This is contains duplicate problem from NEETCODE sheet question 1st and LEETCODE 217 question

<https://leetcode.com/problems/contains-duplicate/description/>

<https://neetcode.io/problems/duplicate-integer?list=neetcode150>

So in this we have 2-3 approaches like

1st is brute force in that approach we have to traverse in array and then take another loop and check that element is present or not like 1st loop i to n tk and 2nd 1 to n tk or if found then array is duplicate and return true else false

But its complexity is n2 not optimal approach .

2nd is using set and length function in this approach what we will do we will find the length of that array with length of array by finding set if both length are equal that means no duplicate so return false else true

3rd is most optimal approach that is hashset in this hashset we will check the apperance of element in this hashset if any element frequency is more than 2 means it contain duplicate and return true

Most optimal as tc is o(n) and sc is also o(1)

So code of this will be

hashset=set()

For i in arr:

If i in hashset:

Return true

hashset.add(i)

Return false

**NOTE : we are using hashset there which only checks the presence of that element but it will not count that how many times element arise in array but in hashmap or dictionary we are able to find the frequency of element also**