Name: Vinayak Kumar Singh **Subject**: Java Programming Lab

Register No: 23MCA1030

Exercise 8 Arrays and ArrayList

1. Write a Java program to find the maximum element in an array.

Code:

```
public class MaxInArray {
    public static int findMax(int[] arr) {
        // Initialize maximum element with first element of array
        int max = arr[0];
        // Traverse the array
        for (int i = 1; i < arr.length; i++) {
            // If the current element is greater than the maximum then update the
maximum
            if (arr[i] > max) {
                max = arr[i];
            }
        }
        // Return the maximum element
        return max;
    }
    public static void main(String[] args) {
        // Declare and initialize the array
```

```
        int[] arr = { 25, 11, 7, 75, 56 };
        // Call the function to find the maximum element
        int MaxInArray = findMax(arr);
        System.out.println("23MCA1030 Vinayak Kumar Singh");
        // Print the maximum element
        System.out.println("The maximum element in the array is: " + MaxInArray);
    }
}
```

Output:

```
PROBLEMS  OUTPUT  PORTS  TERMINAL  COMMENTS

d:\Coding\Java\Arrays\2>cd "d:\Coding\Java\Arrays\1\" && javac MaxInArray.java && java MaxInArray
23MCA1030 Vinayak Kumar Singh
The maximum element in the array is: 75
```

2.Implement a Java program to reverse an array without using any additional array.

Code:

```
public class ArrayReverseNoExtraSpace {
    // Function to reverse the array in-place
    public static void reverseArray(int[] arr) {
        int start = 0;
        int end = arr.length - 1;
        // Swap elements at the start and end indices
        while (start < end) {
            int temp = arr[start];
            arr[start] = arr[end];
```

```
        arr[end] = temp;
        start++;
        end--;
    }
}
public static void main(String[] args) {
    int[] arr = {1, 2, 3, 4, 5};
    System.out.println("23MCA1030 Vinayak Kumar Singh");
    System.out.print("Original array: [");
    for (int i = 0; i < arr.length; i++) {
        System.out.print(arr[i]);
        if (i < arr.length - 1) {
            System.out.print(", ");
        }
    }
    System.out.println("]");
    reverseArray(arr);
    System.out.print("Reversed array: [");
    for (int i = 0; i < arr.length; i++) {
        System.out.print(arr[i]);
        if (i < arr.length - 1) {
            System.out.print(", ");
        }
    }
    System.out.println("]");
```

```
    }
}
```

Output:

```
PROBLEMS  OUTPUT  PORTS  TERMINAL  COMMENTS

d:\Coding\Java\Arrays\2>cd "d:\Coding\Java\Arrays\2\" && javac ArrayReverseNoExtraSpace.java && java ArrayReverseNoExtraSpace
23MCA1030 Vinayak Kumar Singh
Original array: [1, 2, 3, 4, 5]
Reversed array: [5, 4, 3, 2, 1]
```

3. Create a Java program to sort an array of integers in ascending order using the bubble sort algorithm.

Code:

```
public class BubbleSort {
    // Function to sort the array using bubble sort algorithm
    public static void bubbleSort(int[] arr) {
        int n = arr.length;
        boolean swapped;
        for (int i = 0; i < n - 1; i++) {
            swapped = false;
            for (int j = 0; j < n - i - 1; j++) {
                if (arr[j] > arr[j + 1]) {
                    // Swap arr[j] and arr[j+1]
                    int temp = arr[j];
                    arr[j] = arr[j + 1];
                    arr[j + 1] = temp;
                    swapped = true;
                }
            }
            // If no two elements were swapped by the inner loop, break
```

```
        if (!swapped) {
            break;
        }
    }
}
public static void main(String[] args) {
    int[] arr = {64, 34, 25, 12, 22, 11, 90};
    System.out.println("23MCA1030 Vinayak Kumar Singh");
    System.out.print("Original array: [");
    for (int i = 0; i < arr.length; i++) {
        System.out.print(arr[i]);
        if (i < arr.length - 1) {
            System.out.print(", ");
        }
    }
    System.out.println("]");
    bubbleSort(arr);
    System.out.print("Sorted array: [");
    for (int i = 0; i < arr.length; i++) {
        System.out.print(arr[i]);
        if (i < arr.length - 1) {
            System.out.print(", ");
        }
    }
    System.out.println("]");
```

```
    }
}
```

