1. Declare a single-dimensional array of 5 integers inside the main method. Traverse the array to print the default values. Then accept records from the user and print the updated values of the array.

**Program.java**

package in.cdac.ques1;

import java.util.Scanner;

public class Program {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

int[] arr = new int[5];

// for(int i=0;i<arr.length;i++) { //to print default values

// System.out.print(arr[i]+" ");

// }

// System.out.println();

// for(int element:arr) { //using for each loop

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

// }

System.out.println();

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

System.out.print("Enter arr["+i+"]: ");

arr[i]=sc.nextInt();

}

for(int i=0;i<arr.length;i++) { //to print updated values

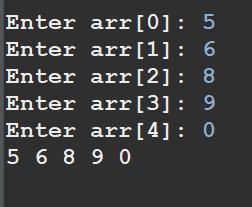
System.out.print(arr[i]+" ");

}

sc.close();

}

}



1. Declare a single-dimensional array of 5 integers inside the main method. Define a method named acceptRecord to get input from the terminal into the array and another method named printRecord to print the state of the array to the terminal.

**Program.java**

package in.cdac.ques2;

import java.util.Scanner;

public class Program {

private static Scanner sc = new Scanner(System.in);

public static void acceptRecord(int[] arr) {

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

System.out.print("Enter arr["+i+"]: ");

arr[i]=sc.nextInt();

}

}

public static void printRecord(int[] arr) {

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

System.out.println("arr["+i+"]: "+arr[i]);

}

}

public static void main(String[] args) {

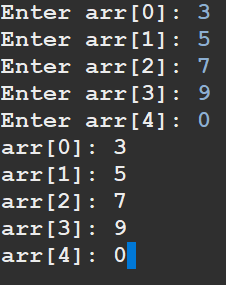
int[] arr = new int[5];

Program.acceptRecord(arr);

Program.printRecord(arr);

}

}



1. Write a program to find the maximum and minimum values in a single-dimensional array of integers.

**Program.java**

package in.cdac.ques3;

import java.util.Arrays;

import java.util.Scanner;

public class Program {

private static Scanner sc = new Scanner(System.in);

public static void acceptRecord(int[] arr) {

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

System.out.print("Enter arr["+i+"]: ");

arr[i]=sc.nextInt();

}

}

public static void printRecord(int[] arr) {

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

System.out.println("arr["+i+"]: "+arr[i]);

}

}

public static void maxValue(int[] arr) {

int[] arrCopy = Arrays.copyOf(arr, 5);

Arrays.sort(arrCopy);

System.out.println("Max value in the array: "+ arrCopy[arrCopy.length-1]);

}

public static void minValue(int[] arr) {

int[] arrCopy = Arrays.copyOf(arr, 5);

Arrays.sort(arrCopy);

System.out.println("Min value in the array: "+ arrCopy[0]);

}

public static void main(String[] args) {

int[] arr = new int[5];

Program.acceptRecord(arr);

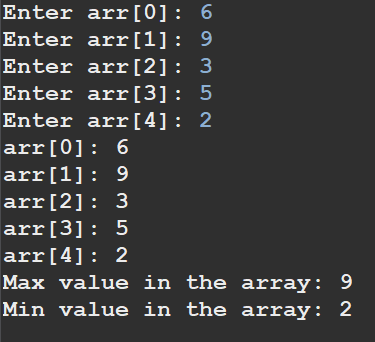
Program.printRecord(arr);

Program.maxValue(arr);

Program.minValue(arr);

}

}



1. Write a program to remove duplicate elements from a single-dimensional array of integers.

**Program.java**

package in.cdac.ques4;

//import java.util.Arrays;

public class Program {

public static void main(String[] args) {

int[] arr = new int[5];

Array.acceptRecord(arr);

Array.removeDuplicates(arr);

}

}

**Array.java**

package in.cdac.ques4;

import java.util.Arrays;

import java.util.Scanner;

public class Array {

private static Scanner sc = new Scanner(System.in);

public static void acceptRecord(int[] arr) {

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

System.out.print("Enter arr["+i+"]: ");

arr[i]=sc.nextInt();

}

}

public static void printRecord(int[] arr) {

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

System.out.println("arr["+i+"]: "+arr[i]);

}

}

public static void removeDuplicates(int[] arr){

int[] arrCopy = Arrays.copyOf(arr, arr.length);

Arrays.sort(arrCopy); //5 2 7 5 2 -> 2 2 5 5 7

int[] anotherCopy = new int[arr.length];

int j=0;

int i=0;

for(i=0;i<arrCopy.length-1;i++) {

if (arrCopy[i]!=arrCopy[i+1]) {

anotherCopy[j]=arrCopy[i]; //2 2 5 5 7 -> 2 5 (store the last element from 1 1 1 i.e 1 in anotherCopy)

j++;

}

}

anotherCopy[j]=arrCopy[i]; // to get the last element

int[] finalCopy = new int[j+1];

for(int k=0;k<finalCopy.length;k++) {

finalCopy[k]=anotherCopy[k]; // created another array to store the elements not the 0s at the last

}

// Arrays.sort(arrCopy);

// System.out.println(Arrays.toString(arrCopy));

System.out.print("Unique elements in the array are: ");

for(int element:finalCopy) {

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

}

}

}

1. Write a program to find the intersection of two single-dimensional arrays.

**Program.java**

package in.cdac.ques5;

public class Program {

public static void main(String[] args) {

int[] arr1 = new int[5];

int[] arr2 = new int[4];

Array.acceptRecord(arr1);

System.out.println();

Array.acceptRecord(arr2);

int[] arr1Unique = Array.removeDuplicates(arr1);

int[] arr2Unique = Array.removeDuplicates(arr2);

Array.findIntersection(arr1Unique, arr2Unique);

}

}

**Array.java**

package in.cdac.ques5;

import java.util.Arrays;

import java.util.Scanner;

public class Array {

private static Scanner sc = new Scanner(System.in);

public static void acceptRecord(int[] arr) {

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

System.out.print("Enter arr["+i+"]: ");

arr[i]=sc.nextInt();

}

}

public static void printRecord(int[] arr) {

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

System.out.println("arr["+i+"]: "+arr[i]);

}

}

public static int[] removeDuplicates(int[] arr){

int[] arrCopy = Arrays.copyOf(arr, arr.length);

Arrays.sort(arrCopy); //5 2 7 5 2 -> 2 2 5 5 7

int[] anotherCopy = new int[arr.length];

int j=0;

int i=0;

for(i=0;i<arrCopy.length-1;i++) {

if (arrCopy[i]!=arrCopy[i+1]) {

anotherCopy[j]=arrCopy[i]; //2 2 5 5 7 -> 2 5 (store the last element from 1 1 1 i.e 1 in anotherCopy)

j++;

}

}

anotherCopy[j]=arrCopy[i]; // to get the last element

int[] finalCopy = new int[j+1];

for(int k=0;k<finalCopy.length;k++) {

finalCopy[k]=anotherCopy[k]; // created another array to store the elements not the 0s at the last

}

// Arrays.sort(arrCopy);

// System.out.println(Arrays.toString(arrCopy));

// System.out.print("Unique elements in the array are: ");

// for(int element:finalCopy) {

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

// }

return finalCopy;

}

public static void findIntersection(int[] arr1, int[] arr2) {

int[] temp = new int[Integer.max(arr1.length,arr2.length)]; // 0 0 0

int k=0;

for(int i=0;i<arr1.length;i++) { // 1 2 3

for(int j=0;j<arr2.length;j++) { // 3 4

if(arr1[i]==arr2[j]) {

temp[k]=arr1[i];

k++;

}

}

}

int[] tempNew = new int[k];

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

tempNew[i]=temp[i]; // to get the starting elements that are not 0 (default is 0 for rest)

}

System.out.print("The common elements from both the arrays are: ");

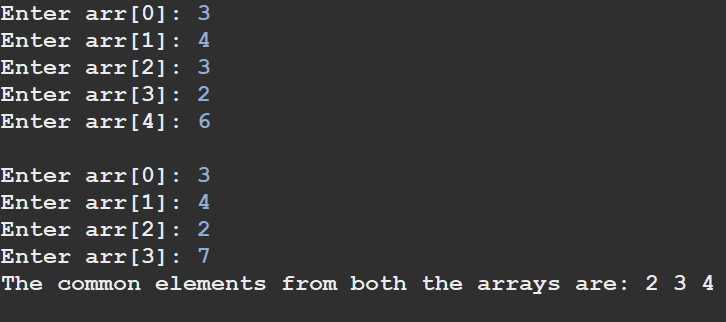
for(int element:tempNew) {

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

}

}

}



1. Write a program to find the missing number in an array of integers ranging from 1 to N.

**Program.java**

package in.cdac.ques6;

import java.util.Scanner;

public class Program {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

int missing=0;

