**Creating a chessboard pattern with JavaScript and DOM**

A chessboard pattern can be created very easily using javascript and the concept of document object module (DOM). Using this method you can create a chessboard with any number of rows and columns as you desire by just tweaking few parameters. Also, the lines of code written using this method will also be way lesser than creating the same using pure HTML and CSS especially when the number of rows and columns is very large.

**Approach**: Create a nested for loop. Let us call the outer loop as "i" and the inner loop as "j".  The outer loop will be used to create rows and the inner loop will be used to create cells in each column. By doing so a N\*M cells will be created where N is the number of rows and M is the number of columns. The combination of i and j value in the inner loop can be used to distinguish between each cell so formed. At the end of the loop, we will have a table created. Also, we need to color the cells with an appropriate color. If the summation of i and j yields an even number then the cell has to be colored white else it has to be colored black. This will create cells of alternative colors of white and black as seen in a chessboard. Creation of table and table cells can use done using DOM and coloration of cells can be done using CSS.

HTML

<html>

<head>

<title>Chess board</title>

<style>

body{

text-align: center;

}

.cell{

height: 30px;

width: 30px;

border : 1.5px solid grey;

border-style: inset;

}

.blackcell{

background-color: black;

}

.whitecell{

background-color: white;

}

</style>

</head>

<body>

<span style="color:green; font-size:30px;" >GeeksforGeeks</span><br><br>

<script type="text/javascript">

// create a center tag to center all the elements

var center = document.createElement('center');

// create a table element

var ChessTable = document.createElement('table');

for (var i = 0; i < 8; i++){

// create a row

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

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

// create a cell

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

// if the sum of cell coordinates is even then color the cell white

if ((i + j) % 2 == 0){

// create a class attribute for all white cells

td.setAttribute('class', 'cell whitecell');

tr.appendChild(td);

}

// if the sum of cell coordinates is odd then color the cell black

else{

// create a class attribute for all black cells

td.setAttribute('class', 'cell blackcell');

// append the cell to its row

tr.appendChild(td);

}

}

// append the row

ChessTable.appendChild(tr);

}

center.appendChild(ChessTable);

// modifiying table attribute properties

ChessTable.setAttribute('cellspacing', '0');

ChessTable.setAttribute('width', '270px');

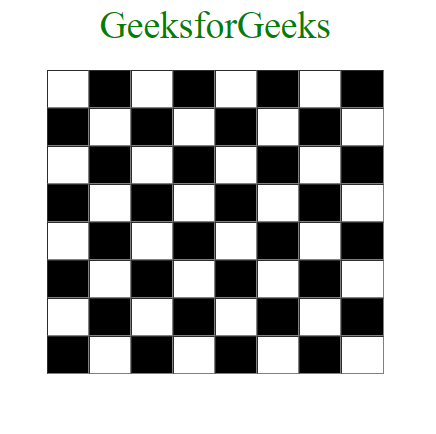
document.body.appendChild(center);

</script>

<br><br>

</body>

</html>

Output

**Code explanation:** An 8 x 8 chessboard will be created for the above code. However, just by modifying the termination condition of i and j, we will be able to create a N x M chessboard with ease. Using Javascript DOM a table element is created initially using createElement(). we know that the i loop is used to create rows, hence a row element will be created during each iteration. similarly, the j loop is responsible for creating cells. Hence table cells are created during each iteration. As discussed before the color of each cell can be decided by the summation of i and j values. If the sum is even then the cell has to be colored white and if it is odd then the cell has to be colored black. This is done by creating and assigning an appropriate class attribute to each cell using setAttribute() and assigning the right color, size, and other properties as you wish using CSS. At the end, all the elements are appended inside the body of the HTML document. Hence, we are able to create a simple chessboard pattern using javascript and DOM very easily.