Question: https://leetcode.com/problems/subarray-sums-divisible-by-k/

So the logic is to get mod of prefix sum by k and store it as key, and their occurrence as value.

Now if key<0 then key=key+k.

If k present in map val of key+count as count.

Code:  
class Solution {

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

int count = 0, sum=0;

HashMap<Integer, Integer> mem = new HashMap<Integer, Integer>();

mem.put(0,1);

for(int i=0; i<nums.length; i++){

sum+=nums[i];

int pKey= sum%k;

pKey = pKey<0 ? pKey+k : pKey;

int val = mem.getOrDefault(pKey, 0);

count+=val;

mem.put(pKey, val+1);

}

return count;

}

}

Github Link :<https://lnkd.in/ecwtJeaz>