

Chapitre 1 :

Les bases de HTML5

I. Introduction au développement web

I.1. L'internet

L'internet est un réseau informatique créé grâce à l'interconnexion de réseaux et d'ordinateurs entre eux. Internet fournit aujourd'hui une multitude de services à travers le monde.

I.2. World Wide Web

Le World Wide Web est communément appelé le Web et abrégé en "WWW" ou "W3" a été inventé en 1989 par Tim Berners-Lee. Le Web actuel est constitué de pages web, physiquement stockées dans des serveurs web structurées en sites web et reliées entre elles par des liens hypertextes. C'est l'un parmi d'autres services offerts par internet (Autres services : courrier électronique, transfert de fichiers, messagerie instantanée,...)

I.3. Fonctionnement du web

Le web fonctionne selon le modèle **client/serveur** qui est régi par le protocole **HTTP** *Hyper Text Transfer Protocol* et s'établit de la manière suivante :

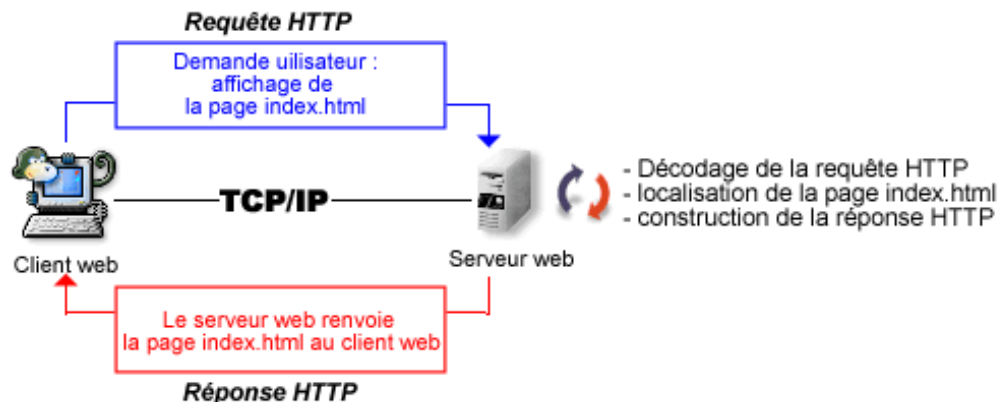


Figure 1: Fonctionnement du Web

- Le navigateur d'un poste client émet une requête http qui va précisément désigner, grâce à l'**URL** qu'elle contient, une page web sur un serveur web ;
- Le serveur traite cette requête et émet une réponse http qui contient la ressource demandée - si le serveur peut la fournir - ou un message d'erreur ;
- Le navigateur interprète la réponse et l'affiche à l'écran.

Quelques définitions

a) HTTP

Le protocole HTTP (HyperText Transfer Protocol), se destine au transfert de données sur Internet, en particulier des pages web au format HTML.

b) Serveur Web

C'est un logiciel permettant à des clients d'accéder à des pages web, à partir d'un navigateur installé sur leur ordinateur distant.

Quelques serveurs Web:

- *Apache* Windows/ UNIX (gratuit) (www.apache.org)
- *Microsoft IIS*(Internet Information Services) :le serveur Web payant de Microsoft
- *Microsoft PWS* (PersonalWeb Server)
- *IplanetWeb Server* : serveur payant de Netscape

c) URL (Uniform Resource Locator)

L'URL est une chaîne de caractères utilisée pour adresser les ressources dans le Web.

Exemple : <http://www.exemple.com/chemin/page.html?q=req>

- **http** : protocole
- **www.exemple.com** : hôte
- **exemple.com** : Nom de domaine
- **/chemin/** : chemin absolu sur le service
- **page.html** : nom de la page Web
- **q=req** : chaîne de requête, transmise à la page

d) Page Web

Une page web est une ressource du World Wide Web :

- créée par des webmasters à l'aide des langages HTML/XHTML, CSS...
- possède une **adresse Web**.
- peut contenir du texte, des images, des tableaux, des formulaires et autres éléments multimédias
- est visualisée par les internautes grâce à des **navigateurs Web**

e) Les navigateurs Web

Le rôle d'un navigateur web est d'analyser le code HTML et CSS des pages web et d'en produire un résultat visuel, facile à lire pour un humain.

