

18/02/2021

HTML 5 – Partie 4

Les tableaux

HTML



MODULE B1.WEB

Introduction.....	2
Les BALISEs POUR LES TABLEAUX	2
structurer son tableau.....	3
Les styles sur tableaux.....	4
Les bordures et la couleur	4
Les sélecteurs hiérarchiques	5
A B	5
Les pseudos classes de selection.....	6
Autres propriétés CSS de mise en forme du tableau	7

INTRODUCTION

Remarque : attention à ne pas confondre les tableaux HTML qui servent à présentation des pages, avec les tableaux utilisés comme structure de données dans les langages de programmation (C, Java...)

En HTML5, les tableaux sont exclusivement réservés à la présentation de données. Ils ne doivent pas être utilisés pour réaliser la disposition de la page. Depuis HTML4, il faut cependant noter la disparition de très nombreux attributs

(width, border, cellpadding, cellspacing, frame et rules, align et valign) de présentation qui doivent maintenant être réalisés

avec des feuilles de style CSS.

valign



Obsolète depuis [HTML4.01](#) ,



Obsolète depuis [HTML5](#)

LES BALISES POUR LES TABLEAUX

Les balises HTML utilisés pour la représentation d'un tableau sont : <table>, <tr>, <td>, <th>, <caption>, <colgroup>, <thead>, <tbody>, <tfoot>

Un tableau (<table>) comporte des lignes (row) <tr>. Ces lignes contiennent des cellules <td> ou des en-têtes <th>.

Exemple d'un tableau de 2 lignes, 2 colonnes. Du CSS a été ajouté pour donner un « aspect tableau ».

```
<div>Exemple 1</div>
<table>
  <tr>
    <td> 1 </td> <td> 2 </td>
  </tr>
  <tr>
    <td> 3 </td> <td> 4 </td>
  </tr>
</table>
```

Exemple 1

1	2
3	4

Pour fusionner des colonnes, la balise <td> et <th> dispose de l'attribut colspan="x" où x correspond au nombre de colonnes que l'on souhaite fusionner horizontalement. Son équivalent verticalement est rowspan="x".

Exemple :

```
<div>Exemple 2</div>
<table>
  <tr>
    <td colspan="3">Titre du tableau</td>
  </tr>
  <tr>
    <td>4</td> <td>5</td> <td>6</td>
  </tr>
</table>

<hr>
<div>Exemple3</div>
<table>
  <tr>
    <td rowspan="2">1</td> <td>2</td> <td>3</td>
  </tr>
  <tr>
    <td>5</td> <td>6</td>
  </tr>
</table>
```

Exemple 2

Titre du tableau		
4	5	6

Exemple3

1	2	3
	5	6

Les cellules d'en-tête (h=header) de colonnes ou de lignes sont définies par la balise `<th></th>` et non `<td>`. Le contenu de la balise `<th>` apparaît en gras et au centre. La balise `<caption>` placée après la balise `<table>` indique la légende du tableau.

Exemple :

```
<table>
  <caption>la légende du tableau</caption>
  <tr>
    <th>Attribut</th>
    <th>Fonction</th>
  </tr>
  <tr>
    <td><i>colspan</i></td> <td>pour la fusion de colonnes</td>
  </tr>
  <tr>
    <td><i>rowspan</i></td>
    <td>pour la fusion de lignes</td>
  </tr>
</table>
```

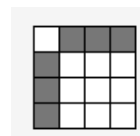
Exemple 4
la légende du tableau

Attribut	Fonction
<i>colspan</i>	pour la fusion de colonnes
<i>rowspan</i>	pour la fusion de lignes

Notons que cette balise permet de définir un en-tête soit de colonnes ou bien de lignes, tout dépend du tableau à réaliser. Pour associer les entêtes à leurs données dans ce type de tableaux, utiliser l'attribut **scope** sur les balises `<th>`. La valeur de cet attribut changera selon que la cellule d'entête concerne :

- La totalité d'une colonne : `scope="col"`.
- La totalité d'une ligne : `scope="row"`.

C'est le cas de tableau avec lecture à double entrée. Dans ce type de tableau on note une cellule vide en haut à gauche.



