

# LAB 10

NAME: Aditya Anand

ROLL NO.: 20124009

BRANCH: IT

S No.	Title	Date Of Implementation	Remarks
1	Program to illustrate array manipulation	28-03-2022	

## Program to illustrate Array Manipulation

### OBJECTIVE:

The objective of the following program is to illustrate various array operations.CODE:

```
import java.util.Scanner;

public class Lab10 {
    public static void main(String[] args){
        int n;
        int sum=0;

        Scanner sc = new Scanner(System.in);

        //reading the number of elements from the that we want to enter for the 1st array
        System.out.print("Enter the size of the 1st array: ");
        n = sc.nextInt();

        int[] arr1 = new int[n];

        System.out.print("Enter "+n+" elements: ");
        for(int i=0; i<n; i++){
            //reading array elements from the user
            arr1[i] = sc.nextInt();
            sum=sum+arr1[i];
        }

        //printing the sum of all elements of array
        System.out.println("Sum = "+sum);

        //printing the average of all elements
        System.out.println("Average = "+(double)sum/(double)n);

        //reading the number of elements from the that we want to enter for the 2nd array
        System.out.print("Enter the size of the 2nd array: ");
        n = sc.nextInt();
        sum=0;

        int[] arr2 = new int[n];

        System.out.print("Enter "+n+" elements: ");
        for(int i=0; i<n; i++){
            //reading array elements from the user
            arr2[i] = sc.nextInt();
            sum=sum+arr2[i];
        }

        //printing the sum of all elements of array
        System.out.println("Sum = "+sum);
```

```

//printing the average of all elements
System.out.println("Average = "+(double)sum/(double)n);

//perform sum and difference of 2 arrays, if their size is same and store the results in the 1st
and 2nd rows of a 2D array
if(arr1.length==arr2.length){
    int [][] op = new int [2][arr2.length];

    System.out.println("Adding the two arrays: ");
    for(int i=0; i<arr1.length; i++){
        op[0][i]=arr1[i]+arr2[i];
    }

    System.out.println("Sum Array: ");
    for(int i=0; i<arr1.length; i++){
        System.out.print(op[0][i]+" ");
    }

    System.out.println("");

    System.out.println("Subtracting the two arrays: ");
    for(int i=0; i<arr1.length; i++){
        op[1][i]=arr1[i]-arr2[i];
    }

    System.out.println("Difference Array: ");

    for(int i=0; i<arr1.length; i++){
        System.out.print(op[1][i]+" ");
    }
    System.out.println("");
}

//reading the number of elements from the that we want to enter
System.out.print("Enter the size of 3rd array: ");
n=sc.nextInt();
int[] arr= new int[100];

System.out.print("Enter the elements of array: ");
for(int i=0; i<n; i++){
    //reading array elements from the user
    arr[i]=sc.nextInt();
}

System.out.print("Array elements are: ");
// accessing array elements
for (int i=0; i<n; i++) {
    System.out.print(arr[i]+" ");
}
System.out.println();
minEle(arr, n);
maxEle(arr, n);

```

```

System.out.print("Enter the element to be searched: ");
int find = sc.nextInt();
linearSearch(arr, find, n);
int rotate_by = n-2;
System.out.println("Rotating the array by "+ rotate_by +" positions\n");
for(int i= 0; i < n-2; i++){
    rotateArr(arr, n);
}
System.out.println("Array elements after rotation are: ");
// accessing array elements
for (int i=0; i<n; i++) {
    System.out.print(arr[i]+" ");
}
System.out.println();

// Reversing the array
reverseArr(arr, 0, n-1);
System.out.println("Array elements after reversal are: ");
for (int i=0; i<n; i++) {
    System.out.print(arr[i]+" ");
}
System.out.println();

sc.close();

}
static void minEle(int arr[], int n){
    int min=arr[0];
    for(int i=1;i<n;i++){
        {
            if(min>arr[i])
                min=arr[i];
        }
        System.out.print("The minimum element is ");

        System.out.println(min);
    }
    // Finding the maximum element of array
    static void maxEle(int arr[], int n){
        int max=arr[0];
        for(int i=1;i<n;i++){
            if(max<arr[i])
                max=arr[i];
        }
        System.out.print("The max element is ");
        System.out.println(max);
    }
    // Linear Searching the element, returning -1 if not present other wise printing index
    public static int linearSearch(int[] arr, int key, int n){
        for(int i=0;i<n;i++){
            if(arr[i] == key){

```

```

        System.out.println("The element is found at index " + i + "\n" );
        return i;
    }
}
System.out.println("Element not found.\n");
return -1;
}

public static void rotateArr(int arr[], int n){
    int i, temp;
    temp = arr[0];
    for (i = 0; i < n - 1; i++){
        arr[i] = arr[i + 1];
    }
    arr[n-1] = temp;
}

public static void reverseArr(int arr[], int start, int end){
    int temp;
    while (start < end){
        temp = arr[start];
        arr[start] = arr[end];
        arr[end] = temp;
        start++;
        end--;
    }
}
}

```

## OUTPUT:

PS C:\Users\beadi\Desktop\JAVA\JAVA LAB\Practical 10>

Enter the size of the 1st array: 3

Enter 3 elements: 1 2 3

Sum = 6

Average = 2.0

Enter the size of the 2nd array: 3

Enter 3 elements: 3 2 1

Sum = 6

Average = 2.0

Adding the two arrays:

Sum Array:

4 4 4

Subtracting the two arrays:

Difference Array:

-2 0 2

Enter the size of 3rd array: 4

Enter the elements of array: 1 2 3 4

Array elements are: 1 2 3 4

The minimum element is 1

The max element is 4

Enter the element to be searched: 2

The element is found at index 1

Rotating the array by 2 positions

Array elements after rotation are:

3 4 1 2

Array elements after reversal are:

2 1 4 3