

1

HTML

Le langage HTML est le langage de base permettant de construire des pages web, que celles-ci soient destinées à être affichées sur un iPhone/Android ou non. Dans notre cas, HTML sera associé à CSS et JavaScript pour construire des applications pour mobiles, qui seront accessibles sur le Web via une URL (applications web).

HTML ou XHTML ? Ici, c'est pareil !

Nous emploierons ici les termes HTML ou XHTML de façon identique. En effet, même si notre page HTML n'est pas formatée strictement comme l'exige XHTML, le navigateur web sait l'interpréter de façon correcte et restituer l'affichage adéquat.

Notions générales

Une page HTML correspond à un fichier texte pouvant être écrit sous n'importe quel éditeur de texte, le plus simple d'utilisation étant dans ce cas le meilleur. Le fichier texte contiendra principalement deux types de données :

- les balises HTML, pour indiquer ce que l'on désire afficher (un paragraphe, une liste, une table, une image, etc.) ;
- les attributs associés aux balises, pour préciser les paramètres de la balise (la taille de l'image, la couleur du texte, etc.).

Balises

Une balise HTML est entourée des caractères < et >, par exemple <p> pour indiquer un paragraphe. Une balise s'applique aux éléments qui la suivent. Pour indiquer la fin, on utilise la même balise, mais son nom est précédé de /. Ainsi, pour la fin du paragraphe, on écrira </p>.

Un paragraphe en HTML

```
<p>Voici un paragraphe écrit en HTML.</p>
```

Certaines balises ne possèdent pas de contenu. Par exemple, la balise
 qui sert à effectuer un retour à la ligne n'en a pas. Elle s'écrira sous la forme suivante.

Une balise sans contenu

```
<br />
```

Attributs

Une balise utilisée sans attribut entraînera un fonctionnement standard. Par exemple, la police de caractères, une couleur, etc., du paragraphe précédent (associés à la balise <p> employée ici) seront standards. Pour avoir un paragraphe de couleur rouge (au lieu du noir standard), nous devons l'indiquer dans les attributs de la balise <p>.

Un paragraphe de couleur rouge

```
<p style="color:red">Un paragraphe de couleur rouge.</p>
```

Chaque balise a ses propres attributs, mais la plupart possèdent les deux attributs `style` et `class` employés pour définir des styles CSS (voir le chapitre suivant).

Nous avons ici utilisé l'attribut `style`, dont la valeur est `color:red`. La forme générale d'écriture d'un attribut est `nom="valeur"`. Des espaces peuvent figurer de part et d'autre du signe =. La valeur de l'attribut est généralement entourée de ' ou " (simple guillemet ou double guillemet). Ils peuvent être omis si la valeur de l'attribut ne comporte pas d'espace. Dans ce cas, le prochain espace ou le < indiquant la fin de la balise permettent au navigateur de connaître la valeur de l'attribut.

Dans le cas précédent, on aurait donc pu aussi écrire :

Pas de guillemets pour les valeurs d'attributs

```
<p style=color:red>Un paragraphe de couleur rouge.</p>
```

Alors que ce qui suit aurait été mal interprété.

Un espace dans la valeur d'un attribut (sans guillemets)

```
<p style=color: red>Un paragraphe de couleur rouge.</p>
```

Dans ce cas, l'attribut `style` aurait la valeur `color:`, tandis que le mot `red` serait interprété comme le nom d'un nouvel attribut (qui n'existe évidemment pas !). Le paragraphe restera donc en couleur standard, c'est-à-dire noir.

Pour inclure un espace dans la valeur d'un attribut, il suffit de l'entourer de guillemets.

Un espace dans la valeur d'un attribut (avec guillemets)

```
<p style="color: red">Un paragraphe de couleur rouge.</p>
```

Dans le cas où plusieurs attributs sont utilisés dans une balise, chaque couple `nom="valeur"` est séparé du suivant par au moins un espace, voire un retour à la ligne. L'ordre dans lequel les attributs sont écrits n'a pas d'importance pour le résultat qui sera affiché.

Forme générale d'une page HTML

Une page HTML doit suivre quelques règles précises pour être correctement interprétée par le navigateur de l'utilisateur (dans notre cas, le navigateur Safari de l'iPhone ou le navigateur Chrome d'un mobile Android).

Format standard d'une page HTML (fichier `index.html`)

```
<html>  
<head>  
</head>  
  
<body>  
Voici une application pour iPhone !  
</body>  
</html>
```

La page HTML indiquée ici correspond à un fichier texte dont le nom est, par exemple, `index.html`. L'extension du fichier sert à connaître le type de son contenu, ici du HTML !

Cette page HTML commence par la balise `<html>` et se termine naturellement par cette même balise (`</html>`). Deux nouvelles balises sont introduites ici.

- La balise `<head>` correspond à la partie déclarative de la page : quels styles seront utilisés ? Quels fichiers de styles sont à inclure ? Quelle sera la taille maximum de l'écran ? Quel titre aura la page ?, etc.
- La balise `<body>` correspond à ce qui sera affiché à l'utilisateur : le texte, les images, les listes, etc.

Visualisation d'une page HTML

La page HTML écrite précédemment peut être visualisée dans n'importe quel navigateur actuel : Internet Explorer, Firefox, Google Chrome ou encore Safari.

Si vous souhaitez visualiser cette page depuis votre ordinateur personnel (PC, Mac, etc.), il suffit de faire glisser le fichier dans votre navigateur. Pour l'afficher sur votre iPhone/Android, il faut tout d'abord installer un serveur web sur votre ordinateur personnel, par exemple un serveur PHP (ou tout autre serveur). Dans cet ouvrage nous avons utilisé deux types de serveurs :

- AppServ sous Windows pour afficher des pages écrites en PHP (et bien sûr aussi HTML) ;
- MAMP sous Mac pour afficher des pages écrites en PHP (et bien sûr aussi HTML) ;
- Ruby on Rails (Windows ou Mac) pour afficher des pages écrites en Ruby (et également en HTML).

Des serveurs Java, Microsoft .Net, etc., fonctionneraient de manière similaire.

Visualisons notre page HTML en ouvrant, par exemple, Safari sur un iPhone et en tapant l'adresse de notre serveur local dans la barre d'adresse (voir figure 1-1).

Le texte inclus dans la balise `<body>` s'affiche, mais dans une taille très petite et pas du tout adaptée à l'iPhone ! Car, en effet, notre page HTML, sauf indication contraire, est écrite pour être affichée dans un navigateur d'un ordinateur de bureau, et non pour le navigateur d'un téléphone portable !

Il suffit d'ajouter la ligne suivante dans la partie `<head>` du code HTML pour afficher correctement la page HTML.

Avoir la taille de la page adaptée à la taille de l'écran

```
<meta name="viewport" content="user-scalable=no,  
width=device-width" />
```

Une fois cette balise `<meta>` insérée dans la partie `<head>`, voici l'affichage obtenu (voir figure 1-2).



Figure 1-1

Une page HTML affichée sur l'iPhone



Figure 1-2

Adaptation de l'affichage à la taille de l'écran

Par la suite, nous conserverons cette balise `<meta>` dans la partie `<head>` de chacune des pages HTML que nous écrirons.

ASTUCE Et pour Android, utile ou pas ?

Android n'a peut-être pas besoin de cette balise `<meta>`. Toutefois, nous l'incluons dans tous nos sources pour la compatibilité avec iPhone et Android.

Le texte

Nous avons précédemment utilisé une balise `<p>` pour afficher un paragraphe de texte. D'autres balises existent permettant d'afficher du texte sous différentes formes.

- `<p>` affiche un paragraphe. Deux paragraphes qui se suivent seront donc espacés verticalement à l'écran.
- `` met le texte en gras (*bold*).
- `<i>` met le texte en italique.
- `
` insère un retour à la ligne. Cette balise ne possède pas de contenu, d'où sa forme d'écriture.
- `<hr />` insère une ligne de séparation horizontale. Cette balise ne possède également pas de contenu.
- `<h1>` affiche un titre de paragraphe. Pour différencier les niveaux de titres, on utilise les autres balises `<h2>`, `<h3>`, `<h4>`, `<h5>` et `<h6>`. Plus l'indice qui suit `h` augmente, plus le titre est petit. L'indice qui suit `h` symbolise donc l'imbrication ou le niveau du titre.

En plus des balises décrites ci-dessus, il existe des codes spéciaux permettant d'écrire certains caractères. Ces codes s'appellent des « entités HTML ». En voici quelques-unes parmi les plus utilisées.

- ` ` ; insère un espace dit « insécable » entre deux mots. En effet, le navigateur ne tient pas compte du formatage du texte que nous tapons dans notre fichier HTML. Les retours à la ligne effectués par l'appui sur la touche *Return* ne sont pas pris en compte (lors de l'affichage), pas plus que les espaces insérés entre les mots (au-delà du premier espace).
- `<` ; : le caractère `<` est un caractère réservé en HTML, signifiant le début ou la fin d'une balise. Pour afficher le signe `<` à l'écran, on utilise donc l'entité `<` ;.
- `>` ; : même chose que pour `<` ;, mais pour afficher le caractère `>` (qui sinon serait interprété comme la fin de la balise).

Voici un exemple contenant quelques balises HTML.

Quelques balises HTML

```
<html>
<head>
  <meta name="viewport" content="user-scalable=no,
    width=device-width" />
</head>
```

```
<body>
<h1>Mode d'emploi des balises</h1>
<p> Commençons par le plus <b>important</b> : </p>
<h2>En gras</h2>
<b>Ceci est en gras</b>
<h2>En italique</h2>
<i>Ceci est en italique</i>
<h2>En gras et en italique</h2>
<b><i>Ceci est en gras et en italique</i></b>
</body>
</html>
```

Voici ce qu'on obtient.

Figure 1-3
Utilisation de balises dans la page



Les images

Les images s'affichent par l'intermédiaire de la balise `` qui ne possède pas de contenu. On l'utilise donc sous la forme ``.