Internal Practical

Problem - 1

Code :

```java
import java.util.*;

class Internal1{
    public static int mostBooked(int n, int[][] meetings) {
        Arrays.sort(meetings, (a, b) -> a[0] - b[0]);

        int[] roomCount = new int[n];
        long[] roomFreeTime = new long[n];

        for (int[] meeting : meetings) {
            int start = meeting[0];
            int duration = meeting[1] - meeting[0];

            int bestRoom = 0;
            long earliestFreeTime = roomFreeTime[0];

            for (int i = 1; i < n; i++) {
                if (roomFreeTime[i] < earliestFreeTime) {
                    earliestFreeTime = roomFreeTime[i];
                    bestRoom = i;
                } else if (roomFreeTime[i] == earliestFreeTime && i < bestRoom) {
                    bestRoom = i;
                }
            }

            if (roomFreeTime[bestRoom] <= start) {
                roomFreeTime[bestRoom] = start + duration;
```

```java
        } else {
            roomFreeTime[bestRoom] = roomFreeTime[bestRoom] + duration;
        }


        roomCount[bestRoom]++;
    }


    int maxMeetings = roomCount[0];
    int resultRoom = 0;


    for (int i = 1; i < n; i++) {
        if (roomCount[i] > maxMeetings) {
            maxMeetings = roomCount[i];
            resultRoom = i;
        }
    }


    return resultRoom;
}

public static void main(String[] args) {
    Scanner s = new Scanner(System.in);
    System.out.print("Enter the number of rooms: ");
    int n = s.nextInt();
    System.out.print("Enter the number of meetings : ");
    int m = s.nextInt();
    int[][] meetings = new int[m][2];
    System.out.println("Enter the meeting times (start and end): ");
    for (int i = 0; i < m; i++) {
        System.out.print("Meeting " + (i + 1) + ": ");
```

```
            meetings[i][0] = s.nextInt();

            meetings[i][1] = s.nextInt();

        }

    System.out.println("Example 1:");

    System.out.println("Input: n = " + n + ", meetings = " + Arrays.deepToString(meetings));

    System.out.println("Output: " + mostBooked(n, meetings));

    System.out.println();

    s.close();

    }

}
```

Output :

```
Enter the number of meetings : 4
Enter the meeting times (start and end):
Meeting 1: 0 10
Meeting 2: 1 5
Meeting 3: 2 7
Meeting 4: 3 4
Example 1:
Input: n = 2, meetings = [[0, 10], [1, 5], [2, 7], [3, 4]]
Output: 0
```

```
Enter the meeting times (start and end):
Meeting 1: 1 20
Meeting 2: 2 10
Meeting 3: 3 5
Meeting 4: 4 9
Meeting 5: 6 8
Example 1:
Input: n = 3, meetings = [[1, 20], [2, 10], [3, 5], [4, 9], [6, 8]]
Output: 1
```

Problem – 2

Code:

```java
import java.util.*;

public class Internal2{
    public static int Trapped(int height[]){
        int n = height.length;

        int leftMax[] = new int[n];
        leftMax[0] = height[0];
        for(int i=1 ; i<n ; i++){
            leftMax[i] = Math.max(height[i] , leftMax[i-1]);
        }

        int rigthMax[] = new int[n];
        rigthMax[n-1] = height[n-1];
        for(int i=n-2 ; i>=0 ; i--){
            rigthMax[i] = Math.max(height[i] , rigthMax[i+1]);
        }

        int Trappedwater = 0;
        for(int i=0 ; i<n ; i++){

            int waterlevel = Math.min(leftMax[i] , rigthMax[i]);
            Trappedwater += waterlevel - height[i];
        }
        return Trappedwater;
    }
    public static void main(String arg[]){
        Scanner s = new Scanner(System.in);
```

```java
        System.out.print("Enter the Length of array : ");

    int n = s.nextInt();

    System.out.print("Enter the elements of array : ");

    int height[] = new int[n];

    for(int i=0 ; i<n ; i++){

        height[i] = s.nextInt();

    }

    System.out.print("Total Trapped water = " + Trapped(height));

    s.close();

  }

}
```

Output :

```
Enter the Length of array : 12
Enter the elements of array : 0 1 0 2 1 0 1 3 2 1 2 1
Total Trapped water = 6
```

```
Enter the Length of array : 6
Enter the elements of array : 4 2 0 3 2 5
Total Trapped water = 9
```