Lab 12

1. Imagine a small movie theater with a seating plan that can be represented as a 2D array. Each seat in the theater can either be occupied ('O') or vacant ('V'). Initially, all seats are vacant.

- Write a Method called displaySeating that takes a 2D array representing the seating arrangement of the theater and prints it in a way that each row is shown on a new line. Each seat should be represented by its respective character ('O' or 'V') with spaces in between.

- Write a Method called bookSeat that takes three parameters: the 2D array of the seating arrangement, and two integers row and col indicating the row and column of the seat to be booked. If the seat is vacant, the Method should mark it as occupied ('O') and return True indicating the seat was successfully booked. If the seat is already occupied, it should return False.

- Write a Method called checkFullRows that takes the 2D array and returns a list of indices of the rows where all seats are occupied.

```
    public static void main(String[] args) {
            // Example to test the MovieTheater class
            char[][] seating = {
                {'V', 'V', 'V', 'V'},
                {'V', 'V', 'V', 'V'},
                {'V', 'V', 'V', 'V'},
                {'V', 'V', 'V', 'V'}
            };
    }
    public static void displaySeating(char[][] seating) {
        for (int i = 0; i < seating.length; i++) {
            for (int j = 0; j < seating[i].length; j++) {
                System.out.print(seating[i][j] + " ");
            }
            System.out.println(); // Moves to the next line after printing
a row
        }
    }

        public static boolean bookSeat(char[][] seating, int row, int col)
{
        if (seating[row][col] == 'V') {
            seating[row][col] = 'O'; // Book the seat by marking it as
occupied
            return true; // Seat successfully booked
        }
        return false; // Seat was already occupied
    }

       public static int[] checkFullRows(char[][] seating) {
        int[] fullRows = new int[seating.length]; // Array to indicate
full (1) or not full (0)
```

```
        for (int i = 0; i < seating.length; i++) {
            boolean isFull = true;
            for (int j = 0; j < seating[i].length; j++) {
                if (seating[i][j] == 'V') {
                    isFull = false;
                    break;
                }
            }

            if (isFull) {
                fullRows[i] = 1; // Assign 1 if the row is full
            } else {
                fullRows[i] = 0; // Assign 0 if the row is not full
            }
        }

        return fullRows;
    }
```

---

2. Create a list of daily average temperatures for a month, you can make up the values and amount. The temperatures are stored in a 1D array of type double, each element representing the temperature of a specific day. Your task is to analyze the temperatures to provide useful insights through various Methods.

- Write a Method called averageTemperature that takes a 1D array of temperatures and returns the average temperature of the month rounded to two decimal places.

- Write a Method called countColdDays that takes a 1D array of temperatures and a temperature threshold as parameters. It returns the count of days where the temperature was below the threshold and returns it.

- Write a Method called increaseTemperatures that takes a 1D array of temperatures and a number to increase by which all temperatures should be increased. The Method should modify the original array.

```
    public static void main(String[] args) {
        double[] temperatures = {20.5, 21.3, 22.1, 23.0, 24.5, 25.0, 26.0,
27.0, 28.0, 29.0, 30.0, 31.0, 32.0, 33.0, 34.0, 35.0, 36.0, 37.0, 38.0,
39.0, 40.0, 41.0, 42.0, 43.0, 44.0, 45.0, 46.0, 47.0, 48.0, 49.0};
    }

    public static double averageTemperature(double[] temperatures) {
        double sum = 0;
        for (int i = 0; i < temperatures.length; i++) {
            sum += temperatures[i];
        }
        return sum / temperatures.length;
    }
    public static int countColdDays(double[] temperatures, double
threshold) {
        int count = 0;
        for (int i = 0; i < temperatures.length; i++) {
            if (temperatures[i] < threshold) {
                count++;
            }
        }
        return count;
    }

    public static void increaseTemperatures(double[] temperatures, double
increaseBy) {
        for (int i = 0; i < temperatures.length; i++) {
            temperatures[i] += increaseBy;
        }
    }
```

3. Create a printData method(s) that can accept as parameters an integer, a double, a 1D integer array, and a 2D integer array. The method(s) should print out the data.

```


    public static void main(String[] args) {
        int num = 10;
        double decimal = 10.5;
        int[] arr = {1, 2, 3, 4, 5};
        int[][] arr2D = {
            {1, 2, 3},
            {4, 5, 6},
            {7, 8, 9}
        };
    }

    public static void printData(int num) {
        System.out.println(num);
    }

    public static void printData(double decimal) {
        System.out.println(decimal);
    }

    public static void printData(int[] arr) {
        for (int i = 0; i < arr.length; i++) {
            System.out.print(arr[i] + " ");
        }
        System.out.println();
    }
    public static void printData(int[][] arr2D) {
        for (int i = 0; i < arr2D.length; i++) {
            for (int j = 0; j < arr2D[i].length; j++) {
                System.out.print(arr2D[i][j] + " ");
            }
            System.out.println();
        }
    }
```
