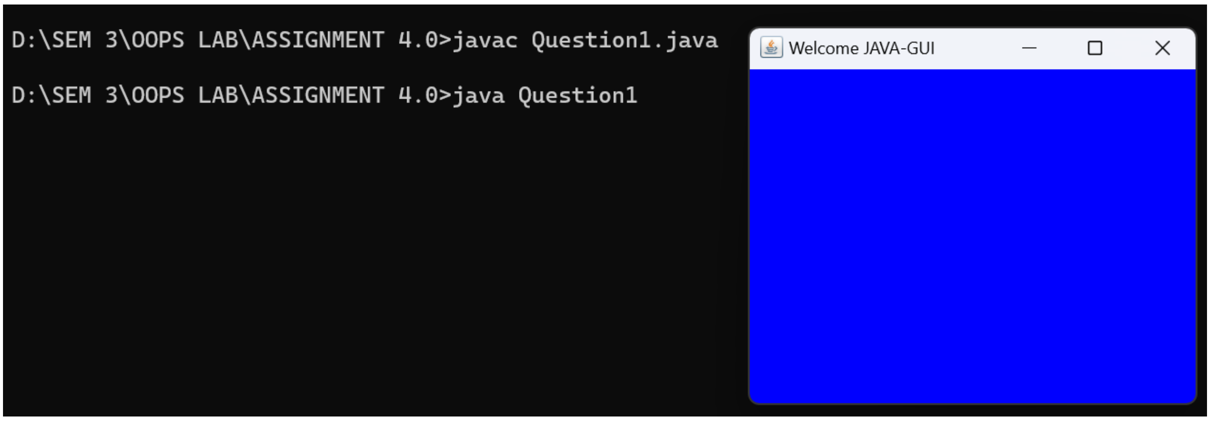
MODULE – 4

1. Write a program to create a Simple JFrame having title “Welcome JAVA-GUI” and background color of frame has to be blue.

CODE:

```
import javax.swing.*;

import java.awt.*;

public class Question1 {

  public static void main(String[] args) {

    JFrame frame = new JFrame();  //creates a frame

    frame.setTitle("Welcome JAVA-GUI");

    frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

    frame.setSize(420,420);

    frame.setVisible(true);

    frame.getContentPane().setBackground(Color.BLUE);

  }

}
```

OUTPUT:



```
D:\SEM 3\OOPS LAB\ASSIGNMENT 4.0>javac Question1.java

D:\SEM 3\OOPS LAB\ASSIGNMENT 4.0>java Question1
```

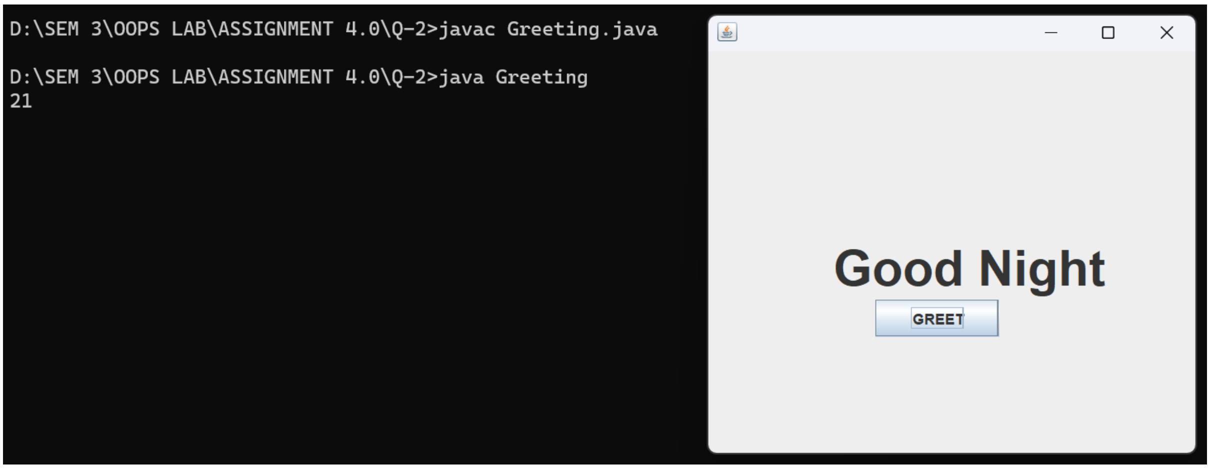
2. Write a java GUI program. Which adds one button “GREET” and one label “Message”to the Frame. By clicking the button will greet user by setting label text as “Good Morning” or “Good AfterNoon” or “Good Evening” or “Good Night” as per current time .

CODE:

```
import java.awt.*;

import javax.swing.*;

import java.awt.event.*;

import java.util.Calendar;


class Myframe extends JFrame implements ActionListener {

   Container c;

   JButton btn;

   Label l;

   Calendar calendar = Calendar.getInstance();


   public Myframe() {


      c = this.getContentPane();

      c.setLayout(null);

      btn = new JButton("GREET");

      l = new Label();

      btn.setBounds(135, 200, 100, 30);

      l.setBounds(100, 150, 300, 50);

      l.setText("Message");

      l.setFont(new Font("MV Boli",Font.BOLD,40));

      btn.addActionListener(this);

      c.add(btn);

      c.add(l);

   }


   public void actionPerformed(ActionEvent e) {
      int timeComp = calendar.get(Calendar.HOUR_OF_DAY);
      System.out.println(timeComp);
      if(timeComp>=4 && timeComp<12)
      {
        l.setText("Good morning");
      }
      if(timeComp>=12 && timeComp<17)
      {
        l.setText("Good Afternoon");
      }
      if(timeComp>=17 && timeComp<20)
      {
        l.setText("Good Evening");
      }
      if((timeComp>=20 && timeComp<=24) || (timeComp>=0 && timeComp<4))
      {
        l.setText("Good Night");
      }
   }
}

class Greeting {
   public static void main(String[] args) {
      Myframe myFrame = new Myframe();
      myFrame.setSize(400, 400);
      myFrame.setLocation(100, 100);
      myFrame.setVisible(true);
      myFrame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

   }
```

```
}
```

OUTPUT:



```
D:\SEM 3\OOPS LAB\ASSIGNMENT 4.0\Q-2>javac Greeting.java

D:\SEM 3\OOPS LAB\ASSIGNMENT 4.0\Q-2>java Greeting
21
```

3. Write a program to create a Menu based application.Main Menu “Formats”.Menuitems – date(dd-mm-yyyy),DayOfMonth, WeekNumber, Hours,DayOfYear.Once clicked on particular item in appropriate format date has to be displayed on Label. (will cover course handout program 13 of demonstrating System clock).

