

Assignment 6

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.

Code :-

```
package project;
import java.util.Scanner;
public class Program1 {

    public static void main(String[] args) {

        // Declare a single-dimensional array of 5 integers
        int[] numbers = new int[5];

        // Print the default values of the array
        System.out.println("Default values in the array:");
        for (int i = 0; i < numbers.length; i++) {
            System.out.println("Index " + i + ": " + numbers[i]);
        }

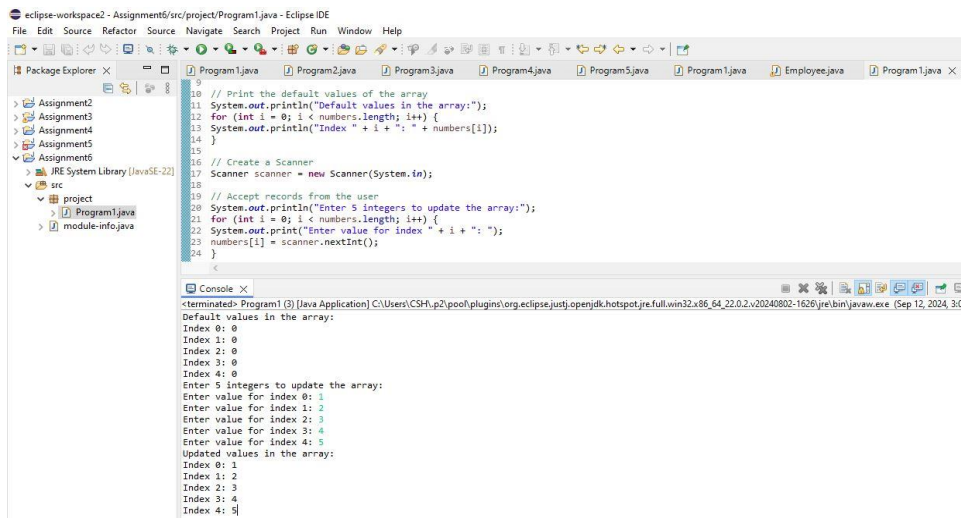
        // Create a Scanner
        Scanner scanner = new Scanner(System.in);

        // Accept records from the user
        System.out.println("Enter 5 integers to update the array:");
        for (int i = 0; i < numbers.length; i++) {
            System.out.print("Enter value for index " + i + ": ");
            numbers[i] = scanner.nextInt();
        }

        // Print the updated values of the array
        System.out.println("Updated values in the array:");
        for (int i = 0; i < numbers.length; i++) {
            System.out.println("Index " + i + ": " + numbers[i]);
        }

        scanner.close();
    }
}
```

Output –

The screenshot shows the Eclipse IDE interface. The Package Explorer on the left shows a project named 'project' with a source folder 'src' containing 'Program1.java' and 'module-info.java'. The main editor displays the code for 'Program1.java'. The code includes a main method that prints default values of an array and a Scanner-based method to accept 5 integers from the user. The Console window at the bottom shows the execution output, including the default array values (all 0s) and the updated array values (1, 2, 3, 4, 5) after user input.

```
10 // Print the default values of the array
11 System.out.println("Default values in the array:");
12 for (int i = 0; i < numbers.length; i++) {
13     System.out.println("Index " + i + ": " + numbers[i]);
14 }
15
16 // Create a Scanner
17 Scanner scanner = new Scanner(System.in);
18
19 // Accept records from the user
20 System.out.println("Enter 5 integers to update the array:");
21 for (int i = 0; i < numbers.length; i++) {
22     System.out.print("Enter value for index " + i + ": ");
23     numbers[i] = scanner.nextInt();
24 }
```

terminated: Program1 (3) [Java Application] C:\Users\CSH\p2\poo\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64_22.0.2\20240802-1626\jre\bin\javaw.exe (Sep 12, 2024 3:07)

Default values in the array:
Index 0: 0
Index 1: 0
Index 2: 0
Index 3: 0
Index 4: 0
Enter 5 integers to update the array:
Enter value for index 0: 1
Enter value for index 1: 2
Enter value for index 2: 3
Enter value for index 3: 4
Enter value for index 4: 5
Updated values in the array:
Index 0: 1
Index 1: 2
Index 2: 3
Index 3: 4
Index 4: 5

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.

