

Program No: 3

Date: 29/11/2025

Program Title : Keep a secret number between 20 and 30 in your program. Ask the user to predict a number between 20 and 30. Use a do-while loop until the user predicts your secret number and displays the count of attempts.

```
/* Program to implement a number guessing game using do-while loop
@Meenadevi Ravikumar
Roll no: 43
Date: 29/11/2025
*/
import java.util.Scanner;

public class NumberGuess {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int secret = 25; // Secret number between 20 and 30
        int guess, attempts = 0;

        do {
            System.out.print("Guess a number between 20 and 30: ");
            guess = sc.nextInt();
            attempts++;
        } while (guess != secret);

        System.out.println("Congratulations! You guessed it in " + attempts + " attempts.");
    }
}
```

Output

```
C:\sem2\java>java NumberGuess
Guess a number between 20 and 30: 21
Guess a number between 20 and 30: 24
Guess a number between 20 and 30: 25
Congratulations! You guessed it in 3 attempts.
```

Program No: 4

Date: 29/11/2025

Program Title : Program to implement searching and sorting elements in an Array.

```
/* Program to implement searching and sorting elements in an Array.
```

```
@Meenadevi Ravikumar
```

```
Roll no: 43
```

```
Date: 29/11/2025
```

```
*/
```

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

```
class C1{  
    int a[];
```

```
    public void read()  
{
```

```
        int n;  
        System.out.print("Enter the number of elements in the array:");
```

```
        Scanner s=new Scanner(System.in);
```

```
        n=s.nextInt();
```

```
        a=new int[n];
```

```
        System.out.print("Enter the elements:");
```

```
        for(int i=0;i<n;i++)
```

```
{
```

```
    a[i]=s.nextInt();
```

```
}
```

```
}
```

```
    public void search(int k)  
{
```

```
        int flag=0,i;
```

```
        for(i=0;i<a.length;i++)
```

```
{
```

```
    if(a[i]==k)
```

```
{
```

```
        flag=1;
```

```
        break;
```

```
}
```

```
}
```

```
if(flag==1)
```

```
{
```

```
    System.out.println("Found at location:"+i);
```

```
}
```

```
        else
        {
            System.out.println("Not Found");
        }
    }
public void sort()
{
    int temp;
    for(int i=0;i<a.length;i++)
    {
        for(int j=0;j<a.length-i-1;j++)
        {
            if(a[j]>a[j+1])
            {
                temp=a[j];
                a[j]=a[j+1];
                a[j+1]=temp;
            }
        }
    }
    for(int i=0;i<a.length;i++)
    {
        System.out.print(a[i] + " ");
    }
}
}

class P4{
    public static void main(String args[])
    {
        C1 obj=new C1();
        int ch,k;
        Scanner s=new Scanner(System.in);
        obj.read();
        do{
            System.out.println("\nMenu: \n 1.Search\n 2.Sort\n 3.Exit\n Enter your choice:");
            ch=s.nextInt();

            switch(ch)
            {
                case 1:System.out.print("Enter the element to search:");
                k=s.nextInt();
                obj.search(k);
                break;
                case 2:obj.sort();
                break;
                case 3:break;
            }
        }
    }
}
```

```
        default:System.out.println("Invalid Input");
    }
}while(ch!=3);
}
}
```

Output

```
C:\sem2\java>javac P4.java

C:\sem2\java>java P4
Enter the number of elements in the array:4
Enter the elements:2
3
5
1

Menu:
1.Search
2.Sort
3.Exit
Enter your choice:
1
Enter the element to search:5
Found at location:2

Menu:
1.Search
2.Sort
3.Exit
Enter your choice:
2
1 2 3 5
```

Program No: 5

Date: 29/11/2025

Program Title : Program to implement a Java program to pass an array in different ways.

```
/* Program to implement a Java program to pass an array in different ways.
```

```
@Meenadevi Ravikumar
```

```
Roll no: 43
```

```
Date: 29/11/2025
```

```
*/
```

```
import java.io.*;
import java.util.*;
class c1
{
    public int[] read(int a[]){
        Scanner s=new Scanner(System.in);
        System.out.println("Enter the no. of elements:");
        int n=s.nextInt();
        int a[]=new int[n];
        System.out.println("Enter the elements:");
        for(int i=0;i<n;i++)
        {
            a[i]=s.nextInt();
        }
        return a;
    }
    public void disp(int a[]){
        System.out.println("The elements are:");
        for(int i=0;i<a.length;i++)
        {
            System.out.print(a[i]+"\t");
        }
        System.out.println("");
    }
}
```

```
class Program5
{
    public static void main(String args[])
    {
        c1 obj=new c1();
        int a[]={};
        a=obj.read(a);
        obj.disp(a);
    }
}
```

Output

```
C:\sem2\java>javac P5.java
```

```
C:\sem2\java>java P5  
Enter the no. of elements:
```

```
4
```

```
Enter the elements:
```

```
3
```

```
2
```

```
4
```

```
1
```

```
The elements are:
```

```
3      2      4      1
```



Program No: 6

Date: 01/12/2025

Program Title : Program to add and multiply two matrices.

```
/* Program to add and multiply two matrices.  
@Meenadevi Ravikumar  
Roll no: 43  
Date: 01/12/2025  
*/  
import java.util.Scanner;  
  
class AddMult{  
    int a[][],b[][],c[][];  
    int r1,c1,r2,c2;  
    public void read()  
{  
  
    Scanner s=new Scanner(System.in);  
    System.out.println("Enter the order of the first matrix:");  
    r1=s.nextInt();  
    c1=s.nextInt();  
    a=new int[r1][c1];  
    System.out.println("Enter the order of the second matrix:");  
    r2=s.nextInt();  
    c2=s.nextInt();  
    b=new int[r2][c2];  
    System.out.println("Enter the elements of the first matrix:");  
    for(int i=0;i<r1;i++)  
    {  
        for(int j=0;j<c1;j++)  
        {  
            a[i][j]=s.nextInt();  
        }  
    }  
    System.out.println("Enter the elements of the second matrix:");  
    for(int i=0;i<r2;i++)  
    {  
        for(int j=0;j<c2;j++)  
        {  
            b[i][j]=s.nextInt();  
        }  
    }  
}  
public void add()  
{
```

```
if(r1!=r2 && c1!=c2)
{
    System.out.println("Addition not possible");
    return;
}
c=new int[r1][c1];
for(int i=0;i<r1;i++)
{
    for(int j=0;j<c1;j++)
    {
        c[i][j]=a[i][j]+b[i][j];
    }
}
System.out.println("Sum of two Matrix:\n");
for(int i=0;i<r1;i++)
{
    for(int j=0;j<c1;j++)
    {
        System.out.print(c[i][j]+" ");
    }
    System.out.println("");
}
}
public void multiply()
{
    if(c1!=r2)
    {
        System.out.println("Multiplication not possible");
        return;
    }
    for(int i=0;i<r1;i++)
    {
        for(int j=0;j<c2;j++)
        {
            c[i][j]=0;
            for(int k=0;k<c1;k++)
            {
                c[i][j]+=a[i][k]*b[k][j];
            }
        }
    }
    System.out.println("Product of two Matrix:\n");
    for(int i=0;i<r1;i++)
    {
        for(int j=0;j<c2;j++)
        {
```

```

        System.out.print(c[i][j]+" ");
    }
    System.out.println("");
}

}

public class Program6{
    public static void main(String args[])
    {
        AddMult obj=new AddMult();
        obj.read();
        obj.add();
        System.out.println("Do you wish to find product of another Matrices?(y/n)");
        Scanner s=new Scanner(System.in);
        char ch=s.next().charAt(0);
        if(ch=='n' || ch=='N')
        {
            obj.multiply();
        }
        else
        {
            obj.read();
            obj.multiply();
        }
    }
}

```

Output

```

C:\sem2>java>javac P6.java
C:\sem2>java P6
Enter the order of the first matrix:
2
2
Enter the order of the second matrix:
2
2
Enter the elements of the first matrix:
2
4
5
1
Enter the elements of the second matrix:
2
1
0
1
Sum of two Matrix:
4 5
5 2
Do you wish to find product of another Matrices?(y/n)
y
Enter the order of the first matrix:
2
2
Enter the order of the second matrix:
2
2
Enter the elements of the first matrix:
1
3
0
1
Enter the elements of the second matrix:
1
3
5
1
Product of two Matrix:
16 6
5 1

```

Program No: 7

Date: 01/12/2025

Program Title : A function takes two integer arguments and returns the maximum. Use this function to find the maximum of three numbers

```
/* A function takes two integer arguments and returns the maximum. Use this function to
find the maximum of three numbers
@Meenadevi Ravikumar
Roll no: 43
Date: 01/12/2025
*/
import java.io.*;
import java.util.*;
class L
{
    int a,b,c;
    public void read()
    {
        Scanner s=new Scanner(System.in);
        System.out.println("Enter 3 numbers:");
        a=s.nextInt();
        b=s.nextInt();
        c=s.nextInt();
    }
    public int large(int n1,int n2)
    {
        return n1>n2?n1:n2;
    }
    public void largest()
    {
        int g;
        g=large(a,b);
        g=large(g,c);
        System.out.println("Largest among three numbers:"+g);
    }
}
public class P7
{
    public static void main(String args[])
    {
        L obj=new L();
        obj.read();
        obj.largest();
    }
}
```

