Practical 3

Q1) Demonstrate N Queens Problem and give a solution

Ans:

nqueen.py

```
"""
nqueen.py
Author: Jagrut Gala
Date: 24-07-2021
Practical: 3
Objective: Demonstrate N Queens Problem and give a solution
"""

global N
N = 8

def generateBoard(size: int) -> list:
    board= list()
    for i in range(size):
        l= []
        for j in range(size):
            l.append(0)
        board.append(l)
    return(board)

def printSolution(board):
    for i in range(N):
        for j in range(N):
            print (board[i][j],end = " ")
        print()

def isSafe(board, row, col):
    # Check this row on left side
    for i in range(col):
        if board[row][i] == 1:
            return False
    # Check upper diagonal on left side
    for i, j in zip(range(row, -1, -1),range(col, -1, -1)):
        if board[i][j] == 1:
            return False

    # Check lower diagonal on left side
    for i, j in zip(range(row, N, 1),range(col, -1, -1)):
        if board[i][j] == 1:
            return False
```

```
    return True

def solveNQUtil(board, col):
    if col >= N:
        return True
    for i in range(N):
        if isSafe(board, i, col):
            # Place this queen in board[i][col]
            board[i][col] = 1
            # recur to place rest of the queens
            if solveNQUtil(board, col + 1) == True:
                return True
            board[i][col] = 0
    return False

def solveNQ():
    board = generateBoard(8)

    if solveNQUtil(board, 0) == False:
        print ("Solution does not exist")
        return False

    printSolution(board)
    return True

# Driver Code
solveNQ()
```