

# Chapitre 01

# Créer et structurer une page web

## Objectifs du chapitre :

- Fonctionnement des sites Web
- Créer et structurer le caneva d'une page HTML5 de base bien formée et validée par une DTD
- Utilisez les balises HTML spécifiques et leurs attributs :
  - Les balises de structuration du document,
  - Les commentaires
  - Les balises de structuration et de hiérarchisation des contenus,
  - Les balises de contenus textuels,
  - Les balises de contenus multimédia externes (images),
  - Les hyperliens
- L'importance de la sémantique lors du choix des balises

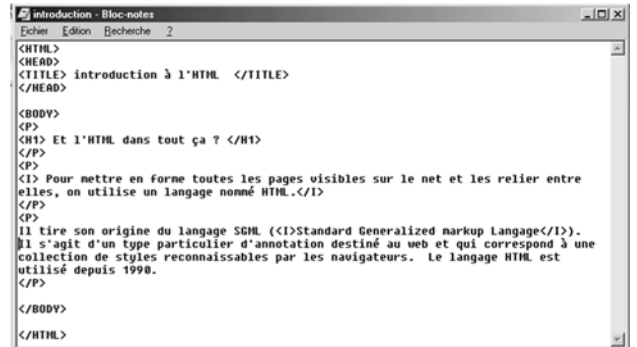
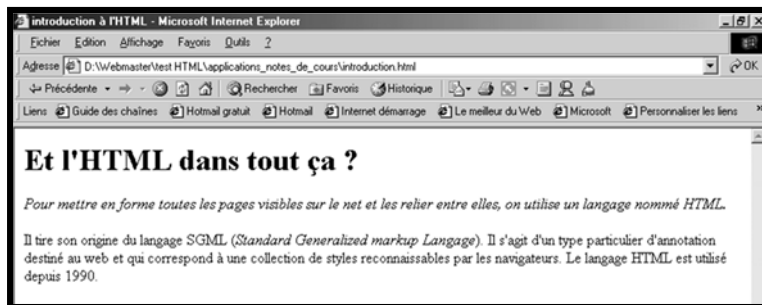
# 1. FONCTIONNEMENT DES SITES WEB ...

## 1.1. Les bases

- De nombreuses personnes confondent (à tort) Internet et le Web. Il faut savoir que **le Web fait partie d'Internet**. Internet est donc un grand ensemble qui comprend, entre autres : le *Web*, les *e-mails*, la *messagerie instantanée*, etc.
- Pour consulter les sites web**, nous lançons un un programme appelé **navigateur**<sup>1</sup>. Celui-ci se base sur le code HTML et CSS des pages pour savoir ce qu'il doit afficher.
- Les langages informatiques HTML et CSS permettent de créer des sites web**, ils sont incontournables et universels car tous les sites web sont basés sur ces langages.  
**Le langage HTML a été inventé par un certain Tim Berners-Lee en 1991...** Tim Berners-Lee suit encore aujourd'hui avec attention l'évolution du Web. Il a créé le World Wide Web Consortium (**W3C**) qui définit les nouvelles versions des langages. Il a par ailleurs créé plus récemment la World Wide Web Foundation, qui analyse et suit l'évolution du Web.

## 1.2. Page Web et source HTML

- Qu'appelle-t-on « page web » ? Les pages web sont des documents informatiques qui peuvent contenir du *texte*, des *images*, des *tableaux*, des *formulaire*s et *d'autres éléments multimédias*.
- Elles sont créées par les webmasters à l'aide des langages HTML et CSS, et sont visualisées par les internautes grâce à un navigateur web.**



## 1.3. Pourquoi deux langages ?

- HTML<sup>2</sup> (HyperText Markup Language)** : a fait son apparition dès **1991** lors du lancement du Web. Son rôle est de **gérer et organiser le contenu**. C'est donc en HTML que vous écrirez ce que vous souhaitez que la page affiche : du texte, des liens, des images...
- CSS (Cascading Style Sheets, aussi appelées Feuilles de style)** : le rôle du CSS est de **gérer l'apparence** de la page web (agencement, positionnement, décoration, couleur, taille du texte...). Ce langage est venu compléter le HTML en **1996**.
- Nous comprendrons mieux à l'aide du site <http://www.csszengarden.com>