Code :-

```
package project;
import java.util.Scanner;
public class Program2 {

    public static void main(String[] args) {

        // Declare a single-dimensional array of 5 integers
        int[] numbers = new int[5];

        // Call method to accept input
        acceptRecord(numbers);

        // Call method to print the state of array
        printRecord(numbers);
    }

    // Method to accept input from the user and update the array
    public static void acceptRecord(int[] array) {
        Scanner scanner = new Scanner(System.in);

        System.out.println("Enter 5 integers to update the array:");
        for (int i = 0; i < array.length; i++) {
            System.out.print("Enter value for index " + i + ": ");
            array[i] = scanner.nextInt();
        }
    }

    // Method to print the state of the array
```

```

public static void printRecord(int[] array) {
System.out.println("Current values in the array:");
for (int i = 0; i < array.length; i++) {
System.out.println("Index " + i + ": " + array[i]);
}
}
}
}

```

Output –

```

1 package project;
2 import java.util.Scanner;
3 public class Program2 {
4
5     public static void main(String[] args) {
6
7         // Declare a single-dimensional array of 5 integers
8         int[] numbers = new int[5];
9
10        // Call method to accept input
11        acceptRecord(numbers);
12
13        // Call method to print the state of array
14        printRecord(numbers);
15    }
16
17    // Method to accept input from the user and update the array
18    public static void acceptRecord(int[] array) {
19        Scanner scanner = new Scanner(System.in);
20
21        System.out.println("Enter 5 integers to update the array:");
22
23        for (int i = 0; i < array.length; i++) {
24            System.out.println("Enter value for index " + i);
25            array[i] = scanner.nextInt();
26        }
27    }
28
29    // Method to print the state of array
30    public static void printRecord(int[] array) {
31        System.out.println("Current values in the array:");
32        for (int i = 0; i < array.length; i++) {
33            System.out.println("Index " + i + ": " + array[i]);
34        }
35    }
36 }

```

```

<terminated> Program2 (3) [Java Application] C:\Users\CSH\p2\pool\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64_22.0.2.v20240802-1620\jre\bin\javaw.exe (Sep 12, 2024, 3:12:07 PM)
Enter 5 integers to update the array:
Enter value for index 0: 2
Enter value for index 1: 4
Enter value for index 2: 6
Enter value for index 3: 8
Enter value for index 4: 10
Current values in the array:
Index 0: 2
Index 1: 4
Index 2: 6
Index 3: 8
Index 4: 10

```

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

Code :-

```

package project;

public class Program3 {

    public static void main(String[] args) {

        int[] numbers = {40, 23, 12, 92, 45, 67, 23};

        int max = findMax(numbers);
        int min = findMin(numbers);

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

    public static int findMax(int[] array) {
        int max = array[0];
        for (int i = 1; i < array.length; i++) {
            if (array[i] > max) {

```

```

max = array[i];
}
}
return max;
}
public static int findMin(int[] array) {
int min = array[0]; // Initialize min to the first element
for (int i = 1; i < array.length; i++) {
if (array[i] < min) {
min = array[i];
}
}
return min;
}
}

```

Output –

The screenshot shows the Eclipse IDE interface. The Package Explorer on the left shows a project named 'project' with several Java files. The main editor displays the source code of 'Program3.java'. The code defines a package 'project', a class 'Program3', and a main method. Inside the main method, an array of integers is declared: `int[] numbers = {40, 23, 12, 92, 45, 67, 23};`. Two static methods are called: `findMax` and `findMin`. The `findMax` method iterates through the array to find the maximum value (92), and the `findMin` method iterates through the array to find the minimum value (12). The results are printed to the console using `System.out.println`. The console output at the bottom shows: `Maximum value: 92` and `Minimum value: 12`.

```

1 package project;
2
3 public class Program3 {
4
5     public static void main(String[] args) {
6
7         int[] numbers = {40, 23, 12, 92, 45, 67, 23};
8
9         int max = findMax(numbers);
10        int min = findMin(numbers);
11
12        System.out.println("Maximum value: " + max);
13        System.out.println("Minimum value: " + min);
14    }
15
16
17    public static int findMax(int[] array) {
18        int max = array[0];
19        for (int i = 1; i < array.length; i++) {
20            if (array[i] > max) {
21                max = array[i];
22            }
23        }
24        return max;
25    }
26
27    public static int findMin(int[] array) {
28        int min = array[0]; // Initialize min to the first element
29        for (int i = 1; i < array.length; i++) {
30            if (array[i] < min) {
31                min = array[i];
32            }
33        }
34        return min;
35    }
36
37 }

