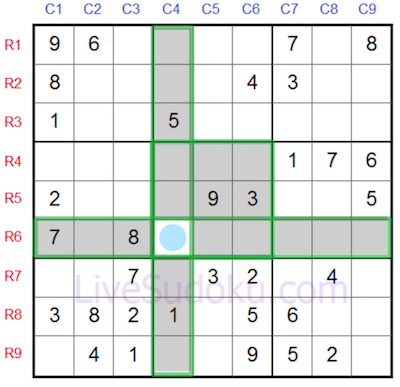
Suduko Game Programmieren

# JavaScript Kurs- Anfänger



# Abdurrahim Tawil-Basel Nabelsi

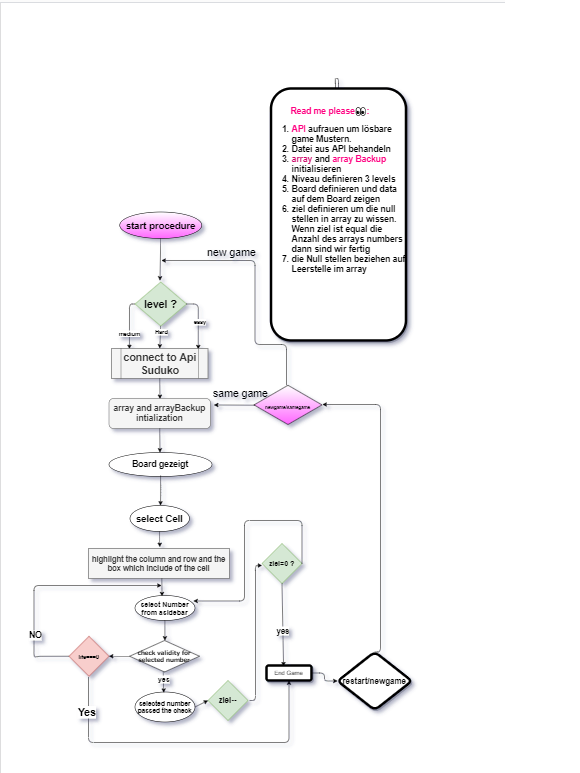


# Plan:

1. Paradigma haben(wie soll es laufen )
2. Data und Information Sammeln(Wie ---!!!? Was benötigt…?
3. Wichtige Punkte Definieren und die wichtigste variablen definieren die man Anfang vorstellt
4. Problem Lösungen vorstellen Flussdiagram erstellen und folgen
5. Ausprobieren
6. Fehler suchen und debuggen
7. Congrat! Du hast viel Fehler Denk nochmal!! ☺

# Lösung Flussdiagramm:





|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| C00 | C01 |  |  |  |  |  |  |  |
| C10 | C11 |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
| C08 |  |  |  |  |  |  |  |  |

Musters holen aus dem API

easy

Medium

Hard

API

showTable

getArrayFromuRL

Start

# Spiel Logik :

1. Jede Reihe (row) enthält eine der Ziffern 1 bis 9 mehr als einmal
2. Jede Spalte(Collum) enthält eine der Ziffern 1 bis 9 mehr als einmal
3. Jede 3\*3 Quadrat (box)enthält eine der Ziffern 1 bis 9 mehr als einmal

# Handlung Prozess :

1-Check Zeile: \_\_\_\_\_\_\_\_

Um zu wissen, dass der Ausgewählte Wert in der Zeile nicht wiederholt wird, müssen wir zuerst die Zeile kennen, wo wir gedrückt haben, und dann den ausgewählten Wert in dieser Zeile überprüfen

2-Check Spalte : |

Um zu wissen, dass der Ausgewählte Wert in der Spalte nicht wiederholt wird, müssen wir zuerst die Spalte kennen, wo wir gedrückt haben, und dann den ausgewählten Wert in dieser Spalte überprüfen

3-Check Validation 3\*3 Quadrate:

Sie müssen auch wissen, zu welcher Box die auf geklickte Zelle gehört und hinzugefügt werden soll

4-check Validation:

Überprüfen den ausgewählten Wert im gedrückten Kästchen[ 3\*3 ]

# Das Spiel besteht aus 4 Main logisch:

addNum (keyboard-asides button)

**Help-functions:**

    function el(css){

        return document.querySelector(css);

    }

    function con(parameter){

        console.log(parameter);

    }

    function getElements(css){

        return document.querySelectorAll(css);

    };

    function cont(t){

        return console.table(t);

    }

* function el(css):css selector
* function con(parameter):console.log
* Funktion getElements(css):um Group selektor von Elements zu haben
* Funktion cont(t) console.Tablle um Tabelle Ansicht zu haben

**Variablen Definieren:**

    let array = [[],[],[],[],[],[],[],[],[]];

    let arrayBackUp = [[],[],[],[],[],[],[],[],[]];

let arrayAddedNumber = [];

let newX =-1; //new x value which we get from

    let newY =-1; //new y value which we get from

    let lives = 3 ;

    let ziel = 0;

    let gameNow = false;

    let zeit =0;

