

Name: VIKASH YADAV

Scholar No. 222120025

MCA 3rd

Sem

Java Programming Assignment

1. WAP to calculate Area Of Circle

```
import java.util.Scanner;

public class CircleAreaCalculator

{ public static void main(String[] args)

{

    Scanner scanner = new Scanner(System.in);

    System.out.print("Enter the radius of the circle: ");

    double radius = scanner.nextDouble();

    double area = Math.PI * Math.pow(radius, 2);

    System.out.println("The area of the circle with radius " + radius + " is: " + area);

}

}
```

2. WAP to calculate Area Of Triangle

```
import java.util.Scanner;

public class TriangleAreaCalculator

{ public static void main(String[] args)

{

    Scanner scanner = new Scanner(System.in);

    System.out.println("Enter the length of the base of the triangle: ");

    double base = scanner.nextDouble();

    System.out.println("Enter the height of the triangle: ");
```

```

        double height = scanner.nextDouble();

        double area = (0.5) * base * height;

        System.out.println("The area of the triangle is: " + area);
    }
}

```

3. WAP to calculate Area Of Rectangle

```

import java.util.Scanner;

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

        System.out.print("Enter the length of the rectangle: ");

        double length = scanner.nextDouble();

        // Prompt the user to enter the width of the rectangle
        System.out.print("Enter the width of the rectangle: ");

        double width = scanner.nextDouble();

        double area = length * width;

        System.out.println("The area of the rectangle is: " + area);
    }
}

```

4. WAP to calculate Area Of Isosceles Triangle

```

import java.util.Scanner;

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

        System.out.print("Enter the length of the equal sides (a): ");
    }
}

```

```

double a = input.nextDouble();

System.out.print("Enter the length of the base (b): ");

double b = input.nextDouble();

double s = (a + a + b) / 2;

double area = Math.sqrt(s * (s - a) * (s - a) * (s - b));

System.out.println("The area of the isosceles triangle is: " + area);

}

}

```

5. WAP to calculate Area Of Parallelogram

```

import java.util.Scanner;

public class ParallelogramAreaCalculator

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

    Scanner scanner = new Scanner(System.in);

    System.out.print("Enter the base length of the parallelogram: ");

    double base = scanner.nextDouble();

    System.out.print("Enter the height of the parallelogram: ");

    double height = scanner.nextDouble();

    double area = base * height;

    System.out.println("The area of the parallelogram is: " + area);

}

}

```

6. WAP to calculate Area Of Rhombus

```

import java.util.Scanner;

public class RhombusAreaCalculator

{ public static void main(String[] args)

{

```

```

Scanner scanner = new Scanner(System.in);

System.out.println("Enter the length of the first diagonal (d1): ");

double d1 = scanner.nextDouble();

System.out.println("Enter the length of the second diagonal (d2): ");

double d2 = scanner.nextDouble();

double area = (d1 * d2) / 2;

System.out.println("The area of the rhombus is: " + area);

}

}

```

7. WAP to calculate Area Of Equilateral Triangle

```

import java.util.Scanner;

import java.lang.Math;

public class EquilateralTriangleArea

{ public static void main(String[] args)

{

Scanner scanner = new Scanner(System.in);

System.out.println("Enter the side length of the equilateral triangle: ");

double side = scanner.nextDouble();

double area = (side * side * Math.sqrt(3)) / 4.0;

System.out.println("The area of the equilateral triangle is: " + area);

}

}

```

8. WAP to calculate Perimeter Of Circle

```

import java.util.Scanner;

public class CirclePerimeterCalculator

{ public static void main(String[] args)

{

```

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

System.out.print("Enter the radius of the circle: ");

double radius = scanner.nextDouble();

double perimeter = 2 * Math.PI * radius;

System.out.println("The perimeter of the circle is: " + perimeter);

}

}
```

9. WAP to calculate Perimeter Of Equilateral Triangle

```
import java.util.Scanner;

public class EquilateralTrianglePerimeter

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

    Scanner input = new Scanner(System.in);

    System.out.print("Enter the length of one side of the equilateral triangle: ");

    double sideLength = input.nextDouble();

    double perimeter = 3 * sideLength;

    System.out.println("The perimeter of the equilateral triangle is: " + perimeter);

}

}
```

10. WAP to calculate Perimeter Of Parallelogram

```
import java.util.Scanner;

public class ParallelogramPerimeterCalculator

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

    Scanner scanner = new Scanner(System.in);

    System.out.print("Enter the length of the base of the parallelogram: ");

    double base = scanner.nextDouble();
```

```

        System.out.print("Enter the length of one of the adjacent sides of the parallelogram: ");

        double side = scanner.nextDouble();

        double perimeter = calculateParallelogramPerimeter(base, side);

        System.out.println("The perimeter of the parallelogram is: " + perimeter);
    }

    public static double calculateParallelogramPerimeter(double base, double side)

    { return 2 * (base + side);

    }

}

```

11. WAP to calculate Perimeter Of Rectangle

```

import java.util.Scanner;

public class RectanglePerimeter

{ public static void main(String[] args)

{

    Scanner scanner = new Scanner(System.in);

    System.out.print("Enter the length of the rectangle: ");

    double length = scanner.nextDouble();

