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.

We'll cover the following

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`.

```
n = 4
board = ["----",
         "----",
         "----",
         "----"
         ]
```

Output

Your function should return the updated list of strings, `board`, such that no queen denoted by `q` shares a row, column, or diagonal with another queen.

```
placeNQueens(n, board) =
       ["-q--",
        "---q",
        "q---",
        "--q-"
        ]
```

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!

```
def isSafe(i, j, board):
  for c in range(len(board)):
    for r in range(len(board)):
      if board[c][r] == 'q' and i==c and j!=r:
        return False
      elif board[c][r] == 'q' and j==r and i!=c:
        return False
      elif (i+j == c+r or i-j == c-r) and board[c][r] == 'q':
        return False
  return True

def placeNQueens(n, board):

  '''
  To check whether index i,j is safe to place queen call isSafe(i, j, board)
  True means it is safe, False means it is not
  '''
  return board
```

Hint 1 of 2

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.