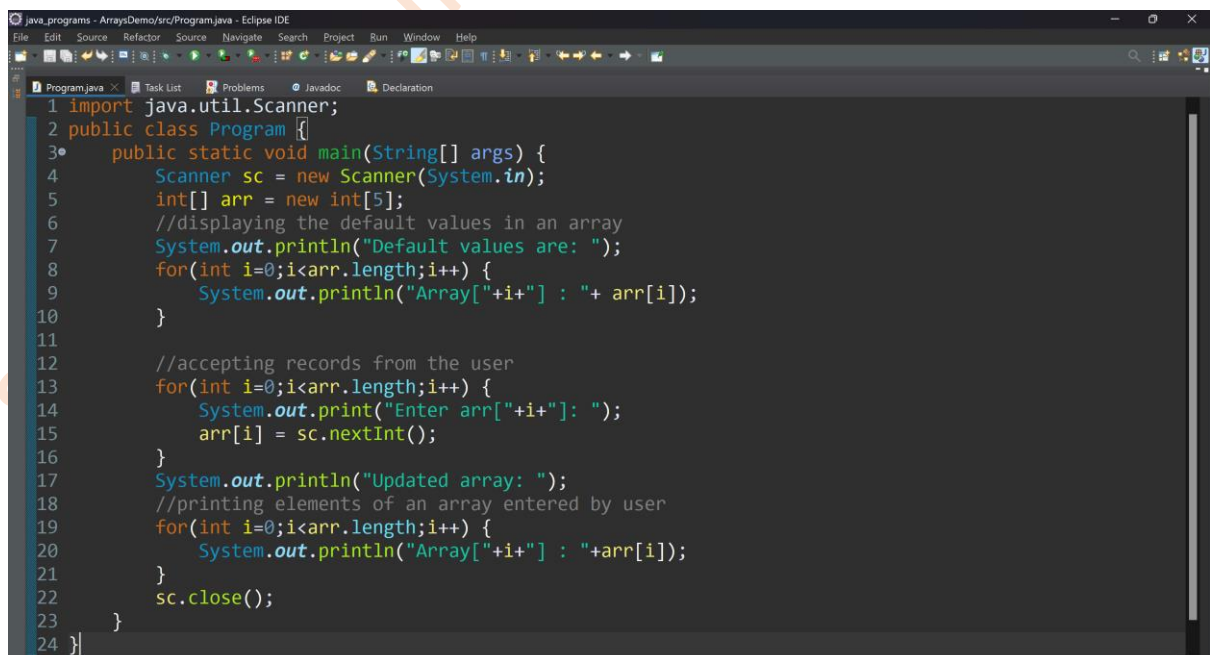


1. Declare a single-dimensional array of 5 integers inside the `main` method. Traverse the array to print the default values. Then accept records from the user and print the updated values of the array.

Code:

```
import java.util.Scanner;
public class Program {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int[] arr = new int[5];
        //displaying the default values in an array
        System.out.println("Default values are: ");
        for(int i=0;i<arr.length;i++) {
            System.out.println("Array["+i+"] : "+ arr[i]);
        }

        //accepting records from the user
        for(int i=0;i<arr.length;i++) {
            System.out.print("Enter arr["+i+"] : ");
            arr[i] = sc.nextInt();
        }
        System.out.println("Updated array: ");
        //printing elements of an array entered by user
        for(int i=0;i<arr.length;i++) {
            System.out.println("Array["+i+"] : "+arr[i]);
        }
        sc.close();
    }
}
```



```
1 import java.util.Scanner;
2 public class Program {
3     public static void main(String[] args) {
4         Scanner sc = new Scanner(System.in);
5         int[] arr = new int[5];
6         //displaying the default values in an array
7         System.out.println("Default values are: ");
8         for(int i=0;i<arr.length;i++) {
9             System.out.println("Array["+i+"] : "+ arr[i]);
10        }
11
12        //accepting records from the user
13        for(int i=0;i<arr.length;i++) {
14            System.out.print("Enter arr["+i+"] : ");
15            arr[i] = sc.nextInt();
16        }
17        System.out.println("Updated array: ");
18        //printing elements of an array entered by user
19        for(int i=0;i<arr.length;i++) {
20            System.out.println("Array["+i+"] : "+arr[i]);
21        }
22        sc.close();
23    }
24 }
```

```

Default values are:
Array[0] : 0
Array[1] : 0
Array[2] : 0
Array[3] : 0
Array[4] : 0
Enter arr[0]: 101
Enter arr[1]: 102
Enter arr[2]: 103
Enter arr[3]: 104
Enter arr[4]: 105
Updated array:
Array[0] : 101
Array[1] : 102
Array[2] : 103
Array[3] : 104
Array[4] : 105

