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

**Contains Duplicate**

**Given an integer array *nums*, return *true* if any value appears *at least twice*in the array, and return*false* if every element is distinct.**

**Example 1:**

**Input: nums = [1,2,3,1]**

**Output: true**

**Example 2:**

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

**Output: false**

**Example 3:**

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

**Output: true**

**Constraints:**

**1 <= nums.length <= 105**

**-109 <= nums[i] <= 109**

**Method 1: (Brute Force)**

Take an element and compare with every other element in the array if they are same(true)

Or not(false)

Time Complexity: O(n2)

Space Complexity: O(1)

**Method 2:**

Create an unordered\_map to map the count of each distinct element in the array

if count of an element>1 return “true”, otherwise return “false”.

Time Complexity: O(n) *[traversing the array; insertion in unordered\_map is O(1)]*

Space Complexity: O(n) *[map has n elements]*

bool containsDuplicate(vector<int>& nums) {

        unordered\_map<int,int> mp;

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

            mp[nums[i]]++;

            if(mp[nums[i]]>1)

                return true;

        }

        return false;

    }