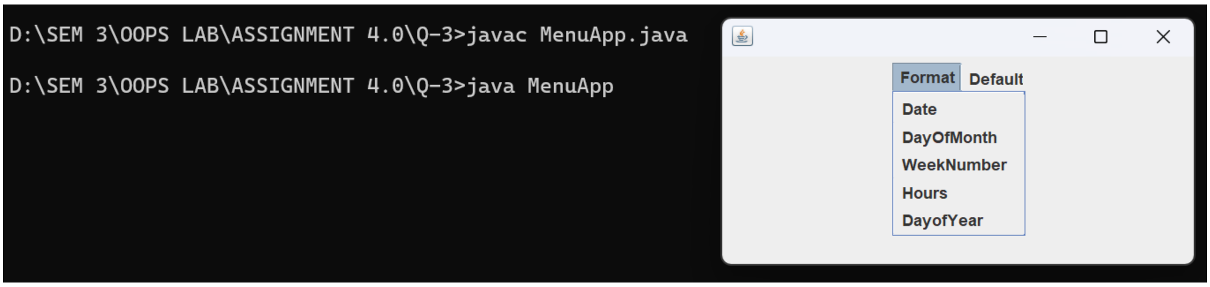
CODE:

```
import javax.swing.*;

import java.awt.*;

import java.awt.event.*;

import java.util.Calendar;

class MenuApp implements ActionListener

{

   JFrame mFrame;

   JLabel label;

   JMenuBar menuBar;

   JMenu menu;

   JMenuItem menuItem1;

   JMenuItem menuItem2;

   JMenuItem menuItem3;

   JMenuItem menuItem4;

   JMenuItem menuItem5;

   Calendar calendar = Calendar.getInstance();
```

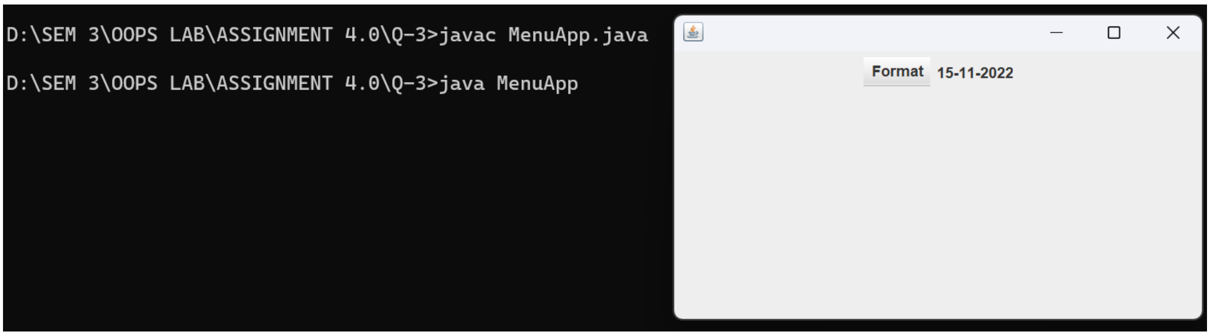
```
public void initGUI()
{
   mFrame = new JFrame();
   label = new JLabel("Default");
   menuBar = new JMenuBar();
   menu = new JMenu("Format");
   menuItem1 = new JMenuItem("Date");
   menuItem2 = new JMenuItem("DayOfMonth");
   menuItem3 = new JMenuItem("WeekNumber");
   menuItem4 = new JMenuItem("Hours");
   menuItem5 = new JMenuItem("DayofYear");
   menu.add(menuItem1);
   menu.add(menuItem2);
   menu.add(menuItem3);
   menu.add(menuItem4);
   menu.add(menuItem5);
   menuBar.add(menu);
   mFrame.add(menuBar);
   mFrame.add(label);
   mFrame.setLayout(new FlowLayout());
   mFrame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
   mFrame.setSize(500,500);
   mFrame.setVisible(true);
   menuItem1.addActionListener(this);
   menuItem2.addActionListener(this);
   menuItem3.addActionListener(this);
   menuItem4.addActionListener(this);
   menuItem5.addActionListener(this);
}
public void actionPerformed(ActionEvent ae)
{
   if(ae.getSource() == menuItem1)
   {
     int month = calendar.get(Calendar.MONTH)+1;
     String data = calendar.get(Calendar.DATE)+"-"+month+"-"+calendar.get(Calendar.YEAR);
     label.setText(data);


   }
   if(ae.getSource() == menuItem2)
   {
     String data=  calendar.get(Calendar.DAY_OF_MONTH)+"";
     label.setText(data);
   }
   if(ae.getSource() == menuItem3)
   {
     String data = calendar.get(Calendar.WEEK_OF_YEAR)+"";
     label.setText(data);
   }
   if(ae.getSource() == menuItem4)
   {
     String data = calendar.get(Calendar.HOUR_OF_DAY)+"";
     label.setText(data);
   }
   if(ae.getSource() == menuItem5)
   {
     String data = calendar.get(Calendar.DAY_OF_YEAR)+"";
     label.setText(data);
   }
}
```

```
  public MenuApp()
  {
    initGUI();
  }
  public static void main(String[] args) {
    MenuApp ap  = new MenuApp();
  }
}
```

OUTPUT:



```
D:\SEM 3\OOPS LAB\ASSIGNMENT 4.0\Q-3>javac MenuApp.java

D:\SEM 3\OOPS LAB\ASSIGNMENT 4.0\Q-3>java MenuApp
```



```
D:\SEM 3\OOPS LAB\ASSIGNMENT 4.0\Q-3>javac MenuApp.java

D:\SEM 3\OOPS LAB\ASSIGNMENT 4.0\Q-3>java MenuApp
```

