

//Prime Number Generation

PROGRAM:

```
import java.io.*;
import java.util.Scanner;

public class PrimeNum
{
    public static void main(String[ ] args)
    {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter an integer: ");
        int n = scanner.nextInt();
        System.out.println("Prime numbers up to " + n + ":");
        for (int i = 2; i <= n; i++)
        {
            if (isPrime(i))
            {
                System.out.println(i + " ");
            }
        }
    }
    public static boolean isPrime(int num)
    {
        if (num <= 1)
        {
            return false;
        }
        for (int i = 2; i * i <= num; i++)
        {
            if (num % i == 0)
            {
                return false;
            }
        }
        return true;
    }
}
```

OUTPUT:

```
D:\II BCA>set path=C:\jdk1.5.0\bin
```

```
D:\II BCA >javac PrimeNum.java
```

```
D:\II BCA >java PrimeNum
```

```
Enter an integer: 20
```

```
Prime numbers up to 20:
```

```
2 3 5 7 11 13 17 19
```

//Matrix Multiplication.

PROGRAM:

```
import java.io.*;
import java.util.Scanner;

public class MatrixMultiplication
{
    public static void main(String[ ] args)
    {
        int i,j,k;
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter the number of rows and columns in the first matrix: ");
        int r1 = scanner.nextInt();
        int c1 = scanner.nextInt();
        System.out.print("Enter the number of rows and columns in the second matrix: ");
        int r2 = scanner.nextInt();
        int c2 = scanner.nextInt();
        int a[ ][ ] = new int[r1][c1];
        int b[ ][ ] = new int[r2][c2];
        int c[ ][ ] = new int[r1][c1];
        if(c1 == r2)
        {
            System.out.println("\n Enter first matrix:\n");
            for(i=0 ; i<r1 ; ++i)
                for(j=0 ; j<c1 ; ++j)
                    a[i][j] = scanner.nextInt();
            System.out.println("\n Enter second matrix:\n");
            for(i=0 ; i<r2 ; ++i)
                for(j=0 ; j<c2 ; ++j)
                    b[i][j] = scanner.nextInt();
            System.out.println("\n The Multiplication result matrix is:\n");
            for(i=0 ; i<r1 ; ++i)
            {
                for(j=0 ; j<c2 ; ++j)
                {
                    c[i][j]=0;
                    for(k=0 ; k<c1 ; ++k)
                        c[i][j]=c[i][j]+(a[i][k]*b[k][j]);
                    System.out.print("\t"+c[i][j]);
                }
            }
            System.out.println("\n");
        }
    }
}
```

```
else
    System.out.println("\nSorry!!!! Matrix multiplication can't be done");
}
}
```

OUTPUT:

```
D:\II BCA >set path=C:\jdk1.5.0\bin
D:\II BCA >javac MatrixMultiplication.java
D:\II BCA >java MatrixMultiplication
Enter the number of rows and columns in the first matrix:
2
2
Enter the number of rows and columns in the second matrix:
2
2
Enter the elements of the first matrix:
1
3
5
7
Enter the elements of the second matrix:
2
4
6
8
Result of matrix multiplication:
20 28
52 76
```

//Displays the number of characters, lines and words in a text

PROGRAM:

```
import java.io.*;
import java.util.Scanner;

public class TextStat
{
    public static void main(String[ ] args)
    {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter a text: ");
        String text = scanner.useDelimiter("Press ctrl+Z").next();
        int cCount = text.length();
        int lCount = countLines(text);
        int wCount = countWords(text);
        System.out.println("Text statistics:");
        System.out.println("Characters: " + cCount);
        System.out.println("Lines: " + lCount);
        System.out.println("Words: " + wCount);
    }
    public static int countLines(String text)
    {
        String[ ] lines = text.split("\\r?\\n");
        return lines.length;
    }

    public static int countWords(String text)
    {
        String[ ] words = text.split("\\s+");
        return words.length;
    }
}
```

OUTPUT:

```
D:\II BCA >set path=C:\jdk1.5.0\bin
```

```
D:\II BCA >javac TextStat.java
```

```
D:\II BCA >java TextStat
```

```
Enter a text:
```

```
All is Well
```

```
The Lion is always Lion
```

```
^Z
```

```
Text statistics:
```

```
Characters: 37
```

```
Lines: 2
```

```
Words: 8
```

