**[Count Number of Pairs With Absolute Difference K](https://leetcode.com/problems/count-number-of-pairs-with-absolute-difference-k/)**

Given an integer array nums and an integer k, return *the number of pairs* (i, j) *where* i < j *such that* |nums[i] - nums[j]| == k.

**Example 1:**

**Input:** nums = [1,2,2,1], k = 1

**Output:** 4

**Example 2:**

**Input:** nums = [1,3], k = 3

**Output:** 0

**Example 3:**

**Input:** nums = [3,2,1,5,4], k = 2

**Output:** 3

**Step 1 :**  First we should split the modulo eqn into two eqn

**| nums[i] - nums[j] | = k**

**nums[i] - nums[j] = k  ==> nums[i] = nums[j] + k**

**nums[j] - nums[i] = k  ==> nums[i] = nums[j] – k**

**Step 2 :** That’s it . For each iteration check wheter the number exist , if exist , then add that occurrence into the answer.

**Step 3 :** put the current number in the Map.

class Solution {

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

        int cnt = 0;

        int n = nums.length;

        Map<Integer , Integer> map = new HashMap<>();

        for(int i = 0 ; i < n ; i++){

            int req1 = nums[i] + k;

            int req2 = nums[i] - k;

            if(map.containsKey(req1)){

                cnt+= map.get(req1);

            }

            if(k != 0 && map.containsKey(req2)){

                cnt+= map.get(req2);

            }

            map.put(nums[i] , map.getOrDefault(nums[i] , 0) + 1);

        }

        return cnt;

    }

}