4. Write a program to create a GUI frame that displays the student information.

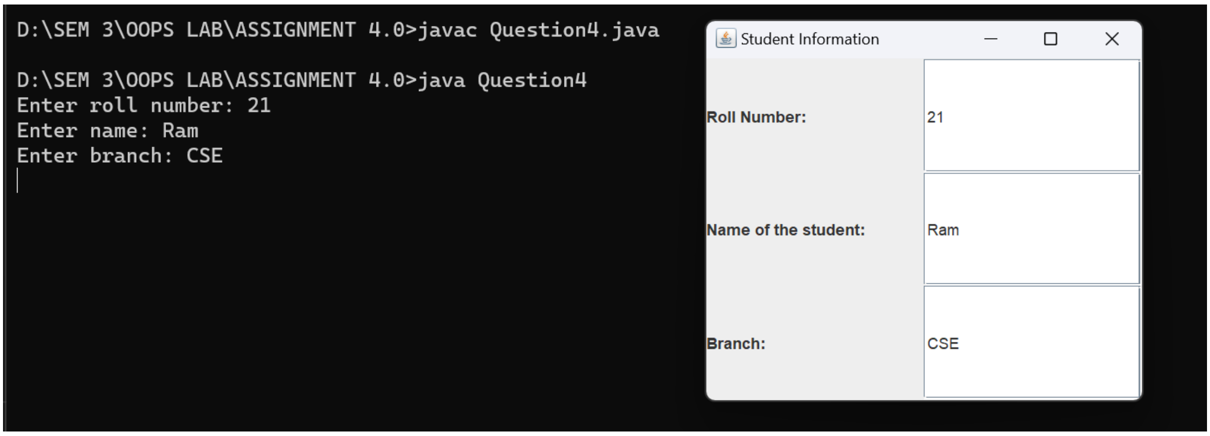
CODE:

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


class Student {
  JLabel L1, L2, L3;
  JTextField tf1, tf2, tf3;
```

```
public Student() { initGui(); }

public void initGui() {
    Scanner sc = new Scanner(System.in);

    JFrame frame = new JFrame("Student Information");

    this.L1 = new JLabel("Roll Number: ");
    this.L2 = new JLabel("Name of the student: ");
    this.L3 = new JLabel("Branch: ");

    this.tf1 = new JTextField(20);
    this.tf2 = new JTextField(20);
    this.tf3 = new JTextField(20);

    System.out.print("Enter roll number: ");
    String rollNumber = sc.nextLine();
    System.out.print("Enter name: ");
    String name = sc.nextLine();
    System.out.print("Enter branch: ");
    String branch = sc.nextLine();

    this.tf1.setText(rollNumber);
    this.tf2.setText(name);
    this.tf3.setText(branch);

    Container container = frame.getContentPane();
    container.setLayout(new GridLayout(3, 2));
     // container.setLayout(null);

    container.add(this.L1);
```

```
        container.add(this.tf1);

        container.add(this.L2);

        container.add(this.tf2);

        container.add(this.L3);

        container.add(this.tf3);


        frame.setSize(350, 300);

        frame.setVisible(true);

        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

    }

}


public class Question4 {

    public static void main(String[] args) {

        new Student();

    }

}
```

OUTPUT:



```
D:\SEM 3\OOPS LAB\ASSIGNMENT 4.0>javac Question4.java

D:\SEM 3\OOPS LAB\ASSIGNMENT 4.0>java Question4
Enter roll number: 21
Enter name: Ram
Enter branch: CSE
```

Student Information

Roll Number: 21

Name of the student: Ram

Branch: CSE

5. Write a calculator application having display as shown in below image. (Use BorderLayout for MainFrame, GridLayout for upper panel and flow layout for Button Panel.)

CODE:

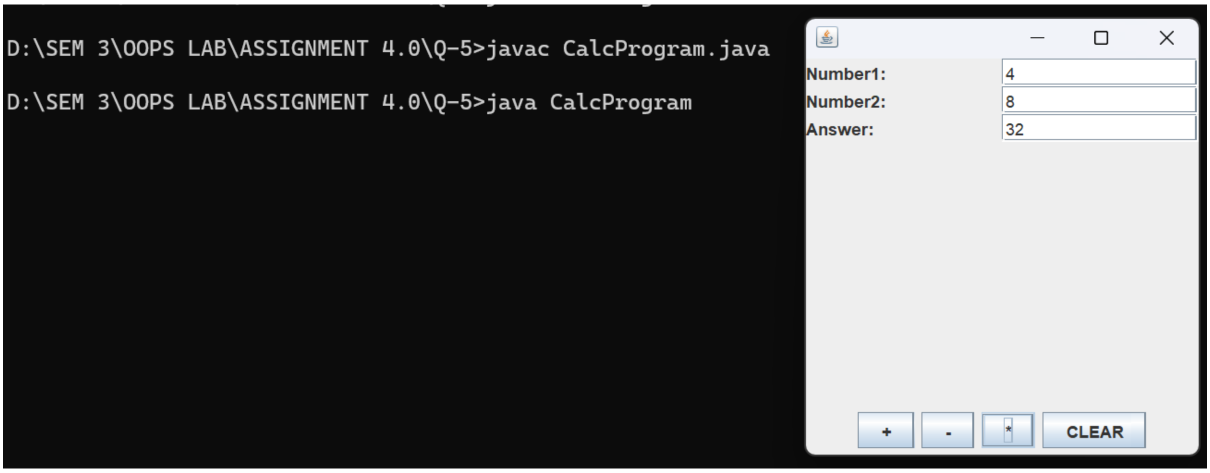
```
import javax.swing.*;

import java.awt.*;
import java.awt.event.*;

class CalcProgram implements ActionListener
{
   JFrame myFrame;
   JButton add;
   JButton sub;
   JButton mul;
   JTextField tf1;
   JTextField tf2;
   JTextField tf3;
   JLabel L1;
   JLabel L2;
   JLabel L3;
   JPanel p1;
   JPanel p2;
   JPanel panel;
   JButton clear;
   public void initGUI()
   {
      myFrame = new JFrame();
      p1 = new JPanel();
      p2 = new JPanel();
      panel = new JPanel();
      sub = new JButton("-");
      Container c = myFrame.getContentPane();
      add = new JButton("+");
      mul = new JButton("*");
      clear = new JButton("CLEAR");
      tf1 = new JTextField(15);
      tf2 = new JTextField(15);
```

