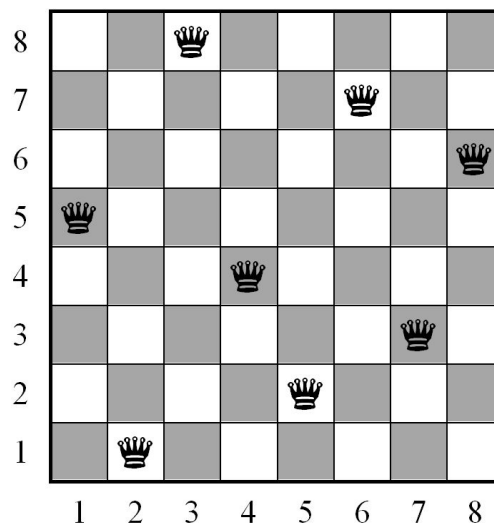


PROJECTS WEEK 3

PROJECT 07 : 8 Queens

Description :



The aim of the project is to place 8 queens on a chessboard without them being able to run into each other in a single move.

- A chessboard is composed of 8x8 squares.
- One square can contain one queen maximum.
- A queen can move in rows, column and diagonal.

The functions will return the number of possibilities of placing 8 queens on a chessboard without them being able to run into each other. Also will display all the possibilities found.

You should use recursivity to solve this problem.

You're program must give the answer in less than 2 seconds.

For the output you can display the chess boards as you want.

Python Bootcamp {2019}



Requirements :

- Program name : `07_queens.py`
- Directory : `week03/projects`
- Function name : `queens`

Hint :

- ❖ backtracking
- ❖ recursivity

EXAMPLE: =====

VALID BOARD :

. Q	[0][0][0][0][0][0][0][1]	00000001
. . Q	[0][0][1][0][0][0][0][0]	00100000
Q	[1][0][0][0][0][0][0][0]	10000000
. Q .	[0][0][0][0][0][1][0][0]	00000100
. Q	[0][1][0][0][0][0][0][0]	01000000
. Q .	[0][0][0][0][1][0][0][0]	00001000
. Q .	[0][0][0][0][0][0][1][0]	00000010
. . . Q . . .	[0][0][0][1][0][0][0][0]	00010000

INVALID BOARD :

. Q	[0][0][0][0][0][0][0][1]	00000001
. Q .	[0][0][0][0][0][1][0][0]	00000100
. . . Q . . .	[0][0][0][0][1][0][0][0]	00010000
. Q .	[0][0][0][0][0][0][1][0]	00000010
. Q .	[0][0][0][0][0][0][1][0]	00000010
. Q .	[0][0][0][0][1][0][0][0]	00001000
. . Q	[0][0][1][0][0][0][0][0]	00100000
. . . . Q . .	[0][0][0][0][1][0][0][0]	00001000

BONUS: Filename \Rightarrow `07_queens_bonus.py` / Function name \Rightarrow `queen_bonus` :
Your function will take an integer in parameters from 1 to 12 that represent the number of the queens and the size of the board (example: 12 \Rightarrow 12 queens on a 12x12 boards).