Output

```
C:\sem2\java>java P7
Enter 3 numbers:
4
7
2
Largest among three numbers:7
```



Program No: 8

Date: 01/12/2025

Program Title : A function takes an integer argument and returns the reverse of the same. Another function takes an integer argument and returns true or false if the number is a palindrome, using the above function.

```
/* A function takes an integer argument and returns the reverse of the same. Another function takes an integer argument and returns true or false if the number is a palindrome, using the above function.
```

@Meenadevi Ravikumar

Roll no: 43

Date: 01/12/2025

*/

```
import java.io.*;
```

```
import java.util.*;
```

```
class C1
```

```
{
```

```
    public int reverse(int a)
```

```
{
```

```
    int n=a,temp=0,rev=0;
```

```
    while(n!=0)
```

```
{
```

```
    temp=n%10;
```

```
    rev=(rev*10)+temp;
```

```
    n/=10;
```

```
}
```

```
    return rev;
```

```
}
```

```
    public Boolean palindrome(int a)
```

```
{
```

```
        return a==reverse(a);
```

```
}
```

```
}
```

```
public class P8
```

```
{
```

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

```
{
```

```
        C1 obj=new C1();
```

```
        Scanner s=new Scanner(System.in);
```

```
        System.out.println("Enter a number:");
```

```
        int num=s.nextInt();
```

```
        int rev=obj.reverse(num);
```

```
        System.out.println("Reversed number:"+rev);
```

```
        if(obj.palindrome(num))
```

```
{
```

```
        System.out.println("It is a palindrome");
    }
else
{
    System.out.println("It is not a palindrome");
}
}
```

Output

```
C:\sem2\java>javac P8.java
```

```
C:\sem2\java>java P8
```

```
Enter a number:
```

```
76543
```

```
Reversed number:34567
```

```
It is not a palindrome
```

```
C:\sem2\java>java P8
```

```
Enter a number:
```

```
8989898
```

```
Reversed number:8989898
```

```
It is a palindrome
```

Program No: 9

Date: 03/12/2025

Program Title : The 15-puzzle (also known as the sliding puzzle) consists of a 4×4 grid containing 15 numbered tiles (from 1 to 15) and one empty space. The objective is to rearrange the tiles by sliding them horizontally or vertically into the empty space to achieve the correct ascending order.

```
/* Program to implement the 15-puzzle  
@Meenadevi Ravikumar
```

Roll no: 43

Date: 03/12/2025

*/

```
import java.io.*;
```

```
import java.util.*;
```

```
class Puzzle
```

```
{
```

```
    int a[];
```

```
    int x;
```

```
    int count;
```

```
    Puzzle()
```

```
{
```

```
    count=0;
```

```
    a = new int[16];
```

```
    for(int i = 0; i < 15; i++)
```

```
{
```

```
    a[i] = i + 1;
```

```
}
```

```
    a[15] = 0;
```

```
    Random r = new Random();
```

```
    for (int i = 0; i < 15; i++) {
```

```
        int j = r.nextInt(15);
```

```
        int temp = a[i];
```

```
        a[i] = a[j];
```

```
        a[j] = temp;
```

```
}
```

```
    x = 15;
```

```
}
```

```
    public void draw()
```

```
{
```

```
    for(int i=0;i<16;i++)
```

```
{
```

```
    if(a[i]==0)
```

```
{
```

```
        System.out.printf("%-4c",'_');
```

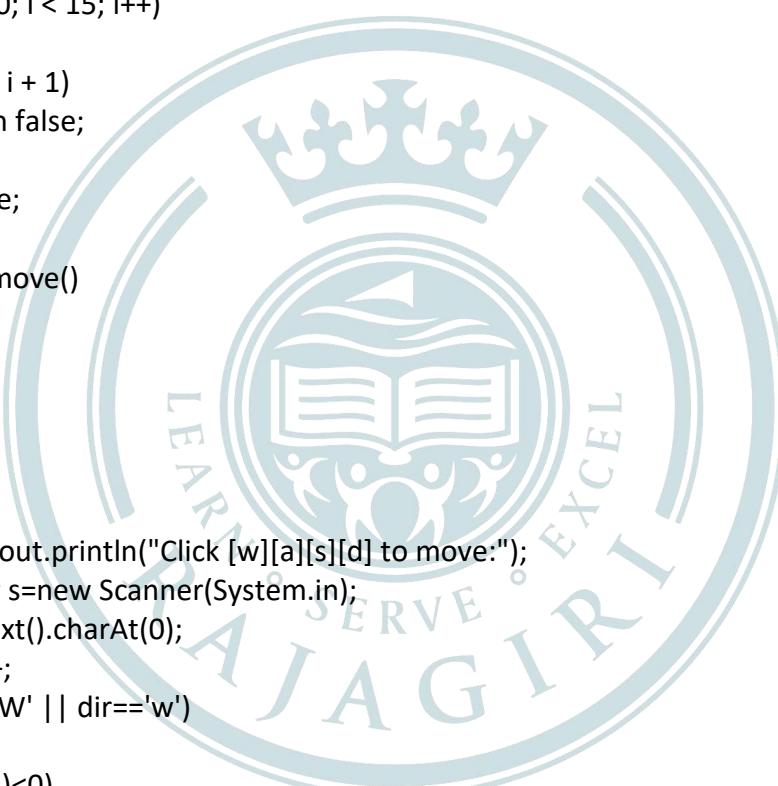
```
}
```

```

        else
        {
            System.out.printf("%-4d",a[i]);
        }
        if((i+1)%4==0)
        {
            System.out.println();
        }

    }
}
public boolean won()
{
    for(int i = 0; i < 15; i++)
    {
        if(a[i] != i + 1)
            return false;
    }
    return true;
}
public void move()
{
    char dir;
    int temp;
    do
    {
        draw();
        System.out.println("Click [w][a][s][d] to move:");
        Scanner s=new Scanner(System.in);
        dir=s.next().charAt(0);
        count++;
        if(dir=='W' || dir=='w')
        {
            if((x-4)<0)
            {
                System.out.println("Unable to move up!!!");
            }
            else
            {
                temp=a[x];
                a[x]=a[x-4];
                a[x-4]=temp;
                x=x-4;
            }
        }
        if(dir=='A' || dir=='a')
        {

```



```
if(x%4==0)
{
    System.out.println("Unable to move left!!!");
}
else
{
    temp=a[x];
    a[x]=a[x-1];
    a[x-1]=temp;
    x=x-1;
}
}
if(dir=='S' || dir=='s')
{
    if((x+4)>15)
    {
        System.out.println("Unable to move down!!!");
    }
    else
    {
        temp=a[x];
        a[x]=a[x+4];
        a[x+4]=temp;
        x=x+4;
    }
}
if(dir=='D' || dir=='d')
{
    if((x+1)%4==0)
    {
        System.out.println("Unable to move right!!!");
    }
    else
    {
        temp=a[x];
        a[x]=a[x+1];
        a[x+1]=temp;
        x=x+1;
    }
}
if(won())
{
    System.out.println("Congrats You Won in "+count+" moves");
    break;
}
}while(true);
}
```

```
}
```

```
class P9
```

```
{
```

```
    public static void main(String args[])
    {
        Puzzle obj=new Puzzle();
        obj.move();
    }
}
```

Output

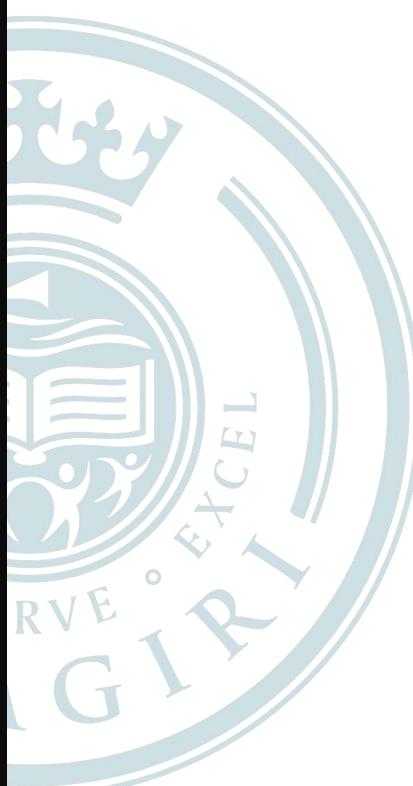
```
C:\sem2>javac P8.java
```

```
C:\sem2>java P8
Enter a number:
76543
Reversed number:34567
It is not a palindrome
```

```
C:\sem2>java P8
Enter a number:
8989898
Reversed number:8989898
It is a palindrome
```

```
C:\sem2>javac P9.java
```

```
C:\sem2>java P9
11 6 12 4
8 2 10 14
13 1 3 5
15 7 9
Click [w][a][s][d] to move:
a
11 6 12 4
8 2 10 14
13 1 3 5
15 7 9
Click [w][a][s][d] to move:
w
11 6 12 4
8 2 10 14
13 1 5
15 7 3 9
Click [w][a][s][d] to move:
d
11 6 12 4
8 2 10 14
13 1 5
15 7 3 9
Click [w][a][s][d] to move:
s
11 6 12 4
8 2 10 14
13 1 5 9
15 7 3
Click [w][a][s][d] to move:
```