    System.out.print("Enter the width of the rectangle: ");

    double width = scanner.nextDouble();

    double perimeter = 2 * (length + width);

    System.out.println("The perimeter of the rectangle is: " + perimeter);

}

}

```

12. WAP to calculate Perimeter Of Square

```

import java.util.Scanner;

public class PerimeterOfSquare {

```

```

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

    System.out.print("Enter the side length of the square: ");

    double sideLength = scanner.nextDouble();

    double perimeter = 4 * sideLength;

    System.out.println("The perimeter of the square is: " + perimeter);

}
}

```

13. WAP to calculate Perimeter Of Rhombus

```

import java.util.Scanner;

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

        System.out.print("Enter the length of one side of the rhombus: ");

        double sideLength = scanner.nextDouble();

        double perimeter = 4 * sideLength;

        System.out.println("The perimeter of the rhombus is: " + perimeter);

    }
}

```

14. WAP to calculate Volume Of Cone

```

import java.util.Scanner;

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

        System.out.print("Enter the radius of the cone's base (in meters): ");
    }
}

```

```

double radius = scanner.nextDouble();

System.out.print("Enter the height of the cone (in meters): ");

double height = scanner.nextDouble();

double volume = (1.0 / 3.0) * Math.PI * Math.pow(radius, 2) * height;

System.out.println("The volume of the cone is: " + volume + " cubic meters");

}

}

```

15. WAP to calculate Volume Of Prism

```

import java.util.Scanner;

public class PrismVolumeCalculator

{ public static void main(String[] args)

{

Scanner scanner = new Scanner(System.in);

System.out.print("Enter the base area of the prism: ");

double baseArea = scanner.nextDouble();

System.out.print("Enter the height of the prism: ");

double height = scanner.nextDouble();

double volume = baseArea * height;

System.out.println("The volume of the prism is: " + volume);

}

}

```

16. WAP to calculate Volume Of Cylinder

```

import java.util.Scanner;

public class CylinderVolumeCalculator

{ public static void main(String[] args)

{

System.out.print("Enter the radius of the cylinder: ");

```



```

double radius = scanner.nextDouble();

System.out.print("Enter the height of the cylinder: ");

double height = scanner.nextDouble();

double volume = calculateCylinderVolume(radius, height);

System.out.println("The volume of the cylinder is: " + volume);
}

public static double calculateCylinderVolume(double radius, double height)

{ double pi = 3.14159;

double volume = pi * Math.pow(radius, 2) * height;

return volume;

}

}

```

17. WAP to calculate Volume Of Sphere

```

import java.util.Scanner;

public class SphereVolumeCalculator

{ public static void main(String[] args)

{

Scanner input = new Scanner(System.in);

System.out.print("Enter the radius of the sphere: ");

double radius = input.nextDouble();

double volume = (4.0 / 3.0) * Math.PI * Math.pow(radius, 3);

System.out.println("The volume of the sphere with radius " + radius + " is: " + volume);

}

}

```

18. WAP to print Fibonacci Series

```

import java.util.Scanner;

```

```

public class FibonacciSeries {

    public static void main(String[] args)

    { Scanner scanner = new

    Scanner(System.in);

    System.out.print("Enter the number of terms for the Fibonacci series: ");

    int n = scanner.nextInt();

    if (n <= 0) {

        System.out.println("Invalid input. Please enter a positive integer.");

    }

    int first = 0, second = 1;

    System.out.print("Fibonacci Series up to " + n + " terms: ");

    System.out.print(first + " " + second + " ");

    for (int i = 2; i < n; i++)

    { int next = first +

    second;

    System.out.print(next + " ");

    first = second;

    second = next;

    }

    }

}

```

19. WAP to calculate Factorial of given Number

```

import java.util.Scanner;

public class FactorialCalculator {

    public static void main(String[] args)

    { Scanner scanner = new

    Scanner(System.in);

    System.out.print("Enter a number: ");

```

```

        int number = scanner.nextInt();

        long factorial = calculateFactorial(number);

        System.out.println("Factorial of " + number + " is: " + factorial);
    }

    public static long calculateFactorial(int
        n){ long factorial = 1;

        for (int i = 1; i <= n; i++)

            { factorial *= i;

            }

        return factorial;

    }

}

```

20. WAP to calculate Average Of N Numbers

```

import java.util.Scanner;

public class A20averageN {

    public static void main(String[] args) {

        // TODO Auto-generated method stub

        Scanner input = new Scanner(System.in);

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

        int n = input.nextInt();

        double sum = 0;
    }
}

```

```

for (int i = 1; i <= n; i++)

{ System.out.print("Enter number " + i + ": ");

    double num = input.nextDouble();

    sum += num;

}

double average = sum / n;

System.out.println("The average is: " + average);

}

}

```

21. WAP to calculate Discount Of Product

```

import java.util.Scanner;

public class A21discountProduct {

    public static void main(String[] args) {

        // TODO Auto-generated method stub

        Scanner scanner = new Scanner(System.in);

        System.out.print("Enter the original price: ");

        double originalPrice= scanner.nextDouble();