//Reference wird als (row-Colum) gegeben:zbs:c01 row:0 colum:1     const bgColor="rgba(148, 156, 37, 0.911)";

    const nivaueHard ="hard";

    const nivaueMedium="medium";

    const nivaueEasy ="easy";

    const urlSudoku ="https://sugoku.herokuapp.com/board?difficulty=";

    let ziel = 0; wie viel 0 habe ich im Array die ich aus API bekam

# Events die das Spiel Kontrolieren:

el("#btnEasy").addEventListener("click", loadGame);

el("#btnMedium").addEventListener("click", loadGame);

el("#btnHard").addEventListener("click", loadGame);

el("#btnStart").addEventListener("click", startGame);

el("#btnStart").disabled = true;

el("#resetbtn").addEventListener("click", restGame);

el("#resetbtn").disabled = true;

document.addEventListener("keydown", getKey);

# Style Funktionen:

clearStyle(): bringt die Zellen auf den Standard style zurück

drawBorder(): zeichnet ein Rahmen zwischen den Zeilen

setStyleToSamevalues():setz die gleiche Style für die ausgewählte Zelle ein

setStyleToSelectedBox():setz die gleiche Style für die ausgewählte Box 3\*3 ein

# get-Funktionen:

getRow(): nimmt der Name der Zelle gibt Zeile Nummer zurück

getColumn():nimmt der Name der Zelle gibt Spalte Nummer zurück

getBoxNo(): Nimmt index der Zelle und gibt die Nummer der Quadrat wo die Zelle sich befindet

getNumberOfBtnId(): gibt den Wert der Taste um die hinzufügen

getKey(e): um die Keycode zu wissen

getNumberToadd(): nimmt die Zahl die hinzugefügt werden soll

# Set-Funktionen:

setGameOverOnBoard(): game over auf dem Brett einsetzen

setPause():Pause Wort auf dem Brett einsetzen und style Ablöschen und Zeit aufhalten.(mit ESCAPE taste)

# Check-Funktionen:

### chkNivaue(): gibt dir die Niveaus aus

### chkRowandColumn (): Überprüfen den Wert in der gleichen Spalte und der gleichen Zeile wenn es den existiert

### chkByBoxNo (): überprüfen den Wert ob es in dem box existiert

### isNumberDone ():überprüfe ob den gewählten wert 9-mal ausgewählt wurde

# Zeit Funktionen:

### startTimer (): bringt die Zellen auf den Standard style zurück

### showTime (): Zeit zeigen

# Add-Funktion:

### addNumber(): bringt die Zellen auf den Standard style zurück

# Remove –Delete –Funktionen:

### removeEventListenerSelectedNumber (): löscht den Eventlistener für 1-9 button in aside

### deleteNumber (): bringt die Zellen auf den Standard style zurück

# Initial-Funktionen

### iniT\_Array (): Array Initialisieren mit 0 werten.

### init\_cell\_of\_div (): fügt ein Eventlistener und löschen für alle Zelle ein

iniT\_SelectedNumber**()**: bringt die Zellen auf den Standard style zurück

# Game-Funktionen:

loadGame **()**: bringt die Zellen auf den Standard style zurück

startGame **():** bringt die Zellen auf den Standard style zurück

restGame **()**: bringt die Zellen auf den Standard style zurück

# show funktion:

UnEditableNumbershow **()**: um die unüberbietbaren Zahlen zu sehen

Undo **()**: um Rueckzug zu machen

Show-table **()**: um das Board anzuzeigen

# Main- Funktion: ☺

ini\_Main **()**: bringt die Zellen auf den Standard style zurück

# Spiel Anweisungen:

Erinnerung:

* jede Reihe enthält eine der Ziffern 1 bis 9 mehr als einmal
* jede Spalte enthält eine der Ziffern 1 bis 9 mehr als einmal
* jede 3\*3 Quadrat enthält eine der Ziffern 1 bis 9 mehr als einmal

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

1. drück mal auf Easy/Hard/Medium um die Muster aus dem API zu holen
2. drück mal auf Start(sudoku-spielen) um das Spiel anzufangen
3. drück mal auf Stop(stop- sudoku) um das Spiel zu pausieren
4. 3 Fehler wirst du rausgeschmissen
5. drück mal auf zurücksetzen um das selbe Spiel vom Anfang an zu beginnen
6. klick auf dem Aside (1-9) um Nummer zu wählen
7. doubleklicken auf dem ausgewählte Nummer in der Zelle um die Nummer zu Loeschen

## Kontrolle-Tasten:

* „Leertaste“ zeigt den organale Nummer die nicht bearbeitbar sind.
* „entfernen“ entfernt die ausgewählte Zahl in der Zelle.
* „ctr+z“ Rückzug .
* „doubleclick“ Zellenmarkierung entfernen.
* „Zahlen-in Tastatur“: sind alle beim drücken aktiv.
* „Escape Taste“ : Spiel pausieren
* „Enter Taste“: Spiel weiter spielen nach der Pause.

So sieht es aus: viel Spaß dabei