Program No: 10

Date: 03/12/2025

Program Title : Program to display from n - 1 and 1-n using recursive functions.

```
/* Program to display from n - 1 and 1-n using recursive functions.
```

```
@Meenadevi Ravikumar
```

```
Roll no: 43
```

```
Date: 03/12/2025
```

```
*/
```

```
import java.io.*;
```

```
import java.util.*;
```

```
class Recursion
```

```
{
```

```
    public void oneToN(int n,int i)
```

```
{
```

```
    if(i>n)
```

```
{
```

```
        return;
```

```
}
```

```
    else
```

```
{
```

```
        System.out.print(i+" ");
```

```
        oneToN(n,i+1);
```

```
}
```

```
}
```

```
    public void nToOne(int n)
```

```
{
```

```
    if(n==0)
```

```
{
```

```
        return;
```

```
}
```

```
    else
```

```
{
```

```
        System.out.print(n+" ");
```

```
        nToOne(n-1);
```

```
}
```

```
}
```

```
}
```

```
class Program10
```

```
{
```

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

```
{
```

```
    Recursion obj=new Recursion();
```

```
    int n,i=1;
```

```
Scanner s=new Scanner(System.in);
System.out.println("Enter the limit:");
n=s.nextInt();
System.out.println("N to 1:");
obj.nToOne(n);
System.out.println("\n1 to N:");
obj.oneToN(n,i);
}
}
```

Output

```
C:\sem2\java>javac P10.java

C:\sem2\java>java P10
Enter the limit:
8
N to 1:
8 7 6 5 4 3 2 1
1 to N:
1 2 3 4 5 6 7 8
```

Program No: 11

Date: 03/12/2025

Program Title : Program for adding, subtracting, multiplying and dividing two numbers using object-oriented concepts.

/* Program for adding, subtracting, multiplying and dividing two numbers using object-oriented concepts.

@Meenadevi Ravikumar

Roll no: 43

Date: 03/12/2025

*/

import java.io.*;

import java.util.*;

class A

{

 int n;

 public void read()

 {

 Scanner s=new Scanner(System.in);

 n=s.nextInt();

 }

 public void display()

 {

 System.out.println(n);

 }

 public void add(A obj)

 {

 System.out.println(n+"+"+obj.n+"=");

 n = n + obj.n;

 display();

 }

 public void subtract(A obj)

 {

 System.out.println(n+"-"+obj.n+"=");

 n = n - obj.n;

 display();

 }

 public void multiply(A obj)

 {

 System.out.println(n+"*"+obj.n+"=");

 n = n * obj.n;

 display();

 }

 public void divide(A obj)

 {

 if(obj.n==0)

 {



```

        System.out.println("Can't Divide by Zero");
    }
    else
    {
        System.out.println(n+"/"+obj.n+"=");
        n = n / obj.n;
        display();
    }
}
public class Program11
{
    public static void main(String args[])
    {
        A a=new A();
        A b=new A();
        int ch;
        Scanner s=new Scanner(System.in);
        do{

            System.out.println("Enter two numbers:");
            a.read();
            b.read();
            System.out.println("1.Addition\n2.Subtraction\n3.Multiplication\n4.Division\n5.Exit");
            System.out.println("Enter your choice:");
            ch=s.nextInt();
            switch(ch)
            {
                case 1:a.add(b);
                break;
                case 2:a.subtract(b);
                break;
                case 3:a.multiply(b);
                break;
                case 4:a.divide(b);
                break;
                case 5:System.out.println("Exiting!!!");
                break;
                default:System.out.println("Invalid Choice");
            }
        }while(ch!=5);
    }
}

```

Output

```
C:\sem2\java>java P11
Enter two numbers:
3
2
1.Addition
2.Subtraction
3.Multiplication
4.Division
5.Exit
Enter your choice:
1
3+2=
5
Enter two numbers:
2
6
1.Addition
2.Subtraction
3.Multiplication
4.Division
5.Exit
Enter your choice:
3
2*6=
12
Enter two numbers:
5
1
1.Addition
2.Subtraction
3.Multiplication
4.Division
5.Exit
Enter your choice:
4
5/1=
5
```



Program No: 12

Date: 06-12-2025

Program Title : Create a class complex having real and imaginary as data members and provide function for read, display, add and multiply.

```
/* Create a class complex having real and imaginary as data members and provide function for read, display, add and multiply.
```

```
@Meenadevi Ravikumar
```

```
Roll no: 43
```

```
Date: 06-12-2025
```

```
*/
```

```
import java.util.*;
```

```
class Complex {  
    private int real;  
    private int imag;  
  
    public void read() {  
        Scanner s = new Scanner(System.in);  
        System.out.print("Enter real part: ");  
        real = s.nextInt();  
        System.out.print("Enter imaginary part: ");  
        imag = s.nextInt();  
    }  
  
    public void display() {  
        if(imag >= 0)  
            System.out.println(real + "+" + imag + "i");  
        else  
            System.out.println(real + "" + imag + "i");  
    }  
  
    public Complex add(Complex c) {  
        Complex result = new Complex();  
        result.real = this.real + c.real;  
        result.imag = this.imag + c.imag;  
        return result;  
    }  
  
    public Complex multiply(Complex c) {  
        Complex result = new Complex();  
        result.real = this.real * c.real - this.imag * c.imag;  
        result.imag = this.real * c.imag + this.imag * c.real;  
        return result;  
    }  
}
```

```
public class Program12 {  
    public static void main(String[] args) {  
        Complex c1 = new Complex();  
        Complex c2 = new Complex();  
        System.out.println("Enter first complex number:");  
        c1.read();  
        System.out.println("Enter second complex number:");  
        c2.read();  
        System.out.print("First complex number: ");  
        c1.display();  
        System.out.print("Second complex number: ");  
        c2.display();  
        Complex sum = c1.add(c2);  
        System.out.print("Sum: ");  
        sum.display();  
        Complex product = c1.multiply(c2);  
        System.out.print("Product: ");  
        product.display();  
    }  
}
```

Output

```
C:\sem2\java>javac P12.java  
  
C:\sem2\java>java P12  
Enter first complex number:  
Enter real part: 2  
Enter imaginary part: 4  
Enter second complex number:  
Enter real part: 6  
Enter imaginary part: 7  
First complex number: 2+4i  
Second complex number: 6+7i  
Sum: 8+11i  
Product: -16+38i
```

Program No: 13

Date: 06-12-2025

Program Title : Create a class for CString having a string data member and provide functions for read, display, compare(return Boolean value),add and concatenate.

```
/* Create a class for CString having a string data member and provide functions for read,  
display, compare(return Boolean value),add and concatenate.
```

```
@Meenadevi Ravikumar
```

```
Roll no: 43
```

```
Date: 06-12-2025
```

```
*/
```

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

```
class CString  
{  
    String str;  
  
    public void read()  
    {  
        Scanner s = new Scanner(System.in);  
        str = s.nextLine();  
    }  
  
    public void display()  
    {  
        System.out.println(str);  
    }  
  
    public boolean compare(CString s)  
    {  
        return str.equals(s.str);  
    }  
  
    public void add(CString s)  
    {  
        System.out.println("Addition Result:");  
        System.out.println(str + s.str);  
    }  
  
    public void concat(CString obj)  
    {  
        System.out.println("Concatenation Result:");  
        str = str + obj.str;  
        display();  
    }  
}
```

```
public class Program13
{
    public static void main(String args[])
    {
        CString a = new CString();
        CString b = new CString();

        System.out.println("Enter a string:");
        a.read();
        System.out.println("Enter another string:");
        b.read();

        System.out.println("First String:");
        a.display();
        System.out.println("Second String:");
        b.display();

        System.out.println("Comparison Result: " + a.compare(b));
        a.add(b);
        a.concat(b);
    }
}
```

Output

```
C:\sem2\java>javac Program13.java
C:\sem2\java>java Program13
Enter a string:
helloo
Enter another string:
hai
First String:
helloo
Second String:
hai
Comparison Result: false
Addition Result:
helloohai
Concatenation Result:
helloohai
```

Program No: 14

Date: 06-12-2025

Program Title : Create a class for a student having (sno, sname, sprogram) [example sno 1,sname-sanjay,sprogram-MCA]. Provide functions for reading and displaying the same information. Also provide a function for comparing two student objects for equality.

```
/* Create a class for a student having (sno, sname, sprogram) [example sno 1,sname-sanjay,sprogram-MCA]. Provide functions for reading and displaying the same information. Also provide a function for comparing two student objects for equality.
```

