**The HTML table basics: tags, attributes and CSS styling**

**Introduction**

The <table> element helps with rendering tables in an HTML document.

Each table row is defined with the <tr> tag (**T**able **R**ow). A table header is defined with the <th> tag (**T**able **H**eader). By default, table headings are bold and centered. A table data/cell is defined with the <td> tag (**T**able **D**ata). In each cell, you can have other HTML elements/tags. You can have only "column table headers" (the first row of the table will be in bold), or you can also have "row headers" (first cell of each row).

**Best practice for making the table accessible:** always add a <caption> tag inside the <table> tag. Data tables very often have brief descriptive text before or after the table that indicates the content of that table. This text should be associated to its respective table using the <caption> element. The <caption> element must be the first thing after the opening <table> tag.

**Second best practice for accessibility:** use a scope attribute with all <th scope = "row or column"> for identifying whether a table header is a column header or a row header.

* + You can read [these recommendations](https://www.w3.org/WAI/tutorials/tables/) for making accessible tables.
  + Link - <https://www.w3.org/WAI/tutorials/tables/>

**Typical example:**

1. <table>
2. **<caption>A typical HTML table</caption>**
3. <tr>
4. <th **scope="col"**>Given Name</th>
5. <th **scope="col"**>Family Name</th>
6. <th **scope="col"**>Age</th>
7. </tr>
8. <tr>
9. <td>Michel</td>
10. <td>Buffa</td>
11. <td>52</td>
12. </tr>
13. <tr>
14. <td>Dark</td>
15. <td>Vador</td>
16. <td>Unknown</td>
17. </tr>
18. <tr>
19. <td>Luke</td>
20. <td>Skywalker</td>
21. <td>Unknown</td>
22. </tr>
23. </table>

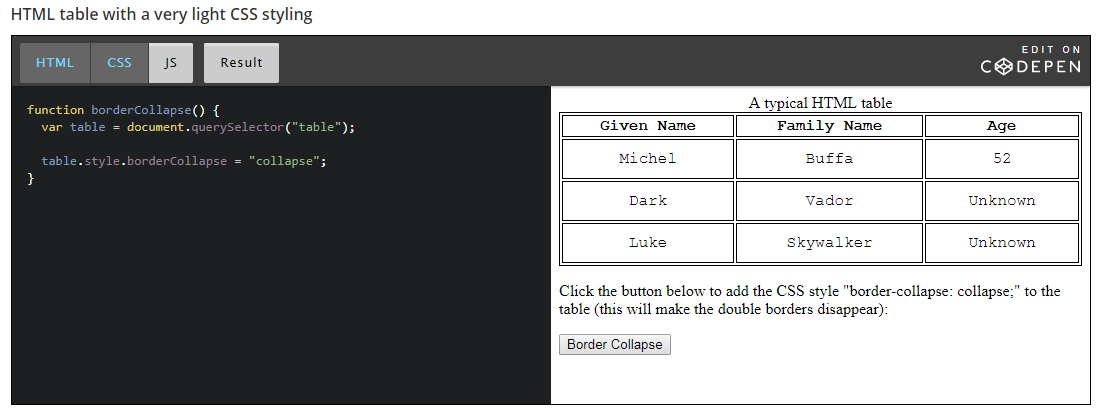
Most of the time, we add some CSS rules for rendering cell/row/table borders and for adjusting spacing between the text in the cells and the cell borders. Let's look at another example

#### HTML table with a very light CSS styling

This is a static table. You can look at the CSS code:

1. table {
2. width:100%;
3. border:1px solid;
4. }

7. tr, th, td {
8. border:1px solid;
9. font-family:courier;
10. }
12. td {
13. text-align:center;
14. padding:10px;
15. }

