**Chess Board Problem**

Let’s create a chessboard and a movable tower using the following HTML

<html lang="en">

<head>

    <meta charset="UTF-8">

    <style>

        table {

            border-collapse: collapse;

        }

        td {

            width: 80px;

            height: 80px;

            border: 1px solid black;

            text-align: center;

            font-size: 50px;

        }

        tr:nth-child(even) td:nth-child(odd), tr:nth-child(odd) td:nth-child(even) {

            background-color: grey;

        }

    </style>

</head>

<body>

    <table>

        <tbody></tbody>

    </table>

    <script>

        const table = document.querySelector('table tbody');

        for (let i = 0; i < 8; i++) {

            const tr = document.createElement('tr');

            for (let j = 0; j < 8; j++) {

                const td = document.createElement('td');

                td.addEventListener('click', placeTower);

                tr.appendChild(td);

            }

            table.appendChild(tr);

        }

        const tower = document.createTextNode('ß');

        table.querySelector('tr:first-child td:first-child').appendChild(tower);

        function placeTower() {

            const x\_old = tower.parentNode.cellIndex;

            const y\_old = tower.parentNode.parentNode.rowIndex;

            const x\_new = this.cellIndex;

            const y\_new = this.parentNode.rowIndex;

            if(x\_old == x\_new || y\_old == y\_new) {

                this.appendChild(tower);

            } else {

                alert('Wrong move');

            }

        }

    </script>

</body>

</html>

**XY Problem**

Integrate the described functionality using JavaScript and the DOM API:

1. The user shall be able to select a grey square by typing in the x and y number.
2. If the user click the fill button, the text from the input box shall be insert into the selected square. Also the CSS properties shall be set.
3. If the user clicks on a filled square, the text and the CSS propertys should be removed from this field.
4. If the user select a non existing square a new column / row should be added to the DOM until the square can be found.

<!DOCTYPE html>

<html>

<head>

    <meta charset="UTF-8" />

    <style>

        \* {

            font: 14px normal Arial, sans-serif;

            color: #000000;

        }

        table {

            margin: 50px auto;

        }

        table,

        td {

            border: 1px solid #aaa;

            border-collapse: collapse;

        }

        th {

            padding: 10px;

            font-weight: bold;

        }

        td {

            background-color: #eeeeee;

            width: 80px;

            height: 80px;

        }

        table:first-child tr td {

            cursor: pointer;

        }

        td[colspan="4"] {

            text-align: center;

        }

    </style>

    <script>

        //running the code after the DOM is constructed

        window.onload = function(){

            //adding addEventListener to the fill button to call onFillClick() function on

"click"

            document.getElementById('mybutton').addEventListener("click", onFillClick);

        }

        function onFillClick() {

            //assigning field\_x and field\_y to xField and yField, respectively

            var xField = document.getElementById("field\_x").value;

            var yField = document.getElementById("field\_y").value;

//get the first tBody

            var firstTableBody = document.getElementsByTagName('tbody')[0];

//get the number of rows in the table

            var tRowsInTable = firstTableBody.rows.length;

//get the number of column in the table

            var tColumnInTable = firstTableBody.rows[0].cells.length;

//get the number of iterations for the loop in the expandTableColumn function

            var iterationsForColumn = ((yField)- (firstTableBody.rows[0].cells.length))\*(fi

rstTableBody.rows.length);

//get the number of iterations for the loop in the expandTableRow function

            var iterationsForRows = xField - (firstTableBody.rows.length);

            console.log(iterationsForRows);

//check if X and Y fields are filled and if filled should be with valid input

            if(xField == '' || yField == '' || xField < 1 || yField < 1){

                alert('Both Fields X and Y should be filled with values greater than 0');

            }

            //if both table rows and columns need to be expanded

            else if((xField > tRowsInTable) && (tColumnInTable < yField)) {

                expandTableRow(xField, yField, firstTableBody, iterationsForRows);

//get the number of iterations for the loop in the expandTableColumn function after expanding table with selected number of rows

                var iterationsForColumn = ((yField)- (firstTableBody.rows[0].cells.length))

\*(firstTableBody.rows.length);

                expandTableColumn(xField, yField, firstTableBody, iterationsForColumn);

//filling the selected square after expanding the table by calling the fillTable function

                fillTable(xField ,yField, firstTableBody);

            }

            //if only table row needs to be expanded

            else if(xField > tRowsInTable) {

                expandTableRow(xField, yField, firstTableBody, iterationsForRows);

//filling the selected square after expanding the table by calling the fill

                fillTable(xField ,yField, firstTableBody);

            }

//if only table column needs to be expanded

            else if(tColumnInTable < yField) {

                expandTableColumn(xField, yField, firstTableBody, iterationsForColumn);

//filling the selected square after expanding the table by calling the fill

                fillTable(xField ,yField, firstTableBody);

            }

//user selected an existing square

            else{

                fillTable(xField, yField, firstTableBody);

            }

        }

        function fillTable(xField, yField, firstTableBody) {

            //get the table row from Field X

            var tRow = firstTableBody.getElementsByTagName('tr')[xField - 1];

//get the table column in the selected row from Field y to get the selected square

            var tColumn = tRow.getElementsByTagName('td')[yField - 1];

            //assigning the text field value to the selected square

            tColumn.innerHTML = document.getElementById('text').value;

//assigning the CSS field value to the selected square

            tColumn.style = document.getElementById('css').value;

//to remove text and css on selected square click

            tColumn.addEventListener('click', function(){

                tColumn.innerHTML = ""; tColumn.style = "";

            })

