

## Lab Assignment 11

**Objective:** Implementations of Arrays in Java.

**Problem Statement:-** A Java program to accessing Array Elements using for Loop.

**Code:-**

```
package lab9;
```

```
public class Student {  
    public int roll_no;  
    public String name;
```

```
Student(int roll_no, String name)  
{  
    this.roll_no = roll_no;  
    this.name = name;  
}  
}
```

```
package lab9;
```

```
public class studentprint {
```

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

```
        // declares an Array of integers.  
        Student[] arr;
```

```
        // allocating memory for 5 objects of type Student.  
        arr = new Student[5];
```

```
        // initialize the elements of the array
```

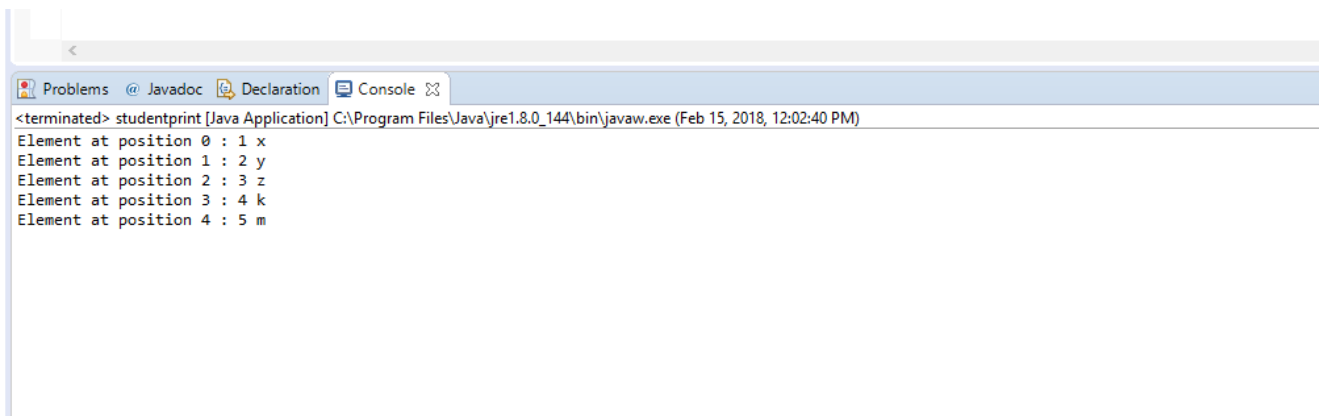
```
        arr[0] = new Student(1,"x");  
        arr[1] = new Student(2,"y");  
        arr[2] = new Student(3,"z");  
        arr[3] = new Student(4,"k");  
        arr[4] = new Student(5,"m");
```

```
        // accessing the elements of the specified array
```

```
        for (int i = 0; i < arr.length; i++)  
            System.out.println("Element at position " + i + " : " +  
                               arr[i].roll_no + " " + arr[i].name);
```

```
    }  
}
```

## **Screenshot:-**



**Explanation:-** An array of objects is created just like an array of primitive type data items in the following way.

```
Student[] arr = new Student[5];
```

The studentArray contains five memory spaces each of size of student class in which the address of five Student objects can be stored. The Student objects have to be instantiated using the constructor of the Student class and their references should be assigned to the array elements in the following way.

```
Student arr = new Student[5];
```

Each element in the array is accessed via its index. The index begins with 0 and ends at (total array size)-1. All the elements of array can be accessed using Java for Loop.

```
// accessing the elements of the specified array
for (int i = 0; i < arr.length; i++)
    System.out.println("Element at index " + i + " : "+ arr[i]);
```

# **Lab Practice**

## **1) Objective:** Implementation of Arrays in Java.

**Problem Statement:-** Write a Java program to sum values of an array.

Input:

```
array [] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};
```

output:

```
The sum is 55
```

Note: Method for finding the sum should be called in the main class.

## **2) Objective:** Implementation of Arrays in Java.

**Problem Statement:-** Write a Java program to print the following grid using array.

Output:

```
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0
```

Note: Method for printing the grid should be called in the main class.

## **3) Objective:** Implementation of Arrays in Java.

**Problem Statement:-** Write a Java program to find the index position of an array element.

Input:

```
int[] array = {14, 25, 14, 56, 15, 36, 56, 77, 18, 29, 49};
```

Sample Output:

```
Index position of 14 is: 0
```

```
Index position of 77 is: 7
```

Note: Method for find the index should be called in the main class.

## **4) Objective:** Implementation of Arrays in Java.

**Problem Statement:-** Write a Java program to insert an element (specific position) into an array.

Input array or Original Array:

[21, 25, 14, 56, 15, 36, 56, 77, 18, 29, 49]

New Array or output array after insert element 5 between 14 and 56:

[21, 25, 14, 5, 56, 15, 36, 56, 77, 18, 29]

Note: Method for insert an element should be called in the main class.