<sup>1</sup> L'installation et la configuration des navigateurs principaux fait partie du programme du cours d'Environnement et Technologie Web (BES 1<sup>ère</sup> année)

<sup>2</sup> **Un peu d'histoire ...** Le HTML est apparu le premier en 1991. Début 2000, le W3C a lancé le XHTML en indiquant que ce serait l'avenir... le langage XHTML, variante du HTML est plus rigoureux et donc plus délicat à manipuler. Le XHTML n'a pas percé comme espéré, retour aux sources en 2009 : le W3C abandonne le XHTML et décide de revenir au HTML pour le faire évoluer. Ces langages se ressemblent beaucoup, aucun n'est vraiment meilleur que l'autre, il s'agit de deux façons de faire différentes. Dans ce cours, nous allons travailler sur la dernière version de HTML (HTML5) qui est aujourd'hui le langage d'avenir que tout le monde est incité à utiliser. Le détail des différentes versions des langages HTML et CSS fait partie du programme du cours d'Environnement et Technologie Web (BES 1<sup>ère</sup> année)

## 1.4 Outils d'édition HTML<sup>3</sup>

### Vous avez besoin ...

D'un éditeur de texte simple Ex : Bloc-notes ou Notepad de Windows ou équivalent  
D'un Browser ou Navigateur Ex : Chrome, Mozilla Firefox, Safari, Internet Explorer ...

### Vous n'avez pas besoin ...

D'être connecté pour écrire ou peaufiner vos pages.

**Il existe cependant des logiciels plus pratiques que le Bloc-Notes, on peut classer ces logiciels de création de site web en deux catégories :**

**Les éditeurs WYSIWIG** (*What You See Is What You Ghet*), les plus simples en apparence, permettent de rédiger le contenu de votre site directement sans avoir à taper la moindre ligne de HTML ou de CSS car il est automatiquement généré.

**Exemples d'éditeurs WYSIWIG** : Dreamweaver, Kompozer, ...

La qualité du code HTML et CSS produit par ces éditeurs est rarement bonne et le site est parfois plus difficile à maintenir. De plus, aucun d'entre eux ne permet d'obtenir toute la souplesse et les possibilités que l'on a en rédigeant soi-même le code HTML/CSS de ses pages.

Pour arriver à utiliser de façon réellement efficace et professionnelle ces différents outils, **la connaissance de l'HTML et du CSS est indispensable.**

Cela vous évite de tomber dans le piège d'une trop grande dépendance par rapport à un outil d'édition **et est indispensable lors de l'intégration future des autres techniques servant à l'édition web** comme le javascript, le PHP, le XML, ...

La solution des webmasters : **les éditeurs SYNTHAXIQUES**

Ce sont des programmes dédiés à l'écriture de code, ces éditeurs spécifiquement conçus ne sont pas indispensables mais accélèrent et facilitent simplement le processus d'application des balises. De plus, on peut en général les utiliser pour de multiples langages, pas uniquement HTML et CSS

**N'utilisez pas Word ou un autre traitement de texte** car ces logiciels ne sont pas adaptés à la manipulation de texte « Brut ».

**Exemples d'éditeurs syntaxiques** : Notepad ++, jEdit, Topstyle, HTML Edit, PSpad, TextWrangler (Mac)...

### Plus Web !

*Brackets, l'éditeur spécialisé front-end à découvrir absolument !*

<http://www.alsacreations.com/outils/lire/1654-brackets-editeur-front-end.html>

<http://samuelbourdon.com/brackets-un-passage-en-douceur/>

<http://www.partage-it.com/selection-extensions-preferees-pour-brackets/>

---

<sup>3</sup> La constitution d'une boîte à outils (logiciels téléchargeables et en ligne) fait partie du programme du cours d'Environnement et Technologie Web (BES 1<sup>ère</sup> année)

## 2. REDIGER SA 1<sup>ERE</sup> PAGE HTML

---

### 2.1. Structure de base d'une page HTML

#### Une page web est constituée de balises.

En HTML, les balises s'écrivent entièrement **en minuscules** (le HTML impose l'utilisation de minuscules pour des raisons d'uniformisation), **sans espace ni accent**.