```
tf3 = new JTextField(15);

L1 = new JLabel("Number1:");
L2 = new JLabel("Number2:");
L3 = new JLabel("Answer:");

p1.add(L1);
p1.add(tf1);
p1.add(L2);
p1.setLayout(new GridLayout(3,2));
p1.add(tf2);
p1.add(L3);
p1.add(tf3);

p2.add(add);
p2.add(sub);
p2.add(mul);
p2.add(clear);

p2.setLayout(new FlowLayout());
c.add(panel);
panel.setLayout(new BorderLayout());
panel.add(p1,BorderLayout.NORTH);
panel.add(p2,BorderLayout.SOUTH);

myFrame.setSize(500,500);
myFrame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
myFrame.setVisible(true);
add.addActionListener(this);
mul.addActionListener(this);
sub.addActionListener(this);
```

```
    clear.addActionListener(this);



}
public void actionPerformed(ActionEvent ae)
{
    String oper,result;
    int num1,num2,res;
    if(ae.getSource()==add)
    {
        oper = tf1.getText();
        num1 = Integer.parseInt(oper);

        oper = tf2.getText();
        num2 = Integer.parseInt(oper);

        res = num1 + num2;
        result = res+"";
        tf3.setText(result);



    }
    if(ae.getSource() == mul)
    {
        oper = tf1.getText();
        num1 = Integer.parseInt(oper);

        oper = tf2.getText();
        num2 = Integer.parseInt(oper);

        res = num1 * num2;
        result = res+"";
        tf3.setText(result);


    }
    if(ae.getSource() == sub)
    {
        oper = tf1.getText();
        num1 = Integer.parseInt(oper);


        oper = tf2.getText();
        num2 = Integer.parseInt(oper);


        res = num1 - num2;
        result = res+"";
        tf3.setText(result);
    }
    if(ae.getSource() == clear)
    {
        tf1.setText(" ");
        tf2.setText(" ");
        tf3.setText(" ");
    }


}
public CalcProgram()
{
    initGUI();
}
public static void main(String[] args) {
    CalcProgram ap = new CalcProgram();
}
```

}

OUTPUT:



```
D:\SEM 3\OOPS LAB\ASSIGNMENT 4.0\Q-5>javac CalcProgram.java

D:\SEM 3\OOPS LAB\ASSIGNMENT 4.0\Q-5>java CalcProgram
```

Number1: 4
Number2: 8
Answer: 32

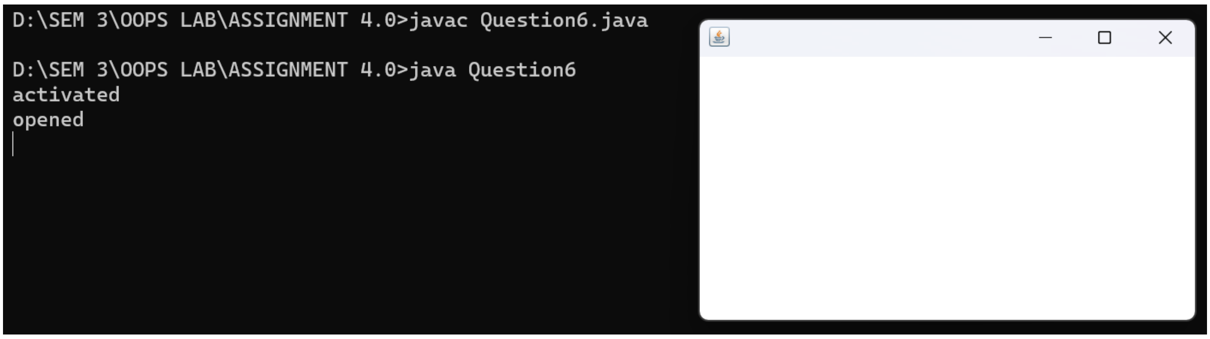
6. Write a program to demonstrate different window event handling.

CODE:

```
import javax.swing.*;

import java.awt.*;

import java.awt.event.*;


public class Question6 extends Frame implements WindowListener {

  Question6() {

    addWindowListener(this);

    setSize (400, 400);

    setLayout (null);

    setVisible (true);

  }


  public static void main(String[] args) {

    new Question6();

  }
```

```
    public void windowActivated (WindowEvent arg0) {
        System.out.println("activated");
    }

    public void windowClosed (WindowEvent arg0) {
        System.out.println("closed");
    }

    public void windowClosing (WindowEvent arg0) {
        System.out.println("closing");
        dispose();
    }

    public void windowDeactivated (WindowEvent arg0) {
        System.out.println("deactivated");
    }

    public void windowDeiconified (WindowEvent arg0) {
        System.out.println("deiconified");
    }

    public void windowIconified(WindowEvent arg0) {
        System.out.println("iconified");
    }

    public void windowOpened(WindowEvent arg0) {
        System.out.println("opened");
    }
}
```

OUTPUT:



```
D:\SEM 3\OOPS LAB\ASSIGNMENT 4.0>javac Question6.java

D:\SEM 3\OOPS LAB\ASSIGNMENT 4.0>java Question6
activated
opened
```

activated

opened

deactivated

activated

deactivated

activated

deactivated

activated

closing

deactivated

closed

7. Write a program to demonstrate mouse events handling.

CODE:

```
import java.awt.*;

import java.awt.event.*;

import javax.swing.*;


public class Question7 extends JFrame implements MouseListener{

   Label l;