## **5) Objective:** Implementation of Arrays in Java.

**Problem Statement:-** Write a Java program to find the maximum and minimum value of an array.

Input:

Original Array: [21, 25, 14, 56, 15, 36, 56, 77, 18, 29, 49, 100]

Output:

Maximum value for the above array = 100

Minimum value for the above array = 14

Note: Method for to find the maximum and minimum should be called in the main class.

## **6) Objective:** Implementation of Arrays in Java.

**Problem Statement:-** Write a Java program to find the duplicate values of an array of integer values.

Input:

int[] array = {1, 2, 5, 5, 6, 6, 7, 2, 9, 10};

Output:

Duplicate Element: 2

Duplicate Element: 5

Duplicate Element: 6

Note: Method for finding the duplicate should be called in the main class.

## **7) Objective:** Implementation of Arrays in Java.

**Problem Statement:-** Write a Java program to reverse an array of integer values.

Original array: [89, 235, 199, 1456, 213, 1458, 258, 1254, 1472, 2365, 1456, 2165, 157, 256]

Output (Reverse array): [256, 157, 2165, 1456, 2365, 1472, 1254, 258, 1458, 213, 1456, 199, 235, 89]

Note: Method for to reverse an array should be called in the main class.

## **8) Objective:** Implementation of Arrays in Java.

**Problem Statement:-** Write a Java program to add two matrices of the same size.

```
Input number of rows of matrix
2
Input number of columns of matrix
2
Input elements of first matrix
1
2
3
4
Input the elements of second matrix
5
6
7
8
Sum of the matrices: -
6      8
10     12
```

Note: Method for add two matrices should be called in the main class.

## **9) Objective:** Implementation of Arrays in Java.

**Problem Statement:-** Write a Java program to find the common elements between two arrays of integers.

Input arrays:-

Array1: [1, 2, 5, 5, 8, 9, 7, 10, 15]

Array2: [1, 0, 6, 15, 6, 4, 7, 0, 9]

Output:-

Common element is: 1

Common element is: 7

Common element is: 15

Note: Method for find the common elements between two arrays should be called in the main class.

## **10) Objective:** Implementation of Arrays in Java.

**Problem Statement:-** Write a Java program to find all pairs of elements in an array whose sum is equal to a specified number.

Input:

```
pairs_value1= ({2, 7, 4, -5, 11, 5, 20}, 15);  
  
pairs_value2= ({14, -15, 9, 16, 25, 45, 12, 8}, 30);
```

Output:-

Pairs of elements and their sum:

4 + 11 = 15

-5 + 20 = 15

Pairs of elements and their sum:

14 + 16 = 30

-15 + 45 = 30

Note: Method for find all pairs of elements should be called in the main class.

## **11)**      **Objective:** Implementation of Arrays in Java.

**Problem Statement:-** Write a Java program to arrange the elements of an given array of integers where all positive integers appear before all the negative integers.

Input:

Original array: [-4, 8, 6, -5, 6, -2, 1, 2, 3, -11]

Output:

New array: [8, 6, 6, 1, 2, -4, -5, -2, 3, -11]

Note: Method for arranges the elements should be called in the main class.

## **12)**      **Objective:** Implementation of Arrays in Java.

**Problem Statement:-** Write a Java program to sort an array of positive integers of an given array, in the sorted array the value of the first element should be maximum, second value should be minimum value, third should be second maximum, fourth second be second minimum and so on.

Input:

Original Array

[10, 20, 30, 40, 50, 60, 70, 80, 90, 100]

Output:

New Array

[100, 10, 90, 20, 80, 30, 70, 40, 60, 50]

## **13)**      **Objective:** Implementation of Arrays in Java.

**Problem Statement:-** Write a Java program to find the two elements from a given array of positive and negative numbers such that their sum is closest to zero.

**Input:-**

```
int arr[] = {1, 5, -4, 7, 8, -6};
```

**Output:**

Two elements whose sum is minimum are 5 and -4

Note: Method to find the two elements should be called in the main class.

## **14)**      **Objective:** Implementation of Arrays in Java.

**Problem Statement:-** Write a Java program to cyclically rotate a given array clockwise by one.

**Input:**

Original array:

```
[10, 20, 30, 40, 50, 60]
```

**Output:**

Rotated array:

```
[60, 10, 20, 30, 40, 50]
```

Note: Method to cyclically rotate a given array should be called in the main class.

## **15)**      **Objective:** Implementation of Arrays in Java.

**Problem Statement:-** Write a Java program to do Row wise sorting in 2D array

**Input :**

```
77 11 22 3
11 89 1 12
32 11 56 7
11 22 44 33
```

**Output :**

```
3 11 22 77
1 11 12 89
7 11 32 56
11 22 33 44
```

Note: Method for do Row wise sorting should be called in the main class.