@Meenadevi Ravikumar

Roll no: 43

Date: 06-12-2025

*/

```
import java.util.*;
```

```
class Student
```

```
{
```

```
    int sno;
```

```
    String sname;
```

```
    String sprogram;
```

```
    public void read()
```

```
{
```

```
        Scanner sc = new Scanner(System.in);
```

```
        System.out.println("Enter student number:");
```

```
        sno = sc.nextInt();
```

```
        sc.nextLine();
```

```
        System.out.println("Enter student name:");
```

```
        sname = sc.nextLine();
```

```
        System.out.println("Enter student program:");
```

```
        sprogram = sc.nextLine();
```

```
}
```

```
    public void display()
```

```
{
```

```
        System.out.println(sno);
```

```
        System.out.println(sname);
```

```
        System.out.println(sprogram);
```

```
}
```

```
    public boolean compare(Student s)
```

```
{
```

```
        return sno == s.sno && sname.equals(s.sname) && sprogram.equals(s.sprogram);
```

```
}
```

```
}
```

```
public class Program14 {
```

```
public static void main(String args[]) {  
    Student a = new Student();  
    Student b = new Student();  
  
    System.out.println("Enter student 1 details:");  
    a.read();  
  
    System.out.println("Enter student 2 details:");  
    b.read();  
  
    System.out.println("Student 1:");  
    a.display();  
  
    System.out.println("Student 2:");  
    b.display();  
  
    System.out.println("Both students equal: " + a.compare(b));  
}  
}
```

Output

```
C:\sem2\java>javac P14.java  
C:\sem2\java>java P14  
Enter student 1 details:  
Enter student number:  
2  
Enter student name:  
meena  
Enter student program:  
mca  
Enter student 2 details:  
Enter student number:  
3  
Enter student name:  
rena  
Enter student program:  
mca  
Student 1:  
2  
meena  
mca  
Student 2:  
3  
rena  
mca  
Both students equal: false
```

Program No: 15

Date: 08/12/2025

Program Title : Program to implement the ISA and HASA relationship.

```
/* Program to implement the ISA and HASA relationship.
```

```
@Meenadevi Ravikumar
```

```
Roll no: 43
```

```
Date: 08/12/2025
```

```
*/
```

```
class Vehicle {
```

```
    void displayType() {
```

```
        System.out.println("I am a Vehicle");
```

```
}
```

```
}
```

```
class Engine {
```

```
    void start() {
```

```
        System.out.println("Engine started");
```

```
}
```

```
}
```

```
class Car extends Vehicle { // ISA: Car is a Vehicle
```

```
    Engine engine; // HASA: Car has an Engine
```

```
    Car() {
```

```
        engine = new Engine();
```

```
}
```

```
    void startCar() {
```

```
        engine.start();
```

```
        System.out.println("Car is ready to go!");
```

```
}
```

```
}
```

```
public class Program15 {
```

```
    public static void main(String[] args) {
```

```
        Car myCar = new Car();
```

```
        myCar.displayType(); // ISA
```

```
        myCar.startCar(); // HASA
```

```
}
```

```
}
```

Output

```
C:\sem2\java>javac P15.java
```

```
C:\sem2\java>java P15
```

```
I am a Vehicle
```

```
Engine started
```

```
Car is ready to go!
```

Program No: 16

Date: 08/12/2025

Program Title : Program to overcome function overriding in Java.

```
/* Program to overcome function overriding in Java.
```

```
@Meenadevi Ravikumar
```

```
Roll no: 43
```

```
Date: 08/12/2025
```

```
*/
```

```
import java.io.*;
import java.util.*;
class Person
{
    int age;
    String name;
    public void read()
    {
        Scanner s=new Scanner(System.in);
        System.out.println("Enter the name and age:");
        name=s.nextLine();
        age=s.nextInt();
    }
    public void disp()
    {
        System.out.println("Name:"+name);
        System.out.println("age:"+age);
    }
}
class Teacher extends Person
{
    String subj;
    public void read()
    {
        super.read();
        Scanner s=new Scanner(System.in);
        System.out.println("Enter the subject:");
        subj=s.nextLine();
    }
    public void disp()
    {
        super.disp();
        System.out.println("Subject:"+subj);
    }
}
```

```
class Program16
{
    public static void main(String args[])
    {
        Person p=new Person();
        Teacher t=new Teacher();
        p.read();
        t.read();
        p.disp();
        t.disp();
    }
}
```

Output

```
C:\sem2\java>javac P16.java
C:\sem2\java>java P16
Enter the name and age:
Meena
19
Enter the name and age:
Sona
32
Enter the subject:
Science
Name:Meena
age:19
Name:Sona
age:32
Subject:Science
```



Program No: 17

Date: 08/12/2025

Program Title : Demonstrate the use of a super keyword

- a. To overcome function overriding
- b. Passing arguments to the base class constructor

```
/* Program to demonstrate the use of a super keyword
```

```
@Meenadevi Ravikumar
```

```
Roll no: 43
```

```
Date: 08/12/2025
```

```
*/
```

```
import java.io.*;  
class A{  
    int a;  
    public A(){  
        a=0;  
    }  
    public A(int a){  
        this.a=a;  
    }  
    void display(){  
        System.out.println("a="+a);  
    }  
}  
class B extends A{  
    int b;  
    public B(){  
        b=0;  
    }  
    public B(int a,int b){  
        super(a);  
        this.b=b;  
    }  
    void display(){  
        super.display();  
        System.out.println("b="+b);  
    }  
}  
class Program17{  
    public static void main(String args[]){  
        B obj=new B(10,20);  
        obj.display();  
    }  
}
```

Output

```
C:\sem2\java>javac P17.java
```

```
C:\sem2\java>java P17  
a=10  
b=20
```



Program No: 18

Date: 08/12/2025

Program Title : Create a class for an employee having eno, ename and esal as data members.

Provide functions for reading, displaying, and comparing by eno. Accept n employees' information in the main function, display the same, and search for an employee whose eno is given.

```
/* Program to Read, Display and Search Employee Records
```

```
@Meenadevi Ravikumar
```

```
Roll no: 43
```

```
Date: 08/12/2025
```

```
*/
```

```
import java.io.*;
```

```
import java.util.*;
```

```
class Employee{
```

```
    int eno;
```

```
    String ename;
```

```
    float esal;
```

```
    public void read(){
```

```
        Scanner s=new Scanner(System.in);
```

```
        System.out.print("Enter eno:");
```

```
        eno=s.nextInt();
```

```
        s.nextLine();
```

```
        System.out.print("Enter name:");
```

```
        ename=s.nextLine();
```

```
        System.out.print("Enter salary:");
```

```
        esal=s.nextFloat();
```

```
}
```

```
    public void display(){
```

```
        System.out.println("eno:"+eno);
```

```
        System.out.println("name:"+ename);
```

```
        System.out.println("salary:"+esal);
```

```
}
```

```
    public int search(int eno){
```

```
        if(this.eno==eno){
```

```
            return 1;
```

```
        }
```

```
        else
```

```
            return 0;
```

```
}
```

```
}
```

```
class Program18{
```

```
    public static void main(String args[]){
```

```
        Employee obj[];
```

```

int n,eno,found=0;
Scanner s=new Scanner(System.in);
System.out.println("Enter the no.of employees:");
n=s.nextInt();
obj = new Employee[n];
for(int i=0;i<n;i++){
    obj[i]=new Employee();
    System.out.println("Enter details of employee "+(i+1)+":");
    obj[i].read();
}
for(int i=0;i<n;i++){
    System.out.println("Details of employee "+(i+1)+":");
    obj[i].display();
}
System.out.println("Enter the eno of employee to be searched:");
eno=s.nextInt();
for(int i=0;i<n;i++){
    found=obj[i].search(eno);
    if(found==1){
        System.out.println("Details of employee searched:");
        obj[i].display();
        break;
    }
}
if(found==0){
    System.out.println("Employee Not Found");
}
}
}

```

Output

```

c:\sem2\java>javac Program18.java
c:\sem2\java>java Program18
Enter the no.of employees:
2
Enter details of employee 1:
Enter eno:1
Enter name:Meena
Enter salary:20000
Enter details of employee 2:
Enter eno:2
Enter name:Sona
Enter salary:30000
Details of employee 1:
eno:1
name:Meena
salary:20000.0
Details of employee 2:
eno:2
name:Sona
salary:30000.0
Enter the eno of employee to be searched:
1
Details of employee searched:
eno:1
name:Meena
salary:20000.0

```

Program No: 19

Date: 10/12/2025

Program Title : Program to implement run-time polymorphism in Java(interface).

```
/* Program to implement run-time polymorphism in Java(interface).
```

```
@Meenadevi Ravikumar
```

```
Roll no: 43
```

```
Date: 10/12/2025
```

```
*/
```

```
interface Animal {
```

```
    void sound();
```

```
}
```

```
class Dog implements Animal {
```

```
    public void sound() {
```

```
        System.out.println("Dog barks");
```

```
}
```

```
}
```

```
class Cat implements Animal {
```

```
    public void sound() {
```

```
        System.out.println("Cat meows");
```

```
}
```

```
}
```

```
public class Program19 {
```

```
    public static void main(String[] args) {
```

```
        Animal a;
```

```
        a = new Dog();
```

```
        a.sound(); // Run-time polymorphism
```

```
        a = new Cat();
```

```
        a.sound(); // Run-time polymorphism
```

```
}
```

```
}
```

