

WETHINKCODE\_

BOOTCAMP

GROUP PROJECT I

---

# Sudoku

---

Boot Camp 2021



## CONTENTS

1	Sudoku	2
2	Introduction	2
3	Objectives	3
4	Instructions	3
5	Mandatory	4
5.1	Level 1 . . . . .	4
5.2	Level 2 . . . . .	4
5.3	Level 3 . . . . .	4
5.4	Bonus . . . . .	5
6	Plagiarism Declaration	5

1	2						8	4
	3						7	
		4				6		
			2		3			
		5				9		
		6		9		5		
	7						2	
				5				

Figure 1: Sudoku Puzzle Example

## 1 SUDOKU

Congratulations! You have made it to the final project for the WeThinkCode\_ Bootcamp.

Your next task is to take on a group project.

Your project is "Sudoku!", to find out more about this game or to play it a little bit for "research" purposes go to: [SudokuOnline.io](https://www.sudokuonline.io)

## 2 INTRODUCTION

Sudoku - pronounced soo-doe-koo - does not require general knowledge, linguistic ability or even mathematical skill. Dubbed the Rubik's Cube of the 21st century, it consists of a grid of 81 squares, divided into nine blocks of nine squares each. Some of the squares contain a figure. The goal is to fill in the empty squares so that the figures 1 to 9 appear just once in every row, column and individual block. The requirement is logic or, for those willing to engage in a fiendish game of trial and error, sheer patience.

The Sudoku story began in 1783 when Leonhard Euler, a Swiss mathematician, devised 'Latin Squares', which he described as 'a new kind of magic squares'. Euler had come up with a grid in which every number or symbol appears once in each row or column. More than two centuries later, the difference for Sudoku players is that the grid is subdivided into blocks of nine. [Source: The Guardian](#)

### 3 OBJECTIVES

The idea is to get you to become familiar with the power of Python and its ability to do complex calculations over different data types and data structures.

Therefore, we wish to remind you, this is **NOT a MATH Problem** but rather a Problem Solving question.

### 4 INSTRUCTIONS

- You are expected to design a function or multiple functions that will solve a Sudoku Puzzle.
- Your function(s) will take an array as input, and return the solution.
- The table shown in this earlier section would be presented as follows:  

```
[0, 0, 0 0, 0, 0 0, 0, 0],
[1, 2, 0 0, 0, 0 0, 8, 4],
[0, 3, 0 0, 0, 0 0, 7, 0],
[0, 0, 4 0, 0, 0 6, 0, 0],
[0, 0, 0 2, 0, 3 0, 0, 0],
[0, 0, 5 0, 0, 0 9, 0, 0],
[0, 0, 6 0, 9, 0 5, 0, 0],
[0, 7, 0 0, 0, 0 0, 0, 0],
[0, 0, 0 0, 5, 0 0, 0, 0]
```

the example is contained inside 'sample.txt' included in the zip folder.

- The "-1" represents a blank space/square. This represents the number you must solve for.
- It is not a coincidence that there is an empty line or spaces after every three (3) numbers or arrays. It is there to help you solve the puzzle.
- You are given the function 'solve\_sudoku', instantiated as:  

```
"""python def sudoku(puzzle): #Your Code Below Here
#Your Code Above Here """
```
- This function is the only function we will call to solve the puzzle.
- Your programme must be able to take input from the command line.




Good Luck!

## 5 MANDATORY

You will have to answer the questions according to levels. You need to complete one level before proceeding to the next level. Each level is designed to test concepts you have learnt throughout the Bootcamp.

### 5.1 Level 1

In order to complete this level you will need to complete the tasks listed below:

1. Read an unsolved puzzle from file 
2. Print a Sudoku Grid from the contents d from file
3. Allow a user to substitute a  value with a number of there own choosing in the puzzle.
4. A user must be able to change a value of a tile by entering a **row & column combination**. The column 0,0 should result in the top left corner being changed. 0,1 will be the tile to its immediate right and 1,0 will be the tile immediately below the top left corner tile.

### 5.2 Level 2


In order to complete **this level**, you must retain all the functionality of Level 1 and add this feature:

1. When the grid has **been** filled with candidate values, the console must output whether the solution is "Correct/Valid" or whether it is "Incorrect/Invalid".
2. The **console** must offer the user a chance to play again.

### 5.3 **Level 3**

In order to complete the third level, you must alter programme in the following ways:

1. There must be a console that asks the User if they wish to play a game (Level 2) or if they want to solve a **puzzle**.

2. When a user selects solve a puzzle, they must be allowed to type in a filename '.txt' (assume it is in the same folder) which will read in an unsolved puzzle from .
3. Your program must automatically solve the puzzle and output the solution to the screen.

#### 5.4 Bonus

Do *NOT* attempt a Bonus unless all three (3) levels have been completed fully.

- Change the colours of the grid and its contents
- Change the output of Level 3 so the user can write to a file (if they wish to do so)
- Design your own Bonus and present it to your evaluators

## 6 PLAGIARISM DECLARATION

Engaging in any cheating or dishonesty in any form of assessment, assignment, test or examination or other WeThinkCode\_ prescribed work is considered cheating and is grounds for disciplinary action. Plagiarism, which is to present work (or a portion of work) as your own when it is not, is considered cheating and is not accepted at WeThinkCode\_.

An evaluator can flag one for plagiarism on one of the following grounds :

- The evaluator (marker) identifies that the student does not understand all or part of the work they have submitted.
- If all or part of the work presented is plagiarised ,i.e. copied from another source without reference.

### Cheating in group projects

The main purpose for a group project is to give students the experience of working in a team, by coming up with a solution to a problem together.

- Each member must be able to show which portion of the project they worked on.
- Failure to do so will result in the student being flagged for cheating which will be grounds for disciplinary action.

- This is to avoid single members doing the majority of the group project at the benefit of a member who is not contributing.
- In this way we are able to ensure fair assessment of each WTC\_student's competence.

Group projects can be approached in two ways.

1. Divide and conquer: This is usually preferred and advised when working on big projects. The project is divided into segments, in which each member of the group can accomplish. Once completed, the group will then integrate the segments to complete the project
2. One for all: This method is usually preferred and advised when a group is working on a small project. The group will work on the solution together from the start of the project until the end. This will require the members to move at a pace in which everyone in the team can keep up with.

NOTE: At the end of each group project, each member should have a general and basic understanding of the project and the solution found. This will include running, testing and explaining the solutions of the project.

## DECLARATION

I hereby declare that the work submitted by me and/or my group members is:

- Original (not plagiarised)
- References listed
- Honest & in Good Faith
- Subject to WeThinkCode\_policies

---

(YOUR NAME)

*WeThinkCode\_ Bootcamper*