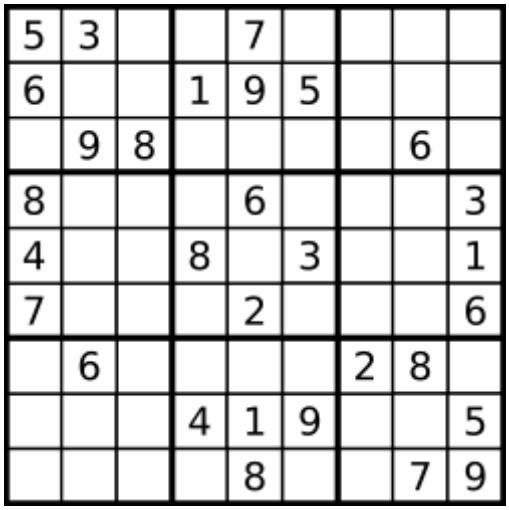
64. **Determine if a 9 x 9 Sudoku board is valid. Only the filled cells need to be validated according to the following rules:**

1. **Each row must contain the digits 1-9 without repetition.**
2. **Each column must contain the digits 1-9 without repetition.**
3. **Each of the nine 3 x 3 sub-boxes of the grid must contain the digits 1-9 without repetition.**

Note:

- **A Sudoku board (partially filled) could be valid but is not necessarily solvable.**
- **Only the filled cells need to be validated according to the mentioned rules.**

Example 1:



5	3			7				
6			1	9	5			
	9	8					6	
8				6				3
4			8		3			1
7				2				6
	6					2	8	
			4	1	9			5
				8			7	9

Input: board =

[["5","3",".",".","7",".",".",".",".","."]

,["6",".",".","1","9","5",".",".",".","."]

,[".","9","8",".",".",".",".",".","6","."]

,["8",".",".",".","6",".",".",".",".","3"]

,["4",".",".","8",".","3",".",".","1"]

,["7",".",".",".",".","2",".",".",".","6"]

,[".","6",".",".",".",".",".","2","8","."]

,[".",".",".","4","1","9",".",".",".","5"]

,[".",".",".",".",".","8",".",".","7","9"]]

Output: true

AIM: To check the Sudoku Board is valid

PROGRAM:

```
def isValidSudoku(board):
    for i in range(9):
        row_set = set()
        for j in range(9):
            if board[i][j] != '.':
                if board[i][j] in row_set:
                    return False
                row_set.add(board[i][j])

    for j in range(9):
        col_set = set()
        for i in range(9):
            if board[i][j] != '.':
                if board[i][j] in col_set:
                    return False
                col_set.add(board[i][j])

    for i in range(0, 9, 3):
        for j in range(0, 9, 3):
            box_set = set()
            for x in range(3):
                for y in range(3):
                    if board[i+x][j+y] != '.':
                        if board[i+x][j+y] in box_set:
                            return False
                        box_set.add(board[i+x][j+y])

    return True

board = [
    ["5","3",".",".","7",".",".",".",".","."],
    ["6",".",".","1","9","5",".",".",".","."],
    [".","9","8",".",".",".",".",".","6","."],
    ["8",".",".",".",".","6",".",".",".","3"],
    ["4",".",".",".","8",".","3",".",".","1"],
    ["7",".",".",".",".","2",".",".",".","6"],
    [".","6",".",".",".",".",".","2","8","."],
    [".",".",".",".","4","1","9",".",".","5"],
    [".",".",".",".",".","8",".",".","7","9"]
]
print(isValidSudoku(board))
```

OUTPUT:

```
True
```

TIME COMPLEXITY : O(1)