Une balise est une information spéciale s'adressant au navigateur. Elle n'est jamais affichée à vos visiteurs, mais elle est en revanche lue et « comprise » par le navigateur. Une balise commence par un chevron ouvrant < et se termine par un chevron fermant >.

Une balise permet de donner une indication à l'ordinateur sur la nature du texte, chaque balise donne un sens précis au texte qu'elle délimite. Par exemple, la balise <title> indique que ce qui suit correspond à un titre.

Vous devez choisir les balises que vous utilisez en fonction de leur signification et non pas en fonction de l'apparence qu'elles donnent au texte. Souvenez-vous que l'apparence peut être modifiée grâce au langage CSS.

#### Il existe deux types de balises :

*Les balises en paires* : ce sont celles que vous rencontrerez le plus souvent. On écrit une première fois la balise, on tape du texte, puis on referme la balise après le texte en récrivant le nom de la balise précédé d'un slash (/).

Par exemple : <title>Le titre de ma page</title>

Ce codage signifie que tout ce qui se trouve entre <title> et </title> est le titre de la page.

Ces balises permettent donc de délimiter une partie de votre texte pour dire à quoi il correspond.

*Les balises auto-fermantes* (ou monoatomiques) : Elles ne délimitent pas de texte, elles servent à insérer une information à un endroit précis<sup>4</sup>.

#### Les attributs

Que la balise soit de type «paire» ou «unique», elle peut prendre en plus ce qu'on appelle des attributs. Le rôle des attributs est de compléter une balise pour donner des informations supplémentaires.

Un attribut s'écrit **obligatoirement en lettres minuscules et ne comporte pas d'espace**, tout comme le nom de la balise.

Il est **immédiatement suivi du signe égal « = » puis de guillemets qui entourent la valeur de l'attribut**<sup>5</sup>. La valeur de l'attribut n'est en revanche pas soumise aux mêmes règles : elle peut contenir des majuscules et des espaces sans problème.

### 2.2. Le code source minimal d'une page

Une page HTML doit comporter un minimum de code pour être « correcte ». Voici le code HTML que toute page doit obligatoirement comporter :

```
<!doctype html>
  <html lang="fr">
  <head>
    <meta charset="utf-8">
    <title>Titre du document</title>
  </head>
  <body>

  </body>
</html>
```

---

<sup>4</sup> Une telle balise devait, pour respecter les normes XHTML, se terminer par un / avant le chevron fermant (exemple : <image />), ce n'est plus le cas depuis le HTML5

<sup>5</sup> Notons que depuis l'HTML5, les guillemets ne sont plus obligatoires.

## Doctype :

```
<!doctype html>
```

Il existe plusieurs versions du langage HTML cohabitant sur le WEB. On doit donc indiquer au navigateur la version retenue pour un document donné en le préfaçant d'un **doctype**. Le doctype est une sorte de grammaire destinée aux navigateurs. C'est non seulement un moyen de **s'assurer que le document sera bien interprété par les navigateurs**, mais aussi un **élément indispensable à sa validation** par l'organisme de référence, le W3C. **Utiliser un DOCTYPE incomplet ou désapprouvé —voire aucun DOCTYPE— déclenche dans ces navigateurs le mode "Quirks"** ("habitudes bizarres") qui va considérer votre balisage comme invalide et démodé, conforme seulement aux normes de la fin des années 90.

Une déclaration de type de document doit apparaître dans le document juste avant l'élément html (qui est l'élément racine de tout document HTML).

**Depuis HTML 5**, un seul DOCTYPE est prévu dont la syntaxe est simplifiée par rapport à celle d'un DOCTYPE classique. Les DOCTYPES des versions plus anciennes de HTML sont plus longues car basées sur le SGML et, par conséquent, exigent une référence à une DTD. Avec HTML 5, ce n'est plus le cas et le DOCTYPE n'est nécessaire que pour activer le mode standard pour les documents écrits selon la syntaxe HTML.