Output

```
C:\sem2\java>javac Program19.java
```

```
C:\sem2\java>java Program19
```

```
Dog barks
```

```
Cat meows
```

Program No: 20

Date: 10/12/2025

Program Title : Create an interface ishape having two prototypes draw and mov, create two classes circle and rectangle which implements the above interface. In the main function only a reference object of ishape is created, depends on the choice entered by the user , it points to the respective memory location.

```
/* Program Demonstrating Interface and Runtime Polymorphism
```

```
@Meenadevi Ravikumar
```

```
Roll no: 43
```

```
Date: 10/12/2025
```

```
*/
```

```
import java.util.*;
```

```
interface ishape{
```

```
    public void draw();
```

```
    public void mov();
```

```
}
```

```
class Circle implements ishape{
```

```
    public void draw()
```

```
{
```

```
    System.out.println("Drawing Circle");
```

```
}
```

```
    public void mov()
```

```
{
```

```
    System.out.println("Moving Circle");
```

```
}
```

```
}
```

```
class Rectangle implements ishape{
```

```
    public void draw()
```

```
{
```

```
    System.out.println("Drawing Rectangle");
```

```
}
```

```
    public void mov()
```

```
{
```

```
    System.out.println("Moving Rectangle");
```

```
}
```

```
}
```

```
class Program20{
```

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

```
{
```

```
    ishape obj;
```

```
    int ch;
```

```
    Scanner s=new Scanner(System.in);
```

```
    System.out.println("Choose a Shape:\n1.Circle \n2.Rectangle");
```

```
ch=s.nextInt();
if(ch==1)
{
    obj=new Circle();
}
else if(ch==2)
{
    obj=new Rectangle();
}
else
{
    System.out.println("Invalid Input");
    return;
}
obj.draw();
obj.mov();
}
```

Output

```
C:\sem2\java>javac Program20.java

C:\sem2\java>java Program20
Choose a Shape:
1.Circle
2.Rectangle
1
Drawing Circle
Moving Circle
```

Program No: 21

Date: 10/12/2025

Program Title : Write a program using multiple catch statements / predefined Exceptions.

```
/* Write a program using multiple catch statements or predefined Exceptions.
```

```
@Meenadevi Ravikumar
```

```
Roll no: 43
```

```
Date: 10/12/2025
```

```
*/
```

```
import java.io.DataInputStream;
import java.io.IOException;

class Program21 {
    public static void main(String args[]) {

        DataInputStream dis = new DataInputStream(System.in);

        try {
            System.out.print("Enter first number: ");
            int a = Integer.parseInt(dis.readLine());

            System.out.print("Enter second number: ");
            int b = Integer.parseInt(dis.readLine());

            int c = a / b; // may cause ArithmeticException
            System.out.println("Result: " + c);
        }
        catch (ArithmaticException e) {
            System.out.println("Error: Division by zero");
        }
        catch (NumberFormatException e) {
            System.out.println("Error: Invalid number format");
        }
        catch (IOException e) {
            System.out.println("Error: Input/Output exception");
        }
        catch (Exception e) {
            System.out.println("General exception occurred");
        }
    }
}
```

Output

```
C:\sem2\java>javac Program21.java
Note: Program21.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.

C:\sem2\java>java Program21
Enter first number: 3
Enter second number: 0
Error: Division by zero
```



Program No: 22

Date: 15/12/2025

Program Title : Write a program to implement a user-defined Exception.

```
/*Program to implement a user-defined Exception.
```

```
@Meenadevi Ravikumar
```

```
Roll no: 43
```

```
Date: 15/12/2025
```

```
*/
```

```
class UDE extends Exception
```

```
{
```

```
    String strError;
```

```
    public UDE()
```

```
    {
```

```
        strError="Unknown Error";
```

```
    }
```

```
    public UDE(String s)
```

```
    {
```

```
        strError=s;
```

```
    }
```

```
    public String showError()
```

```
    {
```

```
        return strError;
```

```
    }
```

```
}
```

```
class Program22
```

```
{
```

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

```
    {
```

```
        java.util.Scanner s = new java.util.Scanner(System.in);
```

```
        System.out.print("Enter value for a: ");
```

```
        int a = s.nextInt();
```

```
        System.out.print("Enter value for b: ");
```

```
        int b = s.nextInt();
```

```
        int c;
```

```
        try
```

```
        {
```

```
            if(b==0)
```

```
                throw new UDE("Dr is zero");
```

```
            c=a/b;
```

```
            System.out.println(c);
```

```
        }
```

```
        catch(UDE e)
```

```
        {
```

```
            System.out.println(e.showError());
```

```
        }
```



```
}
```

```
Output
```

```
C:\sem2\java>javac Program22.java  
C:\sem2\java>java Program22  
Enter value for a: 6  
Enter value for b: 0  
Dr is zero
```



Program No: 23

Date: 15/12/2025

Program Title : Write a program to implement re-throw and finally.

```
/*Program to implement re-throw and finally.  
@Meenadevi Ravikumar  
Roll no: 43  
Date: 15/12/2025  
*/  
class Program23 {  
  
    static void checkNumber(int num) {  
        try {  
            if (num < 0) {  
                throw new ArithmeticException("Negative number not allowed");  
            }  
            System.out.println("Number is valid: " + num);  
        } catch (ArithmaticException e) {  
            System.out.println("Caught exception in checkNumber(): " + e.getMessage());  
            // Re-throwing the exception  
            throw e;  
        } finally {  
            System.out.println("Finally block in checkNumber() executed");  
        }  
    }  
  
    public static void main(String[] args) {  
        try {  
            checkNumber(-5);  
        } catch (ArithmaticException e) {  
            System.out.println("Caught exception in main(): " + e.getMessage());  
        } finally {  
            System.out.println("Finally block in main() executed");  
        }  
    }  
}
```

Output

```
C:\sem2\java>javac Program23.java  
C:\sem2\java>java Program23  
Caught exception in checkNumber(): Negative number not allowed  
Finally block in checkNumber() executed  
Caught exception in main(): Negative number not allowed  
Finally block in main() executed
```

Program No: 24

Date: 15/12/2025

Program Title : Program to Validate Password Using Custom Exception Handling

```
/* Program to Validate Password Using Custom Exception Handling
@Meenadevi Ravikumar
Roll no: 43
Date: 15/12/2025
*/
import java.io.*;
import java.util.*;
class UDE extends Exception
{
    String error;
    public UDE()
    {
        error="Invalid Password";
    }
    public UDE(String s)
    {
        error=s;
    }
    public String showError()
    {
        return error;
    }
}
class Program24
{
    public static void main(String args[])
    {
        String p;
        int scount=0,upper=0,lower=0;
        char prev =' ';
        Scanner s=new Scanner(System.in);
        System.out.println("Rules:\n1.Must be alphanumeric\n2.Atleast one uppercase and one
lowercase letter\n");
        System.out.println("3.Atleast one Special Character( @,*,(,),{},[],= )\n4.Should not have
any consecutive repetition of characters");
        System.out.println("\n5.Atleast 8 characters");
        System.out.print("\nEnter the password:");
        p=s.nextLine();
        try
        {
            for(int i=0;i<p.length();i++)
            {
```

```
if(!Character.isLetter(p.charAt(i)) && !Character.isDigit(p.charAt(i)))
{
    switch(p.charAt(i))
    {
        case '@':
        case '*':
        case '!':
        case '(':
        case ')':
        case '{':
        case '}':
        case '[':
        case ']':
        case '=':scount++;break;
        default:throw new UDE("Password Must be Alphanumeric");
    }
}
if(Character.isUpperCase(p.charAt(i)))
    upper=1;
if(Character.isLowerCase(p.charAt(i)))
    lower=1;
if(i==0)
    prev=p.charAt(i);
else
{
    if(p.charAt(i)==prev)
    {
        throw new UDE("Should not have any consecutive repetition of characters");
    }
    else
    {
        prev=p.charAt(i);
    }
}
if(lower==0)
    throw new UDE("Should have atleast one lowercase letter");
if(upper==0)
    throw new UDE("Should have atleast one Uppercase letter");
if(scount==0)
    throw new UDE("Should have atleast one Special Character");
if(p.length()<8)
    throw new UDE("Atleast 8 characters needed");
else
{
    System.out.println("Valid Password");
}
```

```
        }
    catch(UDE e)
    {
        System.out.println(e.showError());
    }

}
}
```

Output

```
C:\sem2\java>java Program24
Rules:
1.Must be alphanumeric
2.Atleast one uppercase and one lowercase letter

3.Atleast one Special Character( @,*,(,),{},[],= )
4.Should not have any consecutive repetition of characters

5.Atleast 8 characters

Enter the password:ABCDe@12345
Valid Password
```