// int[] arr = new int[] {1,2,3,4,6,7}; //missing number is 3

//0 1 2 3

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

int size = sc.nextInt();

int arr[] = new int[size];

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

System.out.print("arr["+i+"]: ");

arr[i]=sc.nextInt();

}

System.out.println();

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

if(arr[i] != i+1) { //1 != 1 2!=2 4!=3

missing = i+1;

break;

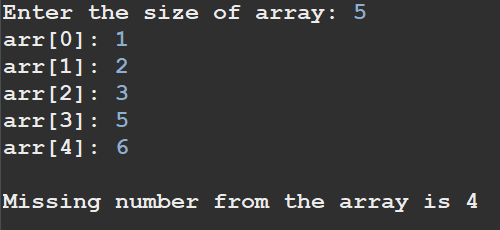
}

}

System.out.println("Missing number from the array is "+missing);

}

}



1. Declare a single-dimensional array as a field inside a class and instantiate it inside the class constructor. Define methods named acceptRecord and printRecord within the class and test their functionality.

**Program.java**

package in.cdac.ques7;

public class Program {

public static void main(String[] args) {

ArrayPrac a1 = new ArrayPrac();

ArrayPrac a2 = new ArrayPrac(7);

System.out.println("Enter elements for 1st array");

a1.acceptRecord();

System.out.println();

System.out.println("Enter elements for 2nd array");

a2.acceptRecord();

System.out.println();

System.out.println("Elements of 1st array: ");

a1.printRecord();

System.out.println();

System.out.println();

System.out.println("Elements of 2nd array: ");

a2.printRecord();

}

}

**ArrayPrac.java**

package in.cdac.ques7;

import java.util.Scanner;

public class ArrayPrac {

private int[] arr;

public ArrayPrac(){

this(4);

}

public ArrayPrac(int size){

arr = new int[size];

}

private Scanner sc = new Scanner(System.in);

public void acceptRecord() {

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

System.out.print("arr["+i+"]: ");

this.arr[i] = sc.nextInt();

}

}

public void printRecord() {

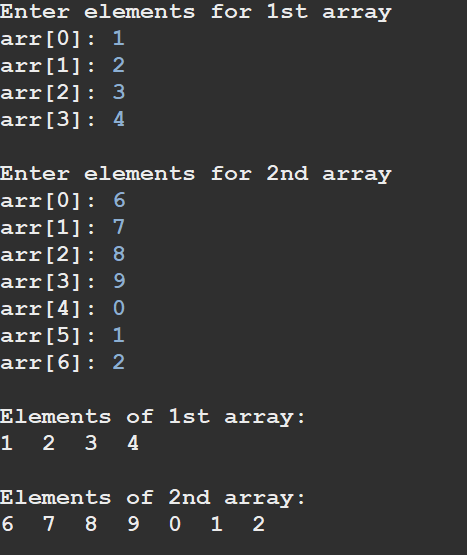
for(int i=0;i<this.arr.length;i++) {

System.out.print(this.arr[i]+" ");

}

}

}



1. Modify the previous assignment to use getter and setter methods instead of acceptRecord and printRecord.

**Program.java**

package in.cdac.ques8;

import java.util.Scanner;

public class Program {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

ArrayPrac a1 = new ArrayPrac(3);

ArrayPrac a2 = new ArrayPrac(5);

System.out.println("Enter elements for 1st array: ");

for(int i=0;i<a1.getArr().length;i++) {

System.out.print("arr["+i+"]: ");

a1.getArr()[i] = sc.nextInt();

}

System.out.println("Enter elements for 2nd array: ");

for(int i=0;i<a2.getArr().length;i++) {

System.out.print("arr["+i+"]: ");

a2.getArr()[i] = sc.nextInt();

