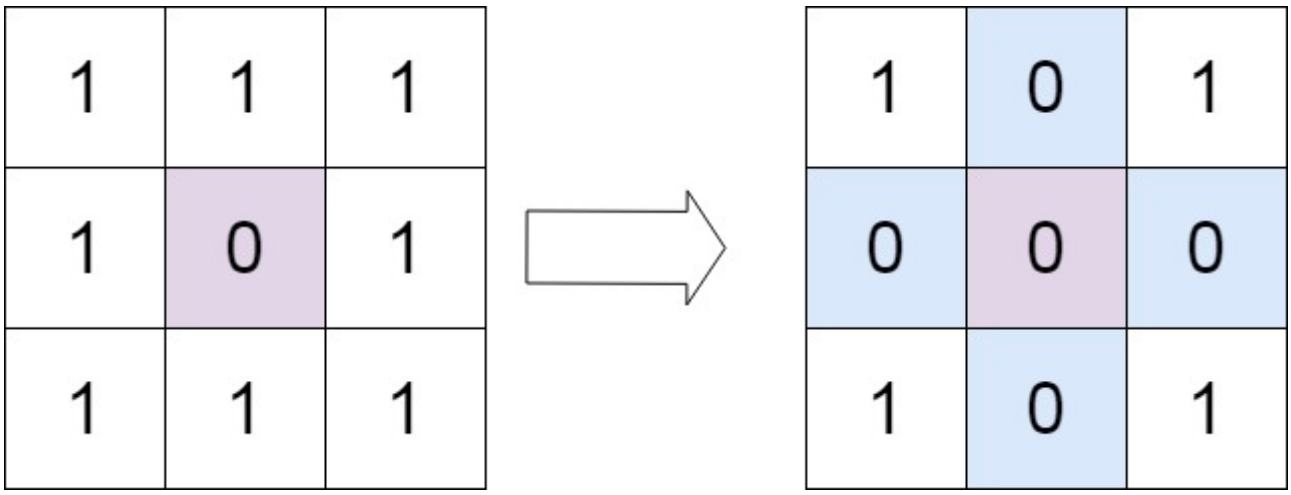
73. Set Matrix Zeroes

Given an `m x n` integer matrix `matrix`, if an element is `0`, set its entire row and column to `0`'s, and return *the matrix*.

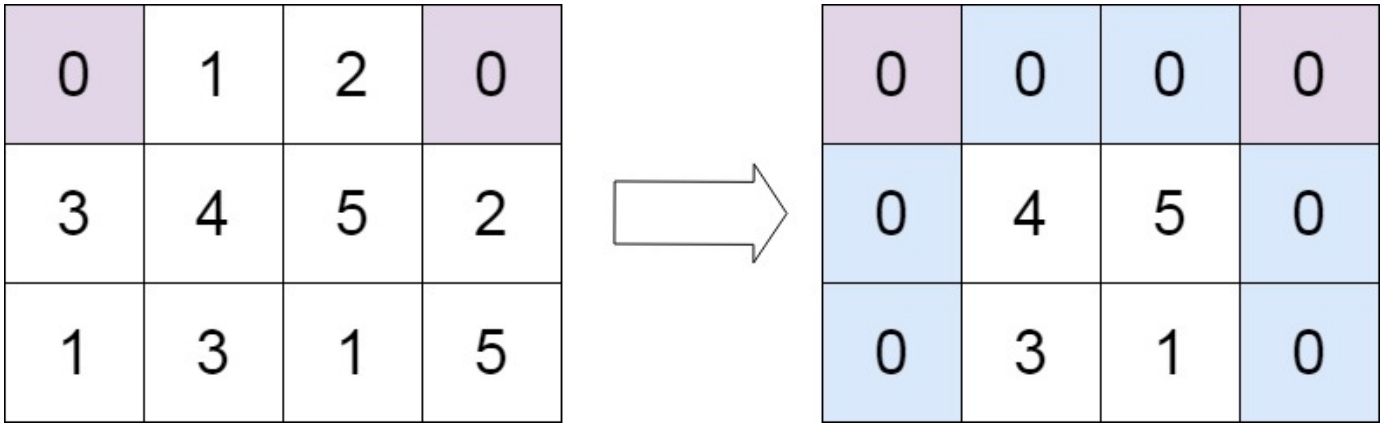
You must do it in place.

Example 1:




```
Input: matrix = [[1,1,1],[1,0,1],[1,1,1]]
Output: [[1,0,1],[0,0,0],[1,0,1]]
```

Example 2:




```
Input: matrix = [[0,1,2,0],[3,4,5,2],[1,3,1,5]]
Output: [[0,0,0,0],[0,4,5,0],[0,3,1,0]]
```

Constraints:

- `m == matrix.length`
- `n == matrix[0].length`

- `1 <= m, n <= 200`
- `-2<sup>31</sup> <= matrix[i][j] <= 2<sup>31</sup> - 1`

Follow up:

- A straightforward solution using `O(mn)` space is probably a bad idea.
- A simple improvement uses `O(m + n)` space, but still not the best solution.
- Could you devise a constant space solution?

```
class Solution:
    def setZeroes(self, matrix: List[List[int]]) -> None:
        """
        Do not return anything, modify matrix in-place instead.
        """
        rows = [False]*len(matrix)
        cols = [False]*len(matrix[0])

        for i in range(len(matrix)):
            for j in range(len(matrix[0])):
                if matrix[i][j]==0:
                    rows[i] = True
                    cols[j] = True

        for i in range(len(matrix)):
            for j in range(len(matrix[0])):
                if rows[i]==True or cols[j]==True:
                    matrix[i][j] = 0
```