   Question7(){

      addMouseListener(this);

      setDefaultCloseOperation(EXIT_ON_CLOSE);

      l=new Label();
```

```
    l.setBounds(20,50,100,20);
    add(l);
    setSize(300,300);
    setLayout(null);
    setVisible(true);
  }
public static void main(String[] args) {
  new Question7();
}
  public void mouseClicked(MouseEvent e) {
    l.setText("Mouse Clicked");
       System.out.println("Mouse clicked");
  }
  public void mouseEntered(MouseEvent e) {
    l.setText("Mouse Entered");
        System.out.println("Mouse entered");
  }
  public void mouseExited(MouseEvent e) {
    l.setText("Mouse Exited");
        System.out.println("Mouse exited");
  }
  public void mousePressed(MouseEvent e) {
    l.setText("Mouse Pressed");
        System.out.println("Mouse pressed");
  }
  public void mouseReleased(MouseEvent e) {
    l.setText("Mouse Released");
        System.out.println("Mouse released");
  }

}
```

OUTPUT:

Mouse entered

Mouse exited

Mouse entered

Mouse exited

Mouse entered

Mouse pressed

Mouse released

Mouse clicked

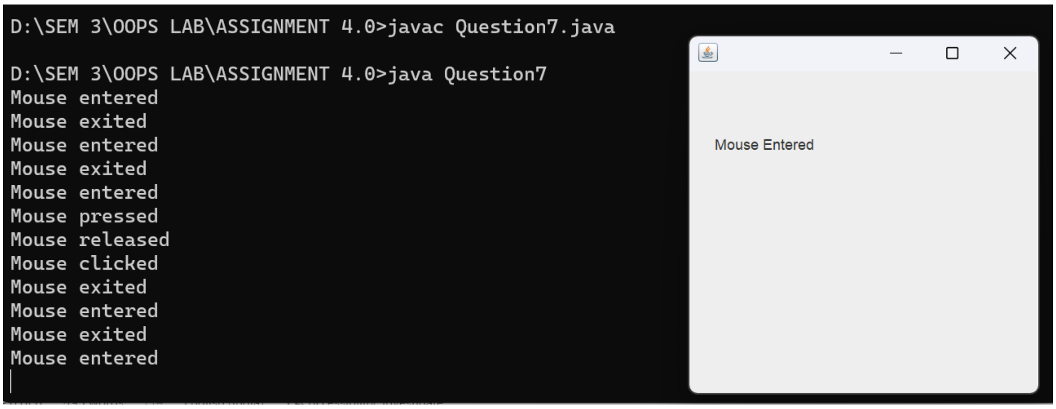
Mouse exited

Mouse entered

Mouse exited

Mouse entered

Mouse exited



```
D:\SEM 3\OOPS LAB\ASSIGNMENT 4.0>javac Question7.java

D:\SEM 3\OOPS LAB\ASSIGNMENT 4.0>java Question7
Mouse entered
Mouse exited
Mouse entered
Mouse exited
Mouse entered
Mouse pressed
Mouse released
Mouse clicked
Mouse exited
Mouse entered
Mouse exited
Mouse entered
```

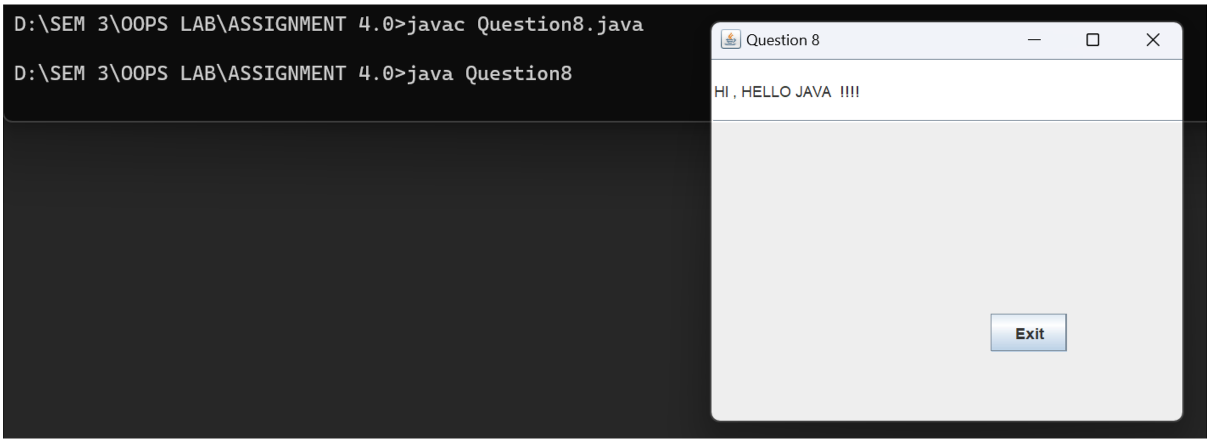
Mouse Entered

8. Write a program to demonstrate keyboard event handling.

CODE:

```
import javax.swing.*;

import java.awt.*;

import java.awt.event.*;


class Question8 implements KeyListener, ActionListener {

  static JFrame frame;

  static JTextField input, output;


  public static void main(String[] args) {

    frame = new JFrame("Question 8");

    frame.setSize(500, 500);

    frame.setLayout(null);

    output = new JTextField();

    output.setBounds(0, 0, 500, 50);
    frame.add(output);
    input = new JTextField();
    input.setBounds(0, 400, 500, 50);
    frame.add(input);
    JButton exit = new JButton("Exit");
    exit.setBounds(220, 200, 60, 30);
    frame.add(exit);
    Question8 obj = new Question8();
    input.addKeyListener(obj);
    exit.addActionListener(obj);
    frame.setVisible(true);
}

public void actionPerformed(ActionEvent ae) {
    frame.dispose();
}

public void keyReleased(KeyEvent e) {
    output.setText("");
    output.setText("Key Released : "+e.getKeyCode());
    if(Character.isLetter(e.getKeyChar()))
        keyTyped(e);
    if(Character.isDigit(e.getKeyChar()))
        keyTyped(e);
}

public void keyPressed(KeyEvent e) {
    output.setText("");
    output.setText("Key Pressed : "+e.getKeyCode());
    if(Character.isLetter(e.getKeyChar()))
        keyTyped(e);
        if(Character.isDigit(e.getKeyChar()))
            keyTyped(e);
    }

    public void keyTyped(KeyEvent e) {
        output.setText("");
        output.setText("Key Typed : "+e.getKeyChar());
    }
}
```

OUTPUT:



```
D:\SEM 3\OOPS LAB\ASSIGNMENT 4.0>javac Question8.java

D:\SEM 3\OOPS LAB\ASSIGNMENT 4.0>java Question8
```

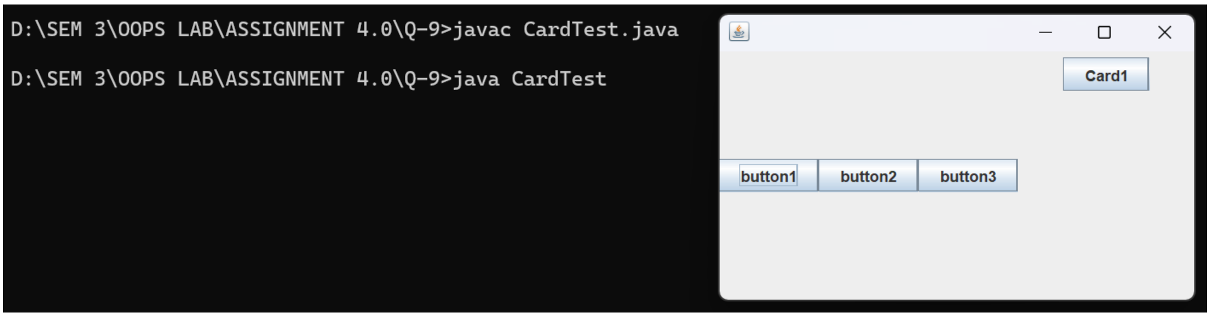
9. Write a program to demonstrate CardLayout (create 3 cards and first card uses BoxLayOut to arrange 3 buttons.)

CODE:

```
import javax.swing.*;

import java.awt.*;

import java.awt.event.*;


class CardTest implements ActionListener
{
    JFrame myFrame;
    JLabel label;
    JButton button11;
JButton button12;
JButton button13;
JButton button1;
JButton button2;
JButton button3;

CardLayout crd;
JPanel panel;
public void initGUI()
{
   myFrame = new JFrame();
   panel = new JPanel();
   button11 = new JButton("button1");
   button12 = new JButton("button2");
   button13 = new JButton("button3");

   button1 = new JButton("Card1");
   button2 = new JButton("Card2");
   button3 = new JButton("Card3");
   JPanel panel1 = new JPanel();
   JPanel panel2 = new JPanel();
   JPanel panel3 = new JPanel();
   JPanel panel4 = new JPanel();
   BoxLayout BX = new BoxLayout(panel1, BoxLayout.LINE_AXIS);
   button1.setAlignmentX(Component.CENTER_ALIGNMENT);
   button1.setAlignmentY(Component.CENTER_ALIGNMENT);

   panel1.add(button11);
   panel1.add(button12);
   panel1.add(button13);
   panel1.setLayout(BX);
   panel2.add(button1);
   panel2.setLayout(new FlowLayout());
   panel1.add(panel2);
   crd = new CardLayout();
   panel.setLayout(crd);
   panel.add("a",panel1);
   panel.add("b",button2);
   panel.add("c",button3);

   button1.addActionListener(this);
   button2.addActionListener(this);
   button3.addActionListener(this);



   myFrame.add(panel);
   myFrame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
   myFrame.setVisible(true);
   myFrame.setSize(500,500);



}
public void actionPerformed(ActionEvent ae)
{
   crd.next(panel);
}
public CardTest()
{
   initGUI();
}
public static void main(String[] args) {
        CardTest ap = new CardTest();
    }
}
```

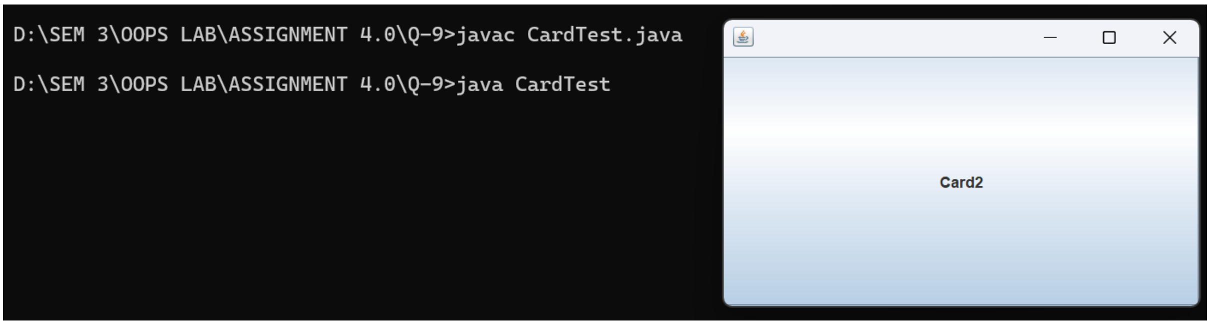
OUTPUT:

Card1 uses box layout having 3 buttons -- button1, button2, button3



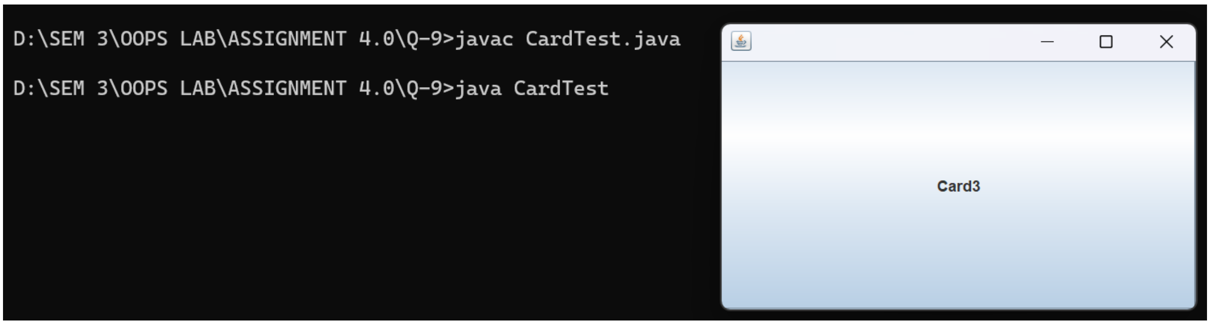
```
D:\SEM 3\OOPS LAB\ASSIGNMENT 4.0\Q-9>javac CardTest.java

