Question: https://leetcode.com/problems/subarray-sum-equals-k/

We can use prefix sum to solve this problem,

For each position we maintain its prefix sum and store that is hashmap, where prefix sum is key and occurence of that sum is it’s value.

Then check if (present prefix sum - target) exists in the hashmap, then there had been such sub-array where the target sum existed.

So count+=map.get(present prefix sum - target);

Code:  
class Solution {

public int subarraySum(int[] nums, int target) {

int count=0, sum=0;

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

mem.put(sum,1);

for(int el: nums){

sum+=el;

if(mem.containsKey(sum-target)){

count+=mem.get(sum-target);

}

mem.put(sum, mem.getOrDefault(sum, 0)+1);

}

return count;

}

}

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