### Lecture Web :

La différence entre chaque doctype est résumée dans un article en ligne de Laurent Denis :

<http://www.alsacreations.com/article/lire/560-dtd-doctype-html-xhtml-comment-choisir.html>

## En-tête et spécification de langue et d'encodage

Les normes du web imposent également d'autres déclarations obligatoires dans l'en-tête de tous documents HTML : **les spécifications de langues et d'encodage**.

```
<html lang="fr">
```

La balise **<html>** permettant d'identifier un document comme une page web. Elle renferme la spécification de la langue du document.

```
<head> et </head>
```

Les balises **<head>** et **</head>** sont des balises d'en-tête renfermant toutes les informations générales sur la page Web, comme son titre, l'encodage (pour la gestion des caractères spéciaux), etc. Les informations que l'en-tête contient ne sont pas affichées sur la page.

```
<meta charset="utf-8">
```

L'encodage est spécifié entre les **<head>**, à l'aide de l'attribut **charset**. Il indique aux navigateurs les caractères potentiellement utilisés dans le texte de la page (la façon dont le fichier est enregistré). C'est lui qui détermine comment les caractères spéciaux vont s'afficher (accents, idéogrammes chinois et japonais, symboles arabes, etc.).

**utf-8<sup>6</sup>** permet d'utiliser la plupart des caractères de la majorité des langues du monde, il est donc intéressant de l'employer<sup>7</sup>. Attention ! Il faut également que votre fichier soit bien enregistré en UTF-8. C'est le cas le plus souvent sous Linux par défaut, mais sous Windows il faut généralement le dire au logiciel.

### Problème d'affichage des accents sur sa page web ?

*C'est qu'il y a un problème avec l'encodage. Vérifiez que la balise indique bien UTF-8 et que votre fichier est enregistré en UTF-8.*

Sous Notepad++, allez dans le menu **Encodage > Encoder en UTF-8 (sans BOM<sup>8</sup>)** pour que votre fichier soit enregistré en UTF-8 dès le début. Cela ne s'applique qu'au fichier actuellement ouvert.

Pour ne pas avoir à le faire pour chaque nouveau fichier : menu **Paramétrage > Préférences**, onglet **Nouveau document/Dossier**. Sélectionnez **UTF-8 sans BOM** dans la liste.

<sup>6</sup> Pour vous rendre compte des immenses possibilités du jeu de caractères UTF-8 : <http://www.columbia.edu/~fdc/utf8/>

<sup>7</sup> Pour un document en langue française, **iso-8859-1** (presque complet pour les langues de l'Europe occidentale) ou **iso-8859-15** (complète le précédent avec quelques caractères supplémentaires tels que €) peuvent être employés.

<sup>8</sup> Le BOM (Byte Order Mark) est un caractère espace insécable de largeur nulle, l'utiliser permet d'éviter l'apparition de caractères invisibles à traiter : <http://www.prelude.me/index.php/2011/01/15/utf-8-avec-ou-sans-bom/>

```
<title> Mon Titre </title>
```

Les balises `<title>` et `</title>` sont des balises de titre : le titre d'une page web est affiché dans la barre de titre du navigateur et dans son onglet. Toute page doit avoir un titre qui décrit ce qu'elle contient. Il est conseillé que le titre soit assez court (moins de 100 caractères en général).

Choisissez-le avec soin car il a beaucoup d'importance pour les moteurs de recherche qui référencent les pages web, comme Google.

## Le contenu de la page

```
<body> et </body>
```

Les balises `<body>` et `</body>` sont des balises pour le corps du document : elles doivent encadrer le contenu de la page Web, tout ce qui se trouve à l'intérieur de cette balise sera affiché à l'écran.

```
</html>
```

Signifie la fin de la page web.