}

System.out.println("Elements of 1st array: ");

for(int i=0;i<a1.getArr().length;i++) { //get the reference of array from ArrayPrac class from getter method

System.out.print(a1.getArr()[i]+" ");

}

System.out.println();

System.out.println("Elements of 2nd array: ");

for(int i=0;i<a2.getArr().length;i++) {

System.out.print(a2.getArr()[i]+" ");

}

sc.close();

}

}

**ArrayPrac.java**

package in.cdac.ques8;

public class ArrayPrac {

private int[] arr;

public ArrayPrac(){

arr = null;

}

public ArrayPrac(int size){

arr = new int[size];

}

public void setArr(int[] arr) {

this.arr = arr;

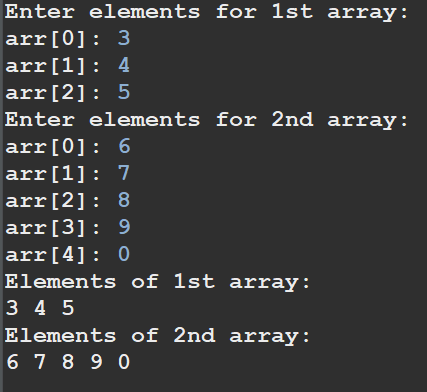
}

public int[] getArr() {

return this.arr;

}

}



1. You need to implement a system to manage airplane seat assignments. The airplane has seats arranged in rows and columns. Implement functionalities to:

* Initialize the seating arrangement with a given number of rows and columns.
* Book a seat to mark it as occupied.
* Cancel a booking to mark a seat as available.
* Check seat availability to determine if a specific seat is available.
* Display the current seating chart.

**Airplane.java**

package in.cdac.ques9;

public class Airplane {

private String[][] seatsArray;

public Airplane(int rows, int cols){

this.seatsArray = new String[rows][cols];

}

public String[][] getSeatsArray() {

return this.seatsArray;

}

}

**AirplaneUtility.java**

package in.cdac.ques9;

import java.util.Scanner;

public class AirplaneUtility {

private static Scanner sc = new Scanner(System.in);

private int r;

private int c;

private Airplane a1 = null;

public void acceptSize() {

System.out.println("Enter no. of rows of seats:");

r = sc.nextInt();

System.out.println("Enter no. of columns of seats:");

c = sc.nextInt();

a1 = new Airplane(r,c);

}

public void initializeArray() { //Set all seats to Available

for(int i=0;i<a1.getSeatsArray().length;i++)

for(int j=0;j<a1.getSeatsArray()[i].length;j++) {

a1.getSeatsArray()[i][j] = "Available";

}

}

public void bookASeat(int row, int col) {

if(a1.getSeatsArray()[row][col]=="Available") {

a1.getSeatsArray()[row][col]="Occupied";

}

else

System.out.println("This seat is already occupied.");

}

public void cancelASeat(int row, int col) {

a1.getSeatsArray()[row][col]="Available";

}

public void printArray() {

String[][] arr = a1.getSeatsArray();

System.out.println("Status of availability of seats: ");

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

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

System.out.print("Row["+i+"]Column["+j+"] "+arr[i][j]+" ");

}

System.out.println();

}

}

public static int menuList(){

System.out.println("0. Exit");

System.out.println("1. Initialize seating arrangement");

System.out.println("2. Show available seats");

System.out.println("3. Book a seat");

System.out.println("4. Cancel a seat");

return sc.nextInt();

}

}

**Program.java**

package in.cdac.ques9;

import java.util.Scanner;

public class Program {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

AirplaneUtility u1 = new AirplaneUtility();

int choice = 0;

int row, col;

while ((choice = AirplaneUtility.menuList()) != 0) {

switch(choice) {

case 1:

u1.acceptSize();

u1.initializeArray();

break;

case 2:

u1.printArray();

break;

case 3:

System.out.print("Enter seat row no. and column no. - \nRow: ");

row = sc.nextInt();

System.out.print("Column: ");

col = sc.nextInt();

u1.bookASeat(row, col);

break;

case 4:

System.out.print("Enter seat row no. and column no. to cancel the seat - \nRow: ");

row = sc.nextInt();

System.out.print("Column: ");

col = sc.nextInt();

u1.cancelASeat(row, col);

break;

}

}

sc.close();

}

}

