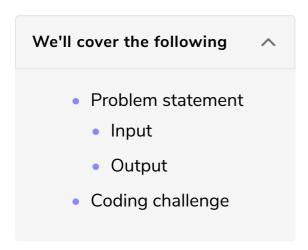
Challenge: Place N Queens on an NxN Chessboard

In this lesson, you will be challenged with a classic recursion problem, placing N queens on an NxN chessboard.



Problem statement

You are given an NxN chessboard, and you are required to place N queens on this chessboard such that no queen is under threat from any other queen.

In chess a queen can move any number of steps horizontally, vertically, or diagonally.

This means that no queen should share a row, column, or diagonal with another queen.

Input

As input, your function will take a number n and a list of strings as board.

Output

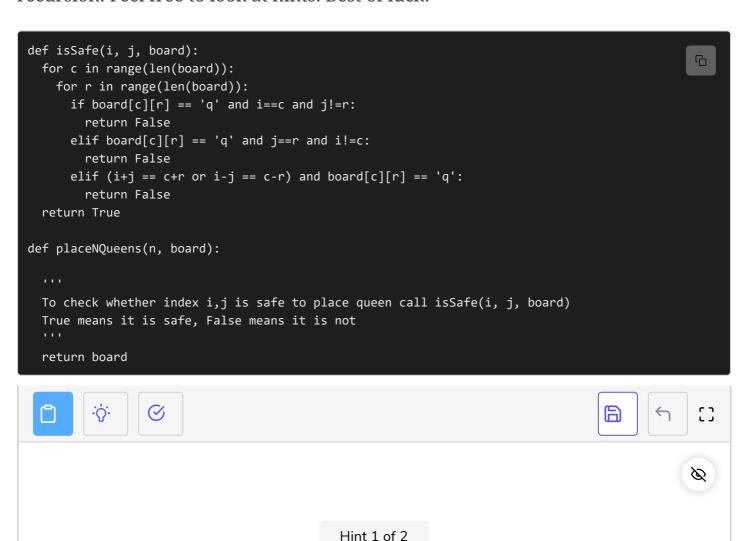
denoted by q shares a row, column, or diagonal with another queen.

Coding challenge

You can use the isSafe(i, j, board) function to test whether the position given by i^{th} row and j^{th} column is safe to place a queen. This function basically checks whether the box position given by row i and column j shares row, column, or diagonal with any other queen you had already placed on the board.

Do not change the prototype of either function as these are being used for testing. You can make your own helper functions though.

This one is a little trickier, but at the same time, it unleashes the real power of recursion. Feel free to look at hints. Best of luck!



Start from first row, keep placing queen on each row one by one.

We will review the solution to this challenge in the next lesson.