N = 8 # Board size (8 for 8-Queens)

def print\_board(board):

for row in board:

print(" ".join("Q" if col else "." for col in row))

print()

def is\_safe(board, row, col):

# Check this column in previous rows

for i in range(row):

if board[i][col]:

return False

# Check upper-left diagonal

for i, j in zip(range(row - 1, -1, -1), range(col - 1, -1, -1)):

if board[i][j]:

return False

# Check upper-right diagonal

for i, j in zip(range(row - 1, -1, -1), range(col + 1, N)):

if board[i][j]:

return False

return True

def solve(board, row):

if row == N:

print\_board(board)

return True # Return True to stop after first solution

# return False to print all possible solutions

for col in range(N):

if is\_safe(board, row, col):

board[row][col] = 1

if solve(board, row + 1):

return True

board[row][col] = 0 # Backtrack

return False

# Initialize 8×8 chess board with 0s

board = [[0 for \_ in range(N)] for \_ in range(N)]

solve(board, 0)