//Random Numbers Generation

PROGRAM:

```
import java.io.*;
import java.util.Random;
import java.util.Scanner;

public class RandomNumGen
{
    public static void main(String[ ] args)
    {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter the lower limit: ");
        int LL = scanner.nextInt();
        System.out.print("Enter the upper limit: ");
        int UL = scanner.nextInt();
        Random random = new Random();
        int RN = random.nextInt(UL - LL + 1) + LL;
        System.out.println("Random number generated: " + RN);
        if (RN < (LL + (UL - LL) / 3))
        {
            System.out.println("Random number is in the lower third of the range.");
        }
        else if (RN < (LL + 2 * (UL - LL) / 3))
        {
            System.out.println("Random number is in the middle third of the range.");
        }
        else
        {
            System.out.println("Random number is in the upper third of the range.");
        }
    }
}
```

OUTPUT:

```
D:\II BCA >set path=C:\jdk1.5.0\bin
```

```
D:\II BCA >javac RandomNumGen.java
```

```
D:\II BCA >java RandomNumGen
```

```
Enter the lower limit: 1
```

```
Enter the upper limit: 10
```

```
Random number generated: 7
```

```
Random number is in the upper third of the range.
```

```
D:\II BCA >java RandomNumGen
```

```
Enter the lower limit: 5
```

```
Enter the upper limit: 10
```

```
Random number generated: 5
```

```
Random number is in the lower third of the range.
```

```
D:\II BCA >java RandomNumGen
```

```
Enter the lower limit: 2
```

```
Enter the upper limit: 8
```

```
Random number generated: 5
```

```
Random number is in the middle third of the range.
```


//String Manipulation

PROGRAM:

```
import java.io.*;
import java.util.Scanner;
public class StringManipulation
{
    public static void main(String[ ] args)
    {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter the first string: ");
        String str1 = scanner.nextLine();
        System.out.print("Enter the second string: ");
        String str2 = scanner.nextLine();
        char[ ] charArray1 = str1.toCharArray();
        char[ ] charArray2 = str2.toCharArray();
        System.out.println("Length of string 1: " + charArray1.length);
        System.out.println("Length of string 2: " + charArray2.length);
        System.out.print("Enter the position to find the character in string 1 (0- " +
            (charArray1.length - 1) + "): ");
        int position = scanner.nextInt();
        scanner.nextLine();
        if (position >= 0 && position < charArray1.length)
        {
            System.out.println("Character at position " + position + " in string 1: " +
                charArray1[position]);
        }
        else
        {
            System.out.println("Invalid position. Please enter a position between 0 and " +
                (charArray1.length - 1));
        }
        char[ ] concatenatedArray = concatenate(charArray1, charArray2);
        System.out.println("Concatenated string: " + new String(concatenatedArray));
    }
    public static char[ ] concatenate(char[ ] charArray1, char[ ] charArray2)
    {
        char[ ] result = new char[charArray1.length + charArray2.length];
        System.arraycopy(charArray1, 0, result, 0, charArray1.length);
        System.arraycopy(charArray2, 0, result, charArray1.length, charArray2.length);
        return result;
    }
}
```

OUTPUT:

```
D:\II BCA >set path=C:\jdk1.5.0\bin
```

```
D:\II BCA >javac StringManipulation.java
```

```
D:\II BCA >java StringManipulation
```

```
Enter the first string: II BCA
```

```
Enter the second string: kumar
```

```
Length of string 1: 6
```

```
Length of string 2: 5
```

```
Enter the position to find the character in string 1 (0-5): 3
```

```
Character at position 3 in string 1: e
```

```
Concatenated string: II BCAkumar
```

//String Operations using String class.

