In [6]:
```
#Number of Quuens
print("Enter Number of Queens : ")
N = int(input())

#chessboard
#NxN matrix with all elements 0
board = [[0]*N  for _ in range (N)]

def is_attack(i,j):
    #checking if there is a queen in row or column
    for k in range(0,N):
        if board [i][k]==1 or board[k][j]==1:
            return True
    #checking if there is a queen in diagonal
    for k in range(0,N):
        for l in range(0,N):
            if (k+l==i+j) or (k-l==i-j):
                if board[k][l]==1:
                    return True
    return False

def N_queen(n):
    #if n = 0, solution found
    if n==0:
        return True
    for i in range(0,N):
        for j in range(0,N):
            '''checking if we can place a queen here or not
            queen will not be placed if place is being attacked or
            already occupied'''
            if(not(is_attack(i,j))) and (board[i][j]!=1):
                board[i][j]=1
                #recursion
                #whether we can put next queen with this arrangement or not
                if N_queen(n-1)==True:
                    return True
                board[i][j] = 0
    return False

N_queen(N)
for i in board:
    print(i)
```

```
Enter Number of Queens :
4
[0, 1, 0, 0]
[0, 0, 0, 1]
[1, 0, 0, 0]
[0, 0, 1, 0]
```

In [ ]:

In [ ]: