

lab 9 - 20-02-2024

Write a program that creates a user interface to perform integer divisions.

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

```
class SwingDemo{
```

```
SwingDemo{
```

```
JFrame jfrm = new JFrame("Divider App");  
jfrm.setSize(275,150);  
jfrm.setLayout(new FlowLayout());  
jfrm.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
```

```
JLabel jlab = new JLabel("Enter the divider and  
divident:");
```

```
JTextField jtf = new JTextField(8);
```

```
JTextField hjtf = new JTextField(8);
```

```
JButton button = new JButton("Calculate");
```

```
JLabel err = new JLabel();
```

```
JLabel alab = new JLabel();
```

```
JLabel blab = new JLabel();
```

```
JLabel anslab = new JLabel();
```

```
jfrm.add(err);
```

```
jfrm.add(jlab);
```



```
jtfm.add(ajtf);  
jtfm.add(bjtf);  
jtfm.add(button);  
jfrm.add(alab);  
jfrm.add(blab);  
jfrm.add(anslab);
```

```
ActionListener l = new ActionListener() {  
    public void actionPerformed(ActionEvent evt) {  
        System.out.println("Action event from a  
        text field");  
    }  
};
```

```
ajtf.addActionListener(l);  
bjtf.addActionListener(l);
```

```
button.addActionListener(new ActionListener() {  
    public void actionPerformed(ActionEvent evt) {  
        try {  
            int a = Integer.parseInt(ajtf.getText());  
            int b = Integer.parseInt(bjtf.getText());  
            int ans = a/b;
```

```
            alab.setText("A = " + a);  
            blab.setText("B = " + b);  
            ansLab.setText("Ans = " + ans);  
            err.setText("B should be NON zero!");  
        }  
    }  
});
```

```
    catch (NumberFormatException e) {  
        alab.setText("");  
        blab.setText("");  
        ansLab.setText("");  
        err.setText("B should be NON zero!");  
    }  
}
```



```

catch(ArithmeticException e) {
    alab.setText("");
    blab.setText("");
    andlab.setText("");
    err.setText("B should be Non zero!");
}
}
}
}
3);
jfm.setVisible(true);
}
}

```

```

public static void main(String args[]) {
    SwingUtilities.invokeLater(new Runnable() {
        public void run() {
            new SwingDemo();
        }
    });
    System.out.println("Sanketh M Hanasi IBM22CS242");
}
}

```

Output :

Enter the divider and dividend :

10 2 Calculate

A=10 B=2 Ans=5

Sanketh M. Hanasi IBM22CS242

* setSize(w, h) : Method that resizes the component so that it has width w , height h .

* setLayout() : Method allows you to set the layout of the container of type `APanel`, to say `FlowLayout`, `BorderLayout`, etc.

* addActionListener() : Event handler to implement/ to define what should be done when an user perform certain operation.

* addWindowListener() : Overriding only the methods of interest.

* setDefaultCloseOperation() : sets the operation that will happen by default when the user initiates a "close" on this frame.

* JFrame : `JFrame` in Java is a class that allows you to create and manage a top level window in a Java application. It serves as the main window for GUI-based Java applications and provides a platform independent way to create graphical user interfaces.

* JLabel : `JLabel` is a class of Java Swing. `JLabel` is used to display a short string or even image icon. It can display text, image, or both. It is inactive to input events such as mouse focus or keyboard focus.

* invokeLater() : It can be used to perform a task asynchronously in AWT Event dispatcher thread.

* `setVisible()` : It is a method that has return type boolean. It sets the currently made attributes to be visible on the screen GUI application mainly Java Swing.

* `SwingUtilities` : The `SwingUtilities` class has 2 important static methods, `invokeAndWait()` and `invokeLater()` to add put references to blocks of code into event queue.

Done
20-2-24