        System.out.print("Enter the discount: ");

        double discountPercentage=scanner.nextDouble();

        double discount = (originalPrice * discountPercentage) / 100.0;

        double discountedPrice = originalPrice - discount;
    }
}

```

```
        System.out.println("Discounted Price: " + discountedPrice);
    }

}
```

22. WAP to calculate Calculate Distance Between Two Points

```
import java.util.Scanner;

public class A22distance2point {

    public static void main(String[] args) {
        // TODO Auto-generated method stub

        Scanner input = new Scanner(System.in);

        System.out.print("Enter the x-coordinate of point 1: ");

        double x1 = input.nextDouble();

        System.out.print("Enter the y-coordinate of point 1: ");

        double y1 = input.nextDouble();

        System.out.print("Enter the x-coordinate of point 2: ");

        double x2 = input.nextDouble();

        System.out.print("Enter the y-coordinate of point 2: ");

        double y2 = input.nextDouble();

        double distance = Math.sqrt(Math.pow(x2 - x1, 2) + Math.pow(y2 - y1, 2));

        System.out.println("The distance between the two points is: " + distance);

    }

}
```

```
}
```

23. WAP to calculate weighted Average

```
import java.util.Scanner;
```

```
public class A23weighted {
```

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

```
        // TODO Auto-generated method stub
```

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

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

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

```
        double sum = 0;
```

```
        double weightSum = 0;
```

```
        for (int i = 1; i <= n; i++)
```

```
        { System.out.print("Enter value " + i + ": ");
```

```
            double value = input.nextDouble();
```

```
            System.out.print("Enter weight for value " + i + ": ");
```

```
            double weight = input.nextDouble();
```

```
            sum += value * weight;
```

```
            weightSum += weight;
```

```
        }
```

```

        double weightedAverage = sum / weightSum;

        System.out.println("The weighted average is: " + weightedAverage);
    }

}

```

24. WAP to calculate CGPA

```

import java.util.Scanner;

public class A24cgpa {

    public static void main(String[] args) {

        // TODO Auto-generated method stub

        Scanner input = new Scanner(System.in);

        System.out.print("Enter the number of subjects: ");

        int n = input.nextInt();

        double totalCredits = 0;

        double totalGradePoints = 0;

        for (int i = 1; i <= n; i++) {

            System.out.print("Enter credits for subject " + i + ": ");

            double credits = input.nextDouble();

            System.out.print("Enter grade for subject " + i + ": ");

            double grade = input.nextDouble();

```

```

        totalCredits += credits;

        totalGradePoints += credits * grade;
    }

    double cgpa = totalGradePoints / totalCredits;

    System.out.println("CGPA: " + cgpa);
}
}

```

25. WAP to calculate Compound Interest

```

import java.util.Scanner;

public class A25compound {

    public static void main(String[] args) {

        // TODO Auto-generated method stub

        Scanner input = new Scanner(System.in);

        System.out.print("Enter the principal amount: ");

        double principal = input.nextDouble();

        System.out.print("Enter the annual interest rate (in percentage): ");

        double rate = input.nextDouble() / 100;

        System.out.print("Enter the number of years: ");

        int time = input.nextInt();
    }
}

```



```

System.out.print("Enter the number of times interest is compounded per year: ");

int n = input.nextInt();

double compoundInterest = principal * Math.pow(1 + (rate / n), n * time) - principal;

System.out.println("The compound interest is: " + compoundInterest);

    }

}

```

26. WAP to calculate Average Marks

```

import java.util.Scanner;

public class A26avgmarks {

    public static void main(String[] args) {

        // TODO Auto-generated method stub

        Scanner input = new Scanner(System.in);

        System.out.print("Enter the number of subjects: ");

        int n = input.nextInt();

        double sum = 0;

        for (int i = 1; i <= n; i++) {

            System.out.print("Enter marks for subject " + i + ": ");

            double marks = input.nextDouble();

```

```

        sum += marks;
    }

    double average = sum / n;

    System.out.println("The average marks are: " + average);
}
}

```

27. WAP for Addition Of Two Numbers

```

import java.util.Scanner;

public class A27add2num {

    public static void main(String[] args) {
        // TODO Auto-generated method stub

        Scanner input = new Scanner(System.in);

        System.out.print("Enter the first number: ");

        double num1 = input.nextDouble();

        System.out.print("Enter the second number: ");

        double num2 = input.nextDouble();

        double sum = num1 + num2;

        System.out.println("The sum is: " + sum);
    }
}

```

```
}
```

28. WAP to find Sum Of N Numbers

```
import java.util.Scanner;
```

```
public class A28sumN {
```

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

```
        // TODO Auto-generated method stub
```

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

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

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

```
        double sum = 0;
```

```
        for (int i = 1; i <= n; i++) {
```

```
            System.out.print("Enter number " + i + ": ");
```

```
            double num = input.nextDouble();
```

```
            sum += num;
```

```
        }
```

```
        System.out.println("The sum is: " + sum);
```

```
    }
```

```
}
```

29. WAP to check given is Armstrong Number

```
import java.util.Scanner;
```

```
public class A29armstrong {
```

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

```
        // TODO Auto-generated method stub
```

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

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

```
        int number = input.nextInt();
```

```
        int originalNumber = number;
```

```
        int sum = 0;
```

```
        while (number > 0) {
```

```
            int digit = number % 10;
```

```
            sum += Math.pow(digit, 3);
```

```
            number /= 10;
```

```
        }
```

```
        if (sum == originalNumber) {
```

```
            System.out.println(originalNumber + " is an Armstrong number.");
```