```

```

<terminated> Program3 (2) [Java Application] C:\Users\CSH\p2\pool\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64_22.0.2.v20240802-1626\jre\bin\javaw.exe (Sep 12, 202
Maximum value: 92
Minimum value: 12

```

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

Code :-

```

package project;

import java.util.Arrays;

import java.util.HashSet;

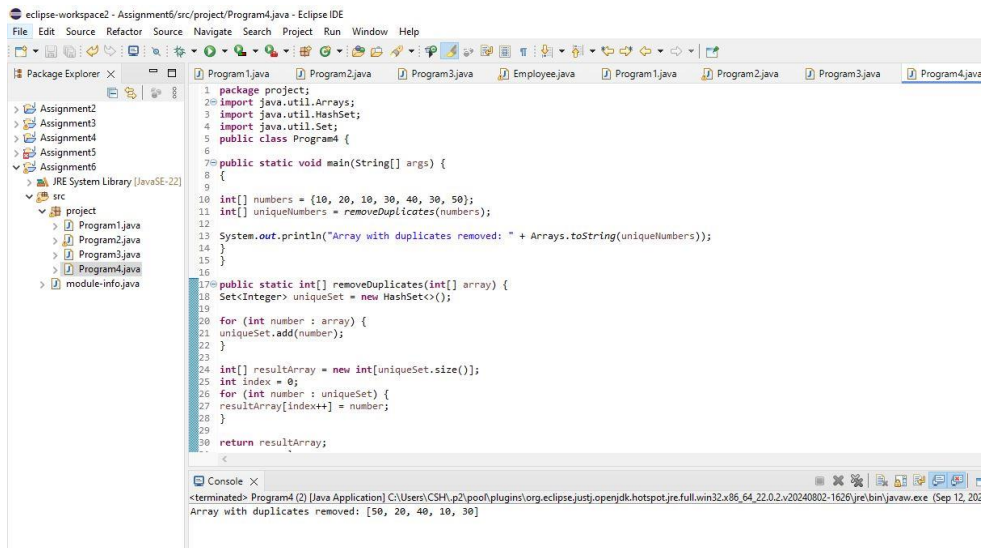
import java.util.Set;

public class Program4 {

```

```
public static void main(String[] args) {  
  
    {  
  
        int[] numbers = {10, 20, 10, 30, 40, 30, 50};  
        int[] uniqueNumbers = removeDuplicates(numbers);  
  
        System.out.println("Array with duplicates removed: " + Arrays.toString(uniqueNumbers));  
    }  
}  
  
public static int[] removeDuplicates(int[] array) {  
    Set<Integer> uniqueSet = new HashSet<>();  
  
    for (int number : array) {  
        uniqueSet.add(number);  
    }  
  
    int[] resultArray = new int[uniqueSet.size()];  
    int index = 0;  
    for (int number : uniqueSet) {  
        resultArray[index++] = number;  
    }  
  
    return resultArray;  
    }  
}
```

Output –



The screenshot shows the Eclipse IDE with a project named 'project' containing several Java files. The 'src' folder is expanded, showing 'Program1.java', 'Program2.java', 'Program3.java', and 'Program4.java'. The 'Program4.java' file is open in the editor, showing the following code:

```
1 package project;
2 import java.util.Arrays;
3 import java.util.HashSet;
4 import java.util.Set;
5 public class Program4 {
6
7     public static void main(String[] args) {
8
9
10        int[] numbers = {10, 20, 10, 30, 40, 30, 50};
11        int[] uniqueNumbers = removeDuplicates(numbers);
12
13        System.out.println("Array with duplicates removed: " + Arrays.toString(uniqueNumbers));
14    }
15 }
16
17 public static int[] removeDuplicates(int[] array) {
18     Set<Integer> uniqueSet = new HashSet<>();
19
20     for (int number : array) {
21         uniqueSet.add(number);
22     }
23
24     int[] resultArray = new int[uniqueSet.size()];
25     int index = 0;
26     for (int number : uniqueSet) {
27         resultArray[index++] = number;
28     }
29
30     return resultArray;
31 }
```