Javascript code -

**Explanations**:

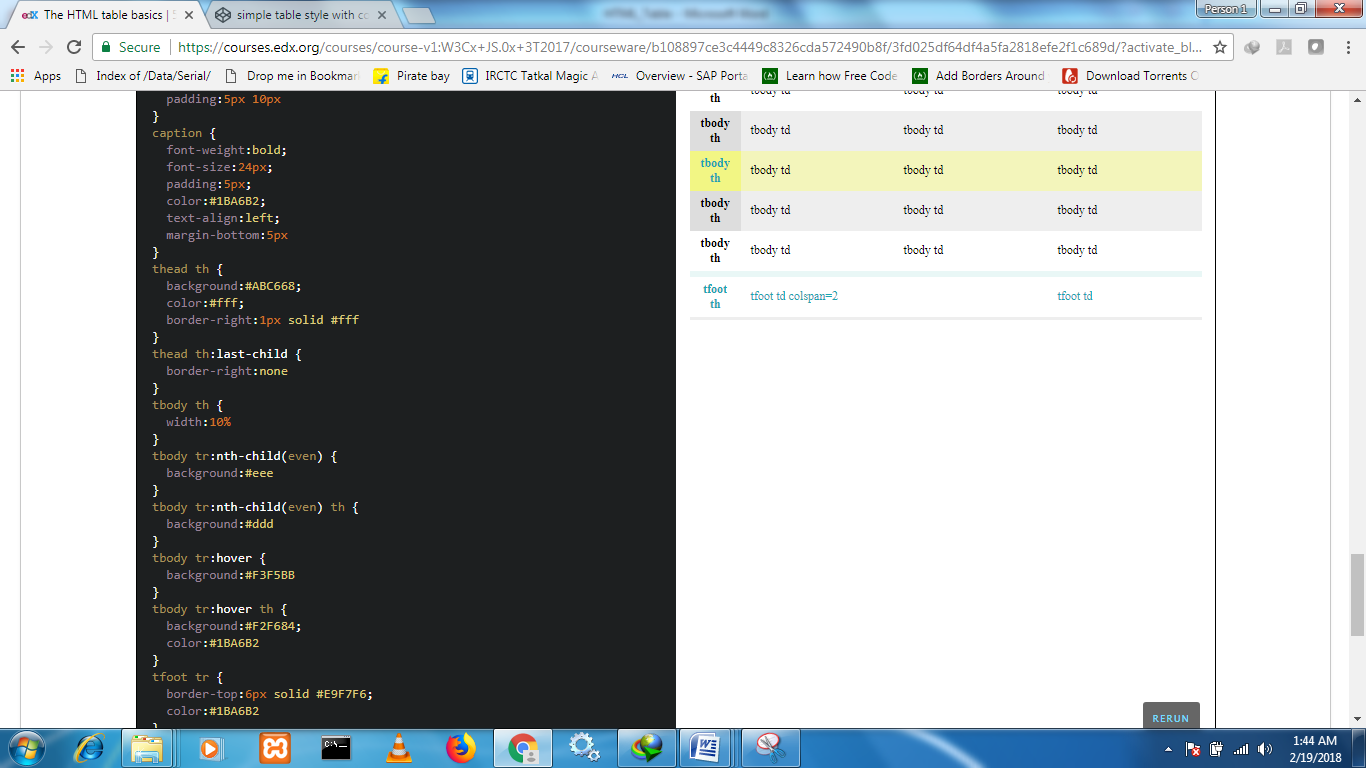
* *Line 1*: this rule says that the table will occupy the width of the window and will have a black, continuous border that is one pixel wide.
* *Line 7*: this rule says that table rows, table cells and table headers will also have a border and will use the font family Courier.
* *Line 12*: this says that all cells will have the text horizontally centered and an internal margin (called padding) of 10px in each direction (top, bottom, left, right).

