

```

import java.util.ArrayList;
import java.util.List;

class NQueens {
    private int[] board;
    private int n;
    private List<List<String>> solutions;

    public List<List<String>> solveNQueens(int n) {
        this.n = n;
        board = new int[n];
        solutions = new ArrayList<>();
        solveNQueensUtil(0);
        return solutions;
    }

    private void solveNQueensUtil(int col) {
        if (col == n) {
            List<String> solution = new ArrayList<>();
            for (int i = 0; i < n; i++) {
                StringBuilder rowString = new StringBuilder();
                for (int j = 0; j < n; j++) {
                    if (board[i] == j) {
                        rowString.append("Q");
                    } else {
                        rowString.append(".");
                    }
                }
                solution.add(rowString.toString());
            }
            solutions.add(solution);
            return;
        }

        for (int row = 0; row < n; row++) {
            if (isSafe(row, col)) {
                board[col] = row;
                solveNQueensUtil(col + 1);
            }
        }
    }

    private boolean isSafe(int row, int col) {
        for (int i = 0; i < col; i++) {
            if (board[i] == row || Math.abs(board[i] - row) == col - i) {
                return false;
            }
        }
        return true;
    }
}

```

```

public static void main(String[] args) {
    int n = 4;
    NQueens solver = new NQueens();
    List<List<String>> solutions = solver.solveNQueens(n);
    System.out.println("Number of solutions for " + n + "-Queens problem: " +
        solutions.size());
    for (List<String> solution : solutions) {
        System.out.println("Solution:");
        for (String row : solution) {
            System.out.println(row);
        }
        System.out.println();
    }
}

```

Output

Solution:

```

.Q..
...Q
Q...
..Q.

```

Solution:

```

..Q.
Q...
...Q
.Q..

```