*Cette formulation est précise et rigoureuse (elle mêle notamment des balises et des parties en majuscules). Pour éviter d'éventuelles erreurs, contentez-vous d'un copier-coller dans vos créations.*

### 2.3. Ajouter un commentaire

Une balise de commentaire n'est visible que si on regarde le code source de la page web que l'on visite. C'est une information généralement destinée au webmaster qui peut servir de repère ou d'aide-mémoire pour ne pas oublier de coder quelque chose.

```
<!-- Ajouter ici les commentaires -->
```

### 2.4. Présentation du code source

- Le code source HTML ne prend pas en compte vos retraits, sauts de ligne ou tabulations. Pour des raisons de lisibilité, **aérez et indentez votre code source autant que possible.**
- **Ajoutez des commentaires** de manière à vous souvenir de l'utilité d'une balise plutôt qu'une autre ou détailler votre code à destination d'un collègue.

## 3. METHODES POUR CREER UN DOCUMENT HTML [NOTIONS DE BASE]

### 3.1. Créer le document

**Etape 1.** Ouvrir l'éditeur de texte (Bloc-notes, Notepad ++)

**Etape 2.** Ecrire le document HTML avec les balises nécessaires

**Etape 3.** Enregistrer le document ...

- a) En veillant à ne pas mettre d'espace, d'accent ou de caractères spéciaux dans le nom de fichier
- b) En rajoutant l'extension.html (ou .htm) derrière le nom du fichier
- c) En veillant à sauvegarder dans un « dossier racine de site » (répertoire qui contiendra tous les éléments de votre site Web).  
Hiérarchisez vos contenus à l'aide de dossiers et sous-dossiers sous peine très vite de ne plus vous y retrouver !

### 3.2. Visualiser le document

**Etape 4.** Double-cliquez sur le fichier et votre navigateur ouvrira la page web

OU

**Etape 5.** Démarrer le navigateur et afficher le document dans votre navigateur via le menu **Ouvrir** de celui-ci ou double-cliquez sur la page HTML à partir de l'explorateur.

**Note :** La majorité des éditeurs (wysiwig et synthaxiques) possèdent une fonction lançant immédiatement la visualisation de vos pages web

## Les différentes versions de navigateurs

- Les navigateurs n'affichent pas toujours les sites web exactement de la même façon *car ils ne connaissent pas toujours les dernières fonctionnalités de HTML et CSS* (Internet Explorer a longtemps été en retard sur certaines fonctionnalités CSS et paradoxalement a aussi été en avance sur quelques autres).
- De plus, plusieurs versions des navigateurs co-existent et chaque version prend en charge de nouvelles fonctionnalités, ainsi, si les utilisateurs ne mettent pas à jour leurs navigateurs cela pose un problème.
- Il est conseillé d'installer plusieurs navigateurs sur son ordinateur et de tester son site web régulièrement au moins sur Google Chrome, Mozilla Firefox, Internet Explorer et Safari. Notez que Safari et Google Chrome affichent les sites web quasiment de la même façon mais c'est toujours plus sûr de vérifier.

## Les navigateurs mobiles

En plus des navigateurs classiques, existent les navigateurs pour smartphone (la plupart des navigateurs sur smartphones sont les mêmes que sur ordinateur, dans une version plus légère adaptée aux mobiles) utilisés par de plus en plus de personnes aujourd'hui, ils sont donc à prendre en compte également. Les navigateurs pour mobiles supportent la plupart des dernières fonctionnalités de HTML et CSS. De plus, le système de mise à jour automatisé des mobiles nous garantit que les utilisateurs auront le plus souvent les dernières versions. Les tablettes tactiles sont équipées des mêmes navigateurs, l'écran étant simplement plus large. Ainsi, l'iPad est équipé de Safari Mobile par défaut mais supporte également Chrome ou d'autres navigateurs du marché, comme les tablettes Android.

### 3.3. Récupérer le code source pour le modifier

Par défaut, lorsque vous double-cliquez sur votre page html, c'est votre navigateur web par défaut qui affiche la page. Il n'est pas possible de la modifier par ce biais.

