

## Non-overlapping Intervals

Difficulty: Medium

Accuracy: 51.92%

Submissions: 32K+

Points: 4

Given a 2D array `intervals[][]` of representing intervals where `intervals[i] = [starti, endi]`. Return the **minimum** number of intervals you need to remove to make the rest of the intervals **non-overlapping**.

### Examples:

**Input:** `intervals[][] = [[1, 2], [2, 3], [3, 4], [1, 3]]`

**Output:** 1

**Explanation:** [1, 3] can be removed and the rest of the intervals are non-overlapping.

**Input:** `intervals[][] = [[1, 3], [1, 3], [1, 3]]`

**Output:** 2

**Explanation:** You need to remove two [1, 3] to make the rest of the intervals non-overlapping.

**Input:** `intervals[][] = [[1, 2], [5, 10], [18, 35], [40, 45]]`

**Output:** 0

**Explanation:** All ranges are already non overlapping.

```
1 // } Driver Code Ends
30
31 class Solution {
32     static int minRemoval(int[][] intervals) {
33         int cnt = 0;
34         Arrays.sort(intervals, (a, b) -> a[1] - b[1]);
35         int end = intervals[0][1];
36         for (int i = 1; i < intervals.length; i++) {
37             if (intervals[i][0] < end)
38                 cnt++;
39             else
40                 end = intervals[i][1];
41         }
42         return cnt;
43     }
44 }
45
```

