**#50dayscodingchallenge**:

**#leetcode** **#leetcodechallenge** **#leetcodestreak** **#leetcode2024** **#leetcode50days**

Just kicked off my coding journey with a fascinating problem - "Divide Array Into Arrays With Max Difference".

The task was to partition an array into multiple arrays, each with a size of 3, ensuring that the difference between any two elements in one array is less than or equal to 'k'.

Solved it using Java and excited about the journey ahead!

**#coding** **#algorithm** **#programming** **#challengeaccepted** **#day1** **#codingcommunity** **#50daysofcodingchallenge** **#java** **#python**

Happy coding!

**Solution:-**

class Solution {

public int[ ][ ] divideArray(int[] nums, int k) {

int[ ][ ] ans = new int[nums.length / 3][3];

Arrays.sort(nums);

for (int i = 2; i < nums.length; i += 3) {

if (nums[i] - nums[i - 2] > k)

return new int[0][];

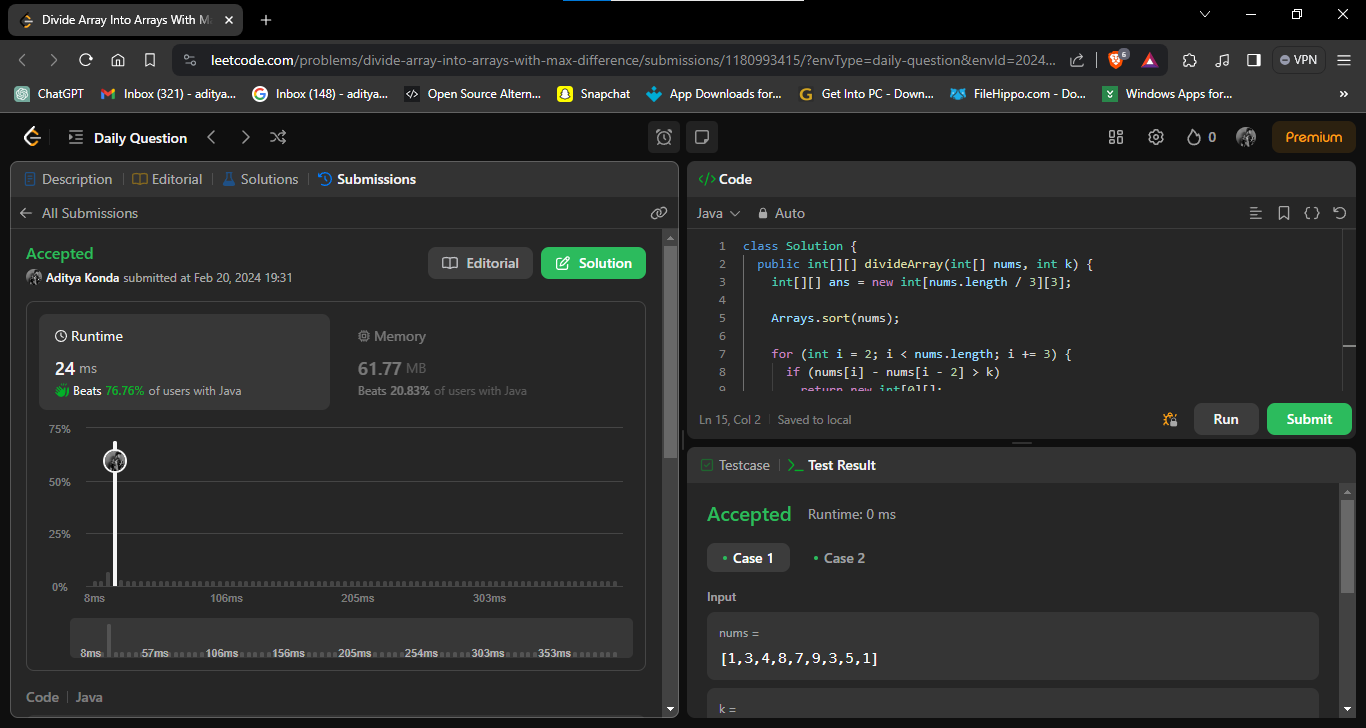
ans[i / 3] = new int[] {nums[i - 2], nums[i - 1], nums[i]};

}

return ans;

}

}



<https://github.com/AdityaKonda6/-50DaysOfCoding>

<https://leetcode.com/problems/divide-array-into-arrays-with-max-difference/?envType=daily-question&envId=2024-02-01>