Exemple :

```
<table>
  <caption>Températures moyennes mensuelles des
    3 plus grandes villes de France.
  </caption>
  <tr>
    <td>&nbsp;</td>
    <th scope="col">Paris</th>
    <th scope="col">Marseille</th>
    <th scope="col">Lyon</th>
  </tr>
  <tr>
    <th scope="row">Juin</th>
    <td>22°C</td>
    <td>28°C</td>
    <td>26°C</td>
  </tr>
  <tr>
    <th scope="row">Juillet</th>
    <td>24°C</td>
    <td>30°C</td>
    <td>28°C</td>
  </tr>
</table>
```

Exemple 5
Températures moyennes
mensuelles des 3 plus grandes
villes de France.

	Paris	Marseille	Lyon
Juin	22°C	28°C	26°C
Juillet	24°C	30°C	28°C

STRUCTURER SON TABLEAU.

Les balises permettant de structurer (respecter cet ordre) de façon logique (n'affecte pas le visuel) les tableaux sont :

- `<thead>` : regrouper les informations concernant l'en-tête du tableau (titre, intitulé des colonnes).

- `<tbody>` pour le corps du tableau, soit l'ensemble des données de celui-ci.
- `<tfoot>` pour le contenu de bas de page par exemple des remarques ou une légende.

Exemple :

```
<table>
  <thead>
    <tr>
      <th colspan="2">la fusion de lignes et
        de colonnes</th>
    </tr>
  </thead>

  <tbody>
    <tr>
      <td><i>colspan</i></td>
      <td>pour la fusion de colonnes</td>
    </tr>
    <tr>
      <td><i>rowspan</i></td>
      <td>pour la fusion de lignes</td>
    </tr>
  </tbody>

  <tfoot>
    <tr>
      <td colspan="2" style="text-align: center;"> Pied de tableau</td>
    </tr>
  </tfoot>
</table>
```

La fusion de lignes et de colonnes	
<i>colspan</i>	Pour la fusion de colonnes
<i>rowspan</i>	Pour la fusion de lignes
Pied de tableau	

La balise `<colgroup>` est utilisée pour grouper des colonnes en vue de leur appliquer une mise en forme à l'ensemble des balises du groupe par l'entremise d'une feuille de style CSS. Elle se positionne juste après la balise `<table>` ou `<caption>` et avant `<tr>`. L'attribut `span="x"` détermine le nombre de colonnes groupées (x=1 par défaut).

Exemple : pour illustrer cette balise, il est difficile de ne pas appliquer un style. Voir attribut "style".

```
<table>
  <colgroup span="1" style="background-color: cyan;"> </colgroup>
  <colgroup span="2" style="background-color: bisque;"> </colgroup>
  <tr>
    <td>1</td><td>2</td><td>3</td>
  </tr>
  <tr>
    <td>4</td><td>5</td><td>6</td>
  </tr>
  <tr>
    <td>7</td><td>8</td><td>9</td>
  </tr>
</table>
```

1	2	3
4	5	6
7	8	9

LES STYLES SUR TABLEAUX

La plupart des styles déjà étudiés s'appliquent également aux tableaux. Mais nous allons revenir sur certaines de ces propriétés ce qui nous permettra de comprendre un certain nombre de concepts liés au CSS.

La plupart des exemples de ce cours seront basé sur l'exemple 5 tu précédent chapitre.

LES BORDURES ET LA COULEUR

Un des principes du CSS est que les enfants héritent des propriétés CSS du parent. Toute balise HTML possède un parent hormis `<html>`. Si ce parent possède un style particulier ce style est naturellement transmis aux balises enfant.

Mais attention certaines propriétés sont héréditaires, d'autres pas. Ce type d'information se trouve dans les spécifications du CSS. Il faut donc bien analyser chaque propriété pour savoir.

Par exemple La propriété « **color** » est une propriété héritable mais « **border** » ne l'ai pas. C'est ce que l'on remarque dans l'exemple suivant la couleur c'est bien propagé sur les balises enfant TR et TH et TD, par contre la bordure ne s'est appliquée que sur la balise table

```
#T1{
  color:red;
  border-style: groove; }
```

	Paris	Marseille	Lyon
Juin	22°C	28°C	26°C
Juillet	24°C	30°C	28°C

Assis si l'on veut appliquer une bordure sur les cellules il faudra répéter la propriété border style sur la balise TD ou TH. Notons au passage que la balise TR ne peut se voir appliquer une bordure.

```
#T1{
  color:red;
  border-style: groove;
}

#T1 td, #T1 th{ // sélecteur hiérarchique
  border-style: groove;
}
```

	Paris	Marseille	Lyon
Juin	22°C	28°C	26°C
Juillet	24°C	30°C	28°C

Nous notons dans l'exemple ci-dessus l'utilisation d'un sélecteur hiérarchique que nous allons maintenant présenter.