1. Clic droit sur la page HTML / **Ouvrir avec...**
2. Sélectionnez le programme que vous employez pour l'édition de votre code source

### 3.4. Un dossier par site

Vous allez être amené à créer plusieurs fichiers .html pour créer votre site, sans parler des fichiers images. Il est donc recommandé de **créer un nouveau dossier pour chaque nouveau site** que vous commencez.

### 3.5. Afficher les extensions des fichiers

Par défaut, Windows masque les extensions des fichiers. Pour afficher les extensions :

Windows < 8 :

1. Menu **Outils (Organiser sous Windows<7) /Options des dossiers** de l'explorateur, onglet **Affichage**
2. Décochez la case « **Masquer les extensions des fichiers dont le type est connu** ».

Sous Windows 8 :

1. Ouvrez l'Explorateur (pour cela ouvrez un dossier ou Ordinateur par exemple).
2. Choisissez l'onglet "Affichage" et Cochez "Extensions de noms de fichiers"

### 3.6. Modifier un document

1. Afficher à nouveau l'éditeur de texte (sans fermer le navigateur), **prenez l'habitude de travailler avec les deux fenêtres ouvertes l'une sur l'autre** et de passer de l'une à l'autre via la barre des tâches de Windows ou le raccourci clavier ALT+Tab.
2. Modifiez les codes HTML
3. Enregistrez vos modifications dans l'éditeur de texte
4. Affichez à nouveau votre navigateur et cliquez sur le bouton **Actualiser (F5) ou Recharger**

### 3.7. Voir le code source des pages que vous visitez

Quand vous naviguez sur le Web, regarder le code source des sites que l'on visite est parfois un bon moyen d'apprendre de nouvelles choses.

a) **Affichage/ Code source** de la page ou **Affichage/Source**.

Notez qu'il est impossible à un webmaster de masquer son code HTML, celui-ci est toujours visible.

### 3.8. En local

Les fichiers de votre site seront pour le moment stockés sur le disque dur de votre ordinateur, on dit qu'on travaille « en local ».

Pour que tous les internautes du monde entier puissent visiter votre site, il faudra envoyer vos fichiers chez un hébergeur web à l'aide d'un logiciel de transfert FTP (ex :FileZilla).

### 3.9. HTML5, CSS3... peut-on tout utiliser dès maintenant ?

Le W3C est toujours en train de travailler sur ses spécificités et de nouveaux changements peuvent encore survenir. Les navigateurs implémentent de plus en plus ces nouveautés mais pas toujours dans le même sens.

Mais même si certaines balises ne sont pas prises en charge et qu'elles ne passent pas partout, il y a toujours moyen d'obtenir des pages web qui fonctionnent, même si leur aspect diffère parfois d'un navigateur à l'autre. **Le fait que les navigateurs plus anciens ne voient pas le contenu de la même manière qu'un navigateur moderne ne doit pas être un frein.**

Vérifiez régulièrement l'avancée de ces compatibilités à l'aide de sites de références comme <http://caniuse.com/>.



## 4. BALISER LE TEXTE DE SA PAGE

### Attention !

Toutes les balises ci-après s'intégreront entre une ouverture de `<body>` et une fermeture de `</body>`

### 4.1. Créer des paragraphes

Le langage HTML ne respecte pas les sauts de ligne que vous pourriez taper, c'est la balise `<p>` qui permet de structurer vos paragraphes.

`<p>` Il existe actuellement des éditeurs spécifiquement conçus pour l'encodage de fichiers HTML ; ils ne sont pas indispensables mais accélèrent et facilitent simplement le processus d'application des balises . `</p>`

Le navigateur sépare chaque paragraphe d'une ligne vide

```
<p>
<br /> Exemples d'éditeurs WYSIWIG :
<br /> Dreamweaver
<br /> Frontpage
</p>
```

La balise `<br />`<sup>9</sup> (pour break), balise auto-fermante, sert simplement à indiquer qu'on insère un retour à la ligne à un endroit. La balise `<br />` est également utilisée pour passer des lignes vides.

### ATTENTION N'abusez pas des `<br />`

Pour décaler le texte et le placer à un endroit précis sur la page, nous utiliserons le langage CSS.