```
        } else {
```

```
            System.out.println(originalNumber + " is not an Armstrong number.");
```

```
        }
```

```
}
```

```
}
```

30. WAP to Reverse A String

```
import java.util.Scanner;
```

```
public class A30reverse {
```

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

```
        // TODO Auto-generated method stub
```

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

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

```
        String str = input.nextLine();
```

```
        String reversed = new StringBuilder(str).reverse().toString();
```

```
        System.out.println("Reversed string: " + reversed);
```

```
    }
```

```
}
```

30. WAP to Reverse A String

```
import java.util.Scanner;
```

```
public class StringReversal {
```

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

```
    { Scanner scanner = new
```

```
        Scanner(System.in);
```

```

        System.out.print("Enter a string: ");

        String input = scanner.nextLine();

        String reversed = reverseString(input);

        System.out.println("Reversed string: " + reversed);
    }

    public static String reverseString(String input)

    {
        StringBuilder reversed = new StringBuilder(input);

        return reversed.reverse().toString();
    }
}

```

31. WAP to find HCF Of Two Numbers

```

import java.util.Scanner;

public class HCFCalculator {

    public static void main(String[] args)

    {
        Scanner input = new Scanner(System.in);

        System.out.print("Enter the first number: ");

        int num1 = input.nextInt();

        System.out.print("Enter the second number: ");

        int num2 = input.nextInt();

        int hcf = calculateHCF(num1, num2);

        System.out.println("The HCF of " + num1 + " and " + num2 + " is " + hcf);
    }

    public static int calculateHCF(int a, int b)

    {
        if (b == 0) {

            return a;

```

```

    } else {
        return calculateHCF(b, a % b);
    }
}
}

```

32. WAP to find LCM Of Two Numbers

```

import java.util.Scanner;

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

        System.out.print("Enter the first number: ");

        int num1 = scanner.nextInt();

        System.out.print("Enter the second number: ");

        int num2 = scanner.nextInt();

        int lcm = findLCM(num1, num2);

        System.out.println("The LCM of " + num1 + " and " + num2 + " is " + lcm);
    }

    public static int findLCM(int a, int b)
    {
        int hcf = findHCF(a, b);

        int lcm = (a * b) / hcf;

        return lcm;
    }

    public static int findHCF(int a, int b)
    {
        while (b != 0) {
            int temp = b;

```

```

        b = a % b;

        a = temp;
    }

    return a;
}
}

```

33. WAP to check given char is Vowel Or Consonant

```

import java.util.Scanner;

import java.lang.*;

public class VowelOrConsonantChecker

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

    Scanner scanner = new Scanner(System.in);

    System.out.print("Enter a character: ");

    char ch = scanner.next().charAt(0);

    ch = Character.toLowerCase(ch);

    if (ch >= 'a' && ch <= 'z') {

        if (ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' ||

ch=='u'){ System.out.println(ch + " is a vowel.");

        } else {

            System.out.println(ch + " is a consonant.");

        }

    } else {

        System.out.println("Invalid input. Please enter a valid alphabet.");

    }

}
}

```



```
}
```

34. WAP to check given number is Perfect Number

```
import java.util.Scanner;

public class PerfectNumberChecker

{ public static void main(String[] args)

{

    Scanner scanner = new Scanner(System.in);

    System.out.print("Enter a number: ");

    int number = scanner.nextInt();

    if (isPerfectNumber(number)) {

        System.out.println(number + " is a perfect number.");

    } else {

        System.out.println(number + " is not a perfect number.");

    }

}

public static boolean isPerfectNumber(int number)

{ if (number <= 1) {

    return false;

}

    int sumOfDivisors = 1;

    for (int i = 2; i <= Math.sqrt(number); i++)

    { if (number % i == 0) {

        sumOfDivisors += i;

        if (i != number / i) {

            sumOfDivisors += number / i;

        }

    }

}
```

```

        }

    }

    return sumOfDivisors == number;

}

}

```

35. WAP for Matrix Multiplication

```

import java.util.Scanner;

public class MatrixMultiplication

{ public static void main(String[] args)

{

    Scanner scanner = new Scanner(System.in);

    System.out.print("Enter the number of rows for matrix A: ");

    int rowsA = scanner.nextInt();

    System.out.print("Enter the number of columns for matrix A: ");

    int colsA = scanner.nextInt();

    System.out.print("Enter the number of rows for matrix B: ");

    int rowsB = scanner.nextInt();

    System.out.print("Enter the number of columns for matrix B: ");

    int colsB = scanner.nextInt();

    if (colsA != rowsB) {

        System.out.println("Matrix multiplication is not possible. The number of columns in A must be
equal to the number of rows in B.");

    }

    System.out.println("Enter the elements of matrix A:");

    int[][] matrixA = new int[rowsA][colsA];

    for (int i = 0; i < rowsA; i++)