Program No: 25

Date: 17/12/2025

Program Title : Implement threading by inheriting Thread class.

```
/* Program to implement threading by inheriting Thread class.
```

```
@Meenadevi Ravikumar
```

```
Roll no: 43
```

```
Date: 17/12/2025
```

```
*/
```

```
class Cthread extends Thread
```

```
{
```

```
    int n;
```

```
    public Cthread(int a)
```

```
{
```

```
    n=a;
```

```
}
```

```
    public void run()
```

```
{
```

```
        int i;
```

```
        for(i=0;i<=n;i++)
```

```
{
```

```
            System.out.println("i="+i+"\t");
```

```
}
```

```
}
```

```
}
```

```
class Program25
```

```
{
```

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

```
{
```

```
    Cthread t1,t2;
```

```
    t1=new Cthread(5);
```

```
    t2=new Cthread(7);
```

```
    System.out.println("B4 Threading");
```

```
    t1.start();
```

```
    System.out.println("BW Threading");
```

```
    t2.start();
```

```
    System.out.println("Finished");
```

```
}
```

```
}
```

Output

```
C:\sem2\java>javac P25.java  
C:\sem2\java>java P25  
B4 Threading  
BW Threading  
Finished  
i=0  
i=1  
i=0  
i=1  
i=2  
i=3  
i=4  
i=5  
i=6  
i=7  
i=2  
i=3  
i=4  
i=5
```



Program No: 26

Date: 17/12/2025

Program Title : Write a Java program to create threads using the Runnable interface.

```
/* Program to create threads using the Runnable interface.
```

```
@Meenadevi Ravikumar
```

```
Roll no: 43
```

```
Date: 17/12/2025
```

```
*/
```

```
import java.io.*;
```

```
class CThread implements Runnable
```

```
{
```

```
    int n;
```

```
    public CThread(int a)
```

```
    {
```

```
        n=a;
```

```
    }
```

```
    public void run()
```

```
    {
```

```
        int i;
```

```
        for(i=0;i<=n;i++)
```

```
        {
```

```
            System.out.println("i="+i+"\t");
```

```
        }
```

```
    }
```

```
}
```

```
class Program26
```

```
{
```

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

```
    {
```

```
        CThread p,q;
```

```
        p=new CThread(5);
```

```
        q=new CThread(7);
```

```
        Thread t1,t2;
```

```
        t1=new Thread(p);
```

```
        t2=new Thread(q);
```

```
        System.out.println("B4 Threading");
```

```
        t1.start();
```

```
        System.out.println("BW Threading");
```

```
        t2.start();
```

```
        System.out.println("Finished");
```

```
    }
```

```
}
```



Output

```
C:\sem2\java>javac P26.java  
C:\sem2\java>java P26  
B4 Threading  
BW Threading  
Finished  
i=0  
i=0  
i=1  
i=1  
i=2  
i=3  
i=2  
i=3  
i=4  
i=5  
i=6  
i=7  
i=4  
i=5
```



Program No: 27

Date: 17/12/2025

Program Title : Write a Java program to create two threads, one to display odd numbers and another to display even numbers up to N.

```
/* Program to create two threads, one to display odd numbers and another to display even numbers up to N.
```

```
@Meenadevi Ravikumar
```

```
Roll no: 43
```

```
Date: 17/12/2025
```

```
*/
```

```
import java.util.Scanner;  
class Cthread extends Thread  
{
```

```
    int start,n;
```

```
    public Cthread(int s,int a)
```

```
    {
```

```
        start=s;
```

```
        n=a;
```

```
    }
```

```
    public void run()
```

```
    {
```

```
        int i;
```

```
        for(i=start;i<=n;i=i+2)
```

```
        {
```

```
            if(i%2==0)
```

```
                System.out.println("Even="+i);
```

```
            else
```

```
                System.out.println("Odd="+i);
```

```
        }
```

```
    }
```

```
}
```

```
class Program27
```

```
{
```

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

```
    {
```

```
        Cthread t1,t2;
```

```
        int n;
```

```
        Scanner s=new Scanner(System.in);
```

```
        System.out.println("Enter the limit:");
```

```
        n=s.nextInt();
```

```
        t1=new Cthread(1,n);
```

```
        t2=new Cthread(0,n);
```

```
        t1.start();
```

```
        t2.start();
```

```
    }
```

```
}
```

Output

```
C:\sem2\java>javac P27.java  
C:\sem2\java>java P27  
Enter the limit:  
4  
Odd=1  
Odd=3  
Even=0  
Even=2  
Even=4
```



Program No: 28

Date: 20/12/2025

Program Title : Search for a given element using n/5 threads.

```
/* Program to search for a given element using n/5 threads.
```

```
@Meenadevi Ravikumar
```

```
Roll no: 43
```

```
Date: 20/12/2025
```

```
*/
```

```
import java.util.*;
```

```
class SearchThread extends Thread {
```

```
    int arr[];
```

```
    int start, end, key;
```

```
    SearchThread(int arr[], int start, int end, int key) {
```

```
        this.arr = arr;
```

```
        this.start = start;
```

```
        this.end = end;
```

```
        this.key = key;
```

```
}
```

```
    public void run() {
```

```
        for (int i = start; i < end; i++) {
```

```
            if (arr[i] == key) {
```

```
                System.out.println("Element found at index: " + i);
```

```
            }
```

```
        }
```

```
}
```

```
}
```

```
class Program28 {
```

```
    public static void main(String args[]) {
```

```
        int index;
```

```
        Scanner s = new Scanner(System.in);
```

```
        System.out.print("Enter number of elements: ");
```

```
        int n = s.nextInt();
```

```
        int arr[] = new int[n];
```

```
        System.out.println("Enter elements:");
```

```
        for (int i = 0; i < n; i++) {
```

```
            arr[i] = s.nextInt();
```

```
}
```

```
System.out.print("Enter element to search: ");
int key = s.nextInt();

int threads = n / 5;

if (threads == 0) {
    threads = 1;
}

for (int i = 0; i < threads; i++) {
    int start = i * 5;
    int end;

    if (i == threads - 1) {
        end = n;
    }
    else {
        end = start + 5;
    }

    SearchThread t = new SearchThread(arr, start, end, key);
    t.start();
}
}
```

Output

```
C:\sem2\java>javac P28.java

C:\sem2\java>java P28
Enter number of elements: 5
Enter elements:
1
5
3
9
3
Enter element to search: 3
Element found at index: 2
Element found at index: 4
```

Program No: 29

Date: 20/12/2025

Program Title : Develop a program to compute square and cube of a given number using two separate threads.

```
/* Program to compute square and cube of a given number using two
separate threads.
@Meenadevi Ravikumar
Roll no: 43
Date: 20/12/2025
*/
import java.io.*;
import java.util.*;
class SquareThread extends Thread {
    int n;

    SquareThread(int n) {
        this.n = n;
    }

    public void run() {
        System.out.println("Square of " + n + " = " + (n * n));
    }
}

class CubeThread extends Thread {
    int n;

    CubeThread(int n) {
        this.n = n;
    }

    public void run() {
        System.out.println("Cube of " + n + " = " + (n * n * n));
    }
}

class Program29 {
    public static void main(String args[]) {
        Scanner s = new Scanner(System.in);

        System.out.print("Enter a number: ");
        int n = s.nextInt();

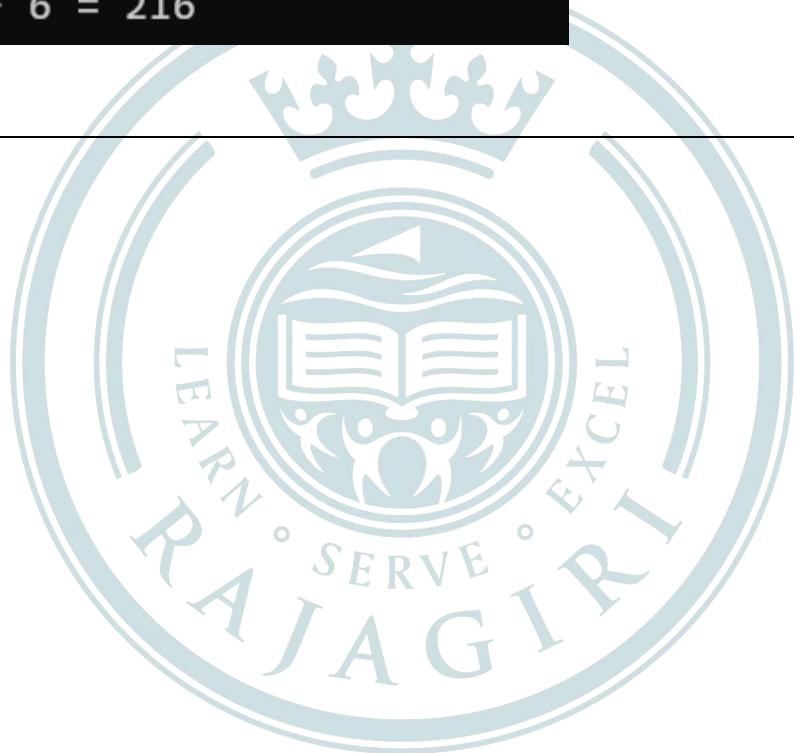
        SquareThread t1 = new SquareThread(n);
        CubeThread t2 = new CubeThread(n);
```

```
    t1.start();
    t2.start();
}
}
```