Il existe plusieurs navigateurs :

- *Internet Explorer* (Windows) ;
- *MozillaFirefox*(Windows, Mac et Linux) ;
- *Opera*(Windows, Mac et Linux) ;
- *Safari* (Mac,Windows)
- *Maxthon*(Windows);
- *Google Chrome* (Windows, Mac et Linux) ;
- *Konqueror*(Linux), etc.
- *Safari Mobile*: iPhone d'Apple
- *Chrome Mobile*: les portables sous Android

II. Structure d'un document HTML5

Le langage HTML (Hypertext Markup Language) est un langage de publication du World Wide Web. Il offre en plus de la gestion du texte, des éléments multimédia et des hyper liens; des langages de Script, des feuilles de style, une meilleure impression et un meilleur accès aux documents pour les utilisateurs.

Un document HTML a pour extension .html, il est composé de texte et de commandes nommées **balises** (tag en anglais). Ces commandes permettent de mettre en forme le texte (Titre, caractère gras, italique, image, liens, etc...).

II.1. Les versions de HTML

Depuis les premiers jours de l'Internet, il y a eu de nombreuses versions de HTML:

Version	Année
HTML	1991
HTML +	1993
HTML 2.0	1995
HTML 3.2	1997
HTML 4.01	1999
XHTML	2000
HTML5	2012

II.2. Structure d'une page HTML5

```
<!DOCTYPE html> <!--Définit le type de document -->
<html>
<!-- Tout contenu html doit être encadré entre les deux balises html.
En général, on a une balise ouvrante et une balise fermante caractérisée par le "/". -->
<head>
<!-- En-tête de la page : type du document, encodage, titre de la page, auteurs, mots clefs,
indications aux moteurs de recherche, styles etc.-->
</head>
<body>
<!-- Le corps du document structure tout ce que l'on voit dans la fenêtre de son navigateur :
titres, textes, images, menus, barres etc.-->
</body>
</html>
```

II.2.1. La déclaration DOCTYPE

La déclaration DOCTYPE est obligatoire dans tout document. Elle précise le type de page qui va être créé. Dans HTML 5, cette déclaration est désormais réduite à sa plus simple expression :

```
<!DOCTYPE html>
```

Un navigateur ne peut afficher une page HTML correctement à 100% que s'il connaît la version HTML et le type utilisé.

Autres déclarations de DOCTYPE :

- *HTML 4.01*
`<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">`
- *XHTML 1.0*
`<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">`

II.2.2. Les balises et leurs attributs

Les balises s'écrivent en un mot en minuscule de préférence entre `<` et `>` sans aucun espace intérieur et doivent être refermées (symbole `/`). Certaines sont auto-fermantes comme ``, `<meta/>`.

- Elles peuvent avoir un ou plusieurs paramètres appelés **attributs** séparés par des espaces
- Un attribut admet un *nom* avec éventuellement une *valeur* introduite par un signe `=` et notée entre guillemets

`<balise attribut1 attribut2=nombre attribut3="valeur3"...>.....</balise>`

- Si un navigateur ne reconnaît pas une balise, il l'ignore et ne l'interprète pas.

II.2.3. Les commentaires

`<!-- Texte -->`

Le texte du commentaire est ignoré par le navigateur.

Exemple :

`<!-- Voici un commentaire HTML qui peut comporter plusieurs lignes
sans problème -->`

II.2.4. L'élément racine `<html>`

L'élément `<html>` est l'élément racine de la page. Il possède les attributs communs dont les plus utiles sont :

- L'attribut **lang** qui indique la langue utilisée par défaut dans la page. Cette valeur sera reconnue par les moteurs de recherche pour leur permettre d'indexer les pages du site en effectuant un tri par langue. Elles n'apparaîtront dans Google par exemple que si l'utilisateur a choisi le bouton France;

- L'attribut **dir** qui indique le sens de lecture du texte de la page. Il peut prendre les valeurs *ltr* pour le texte qui se lit de gauche à droite (langues européennes) ou *rtl* pour le texte qui se lit de droite à gauche (langues orientales : hébreu, arabe).

