# **OBJECT ORIENTED PROGRAMMING AND DESIGN**

**Array**

**1. Sort an Array in Ascending Order**

import java.util.Arrays;

public class SortArray {

public static void main(String[] args) {

int[] arr = {5, 2, 8, 3, 1};

Arrays.sort(arr);

System.out.println("Sorted array:");

for (int value : arr) {

System.out.print(value + " ");

}

}

}

**2. Copy Elements of One Array to Another**

public class CopyArray {

public static void main(String[] args) {

int[] original = {10, 20, 30, 40, 50};

int[] copy = new int[original.length];

System.arraycopy(original, 0, copy, 0, original.length);

System.out.println("Copied array:");

for (int value : copy) {

System.out.print(value + " ");

}

}

}

**3. Check if an Array Contains a Specific Value**

public class ContainsValue {

public static void main(String[] args) {

int[] arr = {5, 7, 9, 2, 4};

int target = 9;

boolean found = false;

for (int value : arr) {

if (value == target) {

found = true;

break;

}

}

System.out.println("Array contains " + target + ": " + found);

}

}

**4. Find the Second Largest Element in an Array**

public class SecondLargest {

public static void main(String[] args) {

int[] arr = {15, 24, 36, 48, 29};

int largest = Integer.MIN\_VALUE;

int secondLargest = Integer.MIN\_VALUE;

for (int value : arr) {

if (value > largest) {

secondLargest = largest;

largest = value;

} else if (value > secondLargest && value != largest) {

secondLargest = value;

}

}

System.out.println("Second largest element: " + secondLargest);

}

}

**5. For each loop example**

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.print("Enter the number of elements in the array: ");

int size = scanner.nextInt();

int[] array = new int[size];

System.out.println("Enter the elements of the array:");

for (int i = 0; i < size; i++) {

System.out.print("Enter element " + (i + 1) + ": ");

array[i] = scanner.nextInt();

}

System.out.println("The elements in the array are:");

for (int element : array) {

System.out.print(element + " ");

}

scanner.close();

}

}

**6. Initializing and Printing a 2D Array**

public class Main {

public static void main(String[] args) {

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

for (int i = 0; i < array.length; i++) {

for (int j = 0; j < array[i].length; j++) {

System.out.print(array[i][j] + " ");

}

System.out.println();

}

}

}

**7. For each loop example**

import java.util.Scanner;

public class TwoDArrayExample {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.print("Enter the number of rows: ");

int rows = scanner.nextInt();

System.out.print("Enter the number of columns: ");

int cols = scanner.nextInt();

int[][] array = new int[rows][cols];

System.out.println("Enter the elements of the array:");

for (int i = 0; i < rows; i++) {

for (int j = 0; j < cols; j++) {

System.out.print("Enter element at [" + i + "][" + j + "]: ");

array[i][j] = scanner.nextInt();

}

}

System.out.println("The elements in the 2D array are:");

for (int[] row : array) {

for (int element : row) {

System.out.print(element + " ");

}

System.out.println(); // Newline after each row

}

scanner.close();

}

}

**8. Summing All Elements in a 2D Array**

public class Main {

public static void main(String[] args) {

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

int sum = 0;

for (int i = 0; i < array.length; i++) {

for (int j = 0; j < array[i].length; j++) {

sum += array[i][j];

}

}

System.out.println("Sum of all elements: " + sum);

}

}

**9. Transposing a 2D Array**

public class Main {

public static void main(String[] args) {

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

int[][] transpose = new int[array[0].length][array.length];

for (int i = 0; i < array.length; i++) {

for (int j = 0; j < array[i].length; j++) {

transpose[j][i] = array[i][j];

}

}

for (int i = 0; i < transpose.length; i++) {

for (int j = 0; j < transpose[i].length; j++) {

System.out.print(transpose[i][j] + " ");

}

System.out.println();

}

}

}

**10. Finding the Maximum Element in a 2D Array**

public class Main {

public static void main(String[] args) {

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

int max = array[0][0];

for (int i = 0; i < array.length; i++) {

for (int j = 0; j < array[i].length; j++) {

if (array[i][j] > max) {

max = array[i][j];

}

}

}

System.out.println("Maximum element in the array: " + max);

}

}

**11. Adding Two 2D Arrays**

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.print("Enter the number of rows: ");

int rows = scanner.nextInt();

System.out.print("Enter the number of columns: ");

int cols = scanner.nextInt();

int[][] matrix1 = new int[rows][cols];

int[][] matrix2 = new int[rows][cols];

int[][] sumMatrix = new int[rows][cols];

System.out.println("Enter values for the first matrix:");

for (int i = 0; i < rows; i++) {

for (int j = 0; j < cols; j++) {

System.out.print("Enter element at [" + i + "][" + j + "]: ");

matrix1[i][j] = scanner.nextInt();

