

208. Write a program to solve a Sudoku puzzle by filling the empty cells. A sudoku solution must satisfy all of the following rules: Each of the digits 1-9 must occur exactly once in each row. Each of the digits 1-9 must occur exactly once in each column. Each of the digits 1-9 must occur exactly once in each of the 9 3x3 sub-boxes of the grid. The '.' character indicates empty cells.

Example 1:

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:

```
[["5","3","4","6","7","8","9","1","2"],
 ["6","7","2","1","9","5","3","4","8"],
 ["1","9","8","3","4","2","5","6","7"],
 ["8","5","9","7","6","1","4","2","3"],
 ["4","2","6","8","5","3","7","9","1"],
 ["7","1","3","9","2","4","8","5","6"],
 ["9","6","1","5","3","7","2","8","4"],
 ["2","8","7","4","1","9","6","3","5"],
 ["3","4","5","2","8","6","1","7","9"]]
```

PROGRAM:-

```
def solveSudoku(board):
```

```
    def is_valid(board, row, col, num):
```

```
        for i in range(9):
```

```
            if board[row][i] == num or board[i][col] == num or board[3 * (row // 3) + i // 3][3 * (col // 3) + i % 3] == num:
```

```
                return False
```

```
        return True
```

```
def solve(board):
```

```
    for i in range(9):
```

```
        for j in range(9):
```

```
            if board[i][j] == '.':
```

```
                for num in '123456789':
```

```
                    if is_valid(board, i, j, num):
```

```
                        board[i][j] = num
```

```
                        if solve(board):
```

```
                            return True
```

```
                        board[i][j] = '.'
```

```
        return False
    return True
```

```
solve(board)
return board
```

Example

```
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(solveSudoku(board))
```

OUTPUT:-

```
[['5', '3', '4', '6', '7', '8', '9', '1', '2'], ['6', '7', '2', '1', '9', '5', '3', '4', '8'], ['1', '9', '8', '3', '4', '2', '5', '6', '7'], ['8', '5', '9', '7', '6', '1', '4', '2', '3'], ['4', '2', '6', '8', '5', '3', '7', '9', '1'], ['7', '1', '3', '9', '2', '4', '8', '5', '6'], ['9', '6', '1', '5', '3', '7', '2', '8', '4'], ['2', '8', '7', '4', '1', '9', '6', '3', '5'], ['3', '4', '5', '2', '8', '6', '1', '7', '9']]
```

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

TIME COMPLEXITY:- $O(9^{81})$