```

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:

```

import java.util.Scanner;

public class Program1 {
    private static Scanner sc = new Scanner( System.in );

    public static void acceptRecord(int[] arr ) {
        if ( arr != null ) {
            for( int i = 0; i < arr.length; i++ ) {
                System.out.print("Enter element [ "+i+" ] : ");
                arr [ i ] = sc.nextInt();
            }
        }
    }

    public static void printRecord( int[] arr ){
        if( arr != null ){
            System.out.print("Array: [ ");
            for( int i = 0; i < arr.length; i++ ){
                System.out.print(arr [i]+" ");
            }
            System.out.println("]");
        }
    }

    public static void main(String[] args) {
        int[] arr = new int[5];
        Program1.acceptRecord(arr);
        Program1.printRecord(arr);
        sc.close();
    }
}

```

```

1 import java.util.Scanner;
2
3 public class Program1 {
4     private static Scanner sc = new Scanner( System.in );
5
6     public static void acceptRecord(int[] arr ) {
7         if ( arr != null ) {
8             for( int i = 0; i < arr.length; i++ ) {
9                 System.out.print("Enter element [ "+i+" ] : ");
10                arr [ i ] = sc.nextInt();
11            }
12        }
13    }
14    public static void printRecord( int[] arr ){
15        if( arr != null ){
16            System.out.print("Array: [ ");
17            for( int i = 0; i < arr.length; i++ ){
18                System.out.print(arr [i]+" ");
19            }
20            System.out.println("]");
21        }
22    }
23 }
24 public static void main(String[] args) {
25     int[] arr = new int[5];
26     Program1.acceptRecord(arr);
27     Program1.printRecord(arr);
28     sc.close();
29 }
30 }

```

```

<terminated> Program1 [Java Application] C:\Program Files\Eclipse\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64_21.0.3.v20240426-1530\jre\bin\javaw.exe (Sep 10, 2024, 12:39:23 PM - 12:39:30 PM) [pid: 261]
Enter element [ 0 ] : 10
Enter element [ 1 ] : 20
Enter element [ 2 ] : 30
Enter element [ 3 ] : 40
Enter element [ 4 ] : 50
Array: [ 10 20 30 40 50 ]

```

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

Code:

```

import java.util.Scanner;
public class Program2 {
    private static Scanner sc = new Scanner( System.in );
    public static void acceptRecord(int[] arr ) {
        if ( arr != null ) {
            for( int i = 0; i < arr.length; i++ ) {
                System.out.print("Enter element [ "+i+" ] : ");
                arr [ i ] = sc.nextInt();
            }
        }
    }
    public static void printRecord( int[] arr ){
        if( arr != null ){
            System.out.print("Array: [ ");
            for( int i = 0; i < arr.length; i++ ){
                System.out.print(arr [i]+" ");
            }
            System.out.println("]");
        }
    }
    public static void maxInArray(int[] arr){

```

```

        if( arr != null ){
            int max=arr[0];
            for( int i = 0; i < arr.length; i++ ){
                if( arr[0] < arr[i] ) {
                    max = arr[i];
                }
            }
            System.out.println("Maximum in an array is "+ max);
        }
    }
}

public static void main(String[] args) {
    int[] arr = new int[5];
    Program2.acceptRecord(arr);
    Program2.printRecord(arr);
    Program2.maxInArray(arr);
    sc.close();
}
}