The console output at the bottom shows the execution of the program:

```
<terminated> Program4 (2) [Java Application] C:\Users\CSH\p2\poo\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64_22.0.2.v20240802-1626\jre\bin\javaw.exe (Sep 12, 2024)
Array with duplicates removed: [50, 20, 40, 10, 30]
```

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

Code :-

```
package project;
```

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

```
import java.util.HashSet;
```

```
import java.util.Set;
```

```
public class Program5 {
```

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

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

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

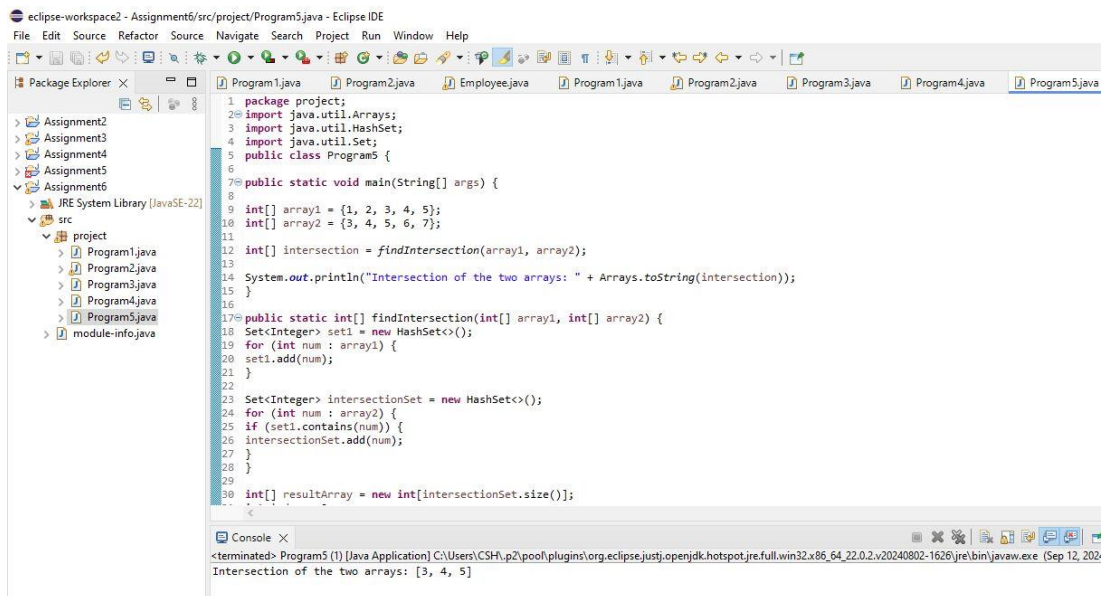
```
        int[] intersection = findIntersection(array1, array2);
```

```
        System.out.println("Intersection of the two arrays: " + Arrays.toString(intersection));
```

```
    }
```

```
public static int[] findIntersection(int[] array1, int[] array2) {  
    Set<Integer> set1 = new HashSet<>();  
    for (int num : array1) {  
        set1.add(num);  
    }  
  
    Set<Integer> intersectionSet = new HashSet<>();  
    for (int num : array2) {  
        if (set1.contains(num)) {  
            intersectionSet.add(num);  
        }  
    }  
  
    int[] resultArray = new int[intersectionSet.size()];  
    int index = 0;  
    for (int num : intersectionSet) {  
        resultArray[index++] = num;  
    }  
  
    return resultArray;  
}
```

Output –



The screenshot shows the Eclipse IDE with the Package Explorer on the left, displaying a project named 'project' under 'Assignment6'. The main editor shows the source code of 'Program5.java'. The code defines a package 'project', imports 'java.util.Arrays' and 'java.util.HashSet', and defines a public class 'Program5' with a main method and a static method 'findIntersection'. The main method creates two arrays, 'array1' and 'array2', and prints the intersection of the two arrays. The 'findIntersection' method uses a HashSet to find the common elements between the two arrays. The console output at the bottom shows the result of the program execution: 'Intersection of the two arrays: [3, 4, 5]'.

```
1 package project;
2 import java.util.Arrays;
3 import java.util.HashSet;
4 import java.util.Set;
5 public class Program5 {
6
7     public static void main(String[] args) {
8
9         int[] array1 = {1, 2, 3, 4, 5};
10        int[] array2 = {3, 4, 5, 6, 7};
11
12        int[] intersection = findIntersection(array1, array2);
13
14        System.out.println("Intersection of the two arrays: " + Arrays.toString(intersection));
15    }
16
17    public static int[] findIntersection(int[] array1, int[] array2) {
18        Set<Integer> set1 = new HashSet<>();
19        for (int num : array1) {
20            set1.add(num);
21        }
22
23        Set<Integer> intersectionSet = new HashSet<>();
24        for (int num : array2) {
25            if (set1.contains(num)) {
26                intersectionSet.add(num);
27            }
28        }
29
30        int[] resultArray = new int[intersectionSet.size()];
31    }
32}
```