        { for (int j = 0; j < colsA; j++) {

```

```

        matrixA[i][j] = scanner.nextInt();
    }
}

System.out.println("Enter the elements of matrix B:");

int[][] matrixB = new int[rowsB][colsB];

for (int i = 0; i < rowsB; i++)
    { for (int j = 0; j < colsB; j++) {
        matrixB[i][j] = scanner.nextInt();
    }
}

int[][] resultMatrix = new int[rowsA][colsB];

for (int i = 0; i < rowsA; i++) {
    for (int j = 0; j < colsB; j++) {
        for (int k = 0; k < colsA; k++) {
            resultMatrix[i][j] += matrixA[i][k] * matrixB[k][j];
        }
    }
}

System.out.println("Resultant Matrix:");

for (int i = 0; i < rowsA; i++) {
    for (int j = 0; j < colsB; j++)
        { System.out.print(resultMatrix[i][j] + " ");
    }

    System.out.println();
}

```

```
}  
}
```

36. WAP for Addition Of Two Matrices

```
import java.util.Scanner;  
  
public class MatrixAddition {  
  
    public static void main(String[] args)  
  
        { Scanner input = new Scanner(System.in);  
  
        System.out.print("Enter the number of rows: ");  
  
        int rows = input.nextInt();  
  
        System.out.print("Enter the number of columns: ");  
  
        int columns = input.nextInt();  
  
        int[][] matrix1 = new int[rows][columns];  
  
        int[][] matrix2 = new int[rows][columns];  
  
        System.out.println("Enter elements for the first matrix:");  
  
        for (int i = 0; i < rows; i++) {  
  
            for (int j = 0; j < columns; j++)  
  
                { matrix1[i][j] = input.nextInt();  
  
                }  
  
        }  
  
        System.out.println("Enter elements for the second matrix:");  
  
        for (int i = 0; i < rows; i++) {  
  
            for (int j = 0; j < columns; j++)  
  
                { matrix2[i][j] = input.nextInt();  
  
                }  
  
        }  
  
    }
```

```

int[][] resultMatrix = new int[rows][columns];

for (int i = 0; i < rows; i++) {

    for (int j = 0; j < columns; j++) {

        resultMatrix[i][j] = matrix1[i][j] + matrix2[i][j];

    }

}

System.out.println("Resultant Matrix:");

for (int i = 0; i < rows; i++) {

    for (int j = 0; j < columns; j++)

        { System.out.print(resultMatrix[i][j] + " ");

        }

    System.out.println();

}

}

```

37. WAP for Subtract of Two Matrices

```

import java.util.Scanner;

public class MatrixSubtraction {

    public static void main(String[] args)

    { Scanner scanner = new

    Scanner(System.in);

    System.out.print("Enter the number of rows: ");

    int rows = scanner.nextInt();

    System.out.print("Enter the number of columns: ");

    int columns = scanner.nextInt();

    int[][] matrix1 = new int[rows][columns];

```

```
System.out.println("Enter the elements of the first matrix:");

for (int i = 0; i < rows; i++) {

    for (int j = 0; j < columns; j++)

        { matrix1[i][j] = scanner.nextInt();

    }

}

int[][] matrix2 = new int[rows][columns];

System.out.println("Enter the elements of the second matrix:");

for (int i = 0; i < rows; i++) {

    for (int j = 0; j < columns; j++)

        { matrix2[i][j] = scanner.nextInt();

    }

}

int[][] resultMatrix = new int[rows][columns];

for (int i = 0; i < rows; i++) {

    for (int j = 0; j < columns; j++)

        { resultMatrix[i][j] = matrix1[i][j] -

        matrix2[i][j];

    }

}

System.out.println("Result of matrix subtraction:");

for (int i = 0; i < rows; i++) {

    for (int j = 0; j < columns; j++)

        { System.out.print(resultMatrix[i][j] + " ");

    }

}

System.out.println();
```

```
    }  
}  
}
```

38. WAP to find Transpose of Matrix

```
import java.util.Scanner;  
  
public class MatrixTranspose {  
  
    public static void main(String[] args)  
  
        { Scanner scanner = new  
  
        Scanner(System.in);  
  
        System.out.print("Enter the number of rows: ");  
  
        int rows = scanner.nextInt();  
  
        System.out.print("Enter the number of columns: ");  
  
        int columns = scanner.nextInt();  
  
        int[][] matrix = new int[rows][columns];  
  
        System.out.println("Enter the elements of the matrix:");  
  
        for (int i = 0; i < rows; i++) {  
  
            for (int j = 0; j < columns; j++)  
  
                { matrix[i][j] = scanner.nextInt();  
  
                }  
  
        }  
  
        int[][] transpose = new int[columns][rows];  
  
        for (int i = 0; i < rows; i++) {  
  
            for (int j = 0; j < columns; j++)  
  
                { transpose[j][i] = matrix[i][j];  
  
                }  
  
        }  
  
    }
```

```

        System.out.println("Original Matrix:");

        printMatrix(matrix);

        System.out.println("Transpose of Matrix:");

        printMatrix(transpose);

    }

    public static void printMatrix(int[][] matrix)

    { for (int i = 0; i < matrix.length; i++) {

        for (int j = 0; j < matrix[i].length; j++)

            { System.out.print(matrix[i][j] + " ");

            }

        System.out.println();

    }

    }

}

```

39. WAP to Check Leap Year Or Not

```

import java.util.Scanner;

public class LeapYearChecker {

    public static void main(String[] args)

    { Scanner scanner = new

    Scanner(System.in);

