75 Days of Code
Day 68
Problem no : Leetcode 136
Problem Title : Single Number
Problem type : Bit manipulation

Given a non-empty array of integers nums, every element appears twice except for one. Find that single one.

You must implement a solution with a linear runtime complexity and use only constant extra space.

Example 1:

Input: nums = [2,2,1]
Output: 1
Example 2:

Input: nums = [4,1,2,1,2]
Output: 4
Example 3:

Input: nums = [1]
Output: 1

Constraints:

1 <= nums.length <= 3 * 104
-3 * 104 <= nums[i] <= 3 * 104

Each element in the array appears twice except for one element which appears only once.

```typescript
function singleNumber(nums: number[]): number {
    if(nums.length==1)return nums[0];

    let ans=nums[0] ;
    for(let num =1;num<nums.length;num++){
        ans = ans ^ nums[num];
    }
    return ans;

};
```

Runtime Details

65 ms

Beats 54.10% of users with TypeScript

Memory

44.92 MB

Beats 90.73% of users with TypeScript