#### Here is another example with more CSS styling (flat design)

#### 

#### And a final example with colored lines, header, footer, legend

#### 



**The HTML table JavaScript API: dynamic tables!**

**Introduction**

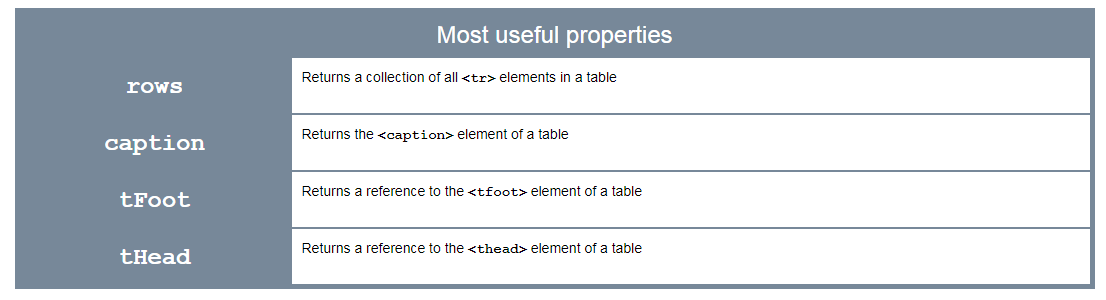
There is a JavaScript API associated with the HTML table elements that makes dynamic table management possible, enabling you to add or delete a row, add or delete a cell, modify the content of the cells, etc.

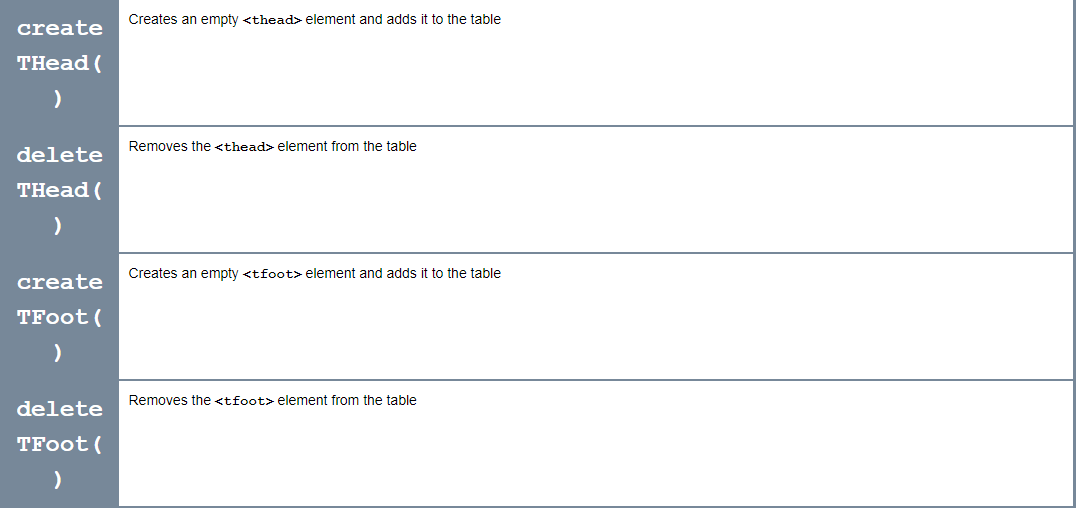
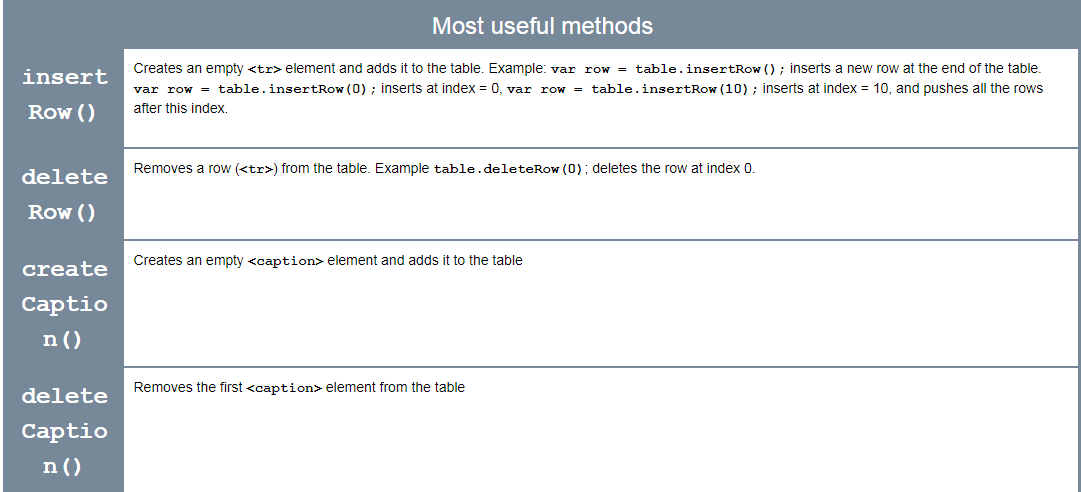
**The Table object (<table>)**

