MS_DOS Connect-4 (CON)

General Requirements & Info

The CreateGamePopup function that is called inside the class constructor expects a JSON object of the basic html structure of the game including classes and id's. This means that any html that is not subject to (constant) structural changes (like the start or gameover screen) has to be added to the JSON structure.

Start screen which will be shown after the onload function has finished executing. This screen shows the name of the game and a prompt to press a button to start. Rules of the game can be shown here as well.

Loading screen which will be shown when the user presses the button prompt on the start screen. This screen will count from 3 to 1 before allowing the game to start.

Gameover screen which will be shown when the game is over, showing some info (win/lose, score etc.). this screen will remain until you click spacebar/enter.

Do not use duplicate functions to do the same thing for different players/AI's (Player1Move, Player2Move etc. becomes PlayerMove(number)).

All functions, classes, id's and class/object variables have the 3 symbol code (found at the top of this document) in front of them, to prevent other games from interfering with this game due to using the same name(onload, onkeydown, onkeyup and onclick are exceptions).

Variables that have to be available to multiple functions or contain game related settings (speed, score etc.) should be made inside the onload function. All variables in the onload are made using the this. syntax instead of using the var, let or const keywords.

Make sure that you stop the game intervals & timeouts from running when the game is closed/the gameover screen is shown to prevent multiple instances of the same intervals & timeouts from running.

Do not focus on the visual/css until the game is properly working.

Connect-4 Mechanics & Info

2 player game

The game has a 7x6 grid.

The starting player is randomly selected.

Players take turns dropping a piece in 1 of the 7 columns.

When a piece is dropped in a column, it will move to the lowest possible position in that column.

Pieces cannot be placed in a column that is full.

The game is over when a player gets a horizontal, vertical or diagonal line of 4 pieces.

Advanced Mechanics & Info

Add an AI, which can be used instead of a player.

The AI follows a set of rules in order:
1 - if it can win in the current turn it will always do so.
2 - if the player has 2 pieces in a line that can go both ways, or 3 pieces in a line that can only go 1 way it will try to break that line.
3 - if the AI has 2 or more pieces in a line it will try to finish that line.
4 - if there is an existing piece on the grid and the piece can still form a line of 4, it will try to create that line.
5 - if none of the above is true it will place a piece at random.