Given an integer array nums, move all 0's to the end of it while maintaining the relative order of the non-zero elements.

Note that you must do this in-place without making a copy of the array.

Example 1:

- **Input:** nums = [0,1,0,3,12]
- **Output:** [1,3,12,0,0]

Example 2:

- **Input:** nums = [0]
- **Output:** [0]

Constraints:

- 1 <= nums.length <= 104
- -231 <= nums[i] <= 231 - 1

Follow up: Could you minimize the total number of operations done?

Approach:
The code moves all non-zero elements of the `nums` array to the front, preserving their order, and sets the remaining elements to zero.

Code:

```
class Solution {
    public void moveZeroes(int[] nums) {
        if(nums.length==0){
            System.out.println(nums[0]);
        }
        int j=0;
        for(int i=0;i<nums.length;i++){
            if(nums[i]!=0){
```

```
                nums[j]=nums[i];
                j++;
            }
        }
        for(int i=j;i<nums.length;i++){
            nums[i]=0;
        }
    }
}
```