Problem 3127: Make a Square with the Same Color

Problem Information

Difficulty: Easy
Acceptance Rate: 52.58%
Paid Only: No
Tags: Array, Matrix, Enumeration

Problem Description

You are given a 2D matrix `grid` of size `3 x 3` consisting only of characters `'B'` and `'W'`. Character `'W'` represents the white color, and character `'B'` represents the black color.

Your task is to change the color of **at most one** cell so that the matrix has a `2 x 2` square where all cells are of the same color.

Return `true` if it is possible to create a `2 x 2` square of the same color, otherwise, return `false`.

Example 1:

Input: grid = [["B","W","B"],["B","W","W"],["B","W","B"]]

Output: true

Explanation:

It can be done by changing the color of the `grid[0][2]`.

Example 2:

Input: grid = [["B","W","B"],["W","B","W"],["B","W","B"]]

Output: false

Explanation:

It cannot be done by changing at most one cell.

Example 3:

Input: grid = [["B","W","B"],["B","W","W"],["B","W","W"]]

Output: true

Explanation:

The `grid` already contains a `2 x 2` square of the same color.

Constraints:

* `grid.length == 3` * `grid[i].length == 3` * `grid[i][j]` is either `'W'` or `'B'`.

Code Snippets

C++:

```
class Solution {
public:
bool canMakeSquare(vector<vector<char>>& grid) {

}
};
```

Java:

```
class Solution {
public boolean canMakeSquare(char[][] grid) {

}
}
```

Python3:

```
class Solution:
def canMakeSquare(self, grid: List[List[str]]) -> bool:
```