**CDAC MUMBAI**

**MODULE 2 OOPJ**

**ASSIGNMENT NO 5**

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:

**package** org.Assignment6;

**import** java.util.Scanner;

**public** **class** ArrayQ1 {

**public** **static** **void** main(String[] args) {

**int**[] arr = **new** **int**[5];

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

System.***out***.println("Default Value");

**for** (**int** index : arr) {

System.***out***.print(index + " ");

}

System.***out***.println();

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

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

arr[index] = sc.nextInt();

}

System.***out***.println("Accpeted value ");

**for** (**int** index : arr) {

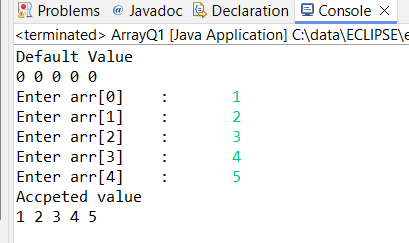
System.***out***.print(index + " ");

}

}

}

Output:



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:

**package** org.Assignment6;

**import** java.util.Scanner;

**class** ArrayDemo {

**private** **int**[] arr;

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

**public** ArrayDemo() {

**this**(5);

}

**public** ArrayDemo(**int** size) {

**this**.arr = **new** **int**[size];

}

**public** **void** acceptRecord() {

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

System.***out***.print("Enter Value of arr[" + index + "] :");

**this**.arr[index] = *sc*.nextInt();

}

}

**public** **void** printRecord() {

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

System.***out***.print(index + " ");

}

}

}

**public** **class** ArrayQ2 {

**public** **static** **void** main(String[] args) {

ArrayDemo array = **new** ArrayDemo();

array.acceptRecord();

array.printRecord();

}

}

Program 2:

**package** org.Assignment6;

**import** java.util.Scanner;

**public** **class** ArrayQ2 {

**public** **static** **void** main(String[] args) {

**int** [] arr=**new** **int**[5];

ArrayQ2.*acceptRecord*(arr);

ArrayQ2.*printRecord*(arr);

}

**public** **static** **void** acceptRecord(**int**[] arr) {

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

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

System.***out***.print("Enter Value of arr[" + index + "] :");

arr[index] = sc.nextInt();

}

}

**public** **static** **void** printRecord(**int** [] arr) {

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

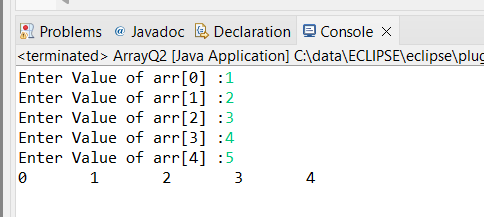
System.***out***.print(index + " ");

}

}

}

Output :



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

Program:

**package** org.Assignment6;

**import** java.util.Scanner;

**public** **class** ArrayQ4 {

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

**public** **static** **void** main(String[] args) {

**int** Size;

System.***out***.print("Enter Size of Array: ");

Size = *sc*.nextInt();

**int**[] arr = **new** **int**[Size];

ArrayQ4.*acceptRecord*(arr);

ArrayQ4.*MaxValue*(arr);

ArrayQ4.*MinValue*(arr);

}

**public** **static** **void** acceptRecord(**int**[] arr) {

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

System.***out***.print("Enter Value of arr[" + index + "] :");

arr[index] = *sc*.nextInt();

}

}

**public** **static** **void** MaxValue(**int**[] arr) {

**int** Max = arr[0];

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

**if** (arr[i] > Max) {

Max = arr[i];

}

}

System.***out***.println("Maximum Value in Array is :" + Max);

}

**public** **static** **void** MinValue(**int**[] arr) {

**int** Min = arr[0];

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

**if** (arr[i] < Min) {

Min = arr[i];

