

EXPERIMENT 5

// Implementing Arrays in Java

One-Dimensional Array

- **Declaration for Static Creation -**

Syntax: data_type identifier[] = {e1 , e2 , , en};

Example: int arr[] = {1,2,3,4};

- **Declaration for Dynamic Creation -**

Syntax: data_type identifier[] = new data_type[size];

Example: int arr[] = new int[4];

Two-Dimensional Array

- **Declaration for Static Creation -**

Syntax: data_type identifier[][] = {{a1 , a2,...,an} , {b1 , b2,...,bn}...};

Example: int arr[][] = {{1,2,3,4} , {5,6,7,8}};

- **Declaration for Dynamic Creation -**

Syntax: data_type identifier[][] = new data_type[d1][d2];

Example: int arr[][] = new int[4][5];

Accessing elements of an array

- By specifying the index of the required element.
For example, arr[1] , arr[1][2].
- By using loops on the array.

PROGRAM 4

// To find out the number of positive, negative and zero elements in a linear array.

// Source Code -

```
import java.util.Scanner;

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

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

```

noe = sc.nextInt();

int arr[ ] = new int[noe];
System.out.println("Enter " + noe + " elements in array:");
for(int i=0; i<noe; i++)
{
    arr[i] = sc.nextInt();
}

int np=0,nn=0,nz=0;
for(int i=0; i<noe; i++)
{
    if(arr[i] > 0)
    {
        np++;
    }
    else if(arr[i] < 0)
    {
        nn++;
    }
    else
    {
        nz++;
    }
}

System.out.println("Number of positive elements = " + np);
System.out.println("Number of negative elements = " + nn);
System.out.println("Number of zero elements = " + nz);
}
}

```

Output -

Enter the number of elements for array: 10

Enter 10 elements in array:

1
-2
3
-4
5

```
-6  
0  
-8  
9  
0  
Number of positive elements = 4  
Number of negative elements = 4  
Number of zero elements = 2
```

PROGRAM 5

// To perform multiplication of two two-dimensional matrices.

// Source Code -

```
import java.util.Scanner;  
  
public class MatMul  
{  
    public static void main(String args[ ])  
    {  
        Scanner sc = new Scanner(System.in);  
  
        int r1,c1,r2,c2,i,j,k;  
  
        System.out.print("Enter the number of rows for matrix 1: ");  
        r1 = sc.nextInt();  
        System.out.print("Enter the number of columns for matrix 1: ");  
        c1 = sc.nextInt();  
        System.out.print("Enter the number of rows for matrix 2: ");  
        r2 = sc.nextInt();  
        System.out.print("Enter the number of columns for matrix 2: ");  
        c2 = sc.nextInt();  
  
        if(c1 != r2)  
        {  
            System.out.println("Matrix Multiplication not possible.");  
        }  
        else  
        {  
            int mat1[ ][ ] = new int[r1][c1];
```

```

int mat2[ ][ ] = new int[r2][c2];
int resmat[ ][ ] = new int[r1][c2];

System.out.print("Enter " + r1*c1 + " elements in matrix 1: ");
for(i=0; i<r1; i++)
{
    for(j=0; j<c1; j++)
    {
        mat1[i][j] = sc.nextInt();
    }
}

System.out.print("Enter " + r2*c2 + " elements in matrix 2: ");
for(i=0; i<r2; i++)
{
    for(j=0; j<c2; j++)
    {
        mat2[i][j] = sc.nextInt();
    }
}

System.out.println("You have entered the following matrices-");
System.out.println("Matrix 1 - ");
for(i=0; i<r1; i++)
{
    for(j=0; j<c1; j++)
    {
        System.out.print(mat1[i][j] + "\t");
    }
    System.out.println();
}

System.out.println("Matrix 2 - ");
for(i=0; i<r2; i++)
{
    for(j=0; j<c2; j++)
    {
        System.out.print(mat2[i][j] + "\t");
    }
    System.out.println();
}

```

```

for(i=0; i<r1; i++)
{
    for(j=0; j<c2; j++)
    {
        int sum=0;
        for(k=0; k<r2; k++)
        {
            sum += mat1[i][k]*mat2[k][j];
        }
        resmat[i][j] = sum;
    }
}

System.out.println("Resultant Matrix - ");
for(i=0; i<r1; i++)
{
    for(j=0; j<c2; j++)
    {
        System.out.print(resmat[i][j] + "\t");
    }
    System.out.println();
}
}

```

Output -

Enter the number of rows for matrix 1: 2
Enter the number of columns for matrix 1: 3
Enter the number of rows for matrix 2: 3
Enter the number of columns for matrix 2: 2
Enter 6 elements in matrix 1: 1 3 5 7 9 11
Enter 6 elements in matrix 2: 2 4 6 8 10 12
You have entered the following matrices-
Matrix 1 -
1 3 5
7 9 11
Matrix 2 -
2 4
6 8

10 12

Resultant Matrix -

70 88

178 232