D:\SEM 3\OOPS LAB\ASSIGNMENT 4.0\Q-9>java CardTest
```

On clicking Card1



```
D:\SEM 3\OOPS LAB\ASSIGNMENT 4.0\Q-9>javac CardTest.java

D:\SEM 3\OOPS LAB\ASSIGNMENT 4.0\Q-9>java CardTest
```

On clicking Card 2



```
D:\SEM 3\OOPS LAB\ASSIGNMENT 4.0\Q-9>javac CardTest.java

D:\SEM 3\OOPS LAB\ASSIGNMENT 4.0\Q-9>java CardTest
```

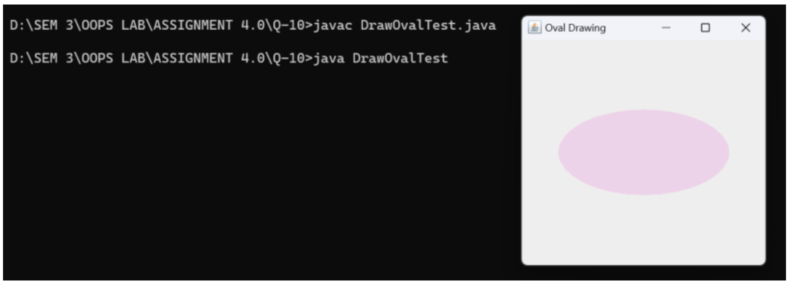
10. Draw an Oval filled with Random Color in middle of the frame and change its color (any random color) after every one second.

CODE:

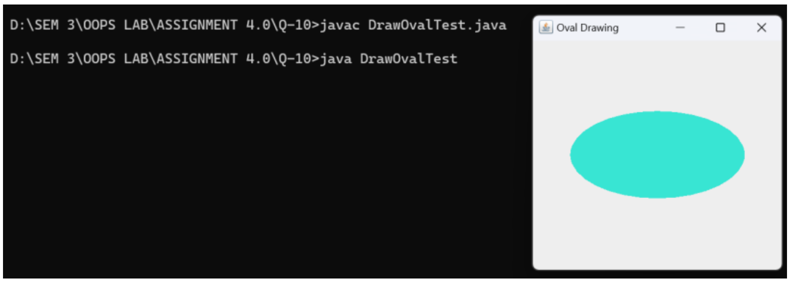
```
import javax.swing.*;
import java.awt.*;
import java.util.*;

class DrawOvalTest extends JFrame
{
   Random rnd = new Random();

   public DrawOvalTest()
   {
      setTitle("Oval Drawing");
      setSize(300,300);
      setContentPane(new PaintPanel());
      setVisible(true);
      setDefaultCloseOperation(EXIT_ON_CLOSE);
   }

   class PaintPanel extends JPanel
   {
      public void paintComponent(Graphics g)
      {
         g.setColor(new Color(rnd.nextInt(255),rnd.nextInt(255),rnd.nextInt(255)));
         g.fillOval((getWidth()-200)/2,(getHeight()-100)/2, 200, 100);
      }
   }


   public static void main(String args[]) throws Exception
```

```
    {
        DrawOvalTest s = new DrawOvalTest();

        Runnable r = new Runnable() {
            public void run()
            {
                s.repaint();
            }
        };

        while(true)
        {
            Thread.sleep(1000);
            SwingUtilities.invokeLater(r);
        }
    }
}
```

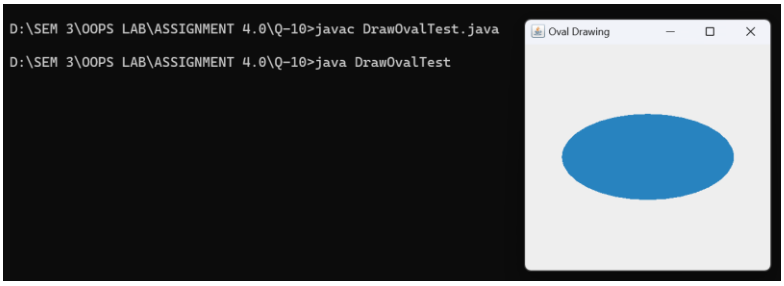
OUTPUT:



```
D:\SEM 3\OOPS LAB\ASSIGNMENT 4.0\Q-10>javac DrawOvalTest.java

D:\SEM 3\OOPS LAB\ASSIGNMENT 4.0\Q-10>java DrawOvalTest
```



```
D:\SEM 3\OOPS LAB\ASSIGNMENT 4.0\Q-10>javac DrawOvalTest.java

D:\SEM 3\OOPS LAB\ASSIGNMENT 4.0\Q-10>java DrawOvalTest
```



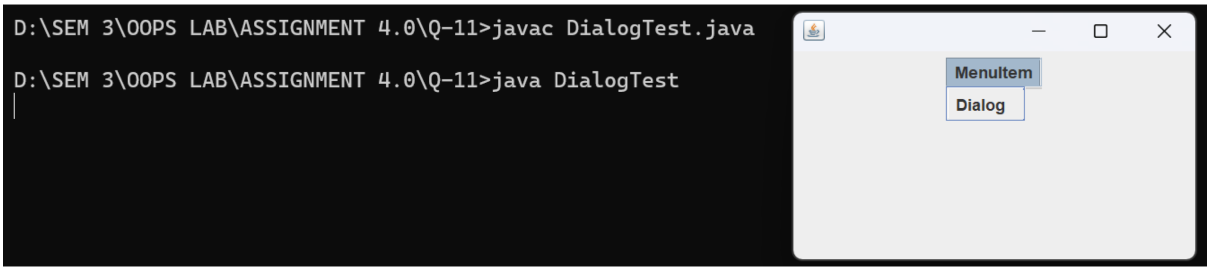
```
D:\SEM 3\OOPS LAB\ASSIGNMENT 4.0\Q-10>javac DrawOvalTest.java