    System.out.print("Enter a year: ");

    int year = scanner.nextInt();

    if (isLeapYear(year)) {

        System.out.println(year + " is a leap year.");

    } else {

        System.out.println(year + " is not a leap year.");

    }

}

```



```

    }
}

public static boolean isLeapYear(int year) {

    // Leap year is divisible by 4, except for years divisible by 100 but not by 400

    return (year % 4 == 0 && year % 100 != 0) || (year % 400 == 0);

}

}

```

40. WAP to find to check Prime Number

```

import java.util.Scanner;

public class PrimeNumberChecker

{ public static void main(String[] args)

{

    Scanner scanner = new Scanner(System.in);

    System.out.print("Enter a number: ");

    int number = scanner.nextInt();

    scanner.close();

    if (isPrime(number)) {

        System.out.println(number + " is a prime number.");

    } else {

        System.out.println(number + " is not a prime number.");

    }

}

}

public static boolean isPrime(int num)

{ if (num <= 1) {

```

```

        return false;
    }

    if (num <= 3)
    { return true;
    }

    if (num % 2 == 0 || num % 3 == 0)
    { return false;
    }

    for (int i = 5; i * i <= num; i += 6) {
        if (num % i == 0 || num % (i + 2) == 0)
        { return false;
        }
    }

    return true;
}

```

41. WAP to find Sum Of Digits Of a given Number

```

import java.util.Scanner;

public class SumOfDigits {

    public static void main(String[] args)

    { Scanner scanner = new

    Scanner(System.in);

    System.out.print("Enter an integer: ");

    int number = scanner.nextInt();

    int sum = 0;

    int originalNumber = number;

```

```

while (number != 0) {

    int digit = number % 10;

    sum += digit;

    number /= 10;

}

System.out.println("The sum of digits in " + originalNumber + " is: " + sum);

}

```

42. WAP to Convert Octal To HexaDecimal

```

import java.util.Scanner;

public class OctalToHexadecimalConverter

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

    Scanner scanner = new Scanner(System.in);

    System.out.print("Enter an octal number: ");

    String octalString = scanner.nextLine();

    int octalNumber = Integer.parseInt(octalString, 8);

    String hexadecimalString = Integer.toHexString(octalNumber);

    System.out.println("Hexadecimal equivalent: 0x" + hexadecimalString.toUpperCase());

}

}

```

43. WAP to Convert Decimal To Hexadecimal

```

import java.util.Scanner;

public class DecimalToHexadecimal

{ public static void main(String[] args)

{

```

```

Scanner scanner = new Scanner(System.in);

System.out.print("Enter a decimal number: ");

int decimalNumber = scanner.nextInt();

String hexadecimal = Integer.toHexString(decimalNumber);

System.out.println("Hexadecimal representation: " + hexadecimal);

}

}

```

44 .WAP to Convert Decimal To Octal

```

import java.util.Scanner;

public class DecimalToOctal

{ public static void main(String[] args)

{

Scanner input = new Scanner(System.in);

System.out.print("Enter a decimal number: ");

int decimalNumber = input.nextInt();

String octalNumber = decimalToOctal(decimalNumber);

System.out.println("Octal representation: " + octalNumber);

}

public static String decimalToOctal(int decimalNumber)

{ if (decimalNumber == 0) {

return "0";

}

StringBuilder octal = new StringBuilder();

while (decimalNumber > 0) {

int remainder = decimalNumber % 8;

octal.insert(0, remainder);

```

```

        decimalNumber /= 8;
    }

    return octal.toString();
}
}

```

45. WAP to Convert Decimal To Binary

```

import java.util.Scanner;

public class DecimalToBinary {

    public static void main(String[] args)

    {
        Scanner scanner = new

        Scanner(System.in);

        System.out.print("Enter a decimal number: ");

        int decimalNumber = scanner.nextInt();

        String binary = decimalToBinary(decimalNumber);

        System.out.println("Binary representation: " + binary);

    }

    public static String decimalToBinary(int decimalNumber)

    {
        if (decimalNumber == 0) {

            return "0";

        }

        StringBuilder binary = new StringBuilder();

        while (decimalNumber > 0) {

            int remainder = decimalNumber % 2;

            binary.insert(0, remainder);

            decimalNumber /= 2;

        }
    }
}

```

```
        return binary.toString();
    }
}
```

46. WAP to Convert Fahrenheit To Celsius

```
import java.util.Scanner;

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

        System.out.print("Enter temperature in Fahrenheit: ");

        double fahrenheit = scanner.nextDouble();

        double celsius = (fahrenheit - 32) * 5.0/9.0;

        System.out.println("Temperature in Celsius: " + celsius);
    }
}
```

47. WAP to Count Vowels In A String

```
import java.util.Scanner;

public class VowelCounter {

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

        System.out.print("Enter a string: ");

        String input = scanner.nextLine().toLowerCase;

        int vowelCount = countVowels(input);

        System.out.println("Number of vowels in the string: " + vowelCount);
    }

    public static int countVowels(String str) {
```

```

int count = 0;

for (int i = 0; i < str.length(); i++)

    { char ch = str.charAt(i);

    if (ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' || ch == 'u')

        { count++;

        }

    }

return count;

}

}

```

48. WAP to Reverse A Number

```

import java.util.Scanner;

public class ReverseNumber {

    public static void main(String[] args)