Output

```
C:\sem2\java>javac P29.java

C:\sem2\java>java P29
Enter a number: 6
Square of 6 = 36
Cube of 6 = 216
```



Program No: 30

Date: 20/12/2025

Program Title : Implement Printer Queue Simulation without synchronization but using multiple threads.

```
/* Program to implement Printer Queue Simulation without synchronization but using
multiple threads.

@Meenadevi Ravikumar
Roll no: 43
Date: 20/12/2025
*/
class PrinterThread extends Thread {
    String documentName;
    int pages;

    PrinterThread(String documentName, int pages) {
        this.documentName = documentName;
        this.pages = pages;
    }

    public void run() {
        System.out.println("Printing started: " + documentName + " (" + pages + " pages)");
        for (int i = 1; i <= pages; i++) {
            System.out.println(documentName + " - Printing page " + i);
            try {
                Thread.sleep(500);
            } catch (Exception e) {
                System.out.println("Printing interrupted: " + documentName);
            }
        }
        System.out.println("Printing finished: " + documentName);
    }
}

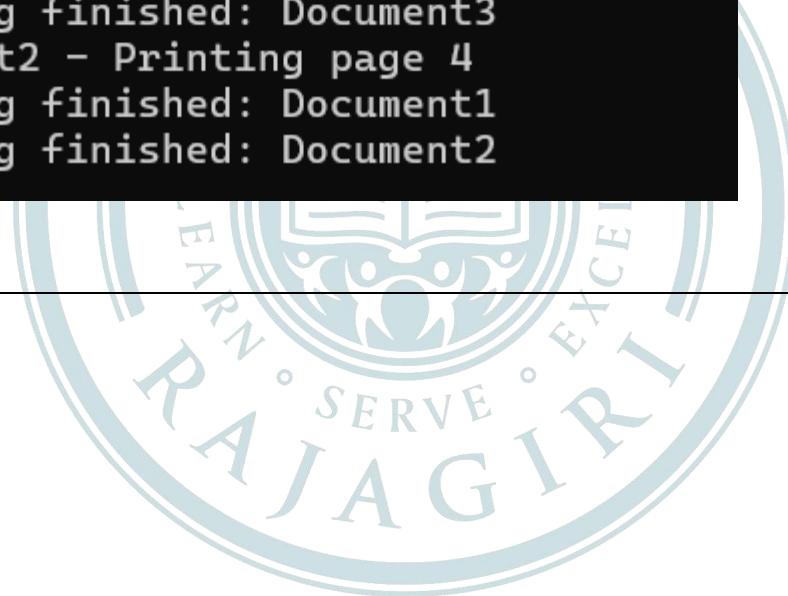
class Program30 {
    public static void main(String args[]) {
        // Simulate multiple print jobs
        PrinterThread job1 = new PrinterThread("Document1", 3);
        PrinterThread job2 = new PrinterThread("Document2", 4);
        PrinterThread job3 = new PrinterThread("Document3", 2);

        // Start all threads (without synchronization)
        job1.start();
        job2.start();
        job3.start();
    }
}
```

Output

```
C:\sem2\java>javac P30.java

C:\sem2\java>java P30
Printing started: Document2 (4 pages)
Printing started: Document3 (2 pages)
Printing started: Document1 (3 pages)
Document1 - Printing page 1
Document2 - Printing page 1
Document3 - Printing page 1
Document1 - Printing page 2
Document2 - Printing page 2
Document3 - Printing page 2
Document2 - Printing page 3
Document1 - Printing page 3
Printing finished: Document3
Document2 - Printing page 4
Printing finished: Document1
Printing finished: Document2
```



Program No: 31	Date: 22/12/2025
Program Title : Implement Printer Queue simulation using multithreading and synchronization	

```
/* Program to implement Printer Queue simulation using multithreading and synchronization
@Meenadevi Ravikumar
Roll no: 43
Date: 22/12/2025
*/
class Printer {
    synchronized void print(String documentName, int pages) {
        System.out.println("Printing started: " + documentName);
        for (int i = 1; i <= pages; i++) {
            System.out.println(documentName + " - Printing page " + i);
            try {
                Thread.sleep(500);
            } catch (Exception e) {
            }
        }
        System.out.println("Printing finished: " + documentName + "\n");
    }
}
class PrinterThread extends Thread {
    Printer printer;
    String documentName;
    int pages;

    PrinterThread(Printer printer, String documentName, int pages) {
        this.printer = printer;
        this.documentName = documentName;
        this.pages = pages;
    }

    public void run() {
        printer.print(documentName, pages);
    }
}
class Program31 {
    public static void main(String args[]) {

        Printer printer = new Printer();

        PrinterThread job1 = new PrinterThread(printer, "Document1", 3);
        PrinterThread job2 = new PrinterThread(printer, "Document2", 4);
        PrinterThread job3 = new PrinterThread(printer, "Document3", 2);
```

```
        job1.start();
        job2.start();
        job3.start();
    }
}
```

Output

```
C:\sem2\java>javac Program31.java

C:\sem2\java>java Program31
Printing started: Document1
Document1 - Printing page 1
Document1 - Printing page 2
Document1 - Printing page 3
Printing finished: Document1

Printing started: Document2
Document2 - Printing page 1
Document2 - Printing page 2
Document2 - Printing page 3
Document2 - Printing page 4
Printing finished: Document2

Printing started: Document3
Document3 - Printing page 1
Document3 - Printing page 2
Printing finished: Document3
```

Program No: 32

Date: 22/12/2025

Program Title : Write a Java program to demonstrate inter-thread communication using `wait()`, `notify()`, and `notifyAll()` by implementing

- a. Producer consumer problem
- b. Readers writer's problem

```
/* Program to demonstrate inter-thread communication
```

```
@Meenadevi Ravikumar
```

```
Roll no: 43
```

```
Date: 22/12/2025
```

```
*/
```

```
import java.io.*;
```

```
class Book
```

```
{
```

```
    int data;  
    boolean f=false;
```

```
    public synchronized void read(int a)
```

```
{
```

```
    try{  
        if(f==true){  
            wait();  
        }  
        data=a;  
        System.out.println("read:"+data);  
        f=true;  
        notify();  
    }
```

```
    catch(Exception e){}
```

```
}
```

```
    public synchronized void write(){
```

```
        try{  
            if(f==false)  
                wait();  
            System.out.println("Write"+data);  
            f=false;  
            notify();  
        }
```

```
        catch(Exception e){}
```

```
}
```

```
}
```

```
class Writer extends Thread{  
    Book o;
```

```
public Writer(Book o){  
    this.o=o;  
}  
public void run(){  
    int i=1;  
    while(true){  
        o.read(i);  
        i++;  
    }  
}  
}  
  
class Reader extends Thread{  
    Book o;  
    public Reader(Book o){  
        this.o=o;  
    }  
    public void run(){  
        while(true){  
            o.write();  
        }  
    }  
}  
  
class ProducerConsumer {  
    int item;  
    boolean available = false;  
  
    public synchronized void produce(int value) {  
        try {  
            if (available) {  
                wait();  
            }  
            item = value;  
            System.out.println("Produced: " + item);  
            available = true;  
            notifyAll();  
        } catch (Exception e) {  
            e.printStackTrace();  
        }  
    }  
  
    public synchronized void consume() {  
        try {  
            if (!available) {  
                wait();  
            }  
            System.out.println("Consumed: " + item);  
        } catch (Exception e) {  
            e.printStackTrace();  
        }  
    }  
}
```



```

        available = false;
        notifyAll();
    } catch (Exception e) {
        e.printStackTrace();
    }
}

class Producer extends Thread {
    ProducerConsumer pc;

    public Producer(ProducerConsumer pc) {
        this.pc = pc;
    }

    public void run() {
        int value = 1;
        while (true) {
            pc.produce(value);
            value++;
        }
    }
}

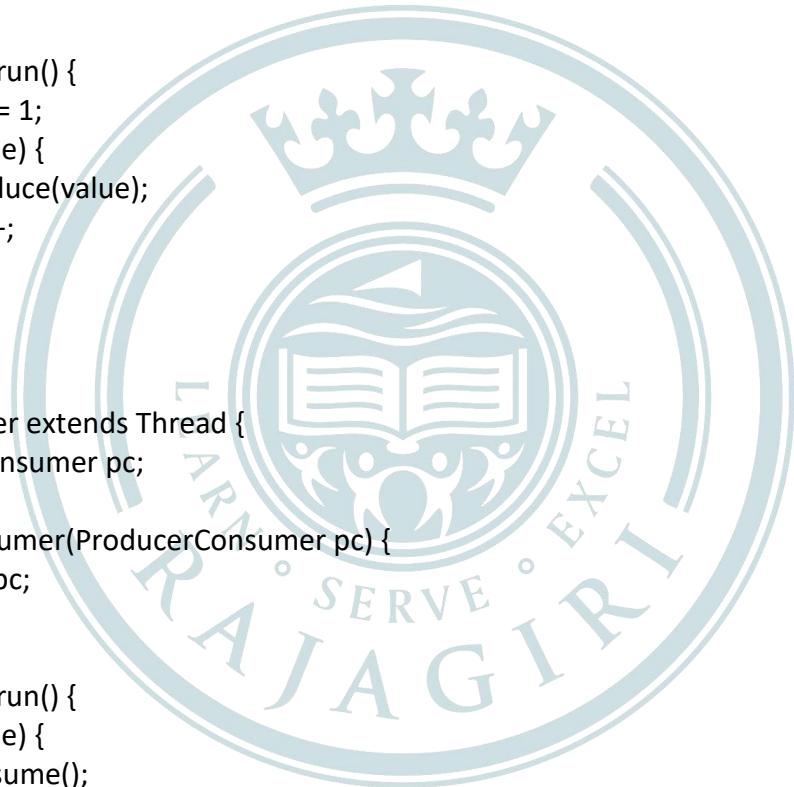
class Consumer extends Thread {
    ProducerConsumer pc;

    public Consumer(ProducerConsumer pc) {
        this.pc = pc;
    }

    public void run() {
        while (true) {
            pc.consume();
        }
    }
}

class Program32 {
    public static void main(String args[]) {
        Book q=new Book();
        Writer p=new Writer(q);
        Reader r=new Reader(q);
        p.start();
        r.start();
        ProducerConsumer pc = new ProducerConsumer();
        Producer producer = new Producer(pc);
    }
}