Exemple

`<html lang="fr" dir="ltr">
<!--suite des éléments inclus -->
</html>`

En pratique, pour des sites ayant un contenu dans une langue européenne, nous omettrons l'attribut `dir`.

II.2.5. Contenu de l'en-tête (<head></head>)

L'en-tête d'un document html contient les *méta-données* d'informations utiles au bon affichage de la page web.

Ces informations sont contenues dans six éléments différents qui ont chacun un rôle bien déterminé.

Il s'agit des éléments <base />, <link />, <meta />, <script>, <style> et <title>.

Exemple :

```
<html>
<head>
<meta charset="UTF-8"/>
<!-- données spécifiques au référencement de la page ainsi que l'encodage
utilisé -->

<meta name="author" content="Nom de l'auteur"/>
<meta name="keywords" content="Mots clefs relatifs à cette page"/>
<meta name="description" content="Descriptif court"/>

<title>Titre de ma page</title>

<link href="modal.css" rel="stylesheet" />
<!-- utiliser la page "modal.css" comme feuille de style pour ma page -->

<base href="http://www.monsite.com" />
<!-- l'adresse de base de tous les fichiers utilisés dans la page quand leur
adresse est transmise par href -->

<script type="text/javascript">
<!-- Scripts JavaScript -->
</script>

<style type="text/css">
<!-- Styles CSS -->
</style>
</head>
...
```

II.2.6. Contenu du corps d'une page web (<body></body>)

Une page web est une disposition particulière de boîtes, ou conteneurs, pouvant contenir d'autres structures html (tableaux, formulaires, listes etc.).

On distingue deux types importants de structures pour le corps d'un document, les structures de type **bloc** et celles de type **inline**.

➤ Les structures de type **bloc** imposent un saut de ligne et s'étendent au maximum en largeur, les structures de ce type se positionnent verticalement. Une structure de type bloc peut contenir, éventuellement, n'importe quel autre élément, y compris d'autres éléments bloc.

Exemples :

- les paragraphes <p>, <pre>

- les listes ,
- les titres <h1> , ..., <h6> ,
- Les tableaux <table>
- les sections <section>
- Les retraits <blockquote>
- Les divisions <div>
- Les coordonnées <address>
- Les formulaires <form>

➤ Les structures de type **inline**, ne peuvent se mettre que côte à côte. Une structure de type inline ne peut contenir que des éléments du même type.

Exemples :

-
- <a>
- , , <code> , ...

II.2.7. Exemple de page web

```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8"/>
<meta name="author" content="Nom de l'auteur"/>
<meta name="keywords" content="Mots clefs relatifs à cette page"/>
<meta name="description" content="Descriptif court"/>
<title>Titre de ma page</title>
</head>
<body>
<h1>Ma première page web</h1>
</body>
</html>
```

C'est cette page web qui va être modifiée dans la suite. Le but est de tester différents éléments de structure.

III. Les balises de formatage

III.1. Titres

Il existe six niveaux de titres (Heading level 1 à 6), pouvant être utilisées pour hiérarchiser le contenu d'une page web en titres.

Exemple d'organisation :

```
...
<body>
<h1>Partie</h1>
```

```
<h2>Chapitre</h2>
<h3>Section</h3>
<h3>Sous-section</h3>
<h5>Sous-sous-section</h5>
<h6>Sous-sous-sous-section</h6>
<body>
...
```

Résultat de l’affichage :

Partie

Chapitre

Section

Sous-section

Sous-sous-section

Sous-sous-sous-section

III.2. Paragraphes

Pour définir des paragraphes, HTML5 offre les balises suivantes :

- La balise **<p>** (paragraph), qui doit être fermée à la fin du paragraphe avec **</p>**.

L’affichage d’un paragraphe ne peut pas être modifié en ajoutant des espaces ou des sauts de ligne, le navigateur les supprime tous lors de l’affichage.

