CORE JAVA PRACTICAL

Practical 8

a. ActionEvent:

Code:

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

class calculate implements ActionListener {
   JLabel l2;
   JTextField t1;

   public calculate() {
      JFrame f = new JFrame();
      JLabel l1 = new JLabel();
      t1 = new JTextField(10);
      JButton b = new JButton("Factorial");
      l2 = new JLabel();

      // Add components to the frame, not the button
      f.add(l1);
      f.add(t1);
      f.add(b);
      f.add(l2);
```

```
        // Set ActionListener for the button
        b.addActionListener(this);

        FlowLayout fl = new FlowLayout();
        f.setLayout(fl);
        f.setSize(500, 500);
        f.setVisible(true);
    }

    public void actionPerformed(ActionEvent e) {
        int c = 1;
        int n = Integer.parseInt(t1.getText());
        for (int i = n; i > 0; i--) {
            c = c * i;
        }
        l2.setText("Factorial of a number is:" + c);
    }
}

class factorial {
    public static void main(String args[]) {
        calculate c = new calculate();
    }
}
```

Output:

b.i. MouseEvent:

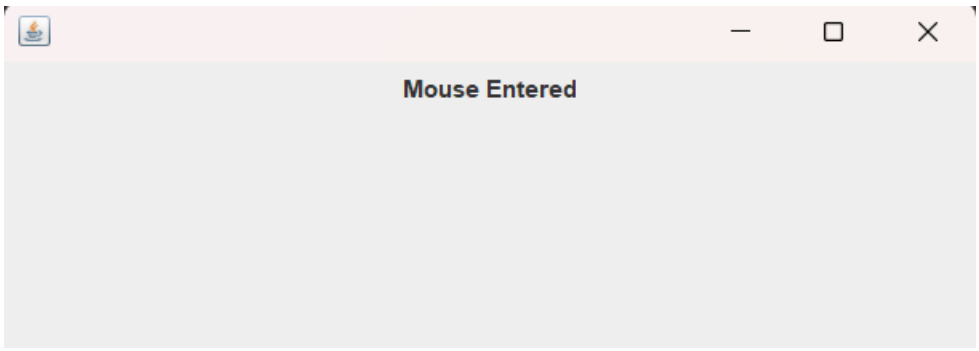
Code:

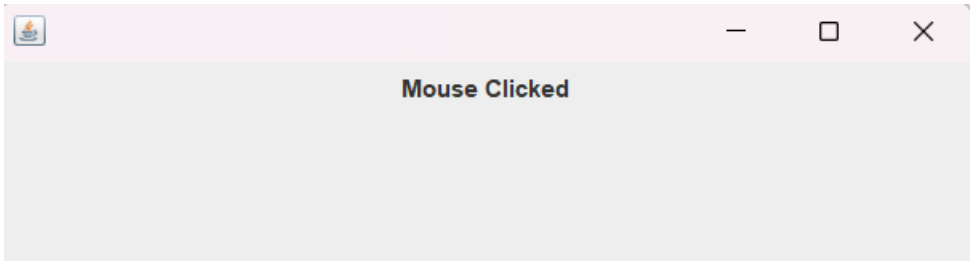
```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
class demomouse implements MouseListener
{
JLabel l;
public demomouse()
{
JFrame f = new JFrame();
l = new JLabel();
f.add(l);
FlowLayout fl = new FlowLayout();
f.setLayout(fl);
f.setSize(500,500);
f.setVisible(true);
f.addMouseListener(this);
}
```

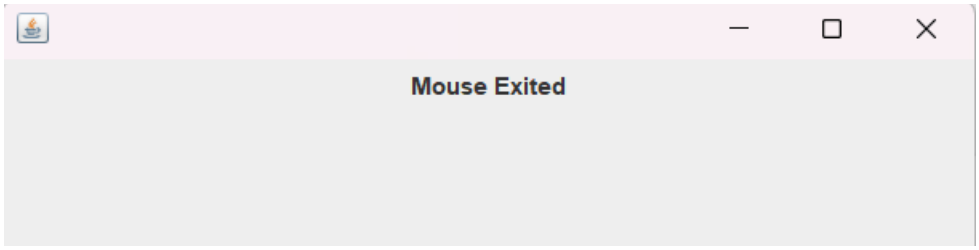
```
public void mousePressed(MouseEvent e)
{
l.setText("Mouse Pressed");
}
public void mouseReleased(MouseEvent e)
{
l.setText("Mouse Released");
}
public void mouseExited(MouseEvent e)
{
l.setText("Mouse Exited");
}
public void mouseEntered(MouseEvent e)
{
l.setText("Mouse Entered");
}
public void mouseClicked(MouseEvent e)
{
l.setText("Mouse Clicked");
}
}
class mousedemo
{
public static void main(String args[])
{
```