        }

//function to expand table row

        function expandTableRow(xField, yField, firstTableBody, iterationsForRows){

            for (var i = 0; i < iterationsForRows; i++) {

//creating table row and adding it to the table body

                var tRow = document.createElement('tr'); firstTableBody.appendChild(tRow);

                for (var j = 0; j < firstTableBody.rows[0].cells.length; j++) {

//creating table column and adding it to the table row

                    var tColumn = document.createElement('td'); tRow.appendChild(tColumn);

                }

            }

        }

//function to expand table column

        function expandTableColumn(xField, yField, firstTableBody, iterationsForColumn){

//variable to use in array for adding columns to the rows

            var temp = 0;

var tRow = firstTableBody.getElementsByTagName('tr');

            for (var i = 0; i < iterationsForColumn; i++) {

                var tColumn = document.createElement('td');

                tRow[temp].appendChild(tColumn);

//increamenting temp and initiallizing it to zero if the temp is equal to the number of rows

                temp++;

                if (temp == firstTableBody.rows.length) {

                    temp = 0;

                }

            }

        }

    </script>

</head>

<body>

<table>

       <tbody>

            <tr>

                <td></td>

                <td></td>

                <td></td>

                <td></td>

            </tr>

            <tr>

                <td></td>

                <td></td>

                <td></td>

                <td></td>

            </tr>

            <tr>

                <td></td>

                <td></td>

                <td></td>

                <td></td>

            </tr>

            <tr>

                <td></td>

                <td></td>

                <td></td>

                <td></td>

            </tr>

        </tbody>

    </table>

    <table>

        <thead>

            <tr>

                <th colspan="4">Fill a field:</th>

            </tr>

        </thead>

        <tbody>

            <tr>

                <td>Text: <br /><input type="text" id="text" value=""></td>

                <td>Field X: <br /><input type="text" id="field\_x" value=""></td>

                <td>Field Y: <br /><input type="text" id="field\_y" value=""></td>

                <td>CSS: <br /><input type="text" id="css" value=""></td>

            </tr>

            <tr>

                <td colspan="4"><button id="mybutton">Fill</button></td>

            </tr>

        </tbody>

    </table>

</body>

</html>

**Box Color Problem**

Change box color from the color table

<!DOCTYPE html>

<html lang="de-DE">

<head>

    <meta charset="UTF-8" />

    <style>

        \* {

            font: 14px normal Arial, sans-serif;

            color: #000000;

        }

        table {

            margin: 50px auto;

        }

        table,

        td {

            border: 1px solid #aaa;

            border-collapse: collapse;

        }

        th {

            padding: 10px;

            font-weight: bold;

        }

        td {

            background-color: #eeeeee;

            width: 80px;

            height: 80px;

        }

        table tr td {

            cursor: pointer;

        }

        div#Log {

            position: absolute;

            top: 50px;

            left: 50%;

            width: 120px;

            height: 310px;

            margin-left: 180px;

            background-color: #eeeeee;

            border: 1px solid #aaa;

            padding: 10px;

            overflow: auto;

        }

    </style>

<script>

        var currentFillColor = '';

        var changeCounter = 0;

        function setField(element) {

            // element contains the current html element

            //Checking if the field was already colored with the selected color and changing its color back to gray

            if (element.style.backgroundColor == document.getElementById('Log').style.color) {

                element.style.backgroundColor = '#eeeeee';

            } else {

                //Coloring the clicked field with the selected color and incrementing the changeCounter variable

                element.style.backgroundColor = currentFillColor;

            }

        }

        function setFillColor(color) { // color should be a string

            currentFillColor = color;

            changeCounter++; //incrementing the changeCounter variable //Checking the changeCounter value and displaying an alert

if(changeCounter == 10){

                alert('You have clicked 10 times on the field!');

                changeCounter = 0; //re-initializing the value of changeCounter to zero

            }

//Checking the selected color and assigning it to the div#Log text color

            document.getElementById('Log').style.color = currentFillColor;

            //Notify the user in div#Log that color has been changed

            document.getElementById('Log').innerHTML += "<div style = 'color: "+currentFillColor+";'>Color changed</div>";

        }

</script>

</head>

<body>

    <div id="Log"></div>

    <table>

        <tbody>

            <tr>

                <td onclick="setField(this)"></td>

                <td onclick="setField(this)"></td>

                <td onclick="setField(this)"></td>

                <td onclick="setField(this)"></td>

            </tr>

            <tr>

                <td onclick="setField(this)"></td>

                <td onclick="setField(this)"></td>

                <td onclick="setField(this)"></td>

                <td onclick="setField(this)"></td>

            </tr>

            <tr>

                <td onclick="setField(this)"></td>

                <td onclick="setField(this)"></td>

                <td onclick="setField(this)"></td>

                <td onclick="setField(this)"></td>

            </tr>

            <tr>

                <td onclick="setField(this)"></td>

                <td onclick="setField(this)"></td>

                <td onclick="setField(this)"></td>

                <td onclick="setField(this)"></td>

            </tr>

        </tbody>

    </table>

    <table>

        <thead>

            <tr>

                <th colspan="4">Choose a color:</th>

            </tr>

        </thead>

        <tbody>

            <tr>

                <td onclick="setFillColor('#CC3333')" style="background color:#CC3333;"></td>

                <td onclick="setFillColor('#3399FF')" style="background-color:#3399FF;"></td>

                <td onclick="setFillColor('#339933')" style="background-color:#339933;"></td>

                <td onclick="setFillColor('#999933')" style="background-color:#999933;"></td>

            </tr>

        </tbody>

    </table>

</body>

</html>