How to generate sudoku game boards by Kristian Colville

Firstly, let me address that I am not a mathematician. I simply have played the game so much that I was bound to have noticed patterns. To help myself create pseudorandom boards I needed to just play sudoku.

Yes, I said pseudorandom. What that means is not truly random. I googled randomness and discovered that I couldn't create a random board with my local machine because eventually, the random numbers would repeat themselves because of the hardware limitations and the seed values given to a computer.

However, I could still create something cool. I began playing sudoku quite quickly after settling on it as a project idea. Here is what I noticed:

1. When you select a number on the board usually whichever game you play on an app the board highlights all the numbers within one of the nine larger grids containing numbers one to nine.
2. It also highlights all the numbers horizontally and vertically for that number you select.

Given this information, I imagined if I could create a board based on this logic. What I found is yes, it is possible.

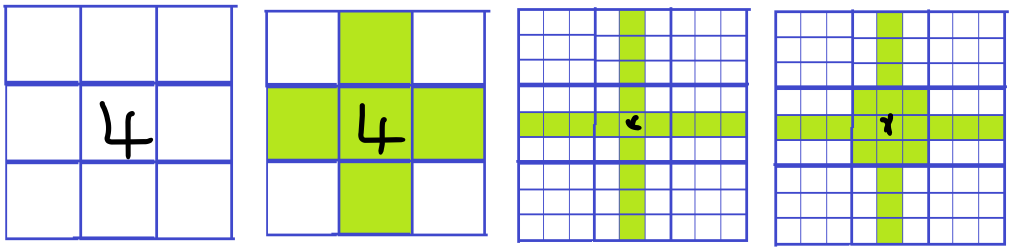
1. Create a variable called **firstValue** and assign it a random number between 1 and 9.
2. Create a variable called **firstPosition** and assign it a random number between 1 and 81.
3. Place **firstValue at the position of firstPosition.**

This is where the fun begins, all that is needed is one random number on the board. Once that is determined you can use the first two things that I noticed to generate an entire board.

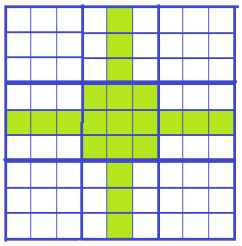
Please don't be fooled, my partner could attest to the number of hours I sat with a painful expression on my face trying to think of how to create this game all by myself. I only began to truly put two and two together when I kept playing the game. Reinventing the wheel is a silly notion and reinventing sudoku is that same concept multiplied by a factor of 81.

My design for the game is relatively straightforward from a high-level perspective, create a number and then create a random position to place that number. I promise you that at a low level it is more complicated. The low level would be the code used.

Once we have placed a number between 1 & 9 on the board, we need the ability for the JavaScript to be able to see all positions within the larger cube containing nine cubes. We also need to give it the ability to know the positions across vertically and horizontally.



The easy fix



Regardless of the first position, it's the same logic all the way around no matter which position is chosen. This helps make the logic for the game only slightly simpler.

In order to build the game, we have to solve the sudoku board. To me, that is very ironic. I have built a few games already and I can tell you I have not seen that same logic yet.

If you google computer science enough you might eventually stumble upon the term critical thinking; A scientific approach is needed here to solve this puzzle.

What do we know so far?

1. We have 1 number on the board and 80 left to figure out.
2. To build the game we have to solve the board.
3. We don't actually need maths to build it surprisingly.
4. We just need sound logic and careful planning.

What options do we have?

1. Data structures.

I put one option there because that's literally all we have. It's all we need.

You get what you ask for

If only that was true in life. However, if I give my computer instructions it will carry out those instructions exactly.

“Computer jump” = does not compute.

You can’t ask a computer with your voice to do specifically what you tell it to do (at least not yet).

You can give a computer or more specifically the program used to write your program a set of instructions to carry out. That program is called an ‘integrated development environment’ or IDE for short.

Using the IDE we can create a script in the JavaScript programming language.