}

}

System.out.println("Enter values for the second matrix:");

for (int i = 0; i < rows; i++) {

for (int j = 0; j < cols; j++) {

System.out.print("Enter element at [" + i + "][" + j + "]: ");

matrix2[i][j] = scanner.nextInt();

}

}

for (int i = 0; i < rows; i++) {

for (int j = 0; j < cols; j++) {

sumMatrix[i][j] = matrix1[i][j] + matrix2[i][j];

}

}

System.out.println("Sum of the two matrices:");

for (int i = 0; i < rows; i++) {

for (int j = 0; j < cols; j++) {

System.out.print(sumMatrix[i][j] + " ");

}

System.out.println();

}

scanner.close();

}

}

**12. Multiplying Two 2D Arrays (Matrix Multiplication)**

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.print("Enter the number of rows for the first matrix: ");

int rowsA = scanner.nextInt();

System.out.print("Enter the number of columns for the first matrix: ");

int colsA = scanner.nextInt();

System.out.print("Enter the number of rows for the second matrix: ");

int rowsB = scanner.nextInt();

System.out.print("Enter the number of columns for the second matrix: ");

int colsB = scanner.nextInt();

if (colsA != rowsB) {

System.out.println("Matrix multiplication is not possible. Number of columns in the first matrix must equal the number of rows in the second matrix.");

return;

}

int[][] matrixA = new int[rowsA][colsA];

int[][] matrixB = new int[rowsB][colsB];

int[][] productMatrix = new int[rowsA][colsB];

System.out.println("Enter values for the first matrix:");

for (int i = 0; i < rowsA; i++) {

for (int j = 0; j < colsA; j++) {

System.out.print("Enter element at [" + i + "][" + j + "]: ");

matrixA[i][j] = scanner.nextInt();

}

}

System.out.println("Enter values for the second matrix:");

for (int i = 0; i < rowsB; i++) {

for (int j = 0; j < colsB; j++) {

System.out.print("Enter element at [" + i + "][" + j + "]: ");

matrixB[i][j] = scanner.nextInt();

}

}

for (int i = 0; i < rowsA; i++) {

for (int j = 0; j < colsB; j++) {

for (int k = 0; k < colsA; k++) {

productMatrix[i][j] += matrixA[i][k] \* matrixB[k][j];

}

}

}

System.out.println("Product of the two matrices:");

for (int i = 0; i < rowsA; i++) {

for (int j = 0; j < colsB; j++) {

System.out.print(productMatrix[i][j] + " ");

}

System.out.println();

}

scanner.close();

}

}

**13. Binary search in Arrays class**

import java.util.Arrays;

public class Main {

public static void main(String[] args) {

int[] array = {2, 4, 6, 8, 10, 12, 14};

Arrays.sort(array);

int index = Arrays.binarySearch(array, 8);

System.out.println("Index of 8: " + index); // Output: Index of 8: 3

}

}

**14. equals() method**

import java.util.Arrays;

public class Main {

public static void main(String[] args) {

int[] array1 = {1, 2, 3, 4, 5};

int[] array2 = {1, 2, 3, 4, 5};

int[] array3 = {5, 4, 3, 2, 1};

int[] array4 = {1, 2, 3, 4};

System.out.println("array1 and array2 are equal: " + Arrays.equals(array1, array2));

System.out.println("array1 and array3 are equal: " + Arrays.equals(array1, array3));

System.out.println("array1 and array4 are equal: " + Arrays.equals(array1, array4));

}

}

**15. fill() method - Filling the Entire Array**

import java.util.Arrays;

public class FillExample {

public static void main(String[] args) {

int[] array = new int[5];

// Fill the entire array with the value 10

Arrays.fill(array, 10);

System.out.println("Array after filling: " + Arrays.toString(array)); // [10, 10, 10, 10, 10]

}

}

**16. fill() method - Filling a Range in the Array**

import java.util.Arrays;

public class FillRangeExample {

public static void main(String[] args) {

int[] array = new int[10];

// Fill a specific range (index 3 to 7) with the value 5

Arrays.fill(array, 3, 7, 5);

System.out.println("Array after filling a range: " + Arrays.toString(array)); // [0, 0, 0, 5, 5, 5, 5, 0, 0, 0]

}

}

**17. sort() method – Sorting an Array of Integers**

import java.util.Arrays;

public class SortExample {

public static void main(String[] args) {

int[] array = {5, 2, 8, 3, 1};

// Sort the entire array

Arrays.sort(array);

System.out.println("Sorted array: " + Arrays.toString(array)); // [1, 2, 3, 5, 8]

}

}

**18. sort() method - Sorting a Specific Range in an Array**

import java.util.Arrays;

public class SortRangeExample {

public static void main(String[] args) {

int[] array = {5, 2, 8, 3, 1, 7, 6};

// Sort a specific range (index 2 to 5)

Arrays.sort(array, 2, 6);

System.out.println("Array after range sort: " + Arrays.toString(array)); // [5, 2, 1, 3, 7, 8, 6]

}

}