

2. 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

Solution

```
/**
 * @param {number[]} nums
 * @return {boolean}
 */

//First method using Map() (exit early if true)
var containsDuplicate = function(nums) {
    const numsSet = new Set()
    for(const i of nums){
        if(numsSet.has(i)){
            return true
        }
        numsSet.add(i)
    }
    return false;
};

//Second method using Map() (Has to map entire array but code is more readable)
var containsDuplicate = function(nums) {
    //create a new hashmap with all the items in the array. Any duplicates will be removed.
    const totalWithoutDuplicates = new Map(nums.map((i) => [i]));

    //check if the size of the initial array is larger than the new hashmap.
    return totalWithoutDuplicates.size !== nums.length;
};

//Third method using Set() (Fastest runtime at 91.95% and very readable code)
var containsDuplicate = function(nums) {
    //Pass the array into a Set() (which removes duplicates) and then compare its size to the original array.
    return new Set(nums).size !== nums.length;
};
```