PROGRAM:

```
import java.io.*;
import java.util.Scanner;
public class StringOperations
{
    public static void main(String[ ] args)
    {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter the first string: ");
        String str1 = scanner.nextLine();
        System.out.print("Enter the second string: ");
        String str2 = scanner.nextLine();
        String conString = str1 + " " + str2;
        System.out.println("Concatenated string: " + conString);
        System.out.print("Enter a substring to search in string 1: ");
        String substring = scanner.nextLine();
        if (str1.contains(substring))
        {
            System.out.println("Substring found in string 1");
        }
        else
        {
            System.out.println("Substring not found in string 1");
        }
        System.out.print("Enter the starting index of the substring to extract from string1: ");
        int sIndex = scanner.nextInt();
        scanner.nextLine();
        System.out.print("Enter the ending index of the substring to extract from string1: ");
        int eIndex = scanner.nextInt();
        scanner.nextLine();
        if (sIndex >= 0 && eIndex >= 0 && sIndex < str1.length() && eIndex <
            str1.length() && sIndex <= eIndex)
        {
            String exSubstring = str1.substring(sIndex, eIndex);
            System.out.println("Extracted substring: " + exSubstring);
        }
        else
        {
            System.out.println("Invalid indices. Please enter valid indices between 0 and " +
                (str1.length() - 1));
        }
    }
}
```

OUTPUT:

D:\II BCA >set path=C:\jdk1.5.0\bin

D:\II BCA >javac StringOperations.java

D:\II BCA >java StringOperations

Enter the first string: Vinod

Enter the second string: kumar

Concatenated string: Vinod kumar

Enter a substring to search in string 1: no

Substring found in string 1

Enter the starting index of the substring to extract from string 1: 0

Enter the ending index of the substring to extract from string 1: 3

Extracted substring: Vin

//String Operations using String Buffer class

PROGRAM:

```
import java.io.*;
import java.util.Scanner;
public class StringBufferOperations
{
    public static void main(String[ ] args)
    {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter a string: ");
        String str = scanner.nextLine( );
        StringBuffer stringBuffer = new StringBuffer(str);
        System.out.println("Length of the string: " + stringBuffer.length());
        stringBuffer.reverse( );
        System.out.println("Reversed string: " + stringBuffer);
        System.out.print("Enter a substring to delete from the string: ");
        String substring = scanner.nextLine();
        int index = stringBuffer.indexOf(substring);
        if (index != -1)
        {
            stringBuffer.delete(index, index + substring.length());
            System.out.println("String after deleting the substring: " + stringBuffer);
        }
        else
        {
            System.out.println("Substring not found in the string");
        }
    }
}
```

OUTPUT:

D:\II BCA >set path=C:\jdk1.5.0\bin

D:\II BCA >javac StringBufferOperations.java

D:\II BCA >java StringBufferOperations

D:\II BCA >java StringBufferOperations

Enter a string: Welcome

Length of the string: 7

Reversed string: emocleW

Enter a substring to delete from the string: oc

String after deleting the substring: emleW

//Multithread Application

PROGRAM:

```
import java.io.*;
import java.util.Random;

class Shared
{
    private int number;
    private boolean isEven;
    private boolean stop;
    public synchronized void setNumber(int number)
    {
        this.number = number;
        this.isEven = (number % 2 == 0);
        notifyAll();
    }
    public synchronized int getNumber( )
    {
        while (number == 0 && !stop)
        {
            try
            {
                wait();
            }
            catch (InterruptedException e)
            {
                Thread.currentThread().interrupt();
            }
        }
        if (stop)
        {
            return 0;
        }
        int temp = number;
        number = 0;
        return temp;
    }
    public synchronized boolean isEven()
    {
        return isEven;
    }
}
```

```

public synchronized void setStop(boolean stop)
{
    this.stop = stop;
    notifyAll();
}
public synchronized boolean isStop( )
{
    return stop;
}
}

class GeneratorThread extends Thread
{
    private Shared shared;
    private Random random;
    private int count;
    public GeneratorThread(Shared shared)
    {
        this.shared = shared;
        this.random = new Random( );
        this.count = 0;
    }
    public void run( )
    {
        while (count < 5)
        {
            int number = random.nextInt(10);
            shared.setNumber(number);
            System.out.println("Generated number: " + number);
            count++;
            try
            {
                Thread.sleep(100);
            }
            catch (InterruptedException e)
            {
                Thread.currentThread( ).interrupt( );
            }
        }
        shared.setStop(true);
    }
}

```



```

class SquareThread extends Thread
{
    private Shared shared;
    public SquareThread(Shared shared)
    {
        this.shared = shared;
    }
    public void run( )
    {
        while (true)
        {
            int number = shared.getNumber( );
            if (shared.isStop( ))
            {
                break;
            }
            if (shared.isEven( ))
            {
                int square = number * number;
                System.out.println("Square of " + number + ": " + square);
            }
        }
    }
}