Output:

```
PROBLEMS  OUTPUT  PORTS  TERMINAL  COMMENTS

d:\Coding\Java\Arrays\3>cd "d:\Coding\Java\Arrays\3\" && javac BubbleSort.java && java BubbleSort
23MCA1030 Vinayak Kumar Singh
Original array: [64, 34, 25, 12, 22, 11, 90]
Sorted array: [11, 12, 22, 25, 34, 64, 90]
```

4. Write a Java program to find the second largest element in an array.

Code:

```
public class SecondLargestInArray {
    public static void main(String[] args) {
        int[] arr = {5, 8, 2, 10, 7, 3};
        int secondLargest = findSecondLargest(arr);
        System.out.println("23MCA1030 Vinayak Kumar Singh");
        System.out.print("Original array is: ");
        printArray(arr);
        System.out.println("\nThe second largest element in array is: " +
secondLargest);
    }
    public static int findSecondLargest(int[] arr) {
        int largest = Integer.MIN_VALUE;
        int secondLargest = Integer.MIN_VALUE;
        // Find the largest element
        for (int num : arr) {
            if (num > largest) {
                secondLargest = largest;
                largest = num;
            }
        }
        // Update the second largest element
```

```
        for (int num : arr) {
            if (num > secondLargest && num != largest) {
                secondLargest = num;
            }
        }
        return secondLargest;
    }
    public static void printArray(int[] arr) {
        for (int i = 0; i < arr.length; i++) {
            if (i > 0) {
                System.out.print(", ");
            }
            System.out.print(arr[i]);
        }
    }
}
```

Output:

```
d:\Coding\Java\Arrays\4>cd "d:\Coding\Java\Arrays\4\" && javac SecondLargestInArray.java && java SecondLargestInArray
23MCA1030 Vinayak Kumar Singh
Original array is: 5, 8, 2, 10, 7, 3
The second largest element in array is: 8
```

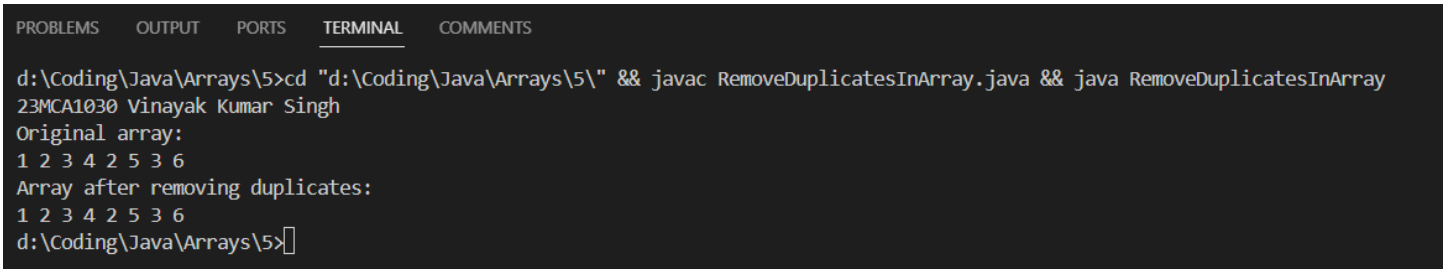
5. Implement a Java program to remove duplicate elements from an array without using any additional data structure.

Code:

```
public class RemoveDuplicatesInArray {
    public static void main(String[] args) {
        int[] arr = {1, 2, 3, 4, 2, 5, 3, 6};
        System.out.println("23MCA1030 Vinayak Kumar Singh");
        System.out.println("Original array: ");
        printArray(arr);
```

```
        int length = removeDuplicates(arr);
        System.out.println("\nArray after removing duplicates: ");
        printArray(arr, length);
    }
    public static int removeDuplicates(int[] arr) {
        if (arr.length == 0) {
            return 0;
        }
        int j = 0;
        for (int i = 1; i < arr.length; i++) {
            if (arr[i] != arr[j]) {
                j++;
                arr[j] = arr[i];
            }
        }
        return j + 1;
    }
    public static void printArray(int[] arr) {
        for (int num : arr) {
            System.out.print(num + " ");
        }
    }
    public static void printArray(int[] arr, int length) {
        for (int i = 0; i < length; i++) {
            System.out.print(arr[i] + " ");
        }
    }
}
```

