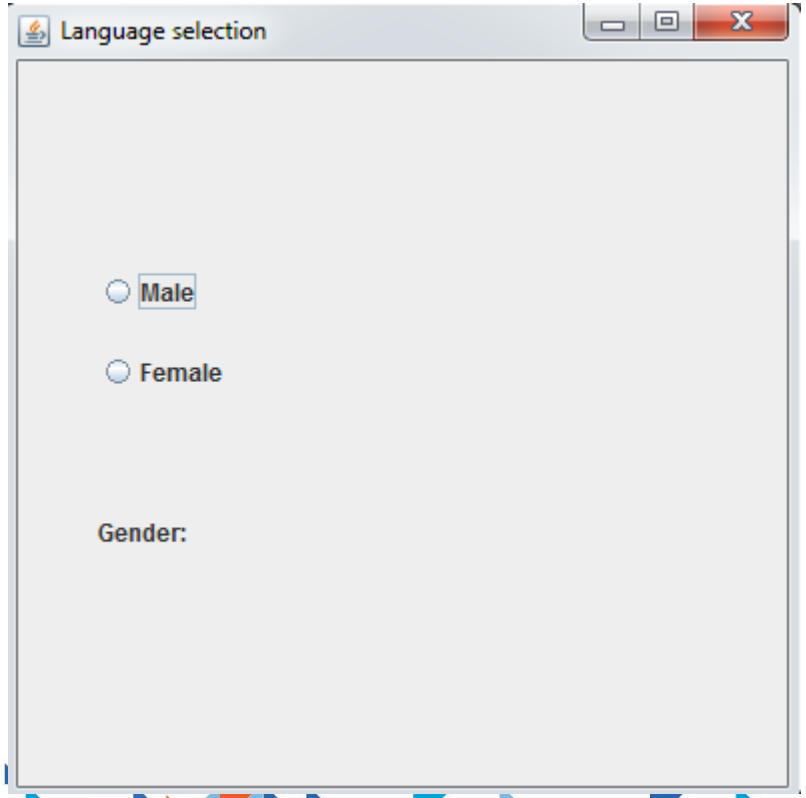
```
JCheckBox box1=new JCheckBox("English");
box1.setBounds(50,200,80,40);
f.add(box1);
JCheckBox box2=new JCheckBox("Dari");
box2.setBounds(140,200,80,40);
f.add(box2);
```



Radio Button

- 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

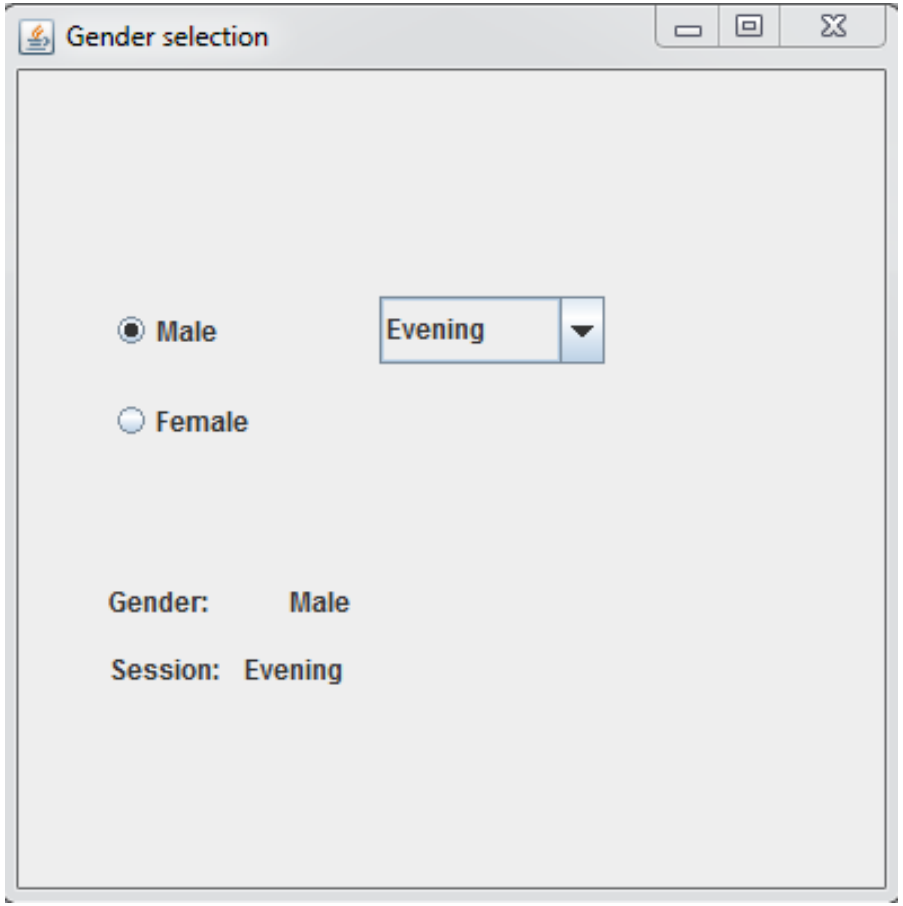
  - JRadioButton r1=**new** JRadioButton(“Male");
  - JRadioButton r2=**new** JRadioButton(“Female");
  - r1.setBounds(75,50,100,30);
  - r2.setBounds(75,100,100,30);
  - ButtonGroup bg=**new** ButtonGroup();
  - bg.add(r1);bg.add(r2);
  - f.add(r1);f.add(r2);



