

Chapitre 2

Les bases de HTML5

Introduction

Le **HTML** (*HyperText Markup Language*) est le format de données conçus pour présenter des pages web qui pouvant être lues par des navigateurs (Exp : *Internet Explorer*, *Mozilla Firefox*, *Opera*, *Google Chrome*, ...).

C'est un langage de présentation de données, et non pas un langage de programmation. Il est figé, c'est-à-dire qu'une fois le document chargé dans le navigateur, il ne répond à aucune action de l'utilisateur sur le contenu de la page. Ce langage est pourvu d'un système de balisage (basé sur les balises) qui va permettre de structurer les documents.

HTML5 (*HyperText Markup Language 5*) est la dernière révision majeure d'HTML (format de données conçu pour représenter les pages web). Cette version a été finalisée le 28 octobre 2014. Dans le langage courant, HTML5 désigne souvent un ensemble de technologies Web (HTML5, CSS3 et JavaScript) permettant notamment le développement d'applications.

I. Qu'est-ce qu'une balise HTML ?

Une balise HTML est l'élément de base du codage HTML d'une page web. Invisibles pour l'internaute (sauf en activant l'affichage du code source), elle est interprétée par le navigateur pour la composition de la page. Elle est toujours délimitée par les signes « < » et « > ».

Une balise peut en plus comporter de zéro à plusieurs **attributs**. Les attributs sont des informations complémentaires qui la caractérisent. Ils se présentent sous la forme :

nomattribut="valeur"

Exemple de balise et son attribut

```
<html lang="fr">  
<h1 id="titre"> Titre de ma page </h1>
```

Chaque balise ouverte doit être fermée, cependant il existe des exceptions. On distingue deux types de balises :

- Les balises simples (mono-balises).
- Les balises doubles.

1. Les balises simples

Ce sont des balises qui sont dites "vides", c'est-à-dire qu'elles ne vont contenir aucune autre balise HTML. Ces balises n'ont pas besoin d'être fermées.

Exemple de balise simple

```

```

2. Les balises doubles

Les balises doubles sont dites ouvrantes/fermantes, c'est-à-dire qu'elles nécessitent deux balises, une ouvrante et une fermante dans lesquelles on va pouvoir mettre d'autres balises ou du texte. La balise fermante est identique à la balise ouvrante, à la différence qu'elle contient un "/".

Exemple de balise double

```
<p> Ici du texte ou toute autre balise. </p>
```

3. Imbrication des balises

Lorsqu'on cumule l'écriture de plusieurs balises ouvrantes (forcément inévitable), il est impératif de respecter la hiérarchie des balises, c'est-à-dire que la première ouverte sera la dernière fermée. Le chevauchement des balises est une erreur grave.

Exemple de balises imbriquées

```
<b><u>Ici les balises sont correctement imbriquées.</u></b>
```

Exemple de balises chevauchées

```
<b><u>Ici les balises ne sont</b>pas correctement imbriquées, c'est une erreur.  
</u>
```

4. Structure d'un document HTML

Document HTML de base

```
<!DOCTYPE html>  
<html>  
  <head>  
    <meta charset="utf-8">  
    <title>Document sans nom</title>  
  </head>  
  <body>  
    ....  
  </body>  
</html>
```

- `<!DOCTYPE html>` définit ce document pour être HTML5
- `<html>` est l'élément racine d'une page HTML
- `<head>` contient des informations méta sur le document
- `<title>` spécifie un titre pour le document
- `<body>` contient le contenu visible de la page
- `<h1>` définit un titre
- `<p>` définit un paragraphe

5. Le DOCTYPE

Il sert d'une part à déclarer quel "*langage*", "*grammaire*", ou *norme* on applique à notre page. Cette déclaration nous permet de soumettre nos pages au validateur du W3C (*World Wide Consortium*) qui vérifie que notre code HTML ne comporte aucune erreur de syntaxe (balises mal fermées, utilisation d'attributs inexistants...).

6. Les commentaires

Les commentaires permettent de laisser des informations ou des descriptions du code et sont utilisables à n'importe quel endroit du document.

