1st Answer:

import java.util.Arrays; public class ArraySort {

public static void main(String[] args) {

int[] numArray = {3, 1, 4, 2, 5}; Arrays.sort(numArray); System.out.println("Sorted numeric array: " +

Arrays.toString(numArray));

String[] strArray = {"apple", "banana", "cherry", "date"}; Arrays.sort(strArray);

System.out.println("Sorted string array: " + Arrays.toString(strArray));

}

}

2nd Answer :

public class ArraySum {

public static void main(String[] args) { int[] numArray = {1, 2, 3, 4, 5}; int sum = 0;

for (int i = 0; i < numArray.length; i++) { sum += numArray[i];

}

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

}

}

3rd answer :

public class gridPrint {

public static void main(String[] args) { String row = "";

for (int i = 0; i < 6; i++) {

for (int j = 0; j < 6; j++) {

if ((i % 2 == 0) == (j % 2 == 0)) {

row += "\* ";

} else {

row += " ";

}

}

System.out.println(row); row = "";

}

}

}

4th answer:

public class ArrayAvg {

public static void main(String[] args) { int[] numArray = {1, 2, 3, 4, 5}; int sum = 0;

for (int i = 0; i < numArray.length; i++) { sum += numArray[i];

}

double avg = (double) sum / numArray.length; System.out.println("Average of the array elements: " + avg);

}

}

5th answer:

import java.util.Arrays;

public class ArraySearch {

public static void main(String[] args) { int[] numArray = {1, 2, 3, 4, 5}; int searchValue = 3;

boolean found = false;

for (int i = 0; i < numArray.length; i++) { if (numArray[i] == searchValue) {

found = true; break;

}

}

array");

if (found) {

System.out.println("Value " + searchValue + " found in the

} else {

System.out.println("Value " + searchValue + " not found in

the array");

}

}

}

6th answer:

public class ArrayIndex {

public static void main(String[] args) { int[] numArray = {1, 2, 3, 4, 5}; int searchValue = 3;

int index = -1;

for (int i = 0; i < numArray.length; i++) { if (numArray[i] == searchValue) {

index = i; break;

}

}

if (index != -1) {

System.out.println("Value " + searchValue + " found at index

" + index);

} else {

System.out.println("Value " + searchValue + " not found in

the array");

}

}

}

7th Answer:

import java.util.Arrays;

public class ArrayRemove {

public static void main(String[] args) { int[] numArray = {1, 2, 3, 4, 5}; int removeValue = 3;

int[] newArray = new int[numArray.length - 1]; int j = 0;

for (int i = 0; i < numArray.length; i++) { if (numArray[i] != removeValue) {

newArray[j] = numArray[i]; j++;

}

}

System.out.println("Original Array : " + Arrays.toString(numArray));

System.out.println("New Array : " + Arrays.toString(newArray));

}

}

8th Answer:

public class ArrayCopy {

public static void main(String[] args) { int[] original = {1, 2, 3, 4, 5}; int[] copy = new int[original.length];

for (int i = 0; i < original.length; i++) { copy[i] = original[i];

}

// Print the copied array for (int i : copy) {

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

}

}

}

9th answer:

import java.util.Arrays;

public class ArrayInsert {

public static void main(String[] args) { int[] arr = {1, 2, 3, 4, 5};

int pos = 3;

int element = 99;

int[] newArr = insertAt(arr, pos, element); System.out.println(Arrays.toString(newArr));

}

public static int[] insertAt(int[] arr, int pos, int element) { int[] newArr = new int[arr.length + 1]; System.arraycopy(arr, 0, newArr, 0, pos);

newArr[pos] = element;

System.arraycopy(arr, pos, newArr, pos + 1, arr.length - pos); return newArr;

}

}

10th answer:

import java.util.Arrays;

public class MinMaxArray {

public static void main(String[] args) { int[] arr = {5, 8, 2, 9, 3, 1, 4};

int min = arr[0]; int max = arr[0];

for (int i = 1; i < arr.length; i++) { if (arr[i] > max) {

max = arr[i];

}

if (arr[i] < min) { min = arr[i];

}

}

System.out.println("Minimum value in the array: " + min); System.out.println("Maximum value in the array: " + max);

}

}

11th answer:

import java.util.\*;

public class Main{

public static void main(String[] args){

Scanner sc = new Scanner(System.in);

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

int size=sc.nextInt();

int[] array=new int[size];

