Idris Ratlamwala
C31 - 2003145

Experiment 6: Programs on Two Dimensional Arrays

Theory :

The simplest of the multi-dimensional array is a **two-dimensional array**.
A simple definition of 2D arrays is: A 2D array is an array of one-dimensional arrays.

In Java, a two-dimensional array is stored in the form of rows and columns and is represented in the form of a matrix.

The general declaration of a two-dimensional array is,
data_type [] [] array_name;

Here,
data_type = data type of elements that will be stored in an array.
array_name = name of the two-dimensional array.

A 2D array can be created using new as follows:
data_type[][] array_name = new data_type[row_size][column_size];

Here,
row_size = number of rows an array will contain.
column_size = number of columns array will contain.

An integer array named 'myarray' of 3 rows and 2 columns can be declared as below.
int [][] myarray = new int[3][2];

Initialize 2d Array
Similar to 1D array, 2 nested for loops can be used to initialize a 2d array.
for(int i=0;i<3;i++) for(int j=0;j<2;j++) myarray[i][j]=sc.nextInt();

A.

Aim : WAP to find Transpose of a Matrix (One class ,only main).

Program :

```
    import java.util.Scanner;

public class Transpose {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter the number of rows and column in matrix :");
        int m=sc.nextInt();
        int n=sc.nextInt();
        int[][] a= new int[m][n];
        int[][] b= new int[n][m];

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

        for(int i=0;i<m;i++) for(int j=0;j<n;j++) a[i][j]=sc.nextInt();

        System.out.println("Transposed matrix :");
        for(int i=0;i<m;i++) for(int j=0;j<n;j++) b[j][i]=a[i][j];

        for(int i=0;i<n;i++) {
            for(int j=0;j<m;j++)
            System.out.print(b[i][j]+" ");
            System.out.println();
        }
    }
}
```

Output :

```
PS C:\Users\IsmailRatlamwala>  & 'c:\Users\IsmailRatlamwala\.vscode\extensions\vscjava.vscode-java-debug-0.36.0\scripts\launcher.bat' 'C:\Pr
ogram Files\Java\jdk-16.0.2\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-Dfile.encoding=UTF-8' '-cp' 'C:\Users\IsmailRatlamwala
\AppData\Local\Temp\vscodesws_6d568\jdt_ws\jdt.ls-java-project\bin' 'Transpose'
Enter the number of rows and column in matrix :2 3
Enter the elements of matrix :
1 2 3
4 5 6
Transposed matrix :
1 4
2 5
3 6
```

B.

Aim : WAP to Pass a 2D Matrix to a function which determines if it is a square matrix. If not, program should come to end else the program should find sum of all diagonal elements of a Matrix.

Program :

```
    import java.util.Scanner;

public class sumOfDiagonal {

    public static int sum(int a[][]) {
        if(a.length==a[0].length){
            int sum=0;
            for(int i=0;i<a.length;i++)
            for(int j=0;j<a[0].length;j++)
            sum=(i==j)?(sum+a[i][j]):sum;
            return sum;
        }
        return -1;
    }
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter the number of rows and column in matrix :");
        int m=sc.nextInt();
        int n=sc.nextInt();
        int[][] a= new int[m][n];

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

        for(int i=0;i<m;i++) for(int j=0;j<n;j++) a[i][j]=sc.nextInt();

        int sum=sum(a);
        if(sum==-1) System.out.println("Given matrix is not a square matrix");
        else System.out.println("Sum of diagonal elements : "+sum);
    }
}
```

Output :

```
Enter the number of rows and column in matrix :2 3
Enter the elements of matrix :
1 2 3
4 5 6
Given matrix is not a square matrix
PS C:\Users\IsmailRatlamwala\Documents\College prog\Oops Labs\expt6> java sumOfDiagonal.java
Enter the number of rows and column in matrix :3 3
Enter the elements of matrix :
1 2 3
4 5 6
7 8 9
Sum of diagonal elements : 15
PS C:\Users\IsmailRatlamwala\Documents\College prog\Oops Labs\expt6>
```