```

The screenshot shows an IDE with the following code in Program2.java:

```

1 import java.util.Scanner;
2 public class Program2 {
3     private static Scanner sc = new Scanner( System.in );
4     public static void acceptRecord(int[] arr ) {
5         if ( arr != null ) {
6             for( int i = 0; i < arr.length; i++ ) {
7                 System.out.print("Enter element [ "+i+" ] : ");
8                 arr [ i ] = sc.nextInt();
9             }
10        }
11    }
12    public static void printRecord( int[] arr ){
13        if( arr != null ){
14            System.out.print("Array: [ ");
15            for( int i = 0; i < arr.length; i++ ){
16                System.out.print(arr [i]+" ");
17            }
18            System.out.println("");
19        }
20    }
21    public static void maxInArray(int[] arr){
22        if( arr != null ){
23            int max=arr[0];
24            for( int i = 0; i < arr.length; i++ ){
25                if( arr[0] < arr[i] ) {
26                    max = arr[i];
27                }
28            }
29            System.out.println("Maximum in an array is "+ max);
30        }
31    }
32    public static void main(String[] args) {
33        int[] arr = new int[5];
34        Program2.acceptRecord(arr);
35        Program2.printRecord(arr);
36        Program2.maxInArray(arr);
37        sc.close();
38    }
39 }

```

The console output shows the following sequence of events:

```

Enter element [ 0 ] : 12
Enter element [ 1 ] : 1
Enter element [ 2 ] : 34
Enter element [ 3 ] : 56
Enter element [ 4 ] : 76
Array: [ 12 1 34 56 76 ]
Maximum in an array is 76

```

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

Code:

```

import java.util.Arrays;
import java.util.Scanner;
public class Program3 {
    private static Scanner sc = new Scanner( System.in );
    public static void acceptRecord(int[] arr ) {
        if ( arr != null) {
            for( int i = 0; i < arr.length; i++ ) {
                System.out.print("Enter element [ "+i+" ] : ");
                arr [ i ] = sc.nextInt();
            }
        }
    }
    public static int[] removeDuplicate(int[] arr) {
        Arrays.sort(arr);
        int[] uniqueArray = new int[arr.length];
        int j = 0;
        for (int i = 0; i < arr.length - 1; i++) {
            if (arr[i] != arr[i + 1]) {
                uniqueArray[j++] = arr[i];
            }
        }
        uniqueArray[j++] = arr[arr.length - 1];
        uniqueArray = Arrays.copyOf(uniqueArray, j);
        return uniqueArray;
    }
    public static void showRecord(int[] uniqueArray) {
        System.out.println("Array without duplicates: " + Arrays.toString(uniqueArray));
    }
    public static void main(String[] args) {
        int[] arr = new int[5];
        int[] newArr = new int[5];
        Program3.acceptRecord(arr);
        newArr = Program3.removeDuplicate(arr);
        Program3.showRecord(newArr);
    }
}

```

```

1 import java.util.Arrays;
2
3
4 public class Program3 {
5     private static Scanner sc = new Scanner(System.in);
6     public static void acceptRecord(int[] arr) {
7         if (arr != null) {
8             for (int i = 0; i < arr.length; i++) {
9                 System.out.print("Enter element [ " + i + " ] : ");
10                arr[i] = sc.nextInt();
11            }
12        }
13    }
14    public static int[] removeDuplicate(int[] arr) {
15        Arrays.sort(arr);
16        int[] uniqueArray = new int[arr.length];
17        int j = 0;
18
19        for (int i = 0; i < arr.length - 1; i++) {
20            if (arr[i] != arr[i + 1]) {
21                uniqueArray[j++] = arr[i];
22            }
23        }
24        uniqueArray[j++] = arr[arr.length - 1];
25        uniqueArray = Arrays.copyOf(uniqueArray, j);
26
27        return uniqueArray;
28    }
29
30    public static void showRecord(int[] uniqueArray) {
31        System.out.println("Array without duplicates: " + Arrays.toString(uniqueArray));
32    }
33    public static void main(String[] args) {
34        int[] arr = new int[5];
35        int[] newArr = new int[5];
36        Program3.acceptRecord(arr);
37        newArr = Program3.removeDuplicate(arr);
38        Program3.showRecord(newArr);
39    }
40 }