Console Output:

```
<terminated> Program5 (1) [Java Application] C:\Users\CSH\AppData\Local\Temp\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64_22.0.2.v20240802-1626\jre\bin\javaw.exe (Sep 12, 2024)
Intersection of the two arrays: [3, 4, 5]
```

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

Code :-

```
package project;
public class Program6 {
    public static void main(String[] args) {

        int[] numbers = {1, 2, 4, 5, 6};
        int N = 6;

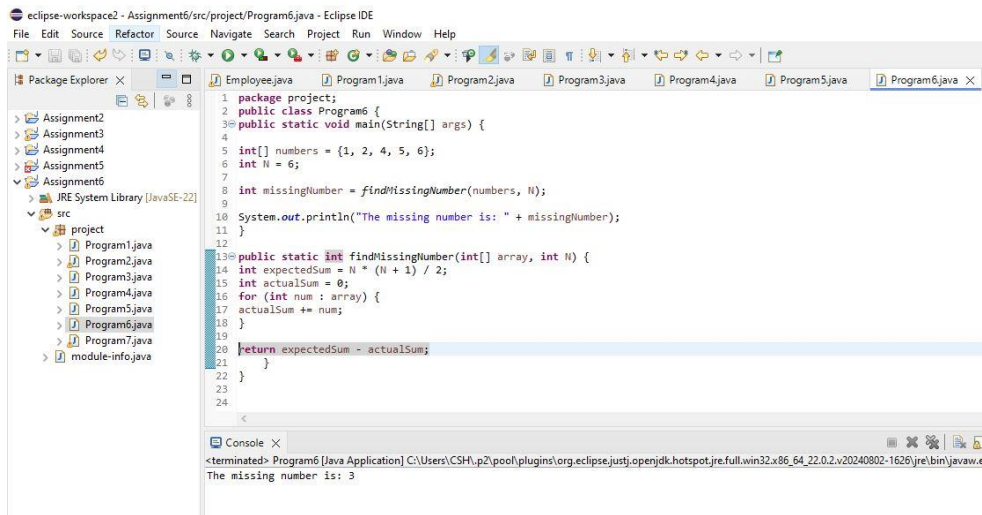
        int missingNumber = findMissingNumber(numbers, N);

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

    public static int findMissingNumber(int[] array, int N) {
        int expectedSum = N * (N + 1) / 2;
        int actualSum = 0;
        for (int num : array) {
            actualSum += num;
        }

        return expectedSum - actualSum;
    }
}
```


Output –



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.

Code :-

```
package project;

import java.util.Scanner;

public class Program7 {
    private int[] numbers;

    public Program7(int size) {
        numbers = new int[5];
    }

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

    public void printRecord() {
        System.out.println("Current values in the array:");
        for (int i = 0; i < numbers.length; i++) {
            System.out.println("Index " + i + ": " + numbers[i]);
        }
    }
}
```

```

public static void main(String[] args) {
    Program7 manager = new Program7(5);
    manager.acceptRecord();
    manager.printRecord();
}
}

```

Output –

The screenshot shows the Eclipse IDE with the Package Explorer on the left displaying a project named 'project' containing several Java files. The main editor shows the source code of Program7.java. The console at the bottom displays the output of the program, showing the user entering 5 integers to update the array and the resulting values in the array.

```

package project;
import java.util.Scanner;

public class Program7 {
    private int[] numbers;

    public Program7(int size) {
        numbers = new int[size];
    }

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

    public void printRecord() {
        System.out.println("Current values in the array:");
    }
}

```

```

<terminated> Program7 [Java Application] C:\Users\CSHL\p2\pool\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64_22.0.2.v20240802-1626\jre\bin\javaw.exe (Sep 12, 2024, 4:38:07 PM)
Enter 5 integers to update the array:
Enter value for index 0: 3
Enter value for index 1: 6
Enter value for index 2: 9
Enter value for index 3: 12
Enter value for index 4: 15
Current values in the array:
Index 0: 3
Index 1: 6
Index 2: 9
Index 3: 12
Index 4: 15

```

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

Code:-

```

package project;
import java.util.Scanner;

public class Program8 {

    private int[] numbers;

    public Program8(int size) {
        numbers = new int[size];
    }

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

    public void setNumbers() {
        Scanner scanner = new Scanner(System.in);
        System.out.println("Enter " + numbers.length + " integers to update the array:");
        for (int i = 0; i < numbers.length; i++) {

```

```

System.out.print("Enter value for index " + i + ": ");
numbers[i] = scanner.nextInt();
}
}

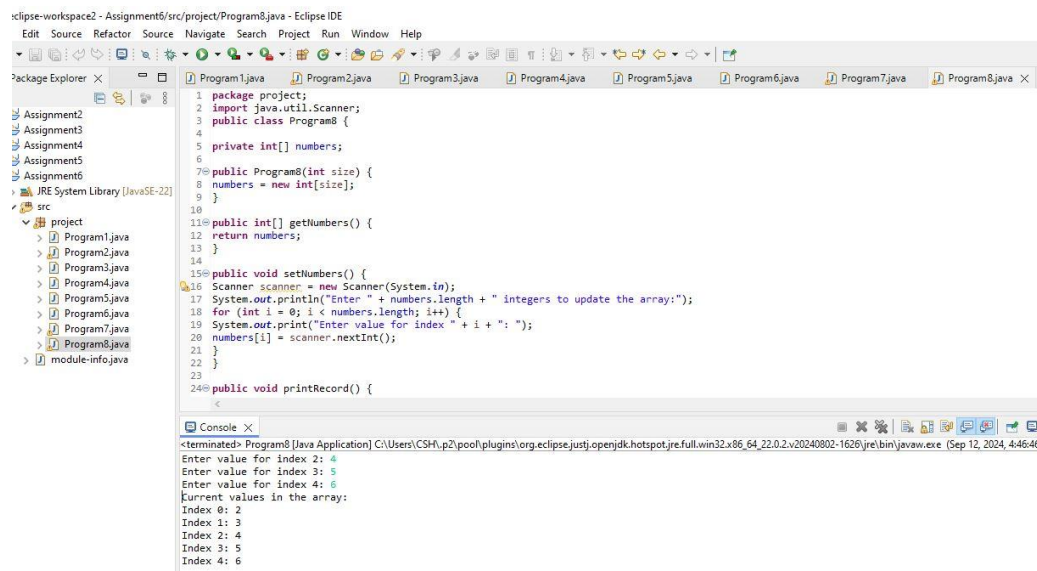
public void printRecord() {
System.out.println("Current values in the array:");
for (int i = 0; i < numbers.length; i++) {
System.out.println("Index " + i + ": " + numbers[i]);
}
}

public static void main(String[] args) {
Program8 manager = new Program8(5);
manager.setNumbers();

manager.printRecord();
}
}

```

Output –



The screenshot shows the Eclipse IDE interface. The Package Explorer on the left lists several projects, with 'Program8.java' selected. The main editor displays the source code of 'Program8.java', which includes a package declaration, imports, a class definition with a private array, and methods for setting and printing array values. The Console window at the bottom shows the program's execution output, including prompts for input and the resulting array state.

```

<terminated> Program8 [Java Application] C:\Users\CSH\p2\plugin\org.eclipse.jdt.openjdk.hotspot.jre.full.win32.x86_64_22.0.2.v20240802-1626\jre\bin\javaw.exe (Sep 12, 2024, 4:46:46)
Enter value for index 2: 4
Enter value for index 3: 5
Enter value for index 4: 6
Current values in the array:
Index 0: 2
Index 1: 3
Index 2: 4
Index 3: 5
Index 4: 6

```

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.

Code :-

```
package project;
import java.util.Scanner;
public class Program9 {

    private String[][] seats;

    public Program9(int rows, int columns) {
        seats = new String[rows][columns];
        initializeSeats();
    }

    private void initializeSeats() {
        for (int i = 0; i < seats.length; i++) {
            for (int j = 0; j < seats[i].length; j++) {
                seats[i][j] = "Available";
            }
        }
    }

    public void bookSeat(int row, int column) {
        if (isSeatValid(row, column)) {
            if (seats[row][column].equals("Available")) {
                seats[row][column] = "Booked";
                System.out.println("Seat (" + row + ", " + column + ") has been booked.");
            }
            else {
                System.out.println("Seat (" + row + ", " + column + ") is already booked.");
            }
        }
    }

    public void cancelBooking(int row, int column) {
        if (isSeatValid(row, column)) {
            if (seats[row][column].equals("Booked")) {
                seats[row][column] = "Available";
                System.out.println("Booking for seat (" + row + ", " + column + ") has been canceled.");
            }
            else {
                System.out.println("Seat (" + row + ", " + column + ") is already available.");
            }
        }
    }

    public boolean isSeatAvailable(int row, int column) {
        if (isSeatValid(row, column)) {
            return seats[row][column].equals("Available");
        }
        return false;
    }

    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] + "\t");
}
System.out.println();
}
}

private boolean isSeatValid(int row, int column) {
    if (row >= 0 && row < seats.length && column >= 0 && column < seats[0].length) {
        return true;
    }
    else {
        System.out.println("Invalid seat position: (" + row + ", " + column + ")");
        return false;
    }
}

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();

    Program9 program = new Program9(rows, columns);

    // Test the functionalities
    program.displaySeatingChart();

    // Book a seat
    System.out.print("Enter row and column to book a seat: ");
    int rowToBook = scanner.nextInt();
    int colToBook = scanner.nextInt();
    program.bookSeat(rowToBook, colToBook);
    program.displaySeatingChart();

    // Cancel a booking
    System.out.print("Enter row and column to cancel a seat booking: ");

    int rowToCancel = scanner.nextInt();
    int colToCancel = scanner.nextInt();
    program.cancelBooking(rowToCancel, colToCancel);
    program.displaySeatingChart();

    // Check seat availability
    System.out.print("Enter row and column to check seat
availability: ");

    int rowToCheck = scanner.nextInt();
    int colToCheck = scanner.nextInt();
    boolean available = program.isSeatAvailable(rowToCheck,
colToCheck);

    System.out.println("Seat (" + rowToCheck + ", " + colToCheck +
") is " + (available ? "Available" : "Booked"));
}

```

```

scanner.close();
}
}

```

Output –

The screenshot shows the Eclipse IDE interface. The Package Explorer on the left lists several assignments. The main editor displays the code for `Program9.java`. The code defines a `Program9` class with a 2D array of seats, an `initializeSeats` method to set all seats to "Available", and a `bookSeat` method to book a seat if it's available. The console at the bottom shows the program's execution: it prompts for the number of rows (5) and columns (5), displays a 5x5 grid of "Available" seats, and then prompts for a row and column to book (2 and 2 respectively).

```

1 package project;
2 import java.util.Scanner;
3 public class Program9 {
4
5     private String[][] seats;
6
7     public Program9(int rows, int columns) {
8         seats = new String[rows][columns];
9         initializeSeats();
10    }
11
12    private void initializeSeats() {
13        for (int i = 0; i < seats.length; i++) {
14            for (int j = 0; j < seats[i].length; j++) {
15                seats[i][j] = "Available";
16            }
17        }
18    }
19
20    public void bookSeat(int row, int column) {
21        if (isSeatValid(row, column)) {
22            if (seats[row][column].equals("Available")) {
23                seats[row][column] = "Booked";
24                System.out.println("Seat (" + row + ", " + column + ") has been booked.");
25            }
26        }
27    }
28
29    private boolean isSeatValid(int row, int column) {
30        return row < seats.length & column < seats[0].length;
31    }
32
33    public static void main(String[] args) {
34        Program9 program9 = new Program9(5, 5);
35        program9.bookSeat(2, 2);
36    }
37 }

```

```

Program9 [Java Application] C:\Users\CSH\p2\poo\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64_22.0.2.v20240802-1626\jre\bin\javaw.exe (Sep 13, 2024, 4:44:24 PM) [pid: 127]
Enter number of rows: 5
Enter number of columns: 5
Current seating chart:
Available Available Available Available Available
Available Available Available Available Available
Available Available Available Available Available
Available Available Available Available Available
Available Available Available Available Available
Enter row and column to book a seat: 2

```