

- 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.

-->>

```
package org.example.demo1;
```

```
import java.util.Scanner;
```

```
public class array {
```

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

```
        int[] numbers = new int[5];
```

```
        System.out.println("Default values of the array:");
```

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

```
            System.out.println("Element " + i + ": " + numbers[i]);
```

```
        }
```

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

```
        System.out.println("\nEnter 5 integer values:");
```

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

```
            System.out.print("Enter value for element " + i + ": ");
```

```
            numbers[i] = scanner.nextInt();
```

```

    }

    System.out.println("\nUpdated values of the array:");
    for (int i = 0; i < numbers.length; i++) {
        System.out.println("Element " + i + ": " + numbers[i]);
    }

    scanner.close();
}
}

```

- 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.

-->>

```
package org.example.demo1;
```

```
import java.util.Scanner;
```

```
public class ArrayEx2 {
```

```
    public static void acceptRecord(int[] numbers) {
        Scanner scanner = new Scanner(System.in);
        System.out.println("Enter 5 integer values:");
```

```
        for (int i = 0; i < numbers.length; i++) {
            System.out.print("Enter value for element " + i
+ ": ");
            numbers[i] = scanner.nextInt();
```

```

        }
    }

    public static void printRecord(int[] numbers) {
        System.out.println("\nCurrent values of the array:");
        for (int i = 0; i < numbers.length; i++) {
            System.out.println("Element " + i + ": " +
numbers[i]);
        }
    }

    public static void main(String[] args) {

        int[] numbers = new int[5];
        acceptRecord(numbers);

        printRecord(numbers);
    }
}

```

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

-->>

```
package org.example.demo1;
```

```

public class MaxMinArray {
    public static void main(String[] args) {
        int[] numbers = { 20, 5, 7, 8, 35, 12 };

        int max = numbers[0];
        int min = numbers[0];
    }
}

```

```

        for (int number : numbers) {
            if (number > max) {
                max = number;
            }
            if (number < min) {
                min = number;
            }
        }

        System.out.println("Maximum value: " + max);
        System.out.println("Minimum value: " + min);
    }
}

```

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

-->>

```
package org.example.demo1;
```

```
import java.util.Arrays;
```

```

public class RemoveDuplicates {
    public static void main(String[] args) {
        int[] numbers = { 10, 20, 10, 30, 40, 30, 50 };
        Arrays.sort(numbers);
    }
}

```

```

int[] uniqueNumbers = new int[numbers.length];
int j = 0;
uniqueNumbers[j++] = numbers[0];

for (int i = 1; i < numbers.length; i++) {
    if (numbers[i] != numbers[i - 1]) {
        uniqueNumbers[j++] = numbers[i];
    }
}

System.out.println("Array after removing
duplicates:");
for (int i = 0; i < j; i++) {
    System.out.print(uniqueNumbers[i] + " ");
}
}

```

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

```

-->>
package org.example.demo1;

import java.util.HashSet;

public class ArrayIntersection {
    public static void main(String[] args) {
        int[] array1 = { 1, 2, 3, 4, 5 };
        int[] array2 = { 3, 4, 5, 6, 7 };

        HashSet<Integer> set = new HashSet<>();
        System.out.println("Intersection of arrays:");

        for (int value : array1) {
            set.add(value);
        }
    }
}

```

```

        for (int value : array2) {
            if (set.contains(value)) {
                System.out.print(value + " ");
            }
        }
    }
}

```

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

```

-->>
package org.example.demo1;

public class MissingNumber {
    public static void main(String[] args) {
        int[] numbers = { 1, 2, 3, 5, 6 };

        int N = numbers.length + 1;
        int sumOfN = N * (N + 1) / 2;
        int sumOfArray = 0;

        for (int number : numbers) {
            sumOfArray += number;
        }

        int missingNumber = sumOfN - sumOfArray;
        System.out.println("Missing number: " +
missingNumber);
    }
}

```

- 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.example.demo1;

import java.util.Scanner;

public class ArrayOperations {
    private int[] numbers;

    public ArrayOperations(int size) {
        numbers = new int[size];
    }

    public void acceptRecord() {
        Scanner scanner = new Scanner(System.in);
        System.out.println("Enter " + numbers.length + "
integers:");
        for (int i = 0; i < numbers.length; i++) {
            numbers[i] = scanner.nextInt();
        }
    }

    public void printRecord() {
        System.out.println("Array contents:");
        for (int number : numbers) {
            System.out.println(number);
        }
    }

    public static void main(String[] args) {
        ArrayOperations arrayOps = new
ArrayOperations(5);
        arrayOps.acceptRecord();
        arrayOps.printRecord();
    }
}

```

- Modify the previous assignment to use getter and setter methods

instead of acceptRecord and printRecord.

```
-->>
package org.example.demo1;

import java.util.Scanner;

public class ArrayOperationsWithGetterSetter {
    private int[] numbers;

    public ArrayOperationsWithGetterSetter(int size) {
        numbers = new int[size];
    }

    public void setNumbers() {
        Scanner scanner = new Scanner(System.in);
        System.out.println("Enter " + numbers.length + "
integers:");
        for (int i = 0; i < numbers.length; i++) {
            numbers[i] = scanner.nextInt();
        }
    }

    public int[] getNumbers() {
        return numbers;
    }

    public static void main(String[] args) {
        ArrayOperationsWithGetterSetter arrayOps = new
ArrayOperationsWithGetterSetter(5);
        arrayOps.setNumbers();

        System.out.println("Array contents:");
        for (int number : arrayOps.getNumbers()) {
            System.out.println(number);
        }
    }
}
```


- 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.example.demo1;
```

```
import java.util.Scanner;
```

```
public class AirplaneSeating {
    private boolean[][] seats;
```

```
    public AirplaneSeating(int rows, int cols) {
        seats = new boolean[rows][cols];
    }
```

```
    public void bookSeat(int row, int col) {
        if (!seats[row][col]) {
            seats[row][col] = true;
            System.out.println("Seat (" + row + ", " + col + ")
booked.");
        } else {
            System.out.println("Seat already booked.");
        }
    }
}
```

```

    public void cancelBooking(int row, int col) {
        if (seats[row][col]) {
            seats[row][col] = false;
            System.out.println("Booking for seat (" + row + ", " +
col + ") canceled.");
        } else {
            System.out.println("Seat was not booked.");
        }
    }

    public boolean checkAvailability(int row, int col) {
        return !seats[row][col];
    }

    public void displaySeatingChart() {
        System.out.println("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) {
        AirplaneSeating airplane = new AirplaneSeating(5, 4);
        Scanner scanner = new Scanner(System.in);

        airplane.bookSeat(2, 3);
        airplane.cancelBooking(2, 3);
        airplane.displaySeatingChart();

        System.out.println("Check if seat (1, 1) is available: " +
airplane.checkAvailability(1, 1));
    }
}

```