Output:

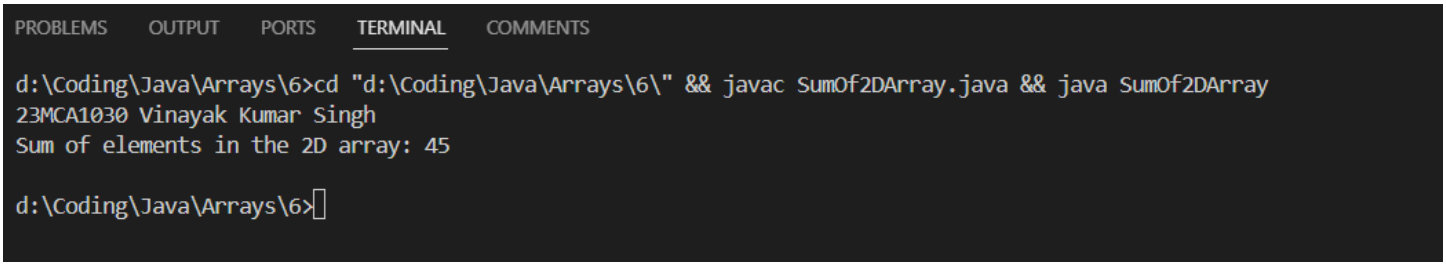


6. Create a Java program to compute the sum of elements in a 2D array.

Code:

```java
public class SumOf2DArray {
    public static int sumOf2DArray(int[][] arr) {
        int sum = 0;
        for (int[] row : arr) {
            for (int num : row) {
                sum += num;
            }
        }
        return sum;
    }
    public static void main(String[] args) {
        int[][] arr = {
            {1, 2, 3},
            {4, 5, 6},
            {7, 8, 9}
        };
        int sum = sumOf2DArray(arr);
        System.out.println("23MCA1030 Vinayak Kumar Singh");
        System.out.println("Sum of elements in the 2D array: " + sum);
    }
}
```

Output:



7. Write a Java program to find the intersection of two arrays.

Code

```
import java.util.HashSet;
import java.util.Set;
public class ArrayIntersection {
    public static void main(String[] args) {
        int[] arr1 = {1, 2, 3, 4, 5};
        int[] arr2 = {4, 5, 6, 7, 8};
        int[] intersection = findIntersection(arr1, arr2);
        System.out.println("23MCA1030 Vinayak Kumar Singh");
        System.out.print("Intersection of the two arrays: ");
        printArray(intersection);
    }
    public static int[] findIntersection(int[] arr1, int[] arr2) {
        Set<Integer> set1 = new HashSet<>();
        Set<Integer> intersection = new HashSet<>();
        // Add all elements of arr1 to set1
        for (int num : arr1) {
            set1.add(num);
        }
        // Get intersection by checking elements of arr2 in set1
        for (int num : arr2) {
            if (set1.contains(num)) {
                intersection.add(num);
```

```
            }
        }
        // Convert the intersection set to an array
        int[] intersectionArray = new int[intersection.size()];
        int index = 0;
        for (int num : intersection) {
            intersectionArray[index++] = num;
        }
        return intersectionArray;
    }
    public static void printArray(int[] arr) {
        for (int num : arr) {
            System.out.print(num + " ");
        }
        System.out.println();
    }
}
```

Output:

```
PROBLEMS  OUTPUT  PORTS  TERMINAL  COMMENTS

d:\Coding\Java\Arrays\7>cd "d:\Coding\Java\Arrays\7\" && javac ArrayIntersection.java && java ArrayIntersection
23MCA1030 Vinayak Kumar Singh
Intersection of the two arrays: 4 5
```