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.

- String session[]={“Morning”, “Evening”};
- JComboBox box=new JComboBox(session);
- box.setBounds(100,100,80,40);
- Fr.add(box);

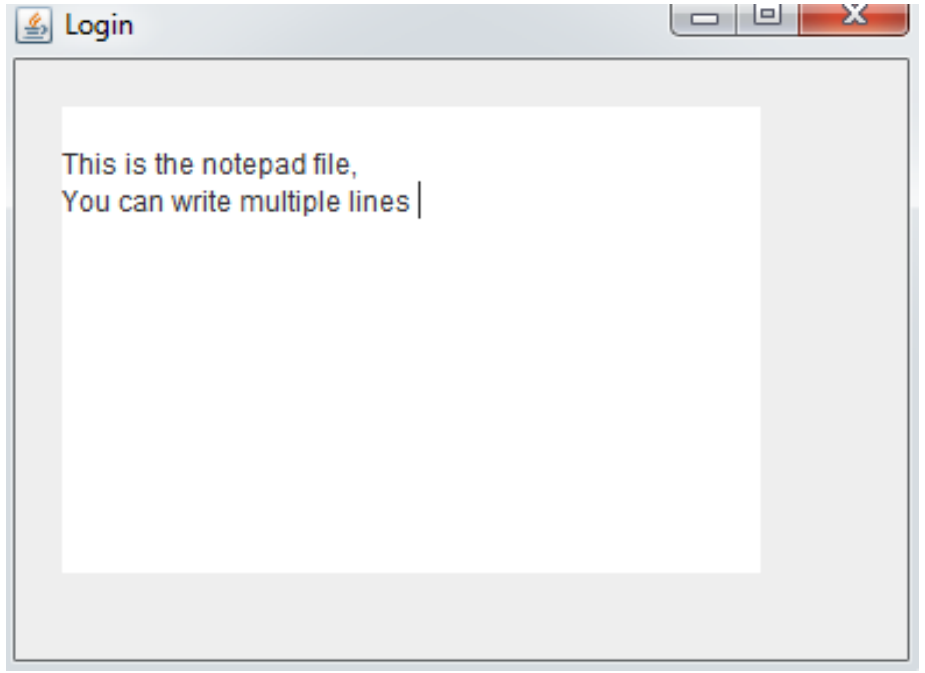


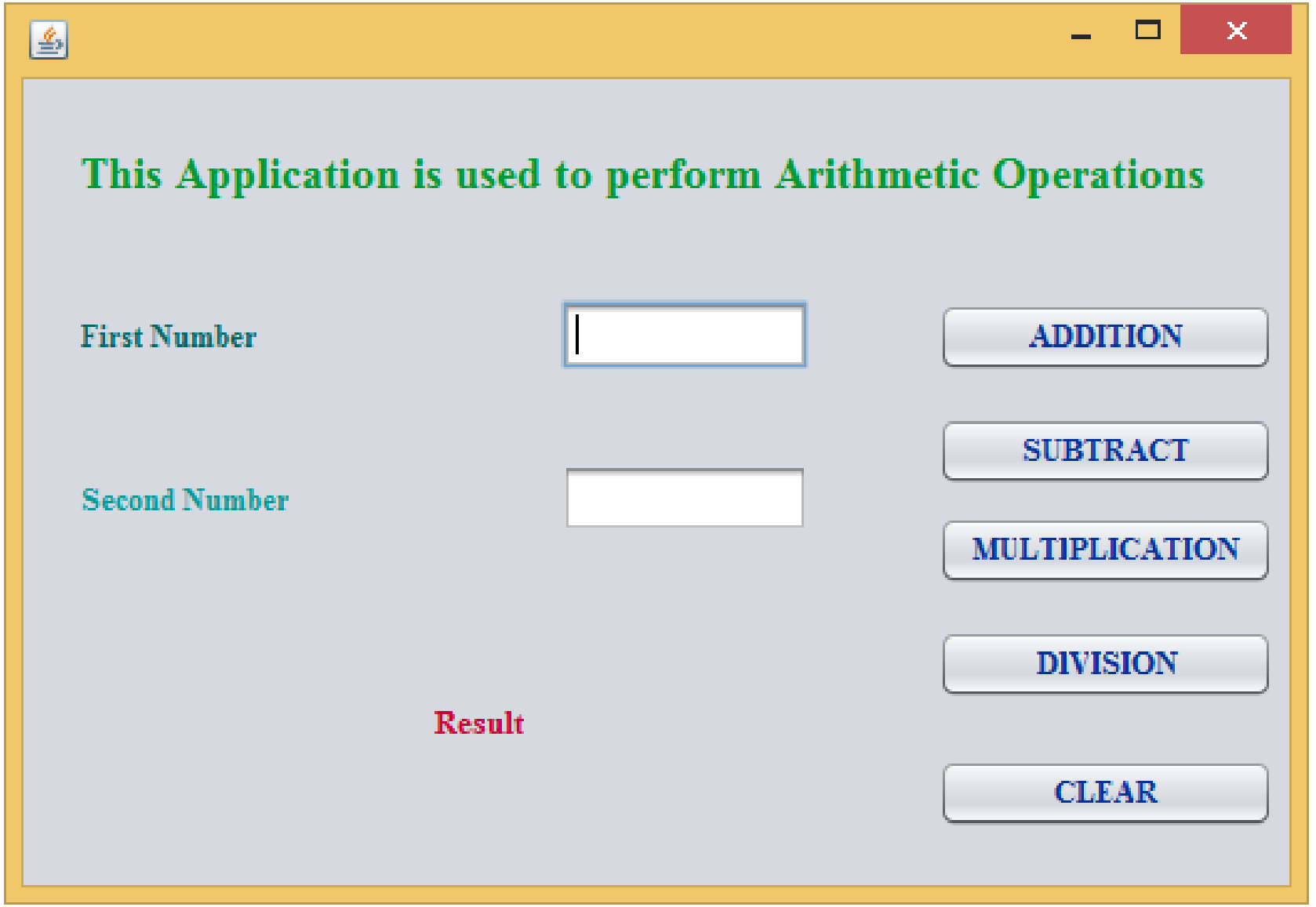
JTextArea

- Text Area is used to provide the space for multiple lines of text as a notepad

- import javax.swing.JTextArea;

```
//
JTextArea area=new JTextArea();
area.setBounds(10,10,300,400);
area.setLineWrap(true);
```




This Application is used to perform Arithmetic Operations
First Number
Second Number
Result
ADDITION
SUBTRACT
MULTIPLICATION
DIVISION
CLEAR

```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    // Addition Button
  String a=TF1.getText();
  String b=TF2.getText();

  int num1=Integer.parseInt(a);
  int num2=Integer.parseInt(b);
  int add= num1 + num2;

  LR.setText("Sum is: "+add);
}

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
    // Subtraction Button
    int num1=Integer.parseInt(TF1.getText());
    int num2=Integer.parseInt(TF2.getText());

    int sub = num1 - num2;
    LR.setText("Subtraction is: "+sub);
}
```

```
    private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
        // Mutliplication Button
        int num1=Integer.parseInt(TF1.getText());
        int num2=Integer.parseInt(TF2.getText());

        int mul = num1 * num2;
        LR.setText("Multiplication is: "+mul);

    }

    private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {
        // Division Button
        int num1=Integer.parseInt(TF1.getText());
        int num2=Integer.parseInt(TF2.getText());

        int div = num1 / num2;
        LR.setText("Division is: "+div);

    }

    private void jButton5ActionPerformed(java.awt.event.ActionEvent evt) {
        // CLEAR Button
        TF1.setText("");
        TF2.setText("");
        LR.setText("");
    }
```

پوهنتون کاردان
KARDAN UNIVERSITY

Thank You…!