    { Scanner scanner = new

    Scanner(System.in);

    System.out.print("Enter a number to reverse: ");

    int num = scanner.nextInt();

    int reversedNum = 0;

    while (num != 0) {

        int digit = num % 10;

        reversedNum = reversedNum * 10 + digit;

        num /= 10;

    }

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

}

```

```
}
```

49. WAP to Reverse An Array

```
import java.util.Scanner;
```

```
public class A49reversearray {
```

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

```
        // TODO Auto-generated method stub
```

```
        int[] array = {1, 2, 3, 4, 5};
```

```
        int start = 0;
```

```
        int end = array.length - 1;
```

```
        while (start < end) {
```

```
            // Swap elements at start and end
```

```
            int temp = array[start];
```

```
            array[start] = array[end];
```

```
            array[end] = temp;
```

```
            start++;
```

```
            end--;
```

```
        }
```

```
        System.out.println("Reversed Array: ");
```

```
        for (int num : array)
```

```
            { System.out.print(num + " ");
```

```
        }
```



```
}  
  
}
```

50. WAP to Insert an Element in an Array

```
package midsem;
```

```
import java.util.Scanner;
```

```
public class A50insetelement {  
  
    public static void main(String[] args) {  
        // TODO Auto-generated method stub  
        int[] array = {1, 2, 3, 4, 5};  
        int elementToInsert = 10;  
        int indexToInsert = 2;  
        if (indexToInsert >= 0 && indexToInsert <= array.length) {  
  
            int[] newArray = new int[array.length + 1];  
            for (int i = 0; i < indexToInsert; i++)  
                { newArray[i] = array[i];  
            }  
  
            newArray[indexToInsert] = elementToInsert;  
  
            for (int i = indexToInsert; i < array.length; i++)  
                { newArray[i + 1] = array[i];  
            }  
        }  
    }  
}
```

```

        array = newArray;

        System.out.println("Array after insertion: ");

        for (int num : array) {

            System.out.print(num + " ");

        }

    } else {

        System.out.println("Invalid index for insertion.");

    }

}

}

```

52. WAP for Linear Search

```

import java.util.Scanner;

public class A52linear {

    public static void main(String[] args) {

        // TODO Auto-generated method stub

        int[] array = {10, 25, 3, 45, 12, 30};

        int target = 45;

        boolean found = false;

        for (int i = 0; i < array.length; i++) {

            if (array[i] == target)

                { found = true;

```

```

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

        break;
    }
}

if (!found) {
    System.out.println("Element " + target + " not found in the array.");
}
}
}

```

53. WAP to Binary Search

```

import java.util.Scanner;

public class A53binary {

    public static void main(String[] args) {
        // TODO Auto-generated method stub

        int[] sortedArray = {3, 6, 9, 12, 15, 18, 21};

        int target = 12;

        int left = 0;

        int right = sortedArray.length - 1;

        boolean found = false;

        while (left <= right) {
            int mid = left + (right - left) / 2;

```

```

        if (sortedArray[mid] == target)

            { found = true;

            System.out.println("Element " + target + " found at index " + mid);

            break;

        }

        if (sortedArray[mid] < target)

            { left = mid + 1;

            } else {

                right = mid - 1;

            }

        }

    }

    if (!found) {

        System.out.println("Element " + target + " not found in the array.");

    }

}

}

```

54. WAP for Merge Sort

```
import java.util.Scanner;
```

```
public class A54merge {
```

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

```

        // TODO Auto-generated method stub

        int[] array = {12, 11, 13, 5, 6, 7};

        for(int i = 0; i < array.length; i++)

            System.out.print(array[i]+" ");

        mergeSort(array, 0, array.length - 1);

        System.out.println();

        for(int i = 0; i < array.length; i++)

            System.out.print(array[i]+" ");

    }

    public static void mergeSort(int[] arr, int left, int right) {

        if (left < right) {

            int mid = left + (right - left) / 2;

            mergeSort(arr, left, mid);

            mergeSort(arr, mid + 1, right);

            merge(arr, left, mid, right);

        }

    }

```

```

        public static void merge(int[] arr, int left, int mid, int right) {

            int n1 = mid - left + 1;

            int n2 = right - mid;

            int[] leftArray = new int[n1];

```

```
int[] rightArray = new int[n2];
```

```
for (int i = 0; i < n1; i++)
```

```
    { leftArray[i] = arr[left + i];
```

```
    }
```

```
for (int j = 0; j < n2; j++)
```

```
    { rightArray[j] = arr[mid + 1 + j];
```

```
    }
```

```
int i = 0, j = 0;
```

```
int k = left;
```

```
while (i < n1 && j < n2) {
```

```
    if (leftArray[i] <= rightArray[j])
```

```
        { arr[k++] = leftArray[i++];
```

```
    } else {
```

```
        arr[k++] = rightArray[j++];
```

```
    }
```

```
}
```

```
while (i < n1) {
```

```
    arr[k++] = leftArray[i++];
```

```
}
```

```
while (j < n2) {
```

```
    arr[k++] = rightArray[j++];
```

```
}
```

```
}  
}
```

55. WAP for Selection Sort

```
package midsem;
```

```
import java.util.Scanner;
```

```
public class A55selection {
```

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