8. Implement a Java program to rotate an array to the right by a given number of steps.

Code:

```
public class ArrayRotate {
    public static void main(String[] args) {
        int[] arr = {1, 2, 3, 4, 5};
        int steps = 2;
        System.out.println("23MCA1030 Vinayak Kumar Singh");
```

```
        System.out.println("Original array: ");
        printArray(arr);
        rotateArray(arr, steps);
        System.out.println("\nArray after rotation: ");
        printArray(arr);
    }
    public static void rotateArray(int[] arr, int steps) {
        int n = arr.length;
        steps = steps % n; // Normalize steps to work for any value
        // Reverse entire array
        reverseArray(arr, 0, n - 1);
        // Reverse first n - steps elements
        reverseArray(arr, 0, n - steps - 1);
        // Reverse remaining steps elements
        reverseArray(arr, n - steps, n - 1);
    }
    public static void reverseArray(int[] arr, int start, int end) {
        while (start < end) {
            int temp = arr[start];
            arr[start] = arr[end];
            arr[end] = temp;
            start++;
            end--;
        }
    }
    public static void printArray(int[] arr) {
        for (int num : arr) {
            System.out.print(num + " ");
        }
        System.out.println();
```

```
    }
}
```

Output:

```
d:\Coding\Java\Arrays\7>cd "d:\Coding\Java\Arrays\8\" && javac ArrayRotate.java && java ArrayRotate
23MCA1030 Vinayak Kumar Singh
Original array:
1 2 3 4 5

Array after rotation:
3 4 5 1 2

d:\Coding\Java\Arrays\8>
```

9. Create a Java program to find the frequency of each element in an array.

Code:

```
import java.util.HashMap;
import java.util.Map;
public class FindFrequency {
    public static Map<Integer, Integer> findFrequency(int[] arr) {
        Map<Integer, Integer> frequencyMap = new HashMap<>();
        for (int num : arr) {
            frequencyMap.put(num, frequencyMap.getOrDefault(num, 0) + 1);
        }
        return frequencyMap;
    }
    public static void main(String[] args) {
        int[] arr = {1, 2, 3, 2, 4, 1, 5, 2, 3};
        Map<Integer, Integer> frequencyMap = findFrequency(arr);
        frequencyMap.forEach((key, value) -> System.out.println(key + " -> " +
value));
    }
}
```

Output:

```
PROBLEMS  OUTPUT  PORTS  TERMINAL  COMMENTS

d:\Coding\Java\Arrays\9>cd "d:\Coding\Java\Arrays\9\" && javac FindFrequency.java && java FindFrequency
1 -> 2
2 -> 3
3 -> 2
4 -> 1
5 -> 1
```

10. Write a Java program to check if two arrays are equal or not.

Code:

```java
public class ArrayEqual {
    public static void main(String[] args) {
        int[] arr1 = {1, 2, 3, 4, 5};
        int[] arr2 = {1, 2, 3, 4, 5};
        int[] arr3 = {5, 4, 3, 2, 1};
        System.out.println("23MCA1030 Vinayak Kumar Singh");
        System.out.println("arr1 and arr2 equal? " + areArraysEqual(arr1, arr2));
        System.out.println("arr1 and arr3 equal? " + areArraysEqual(arr1, arr3));
    }
    public static boolean areArraysEqual(int[] arr1, int[] arr2) {
        // Check if the arrays have the same length
        if (arr1.length != arr2.length) {
            return false;
        }
        // Compare each element of the arrays
        for (int i = 0; i < arr1.length; i++) {
            if (arr1[i] != arr2[i]) {
                return false;
            }
        }
        return true;
    }
}
```

Output:

```
PROBLEMS  OUTPUT  PORTS  TERMINAL  COMMENTS

d:\Coding\Java\Arrays\10>cd "d:\Coding\Java\Arrays\10\" && javac ArrayEqual.java && java ArrayEqual
23MCA1030 Vinayak Kumar Singh
arr1 and arr2 equal? true
arr1 and arr3 equal? false
```

11. Write a Java program to add elements to an ArrayList and display its contents

Code:

```java
import java.util.ArrayList;
public class ArrayListAddQuestion{
    public static void main(String[] args) {
        // Create an ArrayList of Strings
        ArrayList<String> names = new ArrayList<>();
        // Add elements to the ArrayList
        names.add("Aman");
        names.add("Binay");
        names.add("Chirag");
        names.add("Dheeraj");
        // Display contents of the ArrayList
        System.out.println("23MCA1030 Vinayak Kumar Singh");
        System.out.println("Contents of the ArrayList:");
        for (String name : names) {
            System.out.println(name);
        }
    }
}
```

Output:

```
PROBLEMS  OUTPUT  PORTS  TERMINAL  COMMENTS

d:\Coding\Java\Arrays\10>cd "d:\Coding\Java\Arrays\11\" && javac ArrayListAddQuestion.java && java ArrayListAddQuestion
23MCA1030 Vinayak Kumar Singh
Contents of the ArrayList:
Aman
Binay
Chirag
Dheeraj
```

12. Implement a Java program to remove a specific element from an ArrayList.

Code:

```
import java.util.ArrayList;
public class RemoveElementArrayList {
    public static void main(String[] args) {
        // Create an ArrayList
        ArrayList<String> names = new ArrayList<>();
        names.add("Aman");
        names.add("Binay");
        names.add("Chirag");
        names.add("Dheeraj");
        System.out.println("23MCA1030 Vinayak Kumar Singh");
        System.out.println("Original ArrayList: " + names);
        // Remove a specific element
        String elementToRemove = "Chirag";
        boolean isRemoved = names.remove(elementToRemove);
        if (isRemoved) {
            System.out.println("Element \"" + elementToRemove + "\" is removed from the ArrayList.");
        } else {
            System.out.println("Element \"" + elementToRemove + "\" not found in the ArrayList.");
        }
        System.out.println("Updated ArrayList: " + names);
    }
}
```

Output:

```
PROBLEMS  OUTPUT  PORTS  TERMINAL  COMMENTS

d:\Coding\Java\Arrays\12>cd "d:\Coding\Java\Arrays\12\" && javac RemoveElementArrayList.java && java RemoveElementArrayList
23MCA1030 Vinayak Kumar Singh
Original ArrayList: [Aman, Binay, Chirag, Dheeraj]
Element "Chirag" is removed from the ArrayList.
Updated ArrayList: [Aman, Binay, Dheeraj]
```

13. Create a Java program to sort an ArrayList of strings in alphabetical order.

Code:

```java
import java.util.ArrayList;
import java.util.Collections;
public class SortArrayList {
    public static void main(String[] args) {
        // Create an ArrayList of strings
        ArrayList<String> names = new ArrayList<>();
        names.add("Vinayak");
        names.add("Aman");
        names.add("Raj");
        names.add("Binay");
        names.add("Chirag");
        names.add("Tanmay");
        names.add("Dheeraj");
        System.out.println("23MCA1030 Vinayak Kumar Singh");
        System.out.println("Original ArrayList: " + names);
        // Sort the ArrayList in alphabetical order
        Collections.sort(names);
        System.out.println("Sorted ArrayList: " + names);
    }
}
```

Output:

```
PROBLEMS  OUTPUT  PORTS  TERMINAL  COMMENTS

d:\Coding\Java\Arrays\13>cd "d:\Coding\Java\Arrays\13\" && javac SortArrayList.java && java SortArrayList
23MCA1030 Vinayak Kumar Singh
Original ArrayList: [Vinayak, Aman, Raj, Binay, Chirag, Tanmay, Dheeraj]
Sorted ArrayList: [Aman, Binay, Chirag, Dheeraj, Raj, Tanmay, Vinayak]

d:\Coding\Java\Arrays\13>
```

14. Write a Java program to remove all duplicate elements from an ArrayList.

Code:

```
import java.util.ArrayList;
import java.util.LinkedHashSet;

public class RemoveDuplicateArrayList {
    public static void main(String[] args) {
        // Create an ArrayList with duplicates
        ArrayList<String> names = new ArrayList<>();
        names.add("Aman");
        names.add("Aman");
        names.add("Raj");
        names.add("Binay");
        names.add("Chirag");
        names.add("Chirag");
        names.add("Tanmay");
        names.add("Dheeraj");
        System.out.println("Original ArrayList: " + names);
        // Remove duplicates using a LinkedHashSet
        LinkedHashSet<String> uniqueNames = new LinkedHashSet<>(names);
        ArrayList<String> namesWithoutDuplicates = new
ArrayList<>(uniqueNames);
        System.out.println("ArrayList without duplicates: " +
namesWithoutDuplicates);
    }
}
```