```



```
        Consumer consumer = new Consumer(pc);
        producer.start();
        consumer.start();
    }
}
```

Output

```
C:\sem2\java>Write7348
Consumed: 6782
read:7349
Produced: 6783
Write7349
Consumed: 6783
Produced: 6784
read:7350
Write7350
Consumed: 6784
Produced: 6785
read:7351
Consumed: 6785
Write7351
Produced: 6786
read:7352
Consumed: 6786
Produced: 6787
Write7352
Consumed: 6787
read:7353
Produced: 6788
Write7353
```

Program No: 33

Date: 22/12/2025

Program Title : Write a Java program to simulate bank deposit and withdrawal using threads for a single account without synchronization.

```
/* Program to simulate bank deposit and withdrawal using threads for a single account
without synchronization.
@Meenadevi Ravikumar
Roll no: 43
Date: 22/12/2025
*/
class BankAccount {
    int balance = 1000; // Initial balance

    public void deposit(int amount) {
        balance += amount;
        System.out.println("Deposited: " + amount + ", Current Balance: " + balance);
    }

    public void withdraw(int amount) {
        if (balance >= amount) {
            balance -= amount;
            System.out.println("Withdrawn: " + amount + ", Current Balance: " + balance);
        } else {
            System.out.println("Insufficient balance for withdrawal of: " + amount);
        }
    }
}

class DepositThread extends Thread {
    BankAccount account;
    int amount;

    public DepositThread(BankAccount account, int amount) {
        this.account = account;
        this.amount = amount;
    }

    public void run() {
        account.deposit(amount);
    }
}

class WithdrawThread extends Thread {
    BankAccount account;
    int amount;
```

```
public WithdrawThread(BankAccount account, int amount) {  
    this.account = account;  
    this.amount = amount;  
}  
  
public void run() {  
    account.withdraw(amount);  
}  
}  
  
public class Program33 {  
    public static void main(String[] args) {  
        BankAccount account = new BankAccount();  
  
        DepositThread deposit1 = new DepositThread(account, 500);  
        WithdrawThread withdraw1 = new WithdrawThread(account, 700);  
        DepositThread deposit2 = new DepositThread(account, 300);  
        WithdrawThread withdraw2 = new WithdrawThread(account, 900);  
  
        deposit1.start();  
        withdraw1.start();  
        deposit2.start();  
        withdraw2.start();  
    }  
}
```

Output

```
C:\sem2\java>javac Program33.java  
  
C:\sem2\java>java Program33  
Withdrawn: 700, Current Balance: 800  
Deposited: 500, Current Balance: 1500  
Withdrawn: 900, Current Balance: 200  
Deposited: 300, Current Balance: 1100
```

Program No: 34

Date: 24/12/2025

Program Title : Implement thread priority.

```
/* Program to implement thread priority.  
@Meenadevi Ravikumar  
Roll no: 43  
Date: 24/12/2025  
*/  
class PriorityThread extends Thread {  
    private String threadName;  
  
    public PriorityThread(String name) {  
        this.threadName = name;  
    }  
  
    public void run() {  
        for (int i = 1; i <= 5; i++) {  
            System.out.println(threadName + " - Priority: " + getPriority() + " - Count: " + i);  
        }  
    }  
  
    public String getThreadName() {  
        return threadName;  
    }  
  
    public void setThreadName(String name) {  
        this.threadName = name;  
    }  
}  
  
public class Program34 {  
    public static void main(String[] args) {  
        PriorityThread highPriorityThread = new PriorityThread("High Priority Thread");  
        PriorityThread lowPriorityThread = new PriorityThread("Low Priority Thread");  
        PriorityThread normalPriorityThread = new PriorityThread("Normal Priority Thread");  
  
        highPriorityThread.setPriority(Thread.MAX_PRIORITY);  
        lowPriorityThread.setPriority(Thread.MIN_PRIORITY);  
        normalPriorityThread.setPriority(Thread.NORM_PRIORITY);  
  
        highPriorityThread.start();  
        lowPriorityThread.start();  
        normalPriorityThread.start();  
    }  
}
```

Output

```
C:\sem2\java>javac Program34.java
```

```
C:\sem2\java>java Program34
```

```
Normal Priority Thread - Priority: 5 - Count: 1
Normal Priority Thread - Priority: 5 - Count: 2
Normal Priority Thread - Priority: 5 - Count: 3
Normal Priority Thread - Priority: 5 - Count: 4
Normal Priority Thread - Priority: 5 - Count: 5
High Priority Thread - Priority: 10 - Count: 1
Low Priority Thread - Priority: 1 - Count: 1
Low Priority Thread - Priority: 1 - Count: 2
Low Priority Thread - Priority: 1 - Count: 3
Low Priority Thread - Priority: 1 - Count: 4
High Priority Thread - Priority: 10 - Count: 2
High Priority Thread - Priority: 10 - Count: 3
High Priority Thread - Priority: 10 - Count: 4
High Priority Thread - Priority: 10 - Count: 5
Low Priority Thread - Priority: 1 - Count: 5
```



Program No: 43

Date: 24/12/2025

Program Title : Develop a Java program to simulate ATM operations with multiple users accessing the same account

```
/* Program to simulate ATM operations with multiple users accessing the same account

@Meenadevi Ravikumar
Roll no: 43
Date: 24/12/2025
*/
class Bank {
    int balance = 0;

    public synchronized void deposit(int amount) {
        balance += amount;
        System.out.println("Deposited: " + amount + " | Balance: " + balance);
        notifyAll();
    }

    public synchronized void withdraw(int amount) {
        try {
            while (balance < amount) {
                System.out.println("Waiting for deposit...");
                wait(3000);

                if (balance < amount) {
                    System.out.println("Transaction failed: Insufficient balance");
                    return;
                }
            }
            balance -= amount;
            System.out.println("Withdrawn: " + amount + " | Balance: " + balance);
        } catch (Exception e) {
        }
    }
}

class ATM extends Thread {
    Bank bank;
    boolean isDeposit;
    int amount;

    ATM(Bank bank, boolean isDeposit, int amount) {
        this.bank = bank;
        this.isDeposit = isDeposit;
        this.amount = amount;
    }
}
```

```
public void run() {
    if (isDeposit)
        bank.deposit(amount);
    else
        bank.withdraw(amount);
}
}

public class Program43 {
    public static void main(String args[]) {

        Bank bank = new Bank();

        ATM t1 = new ATM(bank, false, 500);
        ATM t2 = new ATM(bank, true, 1000);
        ATM t3 = new ATM(bank, false, 300);

        t1.start();
        t2.start();
        t3.start();
    }
}
```

Output

```
C:\Users\jubin\OneDrive\Desktop\Rajagiri Docs\Sem2\MCA-JAVA-S1\Record Programs>javac Program35.java
C:\Users\jubin\OneDrive\Desktop\Rajagiri Docs\Sem2\MCA-JAVA-S1\Record Programs>java Program35
Waiting for deposit...
Waiting for deposit...
Deposited: 1000 | Balance: 1000
Withdrawn: 300 | Balance: 700
Withdrawn: 500 | Balance: 200
```

Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	



Program No:	Date:
Program Title :	
/* @Meenadevi Ravikumar Roll no: 43 Date: */	
Output	