Exemple

```
<!DOCTYPE html>
<html>
<body>
<h1>Mon premier titre</h1>
<p> Ma      première ligne
Ma deuxième ligne
Ma   troisième ligne </p>
</body>
```

Résultat de l’affichage :

Mon premier titre

Ma première ligne Ma deuxième ligne Ma troisième ligne

- La balise **<pre>** (preformatted Text) définit un texte *préformaté* : Le texte à l’intérieur d’un élément **<pre>** est affiché dans une police à largeur fixe (généralement Courier), et il préserve à la fois les espaces et les sauts de ligne. Il doit être fermée à la fin du paragraphe avec **</pre>**.

Exemple

```
<!DOCTYPE html>
<html>
<body>
<h1>Mon premier titre</h1>
<pre> Ma      première ligne
Ma deuxième ligne
Ma   troisième ligne </pre>
<body>
```

Résultat de l’affichage :

Mon premier titre

Ma première ligne
Ma deuxième ligne
Ma troisième ligne

- La balise **<blockquote>** met en relief un texte qui est cité à partir d’une autre source en provoquant un retrait. L’attribut **cite** indique l’URL de la source de la citation.

Exemple :

```
...
<body>
...
<p>
La balise Blockquote:
<blockquote cite="http://css.mammothland.net/blockquote-retraits-css.php">
L'utilisation de l'élément BLOCKQUOTE pour obtenir le retrait d'un
texte est <br/>
déconseillée en faveur de l'utilisation de feuilles de style.
</blockquote>
</p>
</body>
```

Résultat de l’affichage :

La balise Blockquote:

L'utilisation de l'élément BLOCKQUOTE pour obtenir le retrait d'un texte est
déconseillée en faveur de l'utilisation de feuilles de style.

III.3. Sauter une ligne

HTML 5 offre :

- La balise **
** (line break) qui est une balise orpheline et sert juste à aller à la ligne. Elle doit obligatoirement être mise à l’intérieur d’un paragraphe.
- La balise **<wbr />** (word break opportunity) qui permet au navigateur (mais ne l’y oblige pas) d’effectuer un saut de ligne à l’endroit où elle apparaît : c’est le cas lorsqu’un mot est généralement assez long afin de mieux répartir le texte sur la ligne

Exemple :

```
...  
<body>  
...  
Bonjour, <br /> Bienvenue  
Exemple de texte avec anticonstitu  
tionnellement utilisé pour  
tester le retour à la ligne  
</body>
```

Résultat de l’affichage :

Bonjour,
Bienvenue Exemple de texte avec anticonstitu
tionnellement utilisé pour tester le retour à la ligne

III.4. La mise en valeur

Plusieurs balises sont utilisées pour mettre en valeur des parties d’un texte

Balise	Rôle
(emphase)	Mettre le texte en italique
	Mettre le texte en gras
<mark>	Surligner le texte
<sup> (superscript)	Affiche en exposant le texte qu'elle contient
<sub> (subscript)	Affiche en indice le texte qu'elle contient
<ins> (inserted Text)	Souligner le texte
 (deleted text)	Indique que le texte a été supprimé du document. Cet élément est souvent (mais pas nécessairement) affiché en étant rayé

Exemple :

```
<!DOCTYPE html>  
<html>  
<head> <meta charset ="utf-8" />  
<title> Balises de mise en valeur </title>  
</head>  
  
<body>  
<pre>  
  
Le texte qui suit <em>est écrit en italique</em>  
<strong>Ce Texte est encadré par la balise strong</strong>  
Le texte qui suit est inclus <mark>dans la balise mark</mark>.  
Son affichage est modifié avec un effet surligné jaune.  
Ma couleur favorite est le <del>rouge</del><ins>bleu </ins>
```

```
A<sup>2</sup> = X<sub>1</sub> + X<sub>2</sub><sup>3</sup>
</pre>
</body>
</html>
```

Résultat de l’affichage :

Le texte qui suit *est écrit en italique*

Ce Texte est encadré par la balise strong

Le texte qui suit est inclus dans la balise mark.

Son affichage est modifié avec un effet surligné jaune.

Ma couleur favorite est le ~~rouge~~bleu

$$A^2 = X_1 + X_2^3$$