Output:

```
PROBLEMS  OUTPUT  PORTS  TERMINAL  COMMENTS

Microsoft Windows [Version 10.0.22631.3296]
(c) Microsoft Corporation. All rights reserved.

D:\Coding\Java\Arrays>cd "d:\Coding\Java\Arrays\14\" && javac RemoveDuplicateArrayList.java && java RemoveDuplicateArrayList
Original ArrayList: [Aman, Aman, Raj, Binay, Chirag, Chirag, Tanmay, Dheeraj]
ArrayList without duplicates: [Aman, Raj, Binay, Chirag, Tanmay, Dheeraj]

d:\Coding\Java\Arrays\14>
```

15. Implement a Java program to find the length of the longest subsequence of increasing integers in an ArrayList.

Code:

```
import java.util.ArrayList;
public class LongestIncreasingSubsequence {
    public static int findLongestSubsequenceLength(ArrayList<Integer> nums) {
        int maxLength = 1;  // Length of longest subsequence found so far
        int currentLength = 1;  // Length of current subsequence being considered
        for (int i = 1; i < nums.size(); i++) {
            if (nums.get(i) > nums.get(i - 1)) {
                currentLength++;
                maxLength = Math.max(maxLength, currentLength);
            } else {
                currentLength = 1;  // Reset for new subsequence
            }
        }
        return maxLength;
    }
    public static void main(String[] args) {
        ArrayList<Integer> list = new ArrayList<>() {{
            add(4);
            add(2);
            add(3);
            add(6);
            add(10);
            add(1);
            add(12);
        }};
        int maxLength = findLongestSubsequenceLength(list);
        System.out.println("23MCA1030 Vinayak Kumar Singh");
        System.out.println("Length of longest increasing subsequence: " +
maxLength);
    }
}
```

Output:

```
PROBLEMS  OUTPUT  PORTS  TERMINAL  COMMENTS                                                    Code

Microsoft Windows [Version 10.0.22631.3296]
(c) Microsoft Corporation. All rights reserved.

D:\Coding\Java\Arrays>cd "d:\Coding\Java\Arrays\15\" && javac LongestIncreasingSubsequence.java && java LongestIncreasingSubsequence
23MCA1030 Vinayak Kumar Singh
Length of longest increasing subsequence: 4
```

16. Create a Java program to shuffle the elements of an ArrayList.

Code:

```java
import java.util.ArrayList;
import java.util.Collections;
public class ShuffleArrayList {
    public static void main(String[] args) {
        ArrayList<String> list = new ArrayList<>() {{
            add("Blue");
            add("Yellow");
            add("Purple");
            add("Red");
            add("Green");
        }};
        System.out.println("23MCA1030 Vinayak Kumar Singh");
        System.out.println("Original list: " + list);
        // Using Collections.shuffle() method:
        Collections.shuffle(list);
        System.out.println("Shuffled list (Collections.shuffle()): " + list);
    }
}
```

Output:

```
PROBLEMS  OUTPUT  PORTS  TERMINAL  COMMENTS

d:\Coding\Java\Arrays\16>cd "d:\Coding\Java\Arrays\16\" && javac ShuffleArrayList.java && java ShuffleArrayList
23MCA1030 Vinayak Kumar Singh
Original list: [Blue, Yellow, Purple, Red, Green]
Shuffled list (Collections.shuffle()): [Blue, Green, Yellow, Purple, Red]
```

17. Write a Java program to find the intersection of two ArrayLists.

Code:

```
import java.util.ArrayList;
import java.util.HashSet;
import java.util.Set;
public class IntersectionArrayList {
   public static ArrayList<Integer> findIntersection(ArrayList<Integer> list1,
ArrayList<Integer> list2) {
      Set<Integer> set1 = new HashSet<>(list1);
      Set<Integer> intersection = new HashSet<>();
      for (int num : list2) {
         if (set1.contains(num)) {
            intersection.add(num);
         }
      }
      return new ArrayList<>(intersection);
   }
   public static void main(String[] args) {
      ArrayList<Integer> list1 = new ArrayList<>() {{
         add(1);
         add(2);
         add(3);
         add(4);
         add(5);
      }};
      ArrayList<Integer> list2 = new ArrayList<>() {{
         add(3);
         add(4);
         add(5);
         add(6);
         add(7);
      }};
      ArrayList<Integer> intersection = findIntersection(list1, list2);
      System.out.println("23MCA1030 Vinayak Kumar Singh");
      System.out.println("Intersection: " + intersection);
   }
}
```

Output:

```
PROBLEMS  OUTPUT  PORTS  TERMINAL  COMMENTS

d:\Coding\Java\Arrays\16>cd "d:\Coding\Java\Arrays\17\" && javac IntersectionArrayList.java && java IntersectionArrayList
23MCA1030 Vinayak Kumar Singh
Intersection: [3, 4, 5]
```

18. Implement a Java program to convert an ArrayList to an array.

Code:

```java
import java.util.ArrayList;
import java.util.Arrays;
public class ArrayListToArray {
    public static void main(String[] args) {
        // Create an ArrayList
        ArrayList<String> fruits = new ArrayList<>();
        fruits.add("Apple");
        fruits.add("Banana");
        fruits.add("Cherry");
        fruits.add("Date");
        System.out.println("23MCA1030 Vinayak Kumar Singh");
        System.out.println("ArrayList: " + fruits);
        // Convert ArrayList to an array
        String[] fruitsArray = fruits.toArray(new String[0]);
        System.out.println("Array: " + Arrays.toString(fruitsArray));
    }
}
```

Output:

```
PROBLEMS  OUTPUT  PORTS  TERMINAL  COMMENTS

d:\Coding\Java\Arrays\18>cd "d:\Coding\Java\Arrays\18\" && javac ArrayListToArray.java && java ArrayListToArray
23MCA1030 Vinayak Kumar Singh
ArrayList: [Apple, Banana, Cherry, Date]
Array: [Apple, Banana, Cherry, Date]
```

19. Create a Java program to find the union of two ArrayLists.

Code:

```
import java.util.ArrayList;
import java.util.HashSet;
import java.util.List;
import java.util.Set;
public class UnionArrayList {
    public static List<Integer> findUnion(ArrayList<Integer> list1,
ArrayList<Integer> list2) {
        Set<Integer> set = new HashSet<>();
        set.addAll(list1);
        set.addAll(list2);
        return new ArrayList<>(set);
    }
    public static void main(String[] args) {
        ArrayList<Integer> list1 = new ArrayList<>() {{
            add(1);
            add(2);
            add(3);
            add(4);
            add(5);
        }};
        ArrayList<Integer> list2 = new ArrayList<>() {{
            add(3);
            add(4);
            add(5);
            add(6);
```

```
            add(7);
        }};
        List<Integer> union = findUnion(list1, list2);
        System.out.println("23MCA1030 Vinayak Kumar Singh");
        System.out.println("Union: " + union);
    }
}
```

Output:

```
PROBLEMS  OUTPUT  PORTS  TERMINAL  COMMENTS

d:\Coding\Java\Arrays\19>cd "d:\Coding\Java\Arrays\19\" && javac UnionArrayList.java && java UnionArrayList
23MCA1030 Vinayak Kumar Singh
Union: [1, 2, 3, 4, 5, 6, 7]
```

20. Write a Java program to check if an ArrayList is empty or not.

Code:

```
import java.util.ArrayList;
public class ArrayListEmptyChecker {
  public static void main(String[] args) {
    // Create an empty ArrayList
    ArrayList<String> myList1 = new ArrayList<>();
    // Create a non-empty ArrayList
    ArrayList<Integer> myList2 = new ArrayList<Integer>() {{
      add(1);
      add(2);
      add(3);
    }};
    // Check if the ArrayLists are empty
    System.out.println("23MCA1030 Vinayak Kumar Singh");
    System.out.println("myList1 is empty: " + myList1.isEmpty());
    System.out.println("myList2 is empty: " + myList2.isEmpty());
```

```
    }
}
```

Output:

```
PROBLEMS  OUTPUT  PORTS  TERMINAL  COMMENTS

d:\Coding\Java\Arrays\20>cd "d:\Coding\Java\Arrays\20\" && javac ArrayListEmptyChecker.java && java ArrayListEmptyChecker
23MCA1030 Vinayak Kumar Singh
myList1 is empty: true
myList2 is empty: false
```