```

```

class CubeThread extends Thread
{
    private Shared shared;
    public CubeThread(Shared shared)
    {
        this.shared = shared;
    }

    public void run( )
    {
        while (true)
        {
            int number = shared.getNumber( );
            if (shared.isStop( ))
            {
                break;
            }
        }
    }
}

```

```
        if (!shared.isEven( ))
        {
            int cube = number * number * number;
            System.out.println("Cube of " + number + ": " + cube);
        }
    }
}
```

```
public class MultiThreadApp1
{
    public static void main(String[ ] args)
    {
        Shared shared = new Shared();
        GeneratorThread gThread = new GeneratorThread(shared);
        SquareThread sThread = new SquareThread(shared);
        CubeThread cThread = new CubeThread(shared);
        gThread.start();
        sThread.start();
        cThread.start();
    }
}
```

OUTPUT:

D:\II BCA >set path=C:\jdk1.5.0\bin

D:\II BCA >javac MultiThreadApp1.java

D:\II BCA >java MultiThreadApp1

Generated number: 9

Generated number: 5

Cube of 5: 125

Generated number: 2

Generated number: 6

Generated number: 6

Square of 6: 36

// Inter Thread Communication.

PROGRAM:

```
import java.io.*;

// Thread to print numbers 1 to 10
class Thread1 extends Thread
{
    public void run( )
    {
        for (int i = 1; i <= 10; i++)
        {
            System.out.println("Thread1: " + i);
            try
            {
                Thread.sleep(100);
            }
            catch (InterruptedException e)
            {
                Thread.currentThread().interrupt();
            }
        }
    }
}

// Thread to print numbers 90 to 100
class Thread2 extends Thread
{
    public void run( )
    {
        for (int i = 90; i <= 100; i++)
        {
            System.out.println("Thread2: " + i);
            try
            {
                Thread.sleep(100);
            }
            catch (InterruptedException e)
            {
                Thread.currentThread( ).interrupt( );
            }
        }
    }
}
```

```
public class MultiThreadApp2
{
    public static void main(String[ ] args)
    {
        Thread1 t1 = new Thread1( );
        Thread2 t2 = new Thread2( );
        t1.start();
        t2.start();
    }
}
```

OUTPUT:

D:\II BCA >set path=C:\jdk1.5.0\bin

D:\II BCA >javac MultiThreadApp2.java

D:\II BCA >java MultiThreadApp2

Thread1: 1

Thread2: 90

Thread2: 91

Thread1: 2

Thread2: 92

Thread1: 3

Thread1: 4

Thread2: 93

Thread2: 94

Thread1: 5

Thread2: 95

Thread1: 6

Thread2: 96

Thread1: 7

Thread2: 97

Thread1: 8

Thread1: 9

Thread2: 98

Thread1: 10

Thread2: 99

Thread2: 100

//Exception Handling

PROGRAM:

[illegible]

```
// d. NegativeArraySizeException
try
{
    int[ ] arr = new int[-5];
}
catch (NegativeArraySizeException e)
{
    System.out.println("NegativeArraySizeException caught: " + e.getMessage());
}
}
```


OUTPUT:

D:\II BCA >set path=C:\jdk1.5.0\bin

D:\II BCA >javac ExceptionDemo.java

D:\II BCA >java ExceptionDemo

ArithmeticException caught: / by zero

NumberFormatException caught: For input string: "bca"

ArrayIndexOutOfBoundsException caught: 10

NegativeArraySizeException caught: null

// File Handling

PROGRAM:

```
import java.io.File;
import java.io.IOException;
import java.util.Scanner;

public class FileInfo
{
    public static void main(String[ ] args)
    {
        Scanner scanner = new Scanner(System.in);
        System.out.println("Enter a file name:");
        String fileName = scanner.next();
        File file = new File(fileName);
        System.out.println("File Name: " + file.getName());
        System.out.println("File Exists: " + file.exists());
        System.out.println("File is Readable: " + file.canRead());
        System.out.println("File is Writable: " + file.canWrite());
        if (file.isFile())
        {
            System.out.println("File Type: Regular File");
        }
        else if (file.isDirectory())
        {
            System.out.println("File Type: Directory");
        } else
        {
            System.out.println("File Type: Unknown");
        }
        System.out.println("File Length: " + file.length() + " bytes");
    }
}
```