Commentaires HTML

```
<!-- Voici un premier commentaire sur une ligne. -->
<!-- Voici un deuxième commentaire sur une ligne.
Celui-ci est multilignes. -->
<!------- Cette méthode est déconseillée ! ----->
```

II. Le corps du document (body)

1. Mise en forme

a. Les niveaux de titres

Il existe différents niveaux de titres, ils vont de **1** à **6** et ont chacun leur importance. La balise représentant ces niveaux de titre est `<hX>` où **X** représente le niveau.

Exemples de titres

```
<h1> Titre de ma page </h1>
<h2> Titre de niveau 2 </h2>
<h3> Titre de niveau 3 </h3>
```

b. Les paragraphes de texte

Comme tout document texte, un document **HTML** doit être segmenté en paragraphes. Cette segmentation est possible en précisant où commence chaque paragraphe et où il se termine. C'est le rôle de la balise `<p>`.

Les paragraphes de texte en HTML

```
<p> Voici un premier paragraphe. </p>
<p> Voici un deuxième paragraphe. </p>
<p> Voici un troisième paragraphe. </p>
```

c. La mise en forme de texte

Style	Balise	Affichage
gras	<code>Ce texte s'affichera en gras. </code>	Ce texte s'affichera en gras.
strong	<code>Ce texte s'affichera en gras pour souligner qu'il est important. </code>	Ce texte s'affichera en gras pour souligner qu'il est important.
italique	<code><i>Ce texte s'affichera en italique. </i></code>	<i>Ce texte s'affichera en italique.</i>
emphase	<code>Ce texte s'affichera en italique pour souligner qu'il est important.. </code>	<i>Ce texte s'affichera en italique pour souligner qu'il est important..</i>
souligné	<code><u>Ce texte sera souligné. </u></code> <code><ins>Ce texte sera souligné. </ins></code>	<u>Ce texte sera souligné.</u>
barré	<code><s>Ce texte sera barré. </s></code> <code>Ce texte sera barré. </code>	Ce texte sera barré.
exposant	Ce texte sera en <code><sup></code> exposant. <code></sup></code>	Ce texte sera en ^{exposant} .
indice	Ce texte sera en <code><sub></code> indice. <code></sub></code>	Ce texte sera en _{indice} .
surligner	ce <code><mark></code> mot <code></mark></code> sera surligné	Ce mot sera surligné
plus petit	<code><small>Ce texte est plus petit. </small></code>	Ce texte est plus petit.

d. Modifier la couleur du texte

Pour modifier la couleur du texte on utilise l'attribut « **style** » avec la propriété « **color** » :

Texte en rouge

```
<h1 style="color:#ff0000"> Ce texte sera en rouge. </h1>
```

Les couleurs peuvent être écrites de deux manières :

- En hexadécimal de type RVB et précédées d'un dièse (#) ;

```
Exemples : #ff0000 => rouge, #00ff00 => vert, #0000ff => bleu, ...
```

- Textuelles en anglais US ;

```
Exemples: red, yellow, pink, violet, blue, Orange, SlateBlue, Tomato, MediumSeaGreen, ...
```

e. Modifier la couleur de fond

Pour modifier la couleur du texte on utilise l'attribut « **style** » avec la propriété « **background-color** » :

Fond du texte en bleu

```
<h1 style="background-color :blue"> Ce texte sera en rouge. </h1>
```

f. *Modifier la police du texte*

Pour modifier la police du texte on utilise l'attribut « **style** » avec la propriété « **font-family** » :

Texte en verdana

```
<h1 style="font-family:verdana;"> Ce texte sera en verdana. </h1>
```

On a tendance à écrire des familles de polices plutôt qu'une seule police. En effet, il se peut que sur un système, différent du nôtre, la police choisie ne soit pas installée. On sépare alors les différentes polices par une virgule.

Texte en verdana ou sans-serif

```
<h1 style="font-family:verdana,sans-serif">Ce texte sera en verdana ou en sans-serif si  
verdana n'est pas installée.</h1>
```

g. *Modifier la taille du texte*

Pour modifier la taille du texte on utilise l'attribut « **style** » avec la propriété « **font-size** » :

Texte en taille 10px

```
<h1 style="font-size: 10px"> Ce texte aura 10px de taille. </h1>
```

Par défaut, la valeur en pixels de font-size vaut "**16px**".

Il y a deux manières pour modifier la valeur de cette propriété

/ Valeurs dont la taille est absolue */*

```
font-size: xx-small;  
font-size: x-small;  
font-size: small;  
font-size: medium;  
font-size: large;  
font-size: x-large;  
font-size: xx-large;
```

/ Valeurs relatives */*

```
font-size: larger;  
font-size: smaller;
```

/ Valeurs de longueurs */*

```
font-size: 12px;  
font-size: 0.8em;
```

/ Valeurs proportionnelles */*

```
font-size: 80%;
```

h. *Alignement du texte HTML*

Pour aligner un texte on utilise l'attribut « **style** » avec la propriété « **text-align** » :

Texte centré

```
<h1 style="text-align: center"> Ce texte sera centré </h1>
```

```
/* Exemples de valeurs */  
text-align: left;  
text-align: right;  
text-align: center;  
text-align: justify;
```

2. Les images

Pour afficher une image en **HTML**, on utilise la balise «****». Elle admet des différents attributs, dont les plus importants sont «**src**» et «**alt**».

- **src** : permet de renseigner le chemin de l'image, qui peut être absolu ou relatif.
- **alt** : affiche un texte alternatif à la place de l'image dans le cas où, pour une raison ou une autre, celle-ci ne peut pas être chargée. Il aide de plus les moteurs de recherche à référencer l'image.

Afficher une image

```

```

Les différents formats d'image supportés par cette balise sont :

• GIF • JPG • PNG • BMP (à éviter).

3. Les liens hypertextes

Un lien hypertexte est un élément **HTML** permettant d'envoyer le visiteur vers une nouvelle page. On peut insérer dans cet élément toute sorte de textes, images et autres balises dites "**inline**" (comme les balises ``, ``, etc). Un lien sera représenté dans le code par la balise «**<a>**». L'adresse de destination doit se trouver dans l'attribut «**href**».

L'attribut «**title**», permet de renseigner un petit descriptif sur le contenu de la page cible, c'est un atout non négligeable en matière de référencement et d'accessibilité.

Exemple d'un lien texte

```
<a href="adresse/de/destination.html"> Texte du lien </a>
```

Exemple d'un lien image

```
<a href="adresse/de/destination.html">  
</a>
```

On distingue deux types de liens, les liens externes et les liens internes.

a. Les liens externes

Un lien externe est un lien hypertexte qui renvoie vers une autre page. Cette autre page peut être une page du même.

b. Les liens internes

Un lien interne est un lien qui renvoie vers la même page, mais pas forcément au même endroit de la page.

Attention : A ne pas confondre "*liens internes*" avec "*lien interne au site*" ! Un lien interne au site est en général un lien externe à la page.

Les ancres permettent de naviguer à l'intérieur d'une même page.

Pour faire une ancre, il faut utiliser le signe dièse (#) suivi d'un mot-clef. Ce mot-clef sera repris dans l'attribut « **id** » d'une balise à laquelle on veut faire une ancre :

Exemple d'une ancre interne

```
<a href="#motClef"> Texte de mon ancre </a>  
<h2 id="motClef"> Texte vers lequel l'ancre est renvoyée </h2>
```

Ou bien

```
<a name=" motClef "> Texte vers lequel l'ancre est renvoyée </a>
```

Si on veut faire une ancre vers une page externe, il suffit de rajouter l'adresse de destination dans la balise de lien :

Exemple d'une ancre externe

```
<a href="adresse/de/destination.html#motClef">Texte de mon ancre</a>
```

c. Les liens e-mail

Exemple d'un lien vers une adresse e-mail

```
<a href="mailto:xxx@monsite.com">Ecrivez-moi</a>
```

→ Ce type de liens ouvre le logiciel de messagerie par défaut des visiteurs.

Exemple d'un lien vers une adresse e-mail avec un sujet par défaut

```
<a href="mailto:xxx@monsite.com?subject=Sujet du mail">Ecrivez_moi</a>
```

Exemple d'un lien vers une adresse e-mail avec un corps par défaut

```
<a href="mailto:xxx@monsite.com?body=Corps du mail">Ecrivez-moi</a>
```

Exemple d'un lien vers une adresse e-mail avec un sujet et un corps par défaut

```
<a href="mailto:xxx@monsite.com?subject=Sujet du mail&body=Corps du mail">Ecrivez-moi</a>
```