LES SELECTEURS HIERARCHIQUES

Il permet de désigner un élément contenu par un autre

A B	Descendants	Sélectionne tout élément B qui est descendant d'un élément A.
A > B	Enfants	Sélectionne tout élément B qui est enfant direct d'un élément A.
A + B	Frère adjacent	Sélectionne tout élément B immédiatement précédé d'un élément A.

A B

Reprenons l'exemple précédent. Nous sélectionnons les descendants th et td du sélecteurs d'identification #T1, même chose pour un autre tableau #T2. Cela permet d'appliquer le style choisi a un tableau spécifique.

```
#T1 td, #T1 th{
  border-style: groove;
}

#T2 td, #T2 th{
  border-style: dotted;
}
```

	Paris	Marseille	Lyon
Juin	22°C	28°C	26°C
Juillet	24°C	30°C	28°C

	Paris	Marseille	Lyon
Juin	22°C	28°C	26°C
Juillet	24°C	30°C	28°C

Voici un autre exemple :

Tous les paragraphes <p> inclus dans une balise <div> seront sélectionnés pour la mise en forme, même ceux inclus dans d'autre balise

```
div p {
  color:white;
  background-color: black;
}

<div>
  <p>Contenu du paragraphe </p>
  <header>
    <p> ici aussi c'est un paragraphe</p>
  </header>
  <p> autre paragraphe </p>
</div>
<p> autre paragraphe </p>
```

Contenu du paragraphe
ici aussi c'est un paragraphe
autre paragraphe
autre paragraphe

Exemple A > B :

Tous les paragraphes <p> directement descendant d'une balise <div> seront concernés par la mise en forme, mais pas les autres.

```
div > p {
    color:white;
    background-color: black;
}
<div>
    <p>Contenu du paragraphe </p>
    <header>
        <p> ici aussi c'est un paragraphe</p>
    </header>
    <p> autre paragraphe </p>
</div>
<p> autre paragraphe </p>
```

Contenu du paragraphe

ici aussi c'est un paragraphe

autre paragraphe

autre paragraphe

Exemple A + B : tous les paragraphes <p> frère d'une balise <div> sera concerné par la mise en forme.

```
div + p {
    color: white;
    background-color: black;
}
<div>
    <p>Contenu du paragraphe </p>
    <div>
        <p> ici aussi c'est un paragraphe</p>
    </div>
    <p> autre paragraphe </p>
</div>
<p> autre paragraphe </p>
<p> autre paragraphe </p>
```

Contenu du paragraphe

ici aussi c'est un paragraphe

autre paragraphe**autre paragraphe**

autre paragraphe

LES PSEUDOS CLASSES DE SELECTION

Ce type de pseudo classe sont particulièrement intéressante pour la mise en forme d'un tableau. Il va par exemple être possible de sélectionner une couleur particulière pour une ligne et alterner une autre couleur pour la 2nde ligne.

:root	Représente l'élément racine ou l'élément le plus haut d'un document. Par exemple, en HTML5, la balise <html>
:empty	Correspond aux éléments vides et qui n'ont donc pas d'enfants.
:only-child	Reprend l'enfant unique. Celui-ci n'a donc pas d'élément frère.
:first-child	Sélectionne le premier élément enfant.
:last-child	Sélectionne le dernier élément enfant.
:nth-child(n)	Désigne le nième élément enfant où n est un nombre ou les mots-clés even (pairs) ou odd (impairs).
:nth-last-child(n)	Sélectionne le nième enfant en partant du dernier élément.
:only-of-type	Reprend l'unique élément de ce type.
:first-of-type	Représente le premier élément de ce type.
:last-of-type	Représente le dernier élément de ce type.
:nth-of-type(n)	Désigne le nième élément de ce type où n est un nombre ou les mots-clés even (pairs) ou odd (impairs).
:nth-last-of-type(n)	Sélectionne le nième élément de ce type en partant du dernier élément.