OUTPUT:

D:\II BCA >set path=C:\jdk1.5.0\bin

D:\II BCA >javac FileInfo.java

D:\II BCA >java FileInfo

Enter a file name:

FileInfo.java

File Name: FileInfo.java

File Exists: true

File is Readable: true

File is Writable: true

File Type: Regular File

File Length: 890 bytes

D:\II BCA >java FileInfo

Enter a file name:

rose.txt

File Name: rose.txt

File Exists: false

File is Readable: false

File is Writable: false

File Type: Unknown

File Length: 0 bytes

// Frames and Control

PROGRAM:

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

public class TextEditor extends Frame
{
    private TextArea textArea;
    private Choice fontSizeChoice;
    private Choice fontStyleChoice;
    private Button applyButton;
    public TextEditor( )
    {
        setTitle("Text Editor");
        setSize(600, 400);
        setLayout(new BorderLayout( ));
        textArea = new TextArea( );
        add(textArea, BorderLayout.CENTER);
        Panel controlPanel = new Panel( );
        controlPanel.setLayout(new FlowLayout( ));
        fontSizeChoice = new Choice( );
        for (int i = 8; i <= 72; i += 2)
        {
            fontSizeChoice.add(String.valueOf(i));
        }
        controlPanel.add(new Label("Font Size:"));
        controlPanel.add(fontSizeChoice);
        fontStyleChoice = new Choice( );
        fontStyleChoice.add("Plain");
        fontStyleChoice.add("Bold");
        fontStyleChoice.add("Italic");
        controlPanel.add(new Label("Font Style:"));
        controlPanel.add(fontStyleChoice);
        applyButton = new Button("Apply");
        controlPanel.add(applyButton);
        applyButton.addActionListener(new ActionListener()
        {
            public void actionPerformed(ActionEvent e)
            {
                applyFont();
            }
        });
        add(controlPanel, BorderLayout.SOUTH);
        addWindowListener(new WindowAdapter( )
        {
            public void windowClosing(WindowEvent we)
            {
                dispose( );
            }
        });
    }
}
```

```

        fontSizeChoice.select("12");
        fontStyleChoice.select("Plain");
    }
    private void applyFont( )
    {
        int fontSize = Integer.parseInt(fontSizeChoice.getSelectedItem( ));
        String fontStyle = fontStyleChoice.getSelectedItem( );
        int style = Font.PLAIN;
        if (fontStyle.equals("Bold"))
        {
            style = Font.BOLD;
        }
        else if (fontStyle.equals("Italic"))
        {
            style = Font.ITALIC;
        }
        Font font = new Font("Serif", style, fontSize);
        textArea.setFont(font);
    }
    public static void main(String[] args)
    {
        TextEditor editor = new TextEditor( );
        editor.setVisible(true);
    }
}