Exemple d'un lien vers plusieurs adresses e-mail

```
<a href="mailto:destinataire1@monsite.com;destinataire2@monsite.com">Ecrivez-nous</a>
```

d. Les Protocoles

Un protocole est un ensemble de conventions nécessaires qui permet à deux entités distantes de communiquer entre eux. Il diffère selon le type de données échangées. Il existe de nombreux protocoles disponibles par lien hypertexte, en voici une liste non exhaustive :

Protocole	Signification de l'acronyme	Description	Syntaxe utilisée
HTTP	HyperText Transfer Protocol	Permet de transférer des données sur l'Internet.	<code>..</code>
HTTPS	HyperText Transfer Protocol Secured	Permet de transférer des données sur Internet en permettant de vérifier l'identité du site grâce à un certificate d'authentification.	<code>..</code>
SMTP	Simple Mail Transfer Protocol	Permet de transférer des e-mails.	<code>..</code>
FTP	File Transfer Protocol	Permet de transférer des fichiers.	<code>...</code>
IRC	Internet Relay Chat	Permet de se connecter à un serveur de Chat.	<code>...</code>

e. Informations complémentaires

- Pour éviter d'avoir le cadre bleu autour d'une image lorsqu'elle est dans un lien, il suffit de renseigner l'attribut « border » à "0" :

Image sans bordure bleue

```
<a href="adresse/de/destination.html"> </a>
```

- Si on veut avoir une info bulle qui s'affiche sur le lien lors du passage de la souris, on peut renseigner l'attribut « title » :

Info bulle sur le lien

```
<a href="adresse/de/destination.html" title="Description de la page cible">Texte du lien</a>
```

f. Les listes

Elles servent, comme leur nom l'indique, à créer une liste d'éléments (items), plus précisément à les énumérer, comme dans le cas d'un sommaire par exemple. On distingue cinq types différents de listes, chacun prévu pour un usage bien précis :

- Les listes non ordonnées, ou à puces.
- Les listes ordonnées.
- Les listes de définitions.
- Les listes de répertoires.
- Les listes de menus.

Dans ce présent chapitre nous allons étudier que les deux premiers types.

1. Les listes non ordonnées

Elles sont démarquées par la balise « **** » et chacun des items est entouré de la balise « **** » :

Exemple de liste non ordonnée

```
<p> Ma liste de courses : </p>
<ul>
  <li> Champignons </li>
  <li> Avocat </li>
  <li> Dentifrice </li>
  <li> Céréales </li>
</ul>
```

Ma liste de courses :

- Champignons
- Avocat
- Dentifrice
- Céréales

Les différents types de puces

On renseigne le type des puces avec l'attribut « **style** » et la propriété « **list-style-type** » de la balise ****.

Les types sont :

- **disc** (par défaut au niveau 1 d'imbrication)
- **circle** (par défaut au niveau 2 d'imbrication)
- **square** (par défaut au niveau 3 d'imbrication)
- **none**

Dans l'exemple précédent, nous avons donc vu une liste non ordonnée avec une puce de type disc. Voici à quoi ressemblent les deux autres :

Exemple de liste non ordonnée

```
<p>Ma liste de courses :</p>
<ul style="list-style-type:circle">
  <li>Champignons</li>
  <li>Avocat</li>
  <li>Dentifrice</li>
  <li>Céréales</li>
</ul>
```

Ma liste de courses :

- Champignons
- Avocat
- Dentifrice
- Céréales

Exemple de liste

```
<p> Ma liste de courses : </p>
<ul style="list-style-type:square">
  <li>Champignons</li>
  <li>Avocat</li>
  <li>Dentifrice</li>
  <li>Céréales</li>
</ul>
```

Ma liste de courses :

- Champignons
- Avocat
- Dentifrice
- Céréales

Les puces en image

Pour avoir une puce sous forme d'image, il suffit d'utiliser cette image pour remplacer la puce originale.

Cependant cette méthode n'est possible qu'à l'aide de **CSS** et de la propriété *list-style-image* (Voir chapitre IV).

