

## Chapitre 3

# HTML

### 1. Création d'une page web

Lorsque vous souhaitez créer une page web ou un site internet, il serait logique de penser qu'il suffit d'allumer l'ordinateur et d'ouvrir un éditeur dans lequel taper le code.

Malheureusement, cette méthode n'a jamais fait ses preuves et a plutôt montré que le temps de création de la page ou du site était deux fois plus long.

Il est préférable de préparer sur papier, au brouillon, le dessin de ce qui devrait être obtenu au final, un peu comme le réalisateur qui va utiliser un story-board sur lequel les plans qu'il va devoir filmer ont déjà été réfléchis et dessinés : il sait exactement ce qu'il doit filmer, sous quel angle, avec quel éclairage...

Ensuite, une page web est constituée de mots issus du langage HTML. Certains d'entre eux sont obligatoires et vont être étudiés dans la section suivante. D'autres sont plus spécifiques à ce qui doit être affiché ou non et dépendent du brouillon qui aura été fait.

La structure d'une page web écrite dans le langage HTML correspond aux fondations pour une maison. C'est sur cette structure que vont reposer les styles CSS et le code JavaScript. Il est donc vital de très bien la construire. En même temps, il est évident qu'il y aura parfois des oublis ou simplement un changement d'avis, et il sera toujours possible de revenir sur la structure, de la changer, mais tout cela au détriment d'une certaine perte de temps et parfois de beaucoup d'efforts.

## ■ Remarque

*Comme il est noté plus haut, s'il n'y a pas une préparation et une réflexion sur le résultat à obtenir avant même d'allumer l'ordinateur, le temps de travail sera doublé.*

## 2. Code HTML obligatoire

Une page HTML doit contenir au minimum le code ci-dessous, qui pourra être divisé en deux grandes parties :

- l'en-tête de la page : `<head></head>` ;
- le corps de la page : `<body></body>`.

```
<!DOCTYPE html>
<html>

  <head>
    <title>Le titre de ma page</title>
    <meta http-equiv="Content-Type" content="text/html;
charset=UTF-8">
  </head>

  <body>
    Structure, texte et images...
  </body>

</html>
```

Examinons en détail ce code, car nous le retrouverons **systématiquement**.

### 3. Le doctype

La première balise qui est utilisée dans une page HTML est la balise `<!DOCTYPE html>`.

Jusqu'au HTML5, donc en HTML4 ou XHTML par exemple, le doctype était une balise contenant une centaine de caractères, plutôt complexe et qui était moyennement respectée ou comprise par les développeurs. Aujourd'hui, avec le HTML5, cette balise s'écrit plus simplement. Son contenu permet d'indiquer au navigateur que le code qui suit est du HTML5.

Donc `<!DOCTYPE html>` sera toujours la première chose à écrire dans le code d'une page web en HTML5.

Ensuite, il y a les balises `<html></html>`, deux balises qui vont contenir toute la page et qui indiquent qu'à partir de cette ligne `<html>`, c'est le langage HTML qui est utilisé.

Les balises `<head></head>` ou en-tête de page contiennent des informations de paramétrage qui vont être envoyées au navigateur **avant même** que celui-ci n'affiche quoi que ce soit, et ce afin qu'il se prépare.

Les balises `<body></body>`, pour le corps de la page, vont contenir les éléments que le navigateur doit afficher à l'écran.

L'essentiel du travail d'un webmaster se trouve dans la partie BODY, mais si le HEAD est mal écrit ou qu'il y a des éléments importants qui sont omis, la page pourra mal fonctionner voire ne pas fonctionner du tout.

### 4. La balise `<head>`

Parmi les balises indispensables au bon fonctionnement d'un site, il y a les balises `<title></title>` qui vont se retrouver dans l'en-tête `<head>`.

#### Exemple

```
■ <title>Liste des nouveaux produits</title>
```

Cette dernière, un titre, va afficher le texte contenu entre les balises `<title>` et `</title>`, quelque part dans le navigateur. Cela peut être dans la partie supérieure de la fenêtre du navigateur ou encore dans un onglet.

En plus de donner à la personne qui visite la page une idée de ce qui va être affiché, les informations de la balise `<title>` vont être récupérées par les moteurs de recherche.

Autrement dit, le contenu de la balise `<title>` de l'exemple ci-dessus va informer l'internaute et les moteurs de recherche sur le contenu de la page.

Si un internaute peut se contenter du mot « produits » et comprendre qu'il s'agit de chaussures parce que le site est spécialisé dans la vente de chaussures, les moteurs de recherche, eux, ne peuvent pas deviner de quel type de produit il s'agit. Autrement dit, si un internaute recherche le mot « chaussures » dans un moteur de recherche, le résultat de la recherche n'affichera sûrement pas la page qui a pour titre : « Liste des nouveaux produits ».

Heureusement, le mot « chaussures » sera sûrement écrit dans le corps de la page, et le moteur de recherche proposera tout de même ce site. Mais si le titre était :

```
■ <title>Liste des nouvelles chaussures de ville.</title>
```

ou quelque chose d'avoisinant, il est certain que le moteur de recherche proposerait plus facilement ce site.

#### ■ Remarque

*Autrement dit, utiliser un maximum de mots-clés dans le titre de vos pages aide au référencement.*

La balise qui suit est la balise `<meta></meta>`.

Cette balise contient des informations sur la page qui vont aider le navigateur à, par exemple, afficher correctement la page. On parle de métadonnées.

#### Exemple

```
■ <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
```

Dans ce cas, la balise `<meta>` indique au navigateur que le charset (le jeu de caractères utilisé pour l’affichage) est UTF-8. Autrement dit, cela permet d’afficher tout type de caractères. Sans rentrer dans les détails de l’UTF-8, c’est le format le plus utilisé aujourd’hui pour les sites web, de par sa simplicité et son efficacité.

De nombreuses possibilités sont à notre disposition pour la balise `<meta>`, par exemple :

```
■ <meta name='description' content='Description de la page actuelle...'
```

C’est ce texte de description qui va être retrouvé dans certains moteurs de recherche. Cette description peut ou non être utilisée dans une page HTML. Cela ne changera rien au fonctionnement du site, mais si la description est faite, les outils d’analyse de site pourront récupérer l’information et cela aidera le site à être visible sur Internet.

```
■ <meta name='keywords' content='liste de mots-clés séparés d’une virgule'
```

Ici, la balise `<meta>` contient les keywords (les mots-clés) et indique au moteur de recherche les sujets traités dans cette page, permettant ainsi un affichage plus précis des sites correspondant à la recherche de l’internaute.

#### ■ Remarque

*Il existe certaines polémiques quant à l’utilité de `<meta name='keywords' ...`, les moteurs de recherche ne s’en servant pas forcément. Par contre, tout le texte contenu dans le site sera toujours pris en compte, alors **choisissons en priorité des mots-clés dans les liens et dans le texte.***

## 4.1 Lien avec une feuille de style

D’autres éléments, facultatifs, peuvent être indiqués dans la balise `<head>`.

Par exemple le lien vers une feuille de style.

```
■ <link href="../../../site.css" type="text/css" rel="stylesheet"  
media="screen"/>
```

C'est de cette façon que sera indiqué au navigateur que la page web qu'il est en train d'afficher utilise des styles qui sont écrits dans le fichier `site.css` et que ces styles ont été définis pour l'affichage de la page sur un écran (`screen` : un écran, comme celui de l'ordinateur), en opposition à des styles créés pour une sortie TV ou encore ceux pour l'impression (`print`).

## 4.2 Lien avec un fichier JavaScript

La balise `<head>` contient également l'accès aux scripts, qui seront dans un fichier dédié à JavaScript.

Par exemple :

```
■ <script src="../../../js/outils.js"></script>
```

Ici, `src` (la source) nous indique que le fichier `outils.js`, qui est stocké dans le dossier `js`, est chargé par le navigateur.

À noter que lors de la création d'un site internet, il faut souvent définir plusieurs feuilles de style, ainsi que plusieurs fichiers JavaScript.

### ■ Remarque

*Il est préférable (pour ne pas dire vital) de déclarer dans la balise `<head>` de votre page **d'abord** l'accès aux styles CSS et **ensuite** les scripts JavaScript. Cela va permettre au navigateur d'être beaucoup plus rapide pour afficher la page par rapport à une page dans laquelle les styles et le script seraient écrits de façon désordonnée, avec déclaration d'un script, puis de style, puis à nouveau un script... L'ordre est très important pour l'efficacité du navigateur, même si le fonctionnement est correct dans les deux cas.*

Un dernier point concerne l'insertion de CSS et de JavaScript.

Il est possible de noter directement le style dans la page HTML :

```
■ <style type='text/css'>
  body {
    font-size: 12px;
  }
</style>
```

Dans ce cas, nous utilisons la balise `<style>`, qui va nous permettre de fixer la taille du texte de notre page à 12 pixels.

Dans l'exemple suivant, on insère du code JavaScript directement dans la page en utilisant la balise `<script>` et en précisant le type: "text/javascript".

```
<script type='text/javascript'>
    var nom = 'Bob';
    var age  = 72;
</script>
```

Ici, deux variables sont créées, nom et age, et elles sont chacune initialisées avec une valeur de départ.

Le fait d'insérer du code CSS ou JavaScript directement dans une page fonctionne très bien. Par contre, puisque le code est écrit directement dans la page web, à chaque chargement de la page ce code est envoyé du serveur vers le client.

L'intérêt de stocker les styles dans un fichier à part et les scripts également est que le navigateur demande au serveur de lui envoyer les fichiers la première fois qu'il en a besoin, mais ne les redemande plus pour une seconde page puisqu'il les a déjà chargés et les garde un certain temps dans ce qu'on appelle le cache (un dossier de votre ordinateur contenant tout ce que le navigateur a chargé). De cette façon, le navigateur ne sollicite plus le serveur pour ces fichiers et cela implique que les pages s'affichent beaucoup plus vite puisque certains éléments n'ont plus à être rechargés.

## 5. La balise `<body>`

C'est à l'intérieur de `<body></body>` que sont utilisées toutes les balises qui vont servir à afficher quelque chose à l'écran.

Dans les grandes lignes, nous allons trouver toutes les balises permettant de **structurer la page**, c'est-à-dire de la découper en créant des cases, dans lesquelles il sera possible de mettre le menu du site, ou une zone d'image et une légende en dessous, ou un bloc pour un formulaire.

En fait, la structure des sites a longtemps été dessinée uniquement avec des tableaux. C'étaient des tableaux très simples : il suffisait de mettre dans telle cellule le menu, dans telle autre la liste des articles proposés sur le site, une cellule tout en haut pour une publicité par exemple, etc.

Cette méthode avec les tableaux `<table></table>` (cf. chapitre Mise en page HTML et CSS - section Les tableaux pour plus de détails sur les tableaux) est relativement simple et peut être intéressante pour afficher des données, comme cela serait fait avec un tableur. Le souci est qu'aujourd'hui, il devra également s'afficher correctement sur les smartphones dont les écrans sont plus petits que ceux des ordinateurs. Il ne sera donc pas possible d'afficher le site de la même façon. Autrement dit, il va falloir organiser l'affichage pour que cela fonctionne sur tous les écrans. Il faudra tenir compte également du mode paysage (horizontal) ou portrait (vertical).

Il existe des balises qui sont des blocs (comme un rectangle) et qui peuvent prendre n'importe quelle taille. Ils peuvent être affichés à tous les endroits voulus dans la page, en étant devant ou derrière un autre bloc. Ces blocs sont très pratiques à manipuler et autorisent des choses que les tableaux `<table>` ne permettent pas.

Pour donner quelques balises correspondant à ces blocs, il y a par exemple les balises `<div>`, `<section>`, `<article>`, `<nav>`, `<aside>`, `<header>`, `<footer>`...

Une fois que toutes les zones dans lesquelles sera affiché du texte ou des images, vidéos, infos... sont définies, le contenu pourra être écrit.

Avant de fabriquer une page web, il faut avoir une bonne idée du résultat final désiré. L'étape de création de la structure, de l'organisation des blocs, des dimensions et des positions est sûrement la plus importante et celle qui doit demander le plus d'attention, car tout le reste du site va reposer sur cette structure. Et s'il est nécessaire de changer par la suite l'affichage ou la disposition pour une raison ou une autre, dans certains cas cela se fera assez rapidement, et dans d'autres cas cela pourra demander énormément d'efforts, un peu comme si une personne décidait de changer la place de la cave dans la maison : cela reste possible, mais si on y avait pensé à la construction, cela aurait été beaucoup plus simple et rapide à réaliser.



Une fois la structure en place, il « suffira » de la remplir avec les images et textes souhaités. Pour cela, le webmaster a accès à différentes balises qui lui permettent de créer des listes à puces `<ul></ul>` (*Unordered List* : liste non ordonnée) ou ordonnée `<ol></ol>` (*Ordered List*), ou encore d'autres balises permettant d'écrire un titre en gros et gras `<h1></h1>`.

Il y a également une balise pour créer un paragraphe `<p>`, et à tout cela viendront s'ajouter des styles pour finaliser l'affichage. Tous ces points seront vus en détail au fur et à mesure.

Si vous avez survécu à toute cette théorie, la pratique arrive pour vous soulager un peu avec un exemple concret de mise en page.

## 5.1 Méthode et balises pour structurer une page

Une des balises qu'il est intéressant de savoir manipuler est la balise `<div>`. Elle a été la première du genre pour ensuite être imitée par les autres balises `<section>`, `<article>`, `<nav>`, `<aside>`, `<header>` et `<footer>` qui fonctionnent comme le `<div>`, mais qui ont en plus une identité qui permet de savoir un peu ce qu'elles contiennent.

Par exemple, la balise `<footer>`, de par son nom, indique qu'elle est le pied de quelque chose. Par pied, il faut comprendre zone la plus basse. S'il s'agit d'une page complète, le `<footer>` sera le bas de page. S'il s'agit d'un article, le `<footer>` sera le pied de l'article qui pourra contenir par exemple des informations sur l'auteur de l'article.

Ces balises permettent de définir des blocs, ou rectangles ; des zones dans lesquelles il est possible de mettre tout ce qui doit être affiché ou même d'inclure d'autres blocs.

Après avoir dessiné au brouillon l'allure que devrait avoir le site internet, il est possible de le définir dans la page HTML à l'aide de balises `<div>`.