```
        // TODO Auto-generated method stub
```

```
        int[] array = {64, 25, 12, 22, 11};
```

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

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

```
            System.out.print(array[i]+" ");
```

```
        selectionSort(array);
```

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

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

```
            System.out.print(array[i]+" ");
```

```
    }
```

```
    public static void selectionSort(int[] arr) {
```

```
        int n = arr.length;
```

```
        for (int i = 0; i < n - 1; i++) {
```

```

    int minIndex = i;

    for (int j = i + 1; j < n; j++) {

        if (arr[j] < arr[minIndex])

            { minIndex = j;

            }

        }

    }

    int temp = arr[minIndex];

    arr[minIndex] = arr[i];

    arr[i] = temp;

}

}

}

```

56.WAP for Bubble Sort

```

import java.util.*;

public class A56bubble {

    public static void main(String[] args) {

        // TODO Auto-generated method stub

        int[] array = {64, 34, 25, 12, 22, 11, 90};

        System.out.println();

        for(int i = 0; i < array.length; i++)

            System.out.print(array[i]+" ");
    }
}

```



```

        bubbleSort(array);

        System.out.println();

        System.out.println("Sorted array: " + Arrays.toString(array));

    }

```

```

public static void bubbleSort(int[] arr) {

```

```

    int n = arr.length;

```

```

    for (int i = 0; i < n - 1; i++) {

```

```

        for (int j = 0; j < n - i - 1; j++) {

```

```

            if (arr[j] > arr[j + 1]) {

```

```

                int temp = arr[j];

```

```

                arr[j] = arr[j + 1];

```

```

                arr[j + 1] = temp;

```

```

            }

```

```

        }

```

```

    }

```

```

    }

```

```

}

```

57. WAP for Quick Sort

```

import java.util.*;

```

```

public class A57quick {

```

```
    public static void main(String[] args) {  
        // TODO Auto-generated method stub  
  
        int[] array = {10, 7, 8, 9, 1, 5};  
  
        System.out.println();  
  
        for(int i = 0; i < array.length; i++)  
            System.out.print(array[i]+" ");  
  
  
        quickSort(array, 0, array.length - 1);  
  
        System.out.println();  
  
        System.out.println("Sorted array: " + Arrays.toString(array));  
    }  
}
```

```
public static void quickSort(int[] arr, int low, int high) {  
    if (low < high) {  
        int pi = partition(arr, low, high);  
  
        quickSort(arr, low, pi - 1);  
        quickSort(arr, pi + 1, high);  
    }  
}
```

```
public static int partition(int[] arr, int low, int high) {  
    int pivot = arr[high];  
    int i = low - 1;  
  
    for (int j = low; j < high; j++) {
```

```

        if (arr[j] < pivot)
        { i++;

        // Swap arr[i] and arr[j]

        int temp = arr[i];
        arr[i] = arr[j];
        arr[j] = temp;
        }
    }

    // Swap arr[i+1] and arr[high] (or the pivot)
    int temp = arr[i + 1];
    arr[i + 1] = arr[high];
    arr[high] = temp;

    return i + 1;
}
}

```

58. WAP for Insertion Sort

```

import java.util.*;

public class A58insertion {

    public static void main(String[] args) {

```

```

        // TODO Auto-generated method stub

        int[] array = {12, 11, 13, 5, 6};

        System.out.println();

        for(int i = 0; i < array.length; i++)

            System.out.print(array[i]+" ");


        insertionSort(array);

        System.out.println();

        System.out.println("Sorted array: " + Arrays.toString(array));
    }


    public static void insertionSort(int[] arr) {

        int n = arr.length;


        for (int i = 1; i < n; i++) {

            int key = arr[i];

            int j = i - 1;


            while (j >= 0 && arr[j] > key)

                { arr[j + 1] = arr[j];

                    j--;

                }


            arr[j + 1] = key;

        }

    }
}

```

59. WAP for Heap Sort

```
import java.util.*;

public class A59heap {

    public static void main(String[] args) {

        // TODO Auto-generated method stub

        int[] array = {12, 11, 13, 5, 6, 7};

        System.out.println();

        for(int i = 0; i < array.length; i++)

            System.out.print(array[i]+" ");

        heapSort(array);

        System.out.println();

        System.out.println("Sorted array: " + Arrays.toString(array));

    }

    public static void heapSort(int[] arr) {

        int n = arr.length;

        for (int i = n / 2 - 1; i >= 0; i--) {

            heapify(arr, n, i);

        }

        for (int i = n - 1; i >= 0; i--) {

            int temp = arr[0];
```

```
arr[0] = arr[i];  
arr[i] = temp;  
  
    heapify(arr, i, 0);  
}  
}
```

```
public static void heapify(int[] arr, int n, int i) {  
    int largest = i;  
    int left = 2 * i + 1;  
    int right = 2 * i + 2;  
  
    if (left < n && arr[left] > arr[largest])  
    { largest = left;  
    }  
  
    if (right < n && arr[right] > arr[largest])  
    { largest = right;  
    }  
  
    if (largest != i) {  
        int swap = arr[i];  
        arr[i] = arr[largest];  
        arr[largest] = swap;  
  
        heapify(arr, n, largest);  
    }  
}
```

}

}