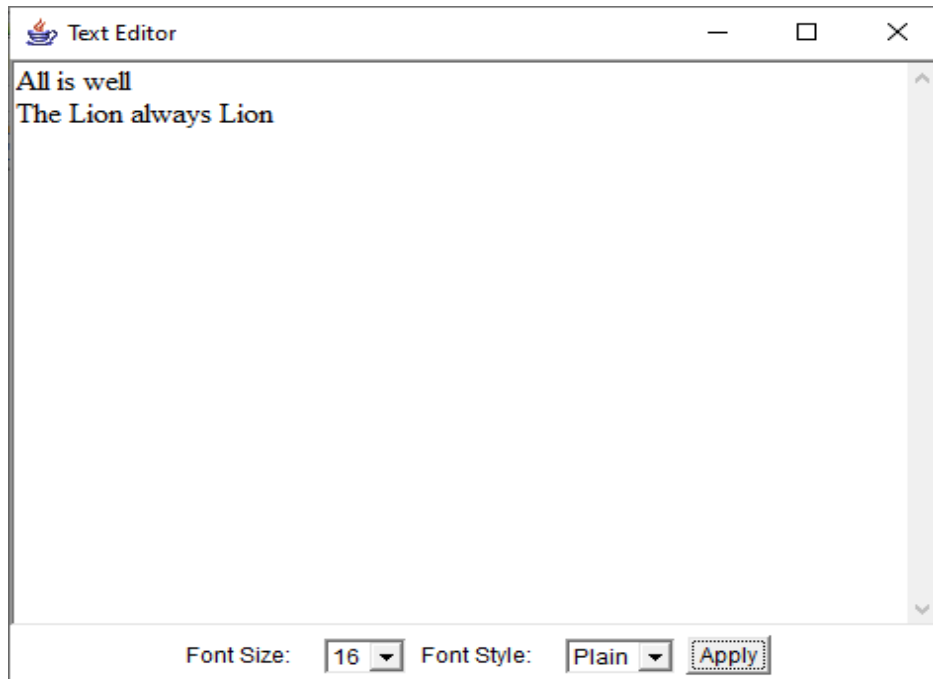
```

OUTPUT:

D:\II BCA>set path=C:\jdk1.5.0\bin

D:\II BCA>javac TextEditor.java

D:\II BCA>java TextEditor





// Mouse Event Handling

PROGRAM :

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

public class MouseEventDemo extends Frame
{
    private String eventName = "Mouse Events";
    public MouseEventDemo( )
    {
        setTitle("Mouse Event Demo");
        setSize(400, 300);
        setLayout(new BorderLayout( ));
        addMouseListener(new MouseAdapter( )
        {
            public void mouseClicked(MouseEvent e)
            {
                eventName = "Mouse Clicked";
                repaint();
            }
            public void mousePressed(MouseEvent e)
            {
                eventName = "Mouse Pressed";
                repaint();
            }
            public void mouseReleased(MouseEvent e)
            {
                eventName = "Mouse Released";
                repaint();
            }
            public void mouseEntered(MouseEvent e)
            {
                eventName = "Mouse Entered";
                repaint( );
            }
            public void mouseExited(MouseEvent e)
            {
                eventName = "Mouse Exited";
                repaint( );
            }
        });
        addWindowListener(new WindowAdapter( )
        {
            public void windowClosing(WindowEvent we)
            {
                dispose();
            }
        });
    }
}
```



```
public void paint(Graphics g)
{
    Font font = new Font("Arial", Font.BOLD, 24);
    g.setFont(font);
    FontMetrics metrics = g.getFontMetrics(font);
    int x = (getWidth( ) - metrics.stringWidth(eventName)) / 2;
    int y = (getHeight( ) - metrics.getHeight()) / 2 + metrics.getAscent();
    g.drawString(eventName, x, y);
}

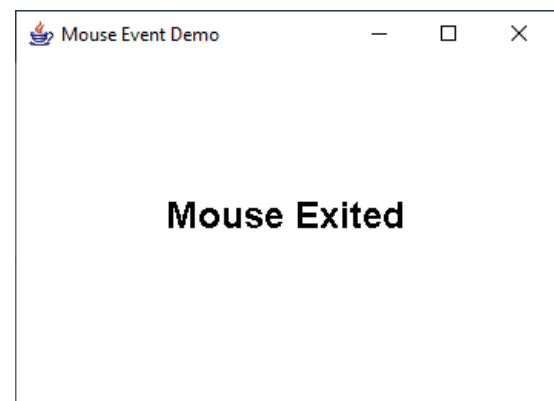
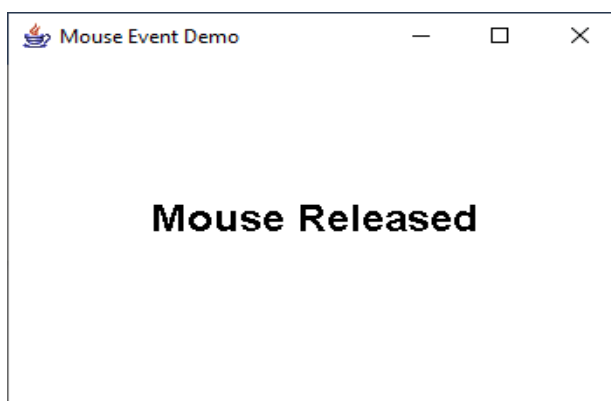
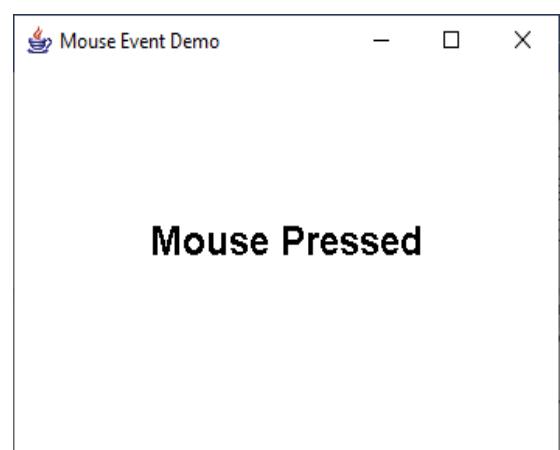
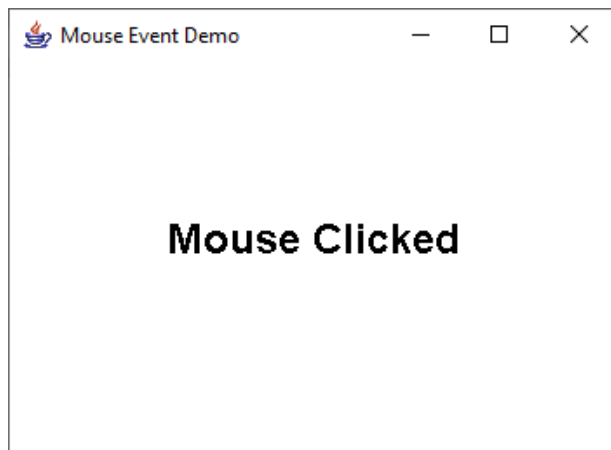
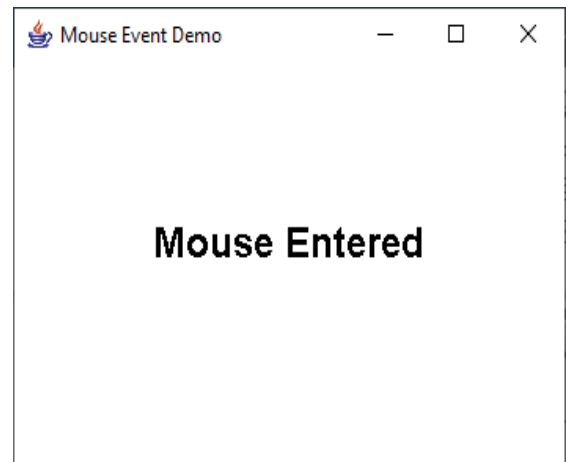
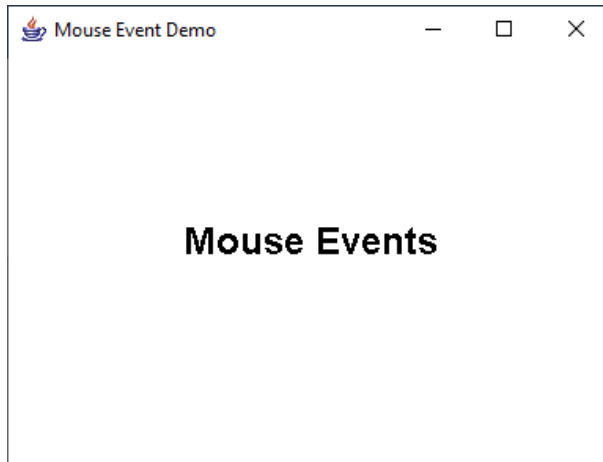
public static void main(String[ ] args)
{
    MouseEventDemo demo = new MouseEventDemo( );
    demo.setVisible(true);
}
}
```