Exemple : Le tableau T3 à la mise en forme suivante : couleur de fond particulière pour la première ligne, blanc pour la première cellule, puis « beige » pour les lignes paires, et « burlywood » pour les lignes impaires

```
#T3 tr:first-child{
    background-color: brown !important;
}
#T3 tr:nth-child(even){
    background-color: beige;
}
#T3 tr:nth-child(odd){
    background-color: burlywood;
```

	Paris	Marseille	Lyon
Juin	22°C	28°C	26°C
Juillet	24°C	30°C	28°C

```

}
#T3 td:first-child{
    background-color: white;
}

```

On note le mot clef CSS « **!important** » signifiant que cette propriété passe avant les autres et doit s'appliquer en priorité. Dans cet exemple l'application de la couleur « Brown » sur la première ligne est prioritaire sur l'application de la couleur beige ou « burlywood ». Il existe encore « **!veryimportant** » qui a encore plus de pouvoir que « **!important** ».

Il est à noter que ce style vient d'être appliqué sur la balise TR. Il est bien entendu possible de même sur la balise TD ce qui ferait une alternance de couleurs non pas sur les lignes mais sur les colonnes.

AUTRES PROPRIETES CSS DE MISE EN FORME DU TABLEAU

- **Exemple 1** : Le tableau est centré sur la page, fait 80% de la largeur et le texte est centré dans les cellules, sans bordure.

```

#T3{
    width: 80%;
    margin: auto;
    text-align: center;
}

```

	Paris	Marseille	Lyon
Juin	22°C	28°C	26°C
Juillet	24°C	30°C	28°C

Par défaut on note une séparation entre les cellules. Si on souhaite coller les cellules il faut utiliser la propriété CSS « **border-collapse** ». On contraire si on veut espacer les cellules on utilisera « **border-spacing** ».

Exemple : border-collapse

```

#T3{
    width: 80%;
    margin: auto;
    text-align: center;
    border-collapse: collapse;
}
#T3 td, #T3 th{
    border: 1px solid black !important;
}

```

	Paris	Marseille	Lyon
Juin	22°C	28°C	26°C
Juillet	24°C	30°C	28°C

Exemple : border-spacing

```

#T4, #T3{
    width: 80%;
    margin: auto;
    text-align: center;
    border-spacing: 5px;
}

```

	Paris	Marseille	Lyon
Juin	22°C	28°C	26°C
Juillet	24°C	30°C	28°C

- **Exemple 2** : on note dans l'exemple précédent que les colonnes n'ont pas des largeurs uniformes, et la hauteur des lignes s'adaptent au contenu. Dans l'exemple qui suit nous aurons une largeur de colonne fixe et la hauteur de ligne fixe également et un alignement vertical sur le haut de la cellule soit <td> ou <th>. Par défaut l'alignement vertical est sur la valeur **middle**

```

#T5{
    height: 200px;
    table-layout: fixed;
}
#T5 td, #T5 th{
    vertical-align: top;
}

```

	Paris	Marseille	Lyon
Juin	22°C	28°C	26°C
Juillet	24°C	30°C	28°C

On note que ce tableau fait 200 pixels de hauteur. Chaque ligne aura une hauteur de $(200 \text{ px} - \text{espacement}) / 3$. Dans tous les cas, la hauteur de ligne ne pourra jamais être inférieure à son contenu. Avec la propriété « **table-layout** » a **fixed** on constate une largeur de colonne uniforme. La valeur par défaut pour cette propriété est auto.

Dans cet exemple le visuel est uniforme. Mais dans le cas d'un trop grand nombre de colonnes la largeur cela ne serait plus le cas.

	Paris	Marseille	Lyon	Strasbourg	Montpellier	Nantes	Nice	Toulouse	Lille	Clermont-Ferrand	Valence	Orange	Dijon	Avignon	Toulon	Pau	Brest	Rennes
Juin	22°C	28°C	26°C															
Juillet	24°C	30°C	28°C															

- **Exemple 3** : la légende du tableau avec `<caption>` s'affiche par défaut en haut du tableau. Il est possible de modifier cela avec la propriété CSS `caption-side`.

```
#T6{
  caption-side:
bottom;
}
```

	Paris	Marseille	Lyon
Juin	22°C	28°C	26°C
Juillet	24°C	30°C	28°C

Par la suite il est possible d'appliquer un style sur la balise `<caption>` selon notre convenance

- **Exemple 4** : il s'agit d'une propriété que nous avons déjà évoquée avec les images. Il s'agit là la pseudo classe « **hover** » qui permet l'ordre le survol d'une ligne ou d'une cellule de changer l'aspect de celle-ci.

```
#T7 tr:hover{
  background-color: #ccffcc;
}
#T7 td:hover{
  background-color: aquamarine;
}
```