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.**

PROGRAM:

```
def isValidSudoku(board):
    # Initialize sets for rows, columns, and 3x3 sub-boxes
    rows = [set() for _ in range(9)]
    cols = [set() for _ in range(9)]
    boxes = [set() for _ in range(9)]

    for r in range(9):
        for c in range(9):
            if board[r][c] != '.':
                num = board[r][c]
                box_index = (r // 3) * 3 + (c // 3)

                if num in rows[r] or num in cols[c] or num in boxes[box_index]:
                    return False

                rows[r].add(num)
                cols[c].add(num)
                boxes[box_index].add(num)

    return True

# Example usage:
board1 = [
    ["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(board1))  # Output: True

board2 = [
    ["8", "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(board2))  # Output: False
```

OUTPUT:-

```
True
False

=== Code Execution Successful ===
```

TIME COMPLEXITY:-O(n)