624. Maximum Distance in Arrays

Medium

Topics

Companies

You are given `m` `arrays`, where each array is sorted in **ascending order**.

You can pick up two integers from two different arrays (each array picks one) and calculate the distance. We define the distance between two integers `a` and `b` to be their absolute difference `|a - b|`.

Return *the maximum distance*.

Example 1:

Input: arrays = 1,2,3],[4,5],[1,2,3
Output: 4
Explanation: One way to reach the maximum distance 4 is to pick 1 in the first or third array and pick 5 in the second array.

Example 2:

Input: arrays = 1],[1
Output: 0

Constraints:

- `m == arrays.length`
- `2 <= m <= 105`
- `1 <= arrays[i].length <= 500`
- `-104 <= arrays[i][j] <= 104`
- `arrays[i]` is sorted in **ascending order**.
- There will be at most `105` integers in all the arrays.

```
class Solution {

    public int maxDistance(List<List<Integer>> arrays) {
        int minAvailable = arrays.get(0).get(0);
        int maxAvailable = arrays.get(0).get(arrays.get(0).size() - 1);
        int maxDistance = 0;

        for(int i =1; i < arrays.size();i++){
            List<Integer> currentArray = arrays.get(i);

            int currentMin = currentArray.get(0);
            int currentMax = currentArray.get(currentArray.size()-1);

            maxDistance = Math.max(Math.abs(currentMin- maxAvailable),
maxDistance);
            maxDistance = Math.max(Math.abs(currentMax- minAvailable),
maxDistance);

            maxAvailable = Math.max(currentMax, maxAvailable);
            minAvailable = Math.min(currentMin, minAvailable);

        }
        return maxDistance;
    }
}
```