```

Console

```

<terminated> Program3 [Java Application] C:\Program Files\Eclipse\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64_21.0.3.v20240426-1530\jre\bin\javaw.exe (Sep 10, 2024, 5:40:26 PM - 5:40:33 PM) [pid: 16440]
Enter element [ 0 ] : 12
Enter element [ 1 ] : 12
Enter element [ 2 ] : 2
Enter element [ 3 ] : 3
Enter element [ 4 ] : 12
Array without duplicates: [2, 3, 12]

```

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

Code:

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

```

public class Program5 {
    public static void main(String[] args) {

        int[] array1 = {1, 2, 3, 4, 5, 8, 6};
        int[] array2 = {3, 4, 5, 6, 7, 8};
        int[] intersectionArr = new int[Math.min(array1.length, array2.length)];

        Arrays.sort(array1);
        Arrays.sort(array2);
        int i = 0, j = 0, k = 0;

        while (i < array1.length && j < array2.length) {
            if (array1[i] == array2[j]) {
                intersectionArr[k++] = array1[i];
                i++;
                j++;
            } else if (array1[i] < array2[j]) {
                i++;
            } else {
                j++;
            }
        }
    }
}

```

```

    }
    intersectionArr = Arrays.copyOf(intersectionArr, k);
    System.out.println("Intersection of the two arrays: " +
Arrays.toString(intersectionArr));
}
}

```

```

1 import java.util.Arrays;
2
3 public class Program5 {
4     public static void main(String[] args) {
5
6         int[] array1 = {1, 2, 3, 4, 5, 8, 6};
7         int[] array2 = {3, 4, 5, 6, 7, 8};
8         int[] intersectionArr = new int[Math.min(array1.length, array2.length)];
9
10        Arrays.sort(array1);
11        Arrays.sort(array2);
12        int i = 0, j = 0, k = 0;
13
14        while (i < array1.length && j < array2.length) {
15            if (array1[i] == array2[j]) {
16                intersectionArr[k++] = array1[i];
17                i++;
18                j++;
19            } else if (array1[i] < array2[j]) {
20                i++;
21            } else {
22                j++;
23            }
24        }
25        intersectionArr = Arrays.copyOf(intersectionArr, k);
26        System.out.println("Intersection of the two arrays: " + Arrays.toString(intersectionArr));
27    }
28 }

```

Console Output: Intersection of the two arrays: [3, 4, 5, 6, 8]

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

Code:

```

public class Program6 {
    public static void main(String[] args) {

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

        int expectedSum = N * (N + 1) / 2;
        int actualSum = 0;
        for (int num : array) {
            actualSum += num;
        }

        int missingNumber = expectedSum - actualSum;

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

```

```

1 public class Program6 {
2     public static void main(String[] args) {
3
4         int[] array = {1, 2, 4, 5, 6};
5         int N = 6;
6
7         int expectedSum = N * (N + 1) / 2;
8         int actualSum = 0;
9         for (int num : array) {
10             actualSum += num;
11         }
12
13         int missingNumber = expectedSum - actualSum;
14
15         System.out.println("The missing number is: " + missingNumber);
16     }
17 }
18

```

Console Output: The missing number is: 3

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:

import java.util.Arrays;

import java.util.Scanner;

```

public class Program7 {
    int[] arr;
    public static Scanner sc = new Scanner(System.in);

    public Program7(int size) {
        arr = new int[size];
    }
    public void acceptRecord() {
        for (int i = 0; i < arr.length; i++) {
            System.out.print("Enter element Array["+i+"]:");
            arr[i] = sc.nextInt();
        }
    }
    public void printRecord() {
        System.out.println("Entered array is : "+Arrays.toString(arr));
    }
    public static void main(String[] args) {
        System.out.print("Enter array size: ");
        int size = sc.nextInt();
        Program7 arr1 = new Program7(size);
        arr1.acceptRecord();
        arr1.printRecord();
        sc.close();
    }
}

```


ASSIGNMENT NO.6

```

1 import java.util.Arrays;
2 import java.util.Scanner;
3
4 public class Program7 {
5     int[] arr;
6     public static Scanner sc = new Scanner(System.in);
7
8     public Program7(int size) {
9         arr = new int[size];
10    }
11    public void acceptRecord() {
12        for (int i = 0; i < arr.length; i++) {
13            System.out.print("Enter element Array["+i+"]:");
14            arr[i] = sc.nextInt();
15        }
16    }
17    public void printRecord() {
18        System.out.println("Entered array is : "+Arrays.toString(arr));
19    }
20
21    public static void main(String[] args) {
22        System.out.print("Enter array size: ");
23        int size = sc.nextInt();
24        Program7 arr1 = new Program7(size);
25        arr1.acceptRecord();
26        arr1.printRecord();
27        sc.close();
28    }
29
30 }

