

PROGRAM 1:

- Here we are being asked for a java program that takes 10 numbers of input from the user, and prints out the number of odd and even numbers.
- We use java's scanner class to scan for user input until a certain outcome is reached.
- we use a loop to ask the user 10 times for a number. We see if modulus of 2 for each number, and if 0, it is an even number. else it is an odd number.

```
import java.util.Scanner; #import the scanner class to allow to read input

public class EvenOddCounter { #here we define the class

    public static void main(String[] args) { #there we have the main method
        Scanner scanner = new Scanner(System.in); #create a scanner object to read input

        int evenCount = 0; #initialise counter for even no.s
        int oddCount = 0; #initialise counter for odd no.s

        System.out.println("Please enter in 10 numbers:"); #will prompt user to enter 10 no.s

        for (int i = 1; i <= 10; i++) { #start a loop that will ask the user 10 times to input a number
            System.out.print("Number " + i + ": ");
            int number = scanner.nextInt(); #read the next no and store it

            if (number % 2 == 0) { #if remainder is even, increment even counter
                evenCount++;
            } else { #else increment odd counter
                oddCount++;
            }
        }

        System.out.println("Total even numbers: " + evenCount); #finally print total number of even
        System.out.println("Total odd numbers: " + oddCount); #and odd numbers
    }
}
```

```
Enter 10 numbers:
Number 1: 66
Number 2: 18
Number 3: 35
Number 4: 999
Number 5: 456
Number 6: 7
Number 7: 3
Number 8: 5
Number 9: 6
Number 10: 7
Total even numbers: 4
Total odd numbers: 6
```

PROGRAM 2:

- Here we asked to compute the standard deviation of N real numbers.
- The user is asked to input the value of n, then asked to input an n number of real numbers.
- To simplify the calculation, i decided to split the standard deviation formula by first getting the part above the fraction, storing this in a variable, then calculating standard deviation.

```
public class StandardDeviationCalculator {  
    public static void main(String[] args) {  
        Scanner scanner = new Scanner(System.in); #create a scanner object to get user input  
  
        System.out.print("Enter the number of elements N: "); #ask user to input number of elements  
        int N = scanner.nextInt(); #store the number of elements provided by user in N  
  
        double[] numbers = new double[N]; create an array to store the N real numbers  
        double sum = 0; #here we will keep track of sum  
  
        System.out.println("Enter " + N + " real numbers:"); # now ask the user to input the N elements  
        for (int i = 0; i < N; i++) {  
            numbers[i] = scanner.nextDouble();  
            sum += numbers[i];  
        }  
  
        double mean = sum / N; #calculate average  
  
        #calculate sum of squared differences from mean  
        double sumOfSquaredDifferences = 0;  
        for (double num : numbers) {  
            sumOfSquaredDifferences += Math.pow(num - mean, 2);  
        }  
  
        # calculate standard deviation  
        double standardDeviation = Math.sqrt(sumOfSquaredDifferences / N);  
  
        #output the result  
        System.out.println("The standard deviation is: " + standardDeviation);  
    }  
}
```

```
Enter the number of elements N: 2  
Enter 2 real numbers:  
26  
83  
The standard deviation is: 28.5
```

PROGRAM 3:

- Here the user is asked to input 10 names, and for these 10 names to be printed back alphabetically
- The names are stored in a string called names. a simple sort is used to sort the contents of names alphabetically.
- The names are then printed out onto the screen one by one alphabetically.

```
import java.util.Scanner;

import java.util.Arrays; #import arrays class to use arrays
import java.util.Scanner;

public class AlphabeticalNames {

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        String[] names = new String[10]; #initialise the string to hold 10 items

        # input 10 names from the user
        System.out.println("Enter 10 names:");
        for (int i = 0; i < names.length; i++) {
            System.out.print("Name " + (i + 1) + ": ");
            names[i] = scanner.nextLine();
        }

        #now lets sort the names alphabetically
        Arrays.sort(names);

        # now print the sorted list of names
        System.out.println("Names in alphabetical order:");
        for (String name : names) {
            System.out.println(name);
        }
    }
}
```

