

□□□□□ Assignment no-7□

□□□□□□

1. Write a program to print elements of Array ?

Soln:-

```
public class PrintArray {  
    public static void main(String[] args) {  
  
        int[] array = {1, 2, 3, 4, 5};  
  
        System.out.println("Elements of the array:");  
        for (int i = 0; i < array.length; i++) {  
            System.out.println(array[i]);  
        }  
    }  
}
```

OutPut:-

PS C:\Users\DELL\Desktop\CDAC\Java OOPJ\Assignment 7> javac PrintArray.java

PS C:\Users\DELL\Desktop\CDAC\Java OOPJ\Assignment 7> java PrintArray

Elements of the array:

1  
2  
3  
4  
5

2) Write a Java program to check the equality of two arrays?

Soln:-

import java.util.Arrays;

```
public class ArrayEqualitycheck {  
    public static void main(String[] args) {  
  
        int[] array1 = {1, 2, 3, 4, 5};  
        int[] array2 = {1, 2, 3, 4, 5};  
  
        boolean areEqual = Arrays.equals(array1, array2);  
  
        if (areEqual) {  
            System.out.println("Arrays are equal.");  
        } else {  
            System.out.println("Arrays are not equal.");  
        }  
    }  
}
```

Output:-

PS C:\Users\DELL\Desktop\CDAC\Java OOPJ\Assignment 7> javac ArrayEqualitycheck.java

PS C:\Users\DELL\Desktop\CDAC\Java OOPJ\Assignment 7> java ArrayEqualitycheck

Arrays are equal.

3) Write a Java program to find all pairs of elements in an integer array whose sum is equal to a given number?

Soln:-

import java.util.\*;

```
public class FindPairsWithSum {
```

```

public static void main(String[] args) {
    int[] array = {2, 4, 3, 5, 7, 8, 9};
    int targetSum = 7;

    findPairsWithSum(array, targetSum);
}

public static void findPairsWithSum(int[] array, int targetSum) {
    Map<Integer, Integer> map = new HashMap<>();
    List<List<Integer>> result = new ArrayList<>();

    for (int num : array) {
        int complement = targetSum - num;
        if (map.containsKey(complement)) {
            List<Integer> pair = new ArrayList<>();
            pair.add(num);
            pair.add(complement);
            result.add(pair);
        }
        map.put(num, num);
    }

    if (result.isEmpty()) {
        System.out.println("No pairs found with the given sum.");
    } else {
        System.out.println("Pairs with sum " + targetSum + ":");
        for (List<Integer> pair : result) {
            System.out.println(pair.get(0) + ", " + pair.get(1));
        }
    }
}
}

```

Output:-

PS C:\Users\DELL\Desktop\CDAC\Java OOP\Assignment 7> javac FindPairsWithSum.java

PS C:\Users\DELL\Desktop\CDAC\Java OOP\Assignment 7> java FindPairsWithSum

Pairs with sum 7:

3, 4

5, 2

4) Write a program to reverse an Array in java .

Soln:-

```

public class ReverseArray {
    public static void main(String[] args) {

        int[] array = {1, 2, 3, 4, 5};

        reverseArray(array);

        System.out.println("Reversed Array:");
        for (int num : array) {
            System.out.print(num + " ");
        }
    }

    public static void reverseArray(int[] array) {
        int length = array.length;
        for (int i = 0; i < length / 2; i++) {

```

```

        int temp = array[i];
        array[i] = array[length - i - 1];
        array[length - i - 1] = temp;
    }
}
}

```

Output:-

PS C:\Users\DELL\Desktop\CDAC\Java OOPJ\Assignment 7> javac ReverseArray.java

PS C:\Users\DELL\Desktop\CDAC\Java OOPJ\Assignment 7> java ReverseArray

Reversed Array:

5 4 3 2 1

5) Find out smallest and largest number in a given Array?

Soln:-

```

public class SmallestLargest {
    public static void main(String[] args) {
        int[] array = {5, 3, 9, 1, 7, 2, 8, 4, 6};

```

```

        int smallest = array[0];
        int largest = array[0];

```

```

        for (int i = 1; i < array.length; i++) {
            if (array[i] < smallest) {
                smallest = array[i];
            }
            if (array[i] > largest) {
                largest = array[i];
            }
        }

```

```

        System.out.println("Smallest number: " + smallest);
        System.out.println("Largest number: " + largest);
    }
}

```

OutPut:-

PS C:\Users\DELL\Desktop\CDAC\Java OOPJ\Assignment 7> javac SmallestLargest.java

PS C:\Users\DELL\Desktop\CDAC\Java OOPJ\Assignment 7> java SmallestLargest

Smallest number: 1

Largest number: 9

6) .Print the third-largest number in an array without sorting it

Input: [ 24,54,31,16,82,45,67]

Output: 54 (82 and 67 are the largest and second-largest)

Soln:-

```

public class ThirdLargetNumber {
    public static void main(String[] args) {
        int[] array = {24, 54, 31, 16, 82, 45, 67};

        int firstLargest = Integer.MIN_VALUE;
        int secondLargest = Integer.MIN_VALUE;
        int thirdLargest = Integer.MIN_VALUE;

        for (int num : array) {
            if (num > firstLargest) {
                thirdLargest = secondLargest;

```

```

        secondLargest = firstLargest;
        firstLargest = num;
    } else if (num > secondLargest && num != firstLargest) {
        thirdLargest = secondLargest;
        secondLargest = num;
    } else if (num > thirdLargest && num != secondLargest && num != firstLargest) {
        thirdLargest = num;
    }
}

if (thirdLargest != Integer.MIN_VALUE) {
    System.out.println("Third Largest number: " + thirdLargest);
} else {
    System.out.println("No third largest number found.");
}
}
}

```

OutPut:-

PS C:\Users\DELL\Desktop\CDAC\Java OOPJ\Assignment 7> javac ThirdLargetNumber.java

PS C:\Users\DELL\Desktop\CDAC\Java OOPJ\Assignment 7> java ThirdLargetNumber

Third Largest number: 54

7)Write a program to merge two arrays of integers by reading one number at a time from each array until one of the

Input: [23,60,94,3,102] and [42,16,74]

Output: [23,42,60,16,94,74,3,102]

Soln:-

```
import java.util.*;
```

```

public class ArrayMerge {
    public static void main(String[] args) {
        int[] array1 = {23, 60, 94, 3, 102};
        int[] array2 = {42, 16, 74};

        int[] mergedArray = mergeArrays(array1, array2);

        System.out.println("Merged Array: " + Arrays.toString(mergedArray));
    }
}

```

```

public static int[] mergeArrays(int[] array1, int[] array2) {
    int index1 = 0;
    int index2 = 0;
    int[] mergedArray = new int[array1.length + array2.length];
    int mergedIndex = 0;
}

```

```

while (index1 < array1.length && index2 < array2.length) {
    mergedArray[mergedIndex++] = array1[index1++];
    mergedArray[mergedIndex++] = array2[index2++];
}

```

```

while (index1 < array1.length) {
    mergedArray[mergedIndex++] = array1[index1++];
}

```

```

while (index2 < array2.length) {
    mergedArray[mergedIndex++] = array2[index2++];
}

```

```

        return mergedArray;
    }
}

```

OutPut:-

PS C:\Users\DELL\Desktop\CDAC\Java OOPJ\Assignment 7> javac ArrayMerge.java

PS C:\Users\DELL\Desktop\CDAC\Java OOPJ\Assignment 7> java ArrayMerge

Merged Array: [23, 42, 60, 16, 94, 74, 3, 102]

8). Write a program which takes an array of integers and prints the running average of 3 consecutive integers. In case the array has fewer than 3 integers, there should be no output.

Input: [5,14,35,89,140]

Output: [18, 46, 88]

(Explanation:  $18 = (5 + 14 + 35) / 3$ ,  $46 = (14 + 35 + 89) / 3$ , ...)

Soln:-

```

public class AveragelnArray {
    public static void main(String[] args) {
        int[] array = {5, 14, 35, 89, 140};

        printRunningAverage(array);
    }

    public static void printRunningAverage(int[] array) {
        if (array.length < 3) {
            System.out.println("Input array has fewer than 3 integers.");
            return;
        }

        System.out.print("Output: [");
        for (int i = 0; i <= array.length - 3; i++) {
            int sum = array[i] + array[i + 1] + array[i + 2];
            double average = sum / 3.0;
            if (i != 0) {
                System.out.print(", ");
            }
            System.out.print((int) average);
        }
        System.out.println("]");
    }
}

```

Output:-

PS C:\Users\DELL\Desktop\CDAC\Java OOPJ\Assignment 7> javac AveragelnArray.java

PS C:\Users\DELL\Desktop\CDAC\Java OOPJ\Assignment 7> java AveragelnArray

Output: [18, 46, 88]

9) Write a program which generates the series 1,4,27,16,125,36

Soln:-

```

public class GenerateSeries {
    public static void main(String[] args) {
        int[] series = generateSeries(6);

        System.out.print("Series: ");
        for (int num : series) {
            System.out.print(num + " ");
        }
    }
}

```

```

public static int[] generateSeries(int length) {
    int[] series = new int[length];
    int count = 1;
    int index = 0;

    while (index < length) {
        if (count % 2 == 1) {

            series[index] = count * count * count;
        } else {

            series[index] = count * count;
        }
        count++;
        index++;
    }

    return series;
}

```

OutPut:-

PS C:\Users\DELL\Desktop\CDAC\Java OOPJ\Assignment 7> javac GenerateSeries.java

PS C:\Users\DELL\Desktop\CDAC\Java OOPJ\Assignment 7> java GenerateSeries

Series: 1 4 27 16 125 36

10) Given an array of integers, print whether the numbers are in ascending order or in descending order or in random order.

Input: [5,14,35,90,139] Output: Ascending

Input: [88,67,35,14,-12] Output: Descending

Input: [65,14,129,34,7] Output: Random

Soln:-

```

public class OrderChecker {
    public static void main(String[] args) {
        int[] array1 = {5, 14, 35, 90, 139};
        int[] array2 = {88, 67, 35, 14, -12};
        int[] array3 = {65, 14, 129, 34, 7};

        printOrder(array1);
        printOrder(array2);
        printOrder(array3);
    }

    public static void printOrder(int[] array) {
        boolean ascending = true;
        boolean descending = true;

        for (int i = 0; i < array.length - 1; i++) {
            if (array[i] < array[i + 1]) {
                descending = false;
            } else if (array[i] > array[i + 1]) {
                ascending = false;
            }
        }

        if (ascending && !descending) {
            System.out.println("Ascending");
        } else if (descending && !ascending) {
            System.out.println("Descending");
        } else {
            System.out.println("Random");
        }
    }
}

```

```
        System.out.println("Random");
    }
}
```

OutPut:-

PS C:\Users\DELL\Desktop\CDAC\Java OOPJ\Assignment 7> javac OrderChecker.java

PS C:\Users\DELL\Desktop\CDAC\Java OOPJ\Assignment 7> java OrderChecker

Ascending

Descending

Random