

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.

```
package com.arrayssolutions;
import java.util.Scanner;

public class ArraysSolutions1 {

    public static void main(String[] args) {
        int[] arr = new int[5]; // Declare an array of 5 integers

        System.out.println("Default values of array:");
        for (int i = 0; i < arr.length; i++) {
            System.out.println(arr[i]);
        }

        Scanner scanner = new Scanner(System.in);
        System.out.println("Enter 5 integers:");
        for (int i = 0; i < arr.length; i++) {
            arr[i] = scanner.nextInt();
        }

        System.out.println("Updated values of array:");
        for (int i = 0; i < arr.length; i++) {
            System.out.println(arr[i]);
        }

        scanner.close();
    }
}
```

2. 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 com.arrayssolutions;
```

```

import java.util.Scanner;

public class ArraysSolutions2 {

    public static void main(String[] args) {
        int[] arr = new int[5];
        acceptRecord(arr);
        printRecord(arr);
    }

    public static void acceptRecord(int[] arr) {
        Scanner scanner = new Scanner(System.in);
        System.out.println("Enter 5 integers:");
        for (int i = 0; i < arr.length; i++) {
            arr[i] = scanner.nextInt();
        }
    }

    public static void printRecord(int[] arr) {
        System.out.println("Array values:");
        for (int val : arr) {
            System.out.println(val);
        }
    }
}

```

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

```

package com.arrayssolutions;
import java.util.Scanner;
public class ArraysSolutions3 {

    public static void main(String[] args) {
        int[] arr = new int[5];
        Scanner scanner = new Scanner(System.in);
    }
}

```

```

System.out.println("Enter 5 integers:");
for (int i = 0; i < arr.length; i++) {
    arr[i] = scanner.nextInt();
}

int max = arr[0], min = 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("Maximum: " + max);
System.out.println("Minimum: " + min);

scanner.close();
}
}

```

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

```

package com.arrayssolutions;
import java.util.Scanner;
import java.util.Arrays;

public class ArraysSolutions4 {

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        int[] arr = new int[5];
        System.out.println("Enter 5 integers:");
        for (int i = 0; i < arr.length; i++) {
            arr[i] = scanner.nextInt();
        }

        Arrays.sort(arr);
        System.out.println("Array after removing duplicates:");
    }
}

```

```

for (int i = 0; i < arr.length - 1; i++) {
    if (arr[i] != arr[i + 1]) {
        System.out.print(arr[i] + " ");
    }
}
System.out.println(arr[arr.length - 1]);

scanner.close();
}
}

```

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

```

package com.arrayssolutions;
import java.util.Scanner;

public class ArraysSolutions5 {

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        int[] arr1 = new int[5];
        int[] arr2 = new int[5];

        System.out.println("Enter 5 integers for first array:");
        for (int i = 0; i < arr1.length; i++) {
            arr1[i] = scanner.nextInt();
        }

        System.out.println("Enter 5 integers for second array:");
        for (int i = 0; i < arr2.length; i++) {
            arr2[i] = scanner.nextInt();
        }

        System.out.println("Intersection of arrays:");
        for (int i = 0; i < arr1.length; i++) {
            for (int j = 0; j < arr2.length; j++) {
                if (arr1[i] == arr2[j]) {

```

```

System.out.println(arr1[i]);
}
}
}
scanner.close();
}
}

```

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

```

package com.arrayssolutions;
import java.util.Scanner;
public class ArraysSolutions6 {

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        int n = 5;
        int[] arr = new int[n - 1];

        System.out.println("Enter " + (n - 1) + " integers between 1 and " + n
            + ":");
        for (int i = 0; i < arr.length; i++) {
            arr[i] = scanner.nextInt();
        }

        int sum = n * (n + 1) / 2;
        int sumArr = 0;
        for (int val : arr) {
            sumArr += val;
        }

        System.out.println("Missing number is: " + (sum - sumArr));
        scanner.close();
    }
}

```

7. 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 com.arrayssolutions;
import java.util.Scanner;

public class ArraysSolutions7 {
    private int[] arr;

    public ArraysSolutions7() {
        arr = new int[5];
    }
    public void acceptRecord() {
        Scanner scanner = new Scanner(System.in);
        System.out.println("Enter 5 integers:");
        for (int i = 0; i < arr.length; i++) {
            arr[i] = scanner.nextInt();
        }
    }
    public void printRecord() {
        System.out.println("Array values:");
        for (int val : arr) {
            System.out.println(val);
        }
    }

    public static void main(String[] args) {
        ArraysSolutions7 obj = new ArraysSolutions7();
        obj.acceptRecord();
        obj.printRecord();
    }
}
```

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

```
package com.arrayssolutions;
import java.util.Scanner;

public class ArraysSolutions8 {
    private int[] arr;

    public ArraysSolutions8() {
        arr = new int[5];
    }

    public void setArray(int[] inputArr) {
        this.arr = inputArr;
    }

    public int[] getArray() {
        return this.arr;
    }

    public static void main(String[] args) {

        ArraysSolutions8 obj = new ArraysSolutions8();
        Scanner scanner = new Scanner(System.in);
        int[] inputArr = new int[5];
        System.out.println("Enter 5 integers:");
        for (int i = 0; i < inputArr.length; i++) {
            inputArr[i] = scanner.nextInt();
        }
        obj.setArray(inputArr);

        System.out.println("Array values:");
        for (int val : obj.getArray()) {
            System.out.println(val);
        }
    }
}
```

9. 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 com.arrayssolutions;

public class ArraysSolutions9 {
    private boolean[][] seats;

    public ArraysSolutions9(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 booked at row " + row + ", col " + col);
        } else {
            System.out.println("Seat is already occupied.");
        }
    }

    public void cancelBooking(int row, int col) {
        if (seats[row][col]) {
            seats[row][col] = false;
        }
    }
}
```



```
    System.out.println("Booking cancelled for seat at row " + row + ",  
col " + col);  
    } else {  
        System.out.println("Seat is already available.");  
    }  
}
```

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

```
public void displaySeatingChart() {  
    for (int i = 0; i < seats.length; i++) {  
        for (int j = 0; j < seats[i].length; j++) {  
            if (seats[i][j]) {  
                System.out.print("[X] ");  
            } else {  
                System.out.print("[ ] ");  
            }  
        }  
        System.out.println();  
    }  
}
```

```
public static void main(String[] args) {  
    ArraysSolutions9 system = new ArraysSolutions9(5, 5);
```

```
    system.bookSeat(1, 2);  
    system.cancelBooking(1, 2);  
    system.displaySeatingChart();
```

```
}  
}
```