}

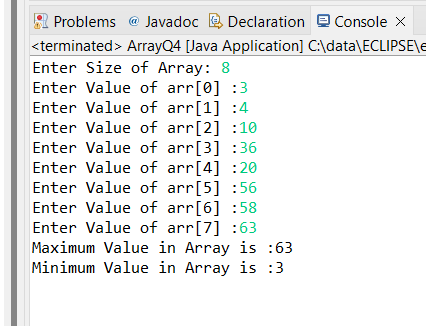
}

System.***out***.println("Minimum Value in Array is :" + Min);

}

}

Output:



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

**package** org.Assignment6;

**import** java.util.Scanner;

**import** java.util.Arrays;

**public** **class** ArrayQ4 {

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

Arrays.*sort*(arr);

**int** j = 0;

**for** (**int** i = 1; i < arr.length; i++) {

**if** (arr[i] != arr[j]) {

j++;

arr[j] = arr[i];

}

}

**return** Arrays.*copyOf*(arr, j + 1);

}

**public** **static** **void** main(String[] args) {

**int**[] arr = {1, 2, 2, 3, 4, 4, 5, 6, 6};

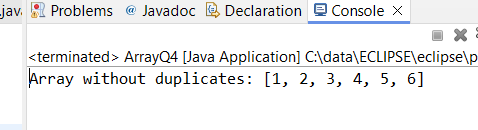
**int**[] result = *removeDuplicates*(arr);

System.***out***.println("Array without duplicates: " + Arrays.*toString*(result));

}

}

Output:



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

**package** org.Assignment6;

**import** java.util.\*;

**public** **class** ArrayQ5 {

**public** **static** List<Integer> findIntersection(**int**[] array1, **int**[] array2) {

Set<Integer> set1 = **new** HashSet<>();

List<Integer> result = **new** ArrayList<>();

**for** (**int** num : array1) {

set1.add(num);

}

**for** (**int** num : array2) {

**if** (set1.contains(num)) {

result.add(num);

}

}

**return** result;

}

**public** **static** **void** main(String[] args) {

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

**int**[] array2 = { 3, 4, 5, 6, 7 };

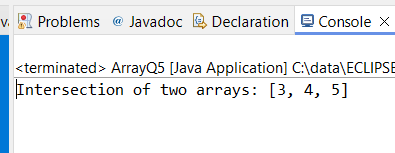
List<Integer> intersection = *findIntersection*(array1, array2);

System.***out***.println("Intersection of two arrays: " + intersection);

}

}

Output:



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

**package** org.Assignment6;

**public** **class** MissingNumber {

**public** **static** **int** findMissingNumber(**int**[] arr, **int** N) {

**int** expectedSum = N \* (N + 1) / 2;

**int** actualSum = 0;

**for** (**int** num : arr) {

actualSum += num;

}

**return** expectedSum - actualSum;

}

**public** **static** **void** main(String[] args) {

**int**[] arr = { 1, 2, 4, 5, 6 };

**int** N = 6;

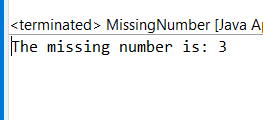
**int** missingNumber = *findMissingNumber*(arr, N);

System.***out***.println("The missing number is: " + missingNumber);

}

}

Output:



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.

**package** org.Assignment6;

**import** java.util.Scanner;

**public** **class** ArrayQ7 {

**private** **int**[] array;

**public** ArrayQ7(**int** size) {

array = **new** **int**[size];

}

**public** **void** acceptRecord() {

Scanner scanner = **new** Scanner(System.***in***);

System.***out***.println("Enter " + array.length + " elements:");

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

array[i] = scanner.nextInt();

}

}

**public** **void** printRecord() {

System.***out***.println("Array elements are:");

**for** (**int** element : array) {

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

}

System.***out***.println();

}

**public** **static** **void** main(String[] args) {

ArrayQ7 handler = **new** ArrayQ7(5);

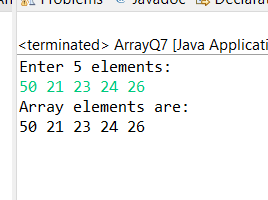
handler.acceptRecord();

handler.printRecord();

}

}

Output:



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

**package** org.Assignment6;

**import** java.util.Scanner;

**public** **class** ArrayQ8 {

**private** **int**[] array;

**public** ArrayQ8(**int** size) {

array = **new** **int**[size];

}

**public** **void** setArray(**int**[] array) {

**this**.array = array;

}

**public** **int**[] getArray() {

**return** array;

}

**public** **static** **void** main(String[] args) {

ArrayQ8 handler = **new** ArrayQ8(5);

Scanner scanner = **new** Scanner(System.***in***);

**int**[] inputArray = **new** **int**[5];

System.***out***.println("Enter 5 elements:");

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

inputArray[i] = scanner.nextInt();

}

handler.setArray(inputArray);

**int**[] array = handler.getArray();

System.***out***.println("Array elements are:");

**for** (**int** element : array) {

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

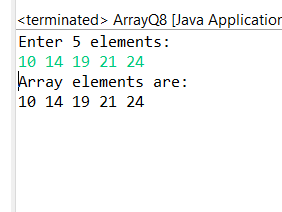
}

System.***out***.println();

}

}

Output:



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.

**package** org.Assignment6;

**import** java.util.Scanner;

**public** **class** AirplaneSeating {

**private** **boolean**[][] seats;

**public** AirplaneSeating(**int** rows, **int** columns) {

seats = **new** **boolean**[rows][columns];

}

**public** **void** bookSeat(**int** row, **int** column) {

**if** (seats[row][column]) {

System.***out***.println("Seat already booked.");

} **else** {

seats[row][column] = **true**;

System.***out***.println("Seat booked successfully.");

}

}

**public** **void** cancelBooking(**int** row, **int** column) {

**if** (!seats[row][column]) {

System.***out***.println("Seat is already available.");

} **else** {

seats[row][column] = **false**;

System.***out***.println("Booking cancelled successfully.");

}

}

**public** **boolean** isSeatAvailable(**int** row, **int** column) {

**return** !seats[row][column];

}

**public** **void** displaySeatingChart() {

System.***out***.println("Current seating chart:");

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

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

System.***out***.print(seats[i][j] ? "X " : "O ");

}

System.***out***.println();

}

}

**public** **static** **void** main(String[] args) {

Scanner scanner = **new** Scanner(System.***in***);

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

**int** rows = scanner.nextInt();

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

**int** columns = scanner.nextInt();

AirplaneSeating airplaneSeating = **new** AirplaneSeating(rows, columns);

**while** (**true**) {

System.***out***.println("\nMenu:");

System.***out***.println("1. Book a seat");

System.***out***.println("2. Cancel a booking");

System.***out***.println("3. Check seat availability");

System.***out***.println("4. Display seating chart");

System.***out***.println("5. Exit");

System.***out***.print("Enter your choice: ");

**int** choice = scanner.nextInt();

**switch** (choice) {

**case** 1:

System.***out***.print("Enter row number: ");

**int** bookRow = scanner.nextInt();

System.***out***.print("Enter column number: ");

**int** bookColumn = scanner.nextInt();

airplaneSeating.bookSeat(bookRow, bookColumn);

**break**;

**case** 2:

System.***out***.print("Enter row number: ");

**int** cancelRow = scanner.nextInt();

System.***out***.print("Enter column number: ");

**int** cancelColumn = scanner.nextInt();

airplaneSeating.cancelBooking(cancelRow, cancelColumn);

**break**;

**case** 3:

System.***out***.print("Enter row number: ");

**int** checkRow = scanner.nextInt();

System.***out***.print("Enter column number: ");

**int** checkColumn = scanner.nextInt();

**boolean** available = airplaneSeating.isSeatAvailable(checkRow, checkColumn);

System.***out***.println("Seat availability: " + (available ? "Available" : "Occupied"));

**break**;

**case** 4:

airplaneSeating.displaySeatingChart();

**break**;

**case** 5:

System.***out***.println("Exiting...");

**return**;

**default**:

System.***out***.println("Invalid choice. Please try again.");

}

}

}

}

Output:

