MineSweeper

JavaScript,html,SASS

Anan Aghbar

Contents

Table of figures

File Structure:

- Js:
  - MineSweeper.js
  - Board.js
  - Cell.js
- Css
  - MineSweeper.sass
  - The compiled css file
- Index.html

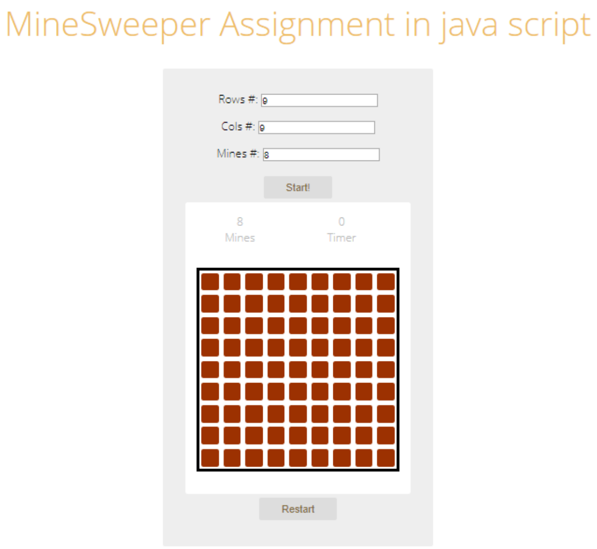
Start Game:

To build a new game the following objects and methods are used:

- New instance from the MineSweeper object in the index.html page
- MineSweeper object has the following properties and functions:
  - Properties:
    - components:
      - board:div.board
      - col:input.col
      - mine:div.mines-count
      - mines:input.mines
      - restartButton:button.restart-button
      - row:input.row
      - startButton:button.start
      - time:div.time
      -
    - dimension
    - initialize
    - isMineSweeperOver
    - leftMineCount
    - mineCount
    - time:
  - Functions:
    - MineSweeperIsover:*ƒ (isWin):to check if the game is over or not.*
    - decrementLeftMineCount:*ƒ (): to decrement mine counter by 1.*
    - increaseLeftMineCount:*ƒ ():* to increment mine counter by 1
    - eventsHandlers:*ƒ (): to assign the events to their handlers functions*
    - findCellByEvent:*ƒ (event): return Cell object of the clicked cell.*
    - generateGame:*ƒ (rows,cols,mines): assign the game dimensions and mines*
    - init:*ƒ (): Initialize the MineSweeper object's elements and properties*
    - isWin:*ƒ (): check the player is win or not*
    - leftClickHandler:*ƒ (event): to process the clicked cell*
    - restartClickHandler:*ƒ (): to rest the game*
    - rightClickHandler:*ƒ (event): to flag/unflag the suspected mine*
    - startClickHandler:*ƒ (): to start new game button*
    - startTimer:*ƒ (): enable the timer*
    - stopTimer:*ƒ (): disable the timer*
    - Board object.

- Board object has the following properties and functions:
    - Properties:
        - cells:*Array Cells objects (game cells*)
        - cols.
        - rows.
        - dimension
        - element:div.board
        - mineCount.
    - Functions:
        - appendClearfixElement:ƒ *(): to append clear fix element between each column and row.*
        - calculate:ƒ *(): to calculate the number of the siblings for each Cell.*
        - draw:ƒ *(): to draw the cells and append the cells elements*
        - getCellSibllings:ƒ *(cell): to get the siblings of the cell*
        - getNotOpenedCells:ƒ *(): to get the unopen cells*
        - getRandomNumbersArr:ƒ *(maxRow): to get an array of random numbers*
        - init:ƒ *(rows,cols,dimension, mineCount): to initialize the board object*
        - open:ƒ *(): to open the board's cells*
        - plantMines:ƒ *(): to distribute the mines on the board's cells randomly.*
- Cell object has the following properties and functions:
    - Properties:
        - element:span.cell
        - isEmpty.
        - isFlagged.
        - isMine.
        - isOpened.
        - mineCount:0.
        - x.
        - y.
    - Functions:
        - open:ƒ *(): open the clicked cell*
        - setEmpty:ƒ *(): set the clicked cell as empty (has no mine)*
        - setFlagged:ƒ *(): set the clicked cell as suspected mine*
        - setMine:ƒ *(): set a mine on the current cell*
        - setMineCount:ƒ *(number): set the mine count*
        - setOpened:ƒ *(): set the open flag.*

Game Screen Shots:d



MineSweeper Assignment in java script

Rows #: 9

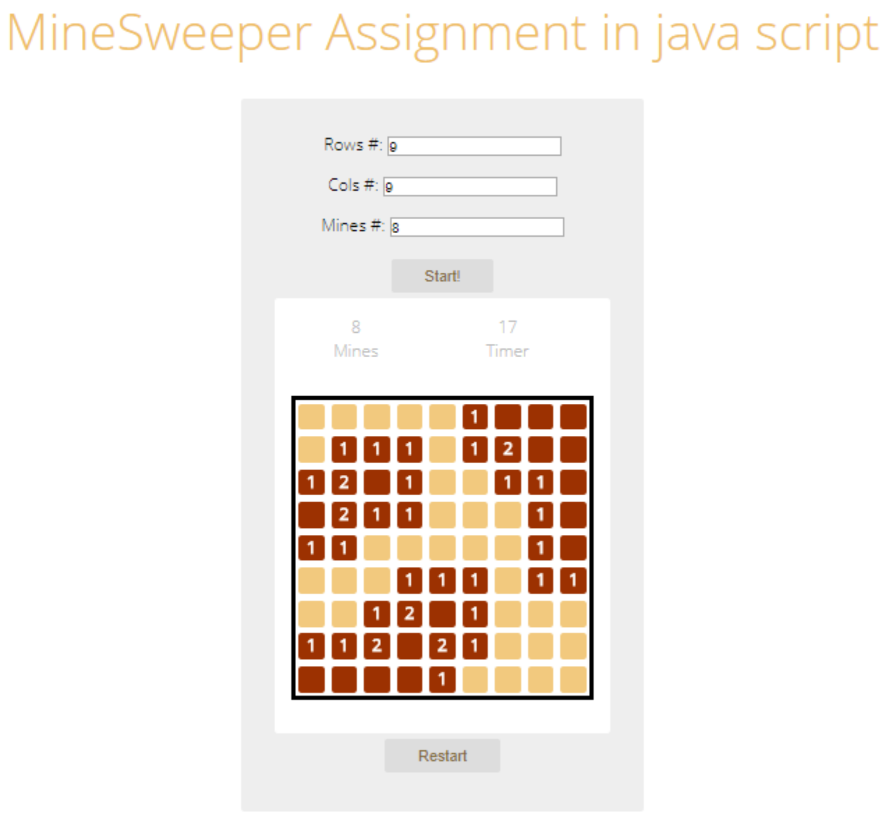
Cols #: 9

Mines #: 8

8 Mines

0 Timer

Figure 1: New Game



MineSweeper Assignment in java script

Rows #: 9

Cols #: 9

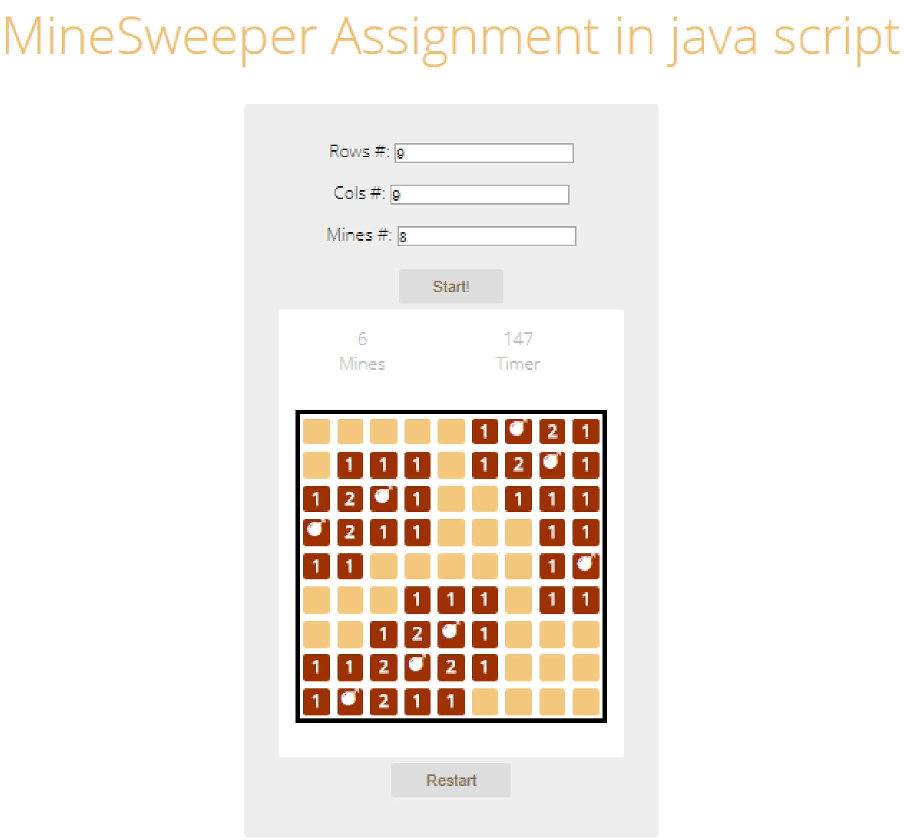
Mines #: 8

8 Mines

17 Timer

					1	■	■	■
	1	1	1		1	2	■	■
1	2	■	1			1	1	■
■	2	1	1				1	■
1	1						1	■
			1	1	1		1	1
		1	2	■	1			
1	1	2	■	2	1			
■	■	■	■	1				

Figure 2: During the game



MineSweeper Assignment in java script

Rows #: 9

Cols #: 9

Mines #: 8

6
Mines

147
Timer

					1	●	2	1
	1	1	1		1	2	●	1
1	2	●	1			1	1	1
●	2	1	1				1	1
1	1						1	●
			1	1	1		1	1
		1	2	●	1			
1	1	2	●	2	1			
1	●	2	1	1				

Figure 3: Game Over