```

```

<terminated> Program7 [Java Application] C:\Program Files\Eclipse\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64_21.0.3.v20240426-1530\jre\bin\javaw.exe (Sep 11, 2024, 11:39:12 AM - 11:39:23 AM) [pid: 272]
Enter array size: 5
Enter element Array[0]:12
Enter element Array[1]:34
Enter element Array[2]:41
Enter element Array[3]:23
Enter element Array[4]:54
Entered array is : [12, 34, 41, 23, 54]

```

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

Code:

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

```

public class Program8 {
    int[] arr;
    public static Scanner sc = new Scanner(System.in);

    public Program8(int size) {
        arr = new int[size];
    }
    public int[] getArr() {
        return arr;
    }
    public int[] setArr() {
        for (int i = 0; i < arr.length; i++) {
            System.out.print("Enter element Array["+i+"]:");
            arr[i] = sc.nextInt();
        }
        return arr;
    }
    public static void main(String[] args) {
        System.out.print("Enter array size: ");
        int size = sc.nextInt();
        Program8 arr1 = new Program8(size);
        arr1.setArr();
    }
}

```

ASSIGNMENT NO.6

```

        System.out.println("Entered array is : "+Arrays.toString(arr1.getArr()));
        sc.close();
    }
}

```

```

1 import java.util.Arrays;
2 import java.util.Scanner;
3
4 public class Program8 {
5     int[] arr;
6     public static Scanner sc = new Scanner(System.in);
7
8     public Program8(int size) {
9         arr = new int[size];
10    }
11    public int[] getArr() {
12        return arr;
13    }
14    public int[] setArr() {
15
16        for (int i = 0; i < arr.length; i++) {
17            System.out.print("Enter element Array["+i+"]:");
18            arr[i] = sc.nextInt();
19        }
20        return arr;
21    }
22    public static void main(String[] args) {
23        System.out.print("Enter array size: ");
24        int size = sc.nextInt();
25        Program8 arr1 = new Program8(size);
26        arr1.setArr();
27        System.out.println("Entered array is : "+Arrays.toString(arr1.getArr()));
28        sc.close();
29    }
30 }

```

Console Output:

```

<terminated> Program8 [Java Application] C:\Program Files\Eclipse\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64_21.0.3.v20240426-1530\jre\bin\javaw.exe (Sep 11, 2024, 11:40:06 AM - 11:40:16 AM) [pid: 311]
Enter array size: 4
Enter element Array[0]:11
Enter element Array[1]:43
Enter element Array[2]:65
Enter element Array[3]:77
Entered array is : [11, 43, 65, 77]

```

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:

```

import java.util.Scanner;

public class Program9 {
    private char[][] seats;
    private int rows, columns;
    public static Scanner sc = new Scanner(System.in);

    public Program9(int rows, int columns) {
        this.rows = rows;
        this.columns = columns;
        seats = new char[rows][columns];
        for (int i = 0; i < rows; i++) {

```

```

        for (int j = 0; j < columns; j++) {
            seats[i][j] = 'A'; //A = available
        }
    }
}

public void displaySeatingChart() {
    System.out.println("Current seating chart:");
    for (int i = 0; i < rows; i++) {
        for (int j = 0; j < columns; j++) {
            System.out.print(seats[i][j] + " ");
        }
        System.out.println();
    }
}

// Method to book a seat
public void bookSeat(int row, int col) {
    if (row < 0 || row >= rows || col < 0 || col >= columns) {
        System.out.println("Invalid seat position!");
    } else if (seats[row][col] == 'X') {
        System.out.println("Seat is already booked.");
    } else {
        seats[row][col] = 'X'; // 'X' indicates the seat is booked
        System.out.println("Seat booked successfully.");
    }
}

// Method to cancel a booking
public void cancelSeat(int row, int col) {
    if (row < 0 || row >= rows || col < 0 || col >= columns) {
        System.out.println("Invalid seat position!");
    } else if (seats[row][col] == 'A') {
        System.out.println("Seat is already available.");
    } else {
        seats[row][col] = 'A'; // Mark the seat as available again
        System.out.println("Booking cancelled successfully.");
    }
}

// Method to check seat availability
public boolean checkSeatAvailability(int row, int col) {
    if (row < 0 || row >= rows || col < 0 || col >= columns) {
        System.out.println("Invalid seat position!");
        return false;
    }
    if (seats[row][col] == 'A') {