```
demomouse dm = new demomouse();
}
}
```

Output:







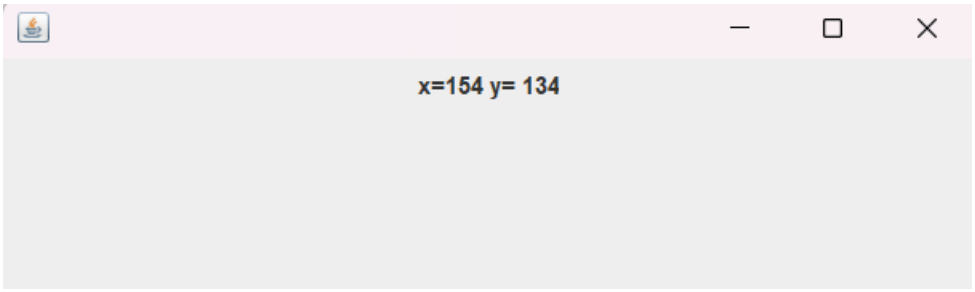
b.ii. MouseMotionListener

code:

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

class MouseMotion implements MouseMotionListener {
   JLabel label;
   JFrame frame;

   public MouseMotion() {
      frame = new JFrame();
      label = new JLabel();
      frame.add(label);
      frame.addMouseMotionListener(this);

      FlowLayout flowLayout = new FlowLayout();
      frame.setLayout(flowLayout);
      frame.setSize(500, 500);
      frame.setVisible(true);
      frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE); //
Added to close the application properly
   }

   public void mouseDragged(MouseEvent e) {
```

```
        Graphics g = frame.getGraphics();
        g.setColor(Color.RED);
        g.fillOval(e.getX(), e.getY(), 5, 5);
    }

    public void mouseMoved(MouseEvent e) {
        label.setText("x=" + e.getX() + " y= " + e.getY());
    }
}

class demomousem {
    public static void main(String args[]) {
        MouseMotion mouseMotion = new MouseMotion();
    }
}
```

Output:



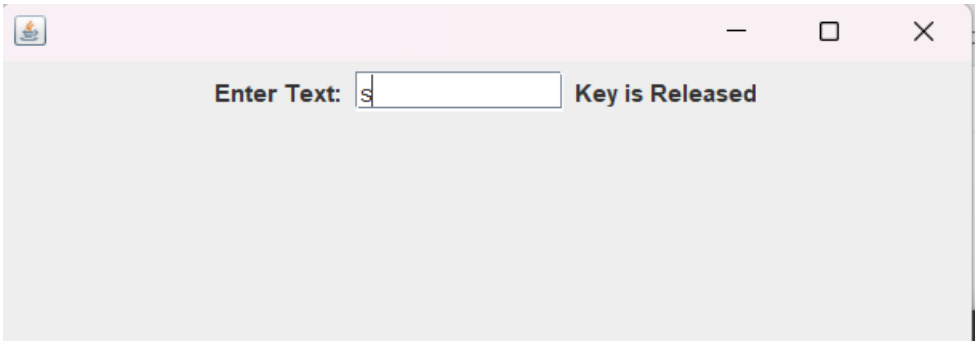
c.. KeyEvent

code:

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

class KeyDemo implements KeyListener {
   JLabel l2;
   JTextField t;

   public KeyDemo() {
      JFrame f = new JFrame();
      JLabel l1 = new JLabel("Enter Text: ");
      t = new JTextField(10);
      l2 = new JLabel();

      // Add KeyListener to the JTextField
      t.addKeyListener(this);

      f.add(l1);
      f.add(t);
      f.add(l2);

