

```

import java.util.Scanner;
public class SudokuSolver {
    private static final int GRID_SIZE = 9;
    public static void main(String[] args) {
        int[][] board = new int[GRID_SIZE][GRID_SIZE];
        Scanner scanner = new Scanner(System.in);
        System.out.println("Enter the Sudoku puzzle to solve (use 0 for empty
cells):");
        for (int i = 0; i < GRID_SIZE; i++) {
            for (int j = 0; j < GRID_SIZE; j++) {
                board[i][j] = scanner.nextInt();
            }
        }
        if (solveBoard(board)) {
            System.out.println("Sudoku puzzle solved successfully!");
            printBoard(board);
        } else {
            System.out.println("Unable to solve the Sudoku puzzle.");
        }
        scanner.close();
    }
    private static boolean solveBoard(int[][] board) {
        for (int row = 0; row < GRID_SIZE; row++) {
            for (int col = 0; col < GRID_SIZE; col++) {
                if (board[row][col] == 0) {
                    for (int num = 1; num <= GRID_SIZE; num++) {
                        if (isValidPlacement(board, num, row, col)) {
                            board[row][col] = num;
                            if (solveBoard(board)) {
                                return true;
                            } else {
                                board[row][col] = 0; // Reset the cell
                            }
                        }
                    }
                    return false;
                }
            }
        }
        return true;
    }
    private static boolean isValidPlacement(int[][] board, int num, int row, int
col) {
        return !isInRow(board, num, row) &&
            !isInCol(board, num, col) &&
            !isInBox(board, num, row, col);
    }
    private static boolean isInRow(int[][] board, int num, int row) {
        for (int i = 0; i < GRID_SIZE; i++) {
            if (board[row][i] == num) {

```

```

        return true;
    }
}
return false;
}
private static boolean isInCol(int[][] board, int num, int col) {
    for (int i = 0; i < GRID_SIZE; i++) {
        if (board[i][col] == num) {
            return true;
        }
    }
    return false;
}
private static boolean isInBox(int[][] board, int num, int row, int col) {
    int localBoxRow = row - row % 3;
    int localBoxCol = col - col % 3;

    for (int i = localBoxRow; i < localBoxRow + 3; i++) {
        for (int j = localBoxCol; j < localBoxCol + 3; j++) {
            if (board[i][j] == num) {
                return true;
            }
        }
    }
    return false;
}
private static void printBoard(int[][] board) {
    for (int row = 0; row < GRID_SIZE; row++) {
        if (row % 3 == 0 && row != 0) {
            System.out.println("-----");
        }
        for (int col = 0; col < GRID_SIZE; col++) {
            if (col % 3 == 0 && col != 0) {
                System.out.print("|");
            }
            System.out.print(board[row][col]);
        }
        System.out.println();
    }
}
}

```