```

```

        System.out.println("Seat is available.");
        return true;
    } else {
        System.out.println("Seat is occupied.");
        return false;
    }
}

public static void main(String[] args) {
    System.out.print("Enter number of rows: ");
    int rows = sc.nextInt();
    System.out.print("Enter number of columns: ");
    int columns = sc.nextInt();

    // Initialize the seating arrangement
    Program9 airplane = new Program9(rows, columns);

    // Menu-driven program for seat booking system
    while (true) {
        System.out.println("\nMenu:");
        System.out.println("1. Display Seating Chart");
        System.out.println("2. Book a Seat");
        System.out.println("3. Cancel a Booking");
        System.out.println("4. Check Seat Availability");
        System.out.println("5. Exit");
        System.out.print("Choose an option: ");
        int choice = sc.nextInt();

        switch (choice) {
            case 1:
                airplane.displaySeatingChart();
                break;
            case 2:
                System.out.print("Enter row to book: ");
                int bookRow = sc.nextInt();
                System.out.print("Enter column to book: ");
                int bookCol = sc.nextInt();
                airplane.bookSeat(bookRow - 1, bookCol - 1);
                break;
            case 3:
                System.out.print("Enter row to cancel: ");
                int cancelRow = sc.nextInt();
                System.out.print("Enter column to cancel: ");
                int cancelCol = sc.nextInt();
                airplane.cancelSeat(cancelRow - 1, cancelCol - 1);
                break;
            case 4:

```

```
System.out.print("Enter row to check: ");
int checkRow = sc.nextInt();
System.out.print("Enter column to check: ");
int checkCol = sc.nextInt();
airplane.checkSeatAvailability(checkRow - 1, checkCol - 1);
break;
case 5:
    System.out.println("Exiting...");
    sc.close();
    System.exit(0);
    break;
default:
    System.out.println("Invalid option. Please try again.");
}}}
```

}

```

Console × Program9.java
<terminated> Program9 [Java Application] C:\Program Files\Eclipse\eclipse\plugins\org.eclipse.justj.open
Enter number of rows: 4
Enter number of columns: 4

Menu:
1. Display Seating Chart
2. Book a Seat
3. Cancel a Booking
4. Check Seat Availability
5. Exit
Choose an option: 1
Current seating chart:
A A A A
A A A A
A A A A
A A A A

Menu:
1. Display Seating Chart
2. Book a Seat
3. Cancel a Booking
4. Check Seat Availability
5. Exit
Choose an option: 2
Enter row to book: 1
Enter column to book: 3
Seat booked successfully.

Menu:
1. Display Seating Chart
2. Book a Seat
3. Cancel a Booking
4. Check Seat Availability
5. Exit
Choose an option: 1
Current seating chart:
A A X A
A A A A
A A A A
A A A A

```

```
Console × Program9.java
<terminated> Program9 [Java Application] C:\Program Files\Eclipse\eclipse\plugins\org.eclipse.justj.openjdk.h

Menu:
1. Display Seating Chart
2. Book a Seat
3. Cancel a Booking
4. Check Seat Availability
5. Exit
Choose an option: 3
Enter row to cancel: 1
Enter column to cancel: 3
Booking cancelled successfully.

Menu:
1. Display Seating Chart
2. Book a Seat
3. Cancel a Booking
4. Check Seat Availability
5. Exit
Choose an option: 1
Current seating chart:
A A A A
A A A A
A A A A
A A A A

Menu:
1. Display Seating Chart
2. Book a Seat
3. Cancel a Booking
4. Check Seat Availability
5. Exit
Choose an option: 2
Enter row to book: 4
Enter column to book: 4
Seat booked successfully.
```