      FlowLayout fl = new FlowLayout();
      f.setLayout(fl);
```

```
        f.setSize(500, 500);
        f.setVisible(true);
        f.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE); // Added to close the application properly
    }

    public void keyTyped(KeyEvent e) {
        l2.setText("Key is Typed");
    }

    public void keyPressed(KeyEvent e) {
        l2.setText("Key is Pressed");
    }

    public void keyReleased(KeyEvent e) {
        l2.setText("Key is Released");
    }
}

class demokey {
    public static void main(String args[]) {
        KeyDemo kd = new KeyDemo();
    }
}
```

Output:



PRACTICAL 9.

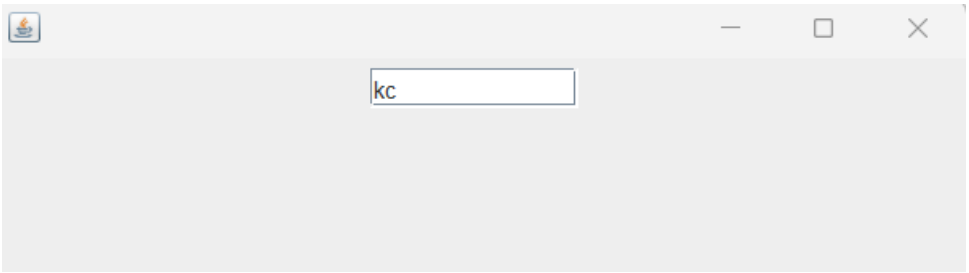
Code:

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

class AdapterDemo extends WindowAdapter implements KeyListener {
    JLabel l;
    JTextField t;

    public AdapterDemo() {
        JFrame f = new JFrame();
        l = new JLabel();
        t = new JTextField(10);  // Initialize the JTextField

        // Add KeyListener to the JTextField
        t.addKeyListener(this);

        f.addWindowListener(this);
```

```
        f.add(t);
        f.add(l);

        FlowLayout fl = new FlowLayout();
        f.setLayout(fl);
        f.setSize(500, 500);
        f.setVisible(true);
    }

    public void windowActivated(WindowEvent e) {
        System.out.println("Activated");
    }

    public void keyPressed(KeyEvent e) {
        System.out.println("Key is Pressed");
    }

    public void keyReleased(KeyEvent e) {
    }

    public void keyTyped(KeyEvent e) {
    }
}

class demoadapter {
```

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

Output:




```
Activated
Key is Pressed
Key is Pressed
Activated
Activated
```

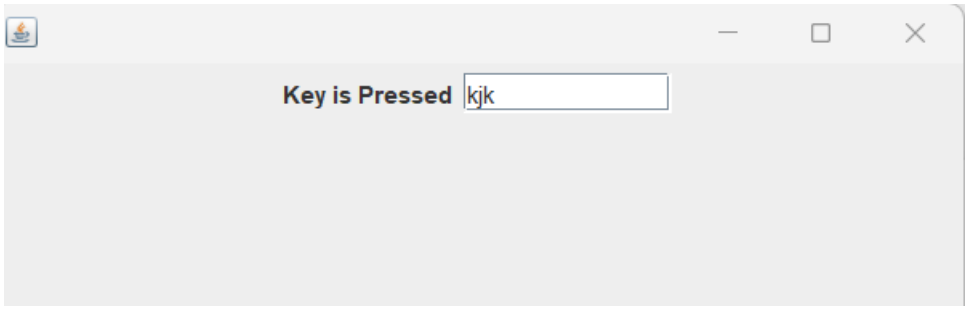
PRACTICAL 10:

CODE:

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

class InnerDemo extends JFrame {
    private JTextField t;
    private JLabel l;

    public InnerDemo() {
        t = new JTextField(10);
        l = new JLabel();

        t.addKeyListener(new DemoKey());
        addMouseListener(new DemoMouse());
        addWindowListener(new DemoWindow());
        add(l);
        add(t);

        FlowLayout fl = new FlowLayout();
        setLayout(fl);
        setSize(500, 500);
        setVisible(true);
    }
```

```
    class DemoKey extends KeyAdapter {
        public void keyPressed(KeyEvent e) {
            l.setText("Key is Pressed");
        }
    }

    class DemoMouse extends MouseAdapter {
        public void mouseClicked(MouseEvent e) {
            l.setText("Mouse is clicked");
        }
    }

    class DemoWindow extends WindowAdapter {
        public void windowOpened(WindowEvent e) {
            System.out.println("Window opened");
        }
    }
}
class demoinner
{
    public static void main(String args[]) {
        InnerDemo id = new InnerDemo();
    }
}
```

Output:




```
C:\corjavapractical>javac demoinner.java

C:\corjavapractical>java demoinner
Window opened
```