Enter 10 names:

Name 1: john

Name 2: michael

Name 3: ciara

Name 4: tom

Name 5: niall

Name 6: sarah

Name 7: mary

Name 8: siobhan

Name 9: caoimhe

Name 10: daniel

Names in alphabetical order

caoimhe

ciara

daniel

john

mary

michael

niall

sarah

siobhan

tom

PROGRAM 4:

- Here we are asked to add 2 3x3 matrices together.
- They are stored using arrays. for each position in each of the matrices, the values are added together, and stored in a 3rd matrix called sumMatrix.

```
import java.util.Scanner;

public class MatrixAddition {

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        int[][] firstMatrix = new int[3][3]; #initialise a 3x3 array to store the first matrix
        int[][] secondMatrix = new int[3][3]; #initialise another one to store the 2nd one
        int[][] sumMatrix = new int[3][3]; ; #initialise another to store sum of the 1st and 2nd matrices

        # ask user to input for the first matrix
        System.out.println("Enter the elements for the first 3x3 matrix:");
        for (int i = 0; i < 3; i++) {
            for (int j = 0; j < 3; j++) {
                firstMatrix[i][j] = scanner.nextInt();
            }
        }

        #ask user to input for the second matrix
        System.out.println("Enter the elements for the second 3x3 matrix:");
        for (int i = 0; i < 3; i++) {
            for (int j = 0; j < 3; j++) {
                secondMatrix[i][j] = scanner.nextInt();
            }
        }

        #now add the two together Adding the two matrices
        for (int i = 0; i < 3; i++) {
            for (int j = 0; j < 3; j++) {
                sumMatrix[i][j] = firstMatrix[i][j] + secondMatrix[i][j];
            }
        }

        #nowe diisplay the result
        System.out.println("Sum of the two matrices is:");
        for (int i = 0; i < 3; i++) {
            for (int j = 0; j < 3; j++) {
                System.out.print(sumMatrix[i][j] + " ");
            }
            System.out.println(); #this will print allow to move to the next row
        }
    }
}
```

```
Enter the elements for the first 3x3 matrix:
3 6 7 2 6 8 9 0 3
Enter the elements for the second 3x3 matrix:
4 5 6 7 8 9 0 1 2
Sum of the two matrices is:
7 11 13
9 14 17
9 1 5
```

PROGRAM 5:

- Here we are asked to get the transpose of a matrix. This means the i th row of the matrix will become a i th column and vice versa.
- Transpose is used to simply get the transpose of the 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: "); #ask the user for the number of rows in the matrix and store it
        int rows = scanner.nextInt();
        System.out.print("Enter the number of columns: "); #now ask for number of columns and store it
        int cols = scanner.nextInt();

        int[][] matrix = new int[rows][cols]; #initliase a 2d array (matrix) which has rows and columns
        int[][] transpose = new int[cols][rows]; #initialise a 2d array (transpose) which has rows and columns

        #aks input the matrix from the user
        System.out.println("Enter the matrix elements:");
        for (int i = 0; i < rows; i++) {
            for (int j = 0; j < cols; j++) {
                matrix[i][j] = scanner.nextInt();
            }
        }

        #now compute the transpose of the matrix
        for (int i = 0; i < rows; i++) {
            for (int j = 0; j < cols; j++) {
                transpose[j][i] = matrix[i][j];
            }
        }

        #now display the transpose of the matrix
        System.out.println("Transpose of the matrix:");
        for (int i = 0; i < cols; i++) {
            for (int j = 0; j < rows; j++) {
                System.out.print(transpose[i][j] + " ");
            }
            System.out.println(); #here print a newline to move to the next row, after printing all the columns of the current row
        }
    }
}
```

```
Enter the number of rows: 2
Enter the number of columns: 2
Enter the matrix elements:
1 2 3 4
Transpose of the matrix:
1 3
2 4
```

PROGRAM 6: