240. Search a 2D Matrix II

Medium 3572 80 Add to List Share

Write an efficient algorithm that searches for a value in an *m* x *n* matrix. This matrix has the following properties:

- Integers in each row are sorted in ascending from left to right.
- Integers in each column are sorted in ascending from top to bottom.

Example:

Consider the following matrix:

```
[
  [1,   4,  7, 11, 15],
  [2,   5,  8, 12, 19],
  [3,   6,  9, 16, 22],
  [10, 13, 14, 17, 24],
  [18, 21, 23, 26, 30]
]
```

Given target = `5`, return `true`.

Given target = `20`, return `false`.

Accepted 350,840 | Submissions 809,344

Seen this question in a real interview before? Yes No

Companies

Related Topics

```java
class Solution {
    public boolean searchMatrix(int[][] matrix, int target) {
        for (int i = 0; i < matrix.length; i++) {
            for (int j = 0; j < matrix[i].length; j++) {
                if (matrix[i][j] == target)
                    return true;
            }
        }

        return false;
    }
}
```

Your previous code was restored from your local storage. Reset to default