D:\SEM 3\OOPS LAB\ASSIGNMENT 4.0\Q-10>java DrawOvalTest
```

11. Write a program to display inputDialog Box on clicking MenuItem – dialog.Than display confirmation dialog box to “asking do you want to exit from application?”.If yes Exit operation.If no than messageDialog displaying “Happy to see u here”.

CODE:

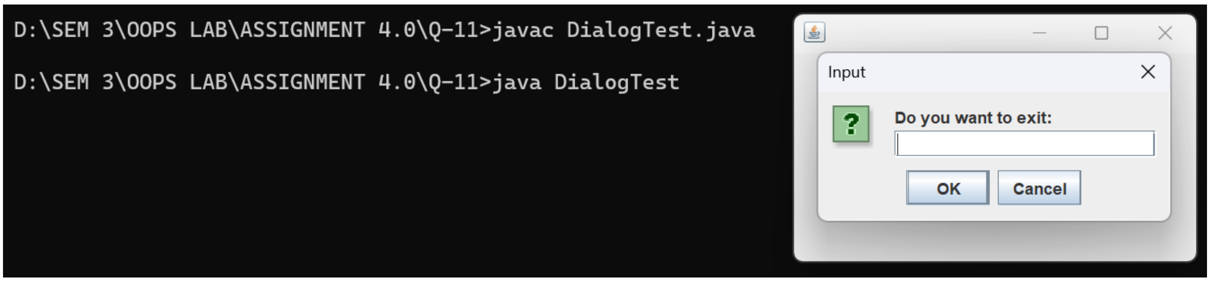
```
import javax.swing.*;

import java.awt.*;

import java.awt.event.*;


class DialogTest implements ActionListener {

   JFrame myFrame;

   JMenuBar menubar;

   JMenu menu;

   JMenuItem menuitem;


   public void initGUI() {

      myFrame = new JFrame();

      menubar = new JMenuBar();

      menuitem = new JMenuItem("Dialog");

      menu = new JMenu("MenuItem");

      menu.add(menuitem);

      menubar.add(menu);

      myFrame.add(menubar);

      myFrame.setSize(500, 500);

      myFrame.setLayout(new FlowLayout());

      myFrame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

      myFrame.setVisible(true);


      menuitem.addActionListener(this);


   }


   public void actionPerformed(ActionEvent ae) {
        String choice;
        choice = JOptionPane.showInputDialog(myFrame, "Do you want to exit: ");
        if (choice.equals("yes")) {
            System.exit(0);
        } else {
            JOptionPane.showMessageDialog(myFrame, "Happy to see u here");
        }

    }

    public DialogTest() {
        initGUI();
    }

    public static void main(String[] args) {
        DialogTest ap = new DialogTest();
    }

}
```

OUTPUT:



```
D:\SEM 3\OOPS LAB\ASSIGNMENT 4.0\Q-11>javac DialogTest.java

D:\SEM 3\OOPS LAB\ASSIGNMENT 4.0\Q-11>java DialogTest
```

On clicking “dialog” :-

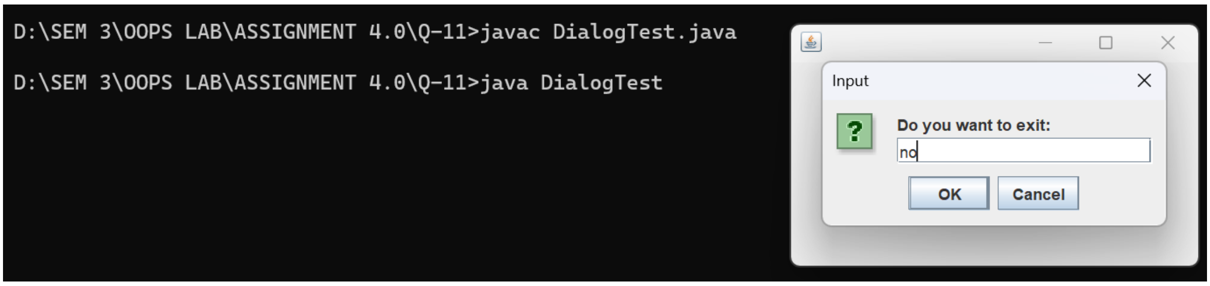


```
D:\SEM 3\OOPS LAB\ASSIGNMENT 4.0\Q-11>javac DialogTest.java

D:\SEM 3\OOPS LAB\ASSIGNMENT 4.0\Q-11>java DialogTest
```

Input

Do you want to exit:

On entering ‘’no’’:-

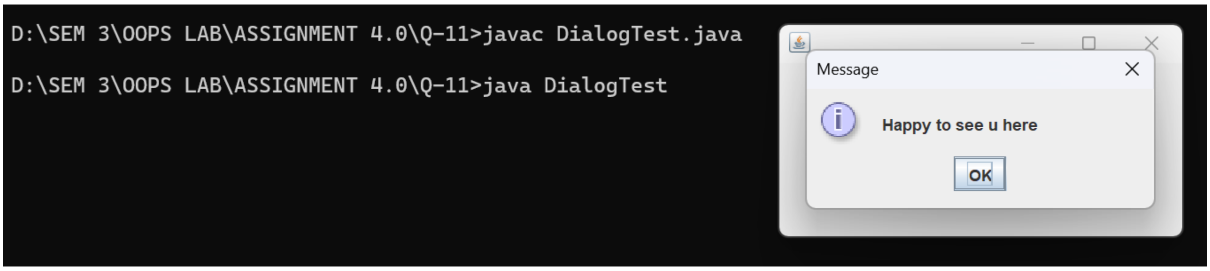


```
D:\SEM 3\OOPS LAB\ASSIGNMENT 4.0\Q-11>javac DialogTest.java

D:\SEM 3\OOPS LAB\ASSIGNMENT 4.0\Q-11>java DialogTest
```

Input

Do you want to exit:

no



```
D:\SEM 3\OOPS LAB\ASSIGNMENT 4.0\Q-11>javac DialogTest.java

D:\SEM 3\OOPS LAB\ASSIGNMENT 4.0\Q-11>java DialogTest
```

Message

Happy to see u here

If we enter ‘’yes’’ in dialog box we exit the application