Exemple de liste non ordonnée

```
<p>Ma liste de courses :</p>
<ul>
  <li>Champignons</li>
  <li>Avocat</li>
  <li>Dentifrice</li>
  <li>Céréales</li>
</ul>
```

Ma liste de courses :

- ▶ Champignons
- ▶ Avocat
- ▶ Dentifrice
- ▶ Céréales

CSS (voir chapitre IV)

```
li {list-style-image : url(puce.gif); }
```

2. Les listes ordonnées

Elles sont introduites par la balise «****» et chacun des items est encadré par la balise «****» :

Exemple de liste ordonnée

```
<ol>
  <li> 1ère place </li>
  <li> 2ème place </li>
  <li> 3ème place </li>
</ol>
```

1. 1ère place
2. 2ème place
3. 3ème place

Il existe cinq manières différentes de numéroté les listes ordonnées. On renseigne le type de numérotation avec l'attribut **type** de la balise ****.

Les types sont :

- **1** : Numérotation par les chiffres arabes (par défaut).
- **I** : Numérotation par les chiffres romains.
- **i** : Numérotation par les chiffres romains minuscules.
- **A** : Numérotation par les lettres de l'alphabet.
- **a** : Numérotation par les lettres de l'alphabet minuscules.

Dans l'exemple précédent, nous avons donc vu une liste ordonnée avec une numérotation par les chiffres arabes. Voici à quoi ressemblent les autres :

Exemple de liste ordonnée

```
<ol type="I">
  <li> 1ère place </li>
  <li> 2ème place </li>
  <li> 3ème place </li>
</ol>
```

- I. 1ère place
- II. 2ème place
- III. 3ème place

Exemple de liste ordonnée

```
<ol type="i">
  <li> 1ère place </li>
  <li> 2ème place </li>
  <li> 3ème place </li>
</ol>
```

- i. 1ère place
- ii. 2ème place
- iii. 3ème place

Il est tout à fait possible, pour une raison ou une autre, de modifier le numéro de départ par lequel va s'incrémenter la numérotation.

Pour cela, il suffit de préciser une valeur à l'attribut **start** de la balise ****. Cette valeur doit dans tous les cas être numérique :

Exemple de liste ordonnée

```
<ol type="1" start="8">
  <li> 8ème place </li>
  <li> 9ème place </li>
  <li> 10ème place </li>
</ol>
```

8. 8ème place
9. 9ème place
10. 10ème place

3. Les listes imbriquées

L'imbrication des listes permet une présentation claire et détaillée d'un menu ou d'un sommaire par exemple. On peut imbriquer les listes non ordonnées et ordonnées sans distinction et même les mélanger :

Exemple de listes imbriquées :

```
<ol type="I">
  <li> La genèse</li>
  <li> La vie
    <ol type="A">
      <li>L'enfance</li>
      <li>L'adolescence</li>
      <li>Le passage à l'âge adulte
        <ul>
          <li> Le jeune adulte </li>
          <li> La crise de la quarantaine</li>
        </ul>
      </li>
      <li> La vieillesse </li>
    </ol>
  </li>
  <li>La mort</li>
</ol>
```

- I. La genèse
- II. La vie
 - A. L'enfance
 - B. L'adolescence
 - C. Le passage à l'âge adulte
 - Le jeune adulte
 - La crise de la quarantaine
 - D. La vieillesse
- III. La mort

III. Les conteneurs HTML

Pour organiser le contenu d'une page web en HTML on pourra utiliser soit des balises génériques ou des balises sémantiques.

4. Les balises génériques

- L'élément HTML **<div>** (qui signifie division du document) est un conteneur générique qui permet d'organiser le contenu sans représenter rien de particulier. Il peut être utilisé afin de grouper d'autres éléments pour leur appliquer un style (en utilisant les attributs *class* ou *id*).

```
<div>Salut</div>
<div>Ceci est une division</div>
```

- L'élément HTML **** est un conteneur générique en ligne (*inline*) pour les contenus phrasés. Il ne représente rien de particulier. Il peut être utilisé pour grouper des éléments afin de les mettre en forme (grâce aux attributs *class* ou *id* et aux règles CSS).

```
<span>Salut</span>
<span>Ceci est un conteneur générique</span>
```

5. Les balises sémantiques

Ces balises ne s'occupent pas de la mise en page. Elles servent seulement à définir les différentes parties d'une page web. Elles peuvent être imbriquées les unes dans les autres.

Balise	fonction
<header>	en-tête de page ou de section
<section>	section de contenu
<article>	article de composition indépendante
<nav>	liens de navigation
<figure>	regroupement des images et de leur description
<footer>	pied de page ou de section
<aside>	un contenu à part (sans relation directe avec le contenu de la page). Exp: pub, etc.
....