OUTPUT :

D:\II BCA>set path=C:\jdk1.5.0\bin

D:\II BCA>javac MouseEventDemo.java

D:\II BCA>java MouseEventDemo



// Simple calculator using Java Swing

PROGRAM:

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

public class Calculator extends JFrame
{
    private JTextField resultField;
    private double number1, number2, result;
    private char operation;
    public Calculator( )
    {
        resultField = new JTextField(20);
        resultField.setEditable(false);
        JPanel buttonPanel = new JPanel();
        buttonPanel.setLayout(new GridLayout(4, 4));
        String[ ] buttons = { "7", "8", "9", "/", "4", "5", "6", "*", "1", "2", "3", "-",
                               "0", ".", "=", "+" };
        for (String button : buttons)
        {
            JButton btn = new JButton(button);
            btn.addActionListener(new ButtonListener( ));
            buttonPanel.add(btn);
        }
        add(resultField, BorderLayout.NORTH);
        add(buttonPanel, BorderLayout.CENTER);
        setSize(200, 200);
        setDefaultCloseOperation(EXIT_ON_CLOSE);
        setVisible(true);
    }
    private class ButtonListener implements ActionListener
    {
        public void actionPerformed(ActionEvent e)
        {
            String command = e.getActionCommand( );
            if (command.equals("="))
            {
                try
                {
                    number2 = Double.parseDouble(resultField.getText( ));
                    switch (operation)
                    {
```