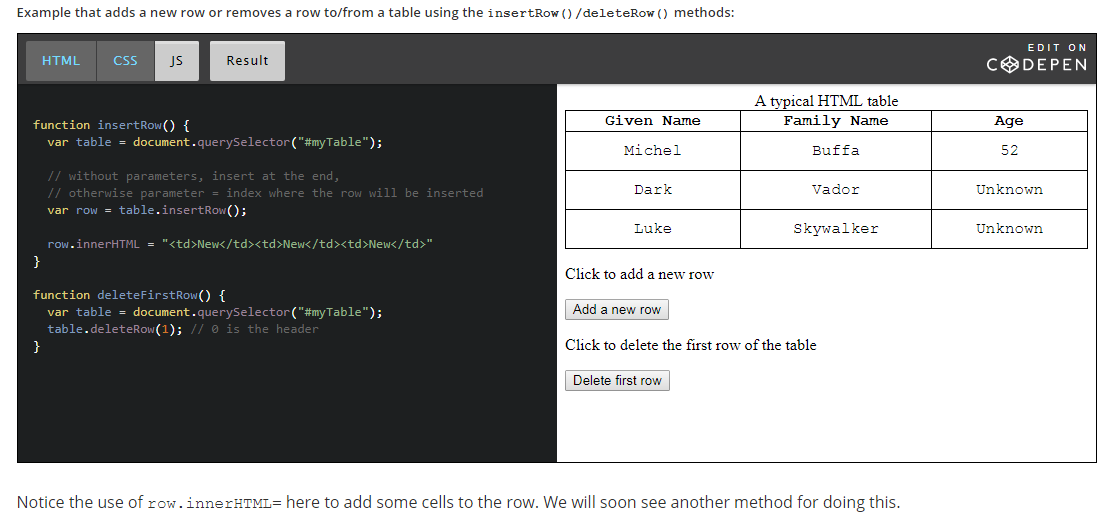
When you look for a table using the DOM API or the selector API, or when you create a table using the DOM API, you get a Table object:

1. var table = document.getElementById("myTable");
3. var table = document.querySelector("#myTable");
5. var table = document.createElement("table"); // creates a new table

Like all objects, an instance of Table will have properties and methods:







**The TableRow object (<tr>)**

When you look for a row using the DOM API or the selector API, or when you create a row using the DOM API, you get a Row object:

1. var row1 = document.getElementById("row1");
3. var row1 = document.querySelector("#row1");
5. var newRow = document.createElement("row"); // creates a new row

You can also access a row from the rows property of a table:

1. > var t = document.createElement("table");
2. undefined
4. > var r1 = t.insertRow(0);
5. undefined
7. > r1.innerHTML="<td>Hello</td>";
8. "<td>Hello</td>"
10. > var r2 = t.insertRow();
11. undefined
13. > r2.innerHTML="<td>Hello 2</td>";
14. "<td>Hello 2</td>"
16. > **var row1 = t.rows[0];**
17. undefined
19. > row1;
20. <tr><td>Hello</td></tr>

Like all objects, a TableRow object has properties and methods. Here are the most useful ones: 