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

System.out.println("Enter the array elements"+(i+1));

array[i]=sc.nextInt();

}

System.out.println("Original array is");

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

System.out.print(array[i]);

}

System.out.println("Reversed array is");

for(int i=size-1;i>=0;i--){

System.out.print(array[i]);

}

}

}

12th answer:

import java.util.\*;

public class Main{

public static void main(String args[]){

int array[]={10,14,10,3,2,10};

boolean flag=false;

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

for(int j=i+1;j<array.length;j++){

if(array[i]==array[j]){

System.out.println("Duplicate value found : "+array[i]);

flag=true;

}

}

}

if(flag==false){

System.out.println("Duplicate value not found");

}

}

}

13th answer:

import java.util.\*;

public class Main{

public static void main(String args[]){

String array[]={"Apple","Orange","Apple","Grapes"};

boolean flag=false;

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

for(int j=i+1;j<array.length;j++){

if(array[i]==array[j]){

System.out.println("Duplicate value found : "+array[i]);

flag=true;

}

}

}

if(flag==false){

System.out.println("Duplicate value not found");

}

}

}

14th answer:

import java.util.\*;

public class Main{

public static void main(String[]args){

String array1[]={"Banana","Apple","Orange"};

String array2[]={"Grapes","Banana","Pineapple"};

for(int i=0;i<array1.length;i++){

for(int j=0;j<array2.length;j++){

if(array1[i]==array2[j]){

System.out.println("Common elements are : "+array1[i]);

}

}

}

}

}

15th answer:

import java.util.\*;

public class Main{

public static void main(String[]args){

int array1[]={10,20,40,50};

int array2[]={20,40,10,30};

for(int i=0;i<array1.length;i++){

for(int j=0;j<array2.length;j++){

if(array1[i]==array2[j]){

System.out.println("Common elements are : "+array1[i]);

}

}

}

}

}

16th answer:

import java.util.Arrays;

import java.util.LinkedHashSet;

public class RemoveDuplicates {

public static void main(String[] args) {

String[] array = {"apple", "banana", "orange", "apple", "kiwi"};

LinkedHashSet<String> set = new LinkedHashSet<>(Arrays.asList(array));

String[] uniqueArray = set.toArray(new String[0]);

System.out.println("Original Array: " + Arrays.toString(array));

System.out.println("Unique Array: " + Arrays.toString(uniqueArray));

}

}

17th answer:

public class SecondLargest {

public static void main(String[] args) {

int[] array = {10, 34, 56, 45, 67, 89, 12};

int largest = Integer.MIN\_VALUE;

int secondLargest = Integer.MIN\_VALUE;

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

if (array[i] > largest) {

secondLargest = largest;

largest = array[i];

} else if (array[i] > secondLargest && array[i] != largest) {

secondLargest = array[i];

}

}

System.out.println("The second largest element in the array is: " + secondLargest);

}

}

18th answer:

public class SecondSmallest {

public static void main(String[] args) {

int[] array = {10, 34, 56, 45, 67, 89, 12};

int smallest = Integer.MAX\_VALUE;

int secondSmallest = Integer.MAX\_VALUE;

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

if (array[i] < smallest) {

secondSmallest = smallest;

smallest = array[i];

} else if (array[i] < secondSmallest && array[i] != smallest) {

secondSmallest = array[i];

}

}

System.out.println("The second smallest element in the array is: " + secondSmallest);

}

}

19th answer:

public class MatrixAddition {

public static void main(String[] args) {

int[][] matrix1 = {{1, 2, 3}, {4, 5, 6}, {7, 8, 9}};

int[][] matrix2 = {{9, 8, 7}, {6, 5, 4}, {3, 2, 1}};

int[][] sum = new int[matrix1.length][matrix1[0].length];

for (int i = 0; i < matrix1.length; i++) {

for (int j = 0; j < matrix1[i].length; j++) {

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

}

}

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

for (int[] row : sum) {

System.out.println(Arrays.toString(row));

}

}

}

20th answer:

import java.util.ArrayList;

import java.util.Arrays;

public class ArrayToArrayList {

public static void main(String[] args) {

String[] array = {"apple", "banana", "orange"};

ArrayList<String> list = new ArrayList<>(Arrays.asList(array));

System.out.println("Array: " + Arrays.toString(array));

System.out.println("ArrayList: " + list);

}

}