```

        case '+' :
            result = number1 + number2;
            break;
        case '-' :
            result = number1 - number2;
            break;
        case '*' :
            result = number1 * number2;
            break;
        case '/' :
            if (number2 != 0)
            {
                result = number1 / number2;
            }
            else
            {
                resultField.setText("Error: Division by zero");
                return;
            }
            break;
    }
    resultField.setText(String.valueOf(result));
}
catch (NumberFormatException ex)
{
    resultField.setText("Error: Invalid input");
}
}
else if (command.equals("+") || command.equals("-") || command.equals("*") ||
        command.equals("/"))
{
    number1 = Double.parseDouble(resultField.getText());
    operation = command.charAt(0);
    resultField.setText("");
} else {
    resultField.setText(resultField.getText() + command);
}
}
}
public static void main(String[] args)
{
    new Calculator();
}
}

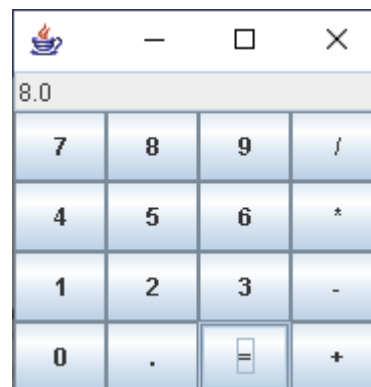
```

OUTPUT:

D:\II BCA >set path=C:\jdk1.5.0\bin

D:\II BCA >javac Calculator.java

D:\II BCA >java Calculator



// Simulation of Traffic light using Java Swing.

PROGRAM:

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

public class TrafficLight extends JFrame
{
    private JRadioButton redButton, yellowButton, greenButton;
    private JLabel messageLabel;
    public TrafficLight()
    {
        // Create a panel for the message label
        JPanel messagePanel = new JPanel();
        messagePanel.setLayout(new FlowLayout());
        messageLabel = new JLabel("", SwingConstants.CENTER);
        messageLabel.setPreferredSize(new Dimension(200, 30));
        messagePanel.add(messageLabel);
        // Create a panel for the radio buttons
        JPanel buttonPanel = new JPanel();
        buttonPanel.setLayout(new FlowLayout());
        redButton = new JRadioButton("Red");
        yellowButton = new JRadioButton("Yellow");
        greenButton = new JRadioButton("Green");
        ButtonGroup group = new ButtonGroup();
        group.add(redButton);
        group.add(yellowButton);
        group.add(greenButton);
        buttonPanel.add(redButton);
        buttonPanel.add(yellowButton);
        buttonPanel.add(greenButton);
        // Add action listeners to the radio buttons
        redButton.addActionListener(new RadioButtonListener());
        yellowButton.addActionListener(new RadioButtonListener());
        greenButton.addActionListener(new RadioButtonListener());
        // Add the panels to the frame
        add(messagePanel, BorderLayout.NORTH);
        add(buttonPanel, BorderLayout.CENTER);
        setSize(250, 150);
        setDefaultCloseOperation(EXIT_ON_CLOSE);
        setVisible(true);
    }
}
```

```
private class RadioButtonListener implements ActionListener
{
    public void actionPerformed(ActionEvent e)
    {
        if (redButton.isSelected())
        {
            messageLabel.setText("STOP");
            messageLabel.setForeground(Color.RED);
        }
        else if (yellowButton.isSelected())
        {
            messageLabel.setText("READY");
            messageLabel.setForeground(Color.YELLOW);
        }
        else if (greenButton.isSelected())
        {
            messageLabel.setText("GO");
            messageLabel.setForeground(Color.GREEN);
        }
    }
}

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

OUTPUT:

D:\II BCA >set path=C:\jdk1.5.0\bin

D:\II BCA >javac TrafficLight.java

D:\II BCA >java TrafficLight