### Placement des balises `<br />`

Vous devez impérativement placer la balise `<br />` à l'intérieur d'une balise de paragraphe `<p></p>`

Si vous ne faites pas cela, aucune erreur ne s'affichera mais votre page sera considérée comme invalide. Une page invalide est une page qui ne respecte pas les règles du langage HTML, sa pérennité risque alors d'être affectée. Tout webmaster un tant soit peu consciencieux doit faire l'effort de respecter ces règles.

### 4.2. Créer des titres

Dans une page web correctement organisée, on trouve des titres, des sous-titres, des sous-sous-titres.

```
<h1>Ceci est un titre de niveau 1</h1>    Plus grand
<h2>Ceci est un titre de niveau 2</h2>
<h3>Ceci est un titre de niveau 3</h3>
<h4>Ceci est un titre de niveau 4</h4>
<h5>Ceci est un titre de niveau 5</h5>
<h6>Ceci est un titre de niveau 6</h6>    Plus petit
```

### Il faut choisir la balise en fonction de l'importance du titre

Lorsqu'on débute, on a tendance à choisir sa balise de titre en fonction de la taille dans lequel le texte est écrit. C'est une erreur, il faut choisir la balise en fonction du « niveau » du titre, c'est-à-dire de son importance car pour modifier la taille du texte c'est le langage CSS qu'il faut utiliser.

### Placement des balises de titre

Les balises de titres doivent être placées à l'extérieur des balises de paragraphe.

<sup>9</sup> Pour rappel, le "/" en fin de balise n'est plus obligatoire en HTML5



## Insérer des caractères spéciaux

```
<p> &#169 ; 2000 Marc Harmin </p>
```

Ce code permet d'obtenir le sigle copyright

### Une liste des différents caractères spéciaux :

<http://www.commentcamarche.net/contents/html/htmlcarac.php3>

<http://www.ccim.be/ccim328/html/speciaux.htm>

#### Remarque :

Nous verrons, lors de l'intégration du code **CSS**, que d'avantage de mises en forme sont possibles.

**C'est volontairement que nous n'utilisons plus les propriétés des balises HTML car celles-ci seront spécifiées dans le code CSS afin de simplifier et d'alléger le code HTML et de respecter la règle de base en web – LA SÉPARATION DU CONTENU ET DE LA MISE EN FORME.**

## 4.4. L'importance de la sémantique

Il est fondamental d'utiliser les bonnes balises au bon endroit, **de mettre en place un balisage qui a du sens.**

Par exemple : <h1> pour le titre principal, <ul> et <li> pour les listes de liens (menus), <blockquote> et <q> pour les citations, ... exemple :

```
<h1>Titre de la page</h1>
<h2>Titre du paragraphe</h2>
Mon Texte.
<ul>
  <li>Idée 1</li>
  <li>Idée 2</li>
</ul>
```

Les balises <i> et <b> n'apportant aucun sens (sémantique) au document contrairement aux éléments <strong> et <em>, seront employées dont le seul but est de créer une mise en forme purement visuelle. Les éléments <strong> et <em> qui sont des **mises en exergue** seront employées en fonction de la force que l'on veut conférer au texte déterminé, ces balises ont un impact direct sur le **référencement**.

### Le respect de la sémantique comporte divers avantages :

- Votre code est nettement plus propre et structuré
- Les moteurs de recherches et les navigateurs analyseront beaucoup plus facilement vos pages, votre contenu sera donc mieux référencé
- La hiérarchie des éléments de votre page est respectée même quand la feuille de style CSS est désactivée
- L'interprétation de la page est facilitée pour les lecteurs braille (qui ne perçoivent pas la présentation réelle)

|| Lecture Web : [http://fr.wikipedia.org/wiki/Web\\_s%C3%A9mantique](http://fr.wikipedia.org/wiki/Web_s%C3%A9mantique)

## 5. CREER UNE LISTE A PUCE OU NUMEROTEE

Les listes à puces sont importantes dans les sites web car elles permettent de structurer les informations, mais participent aussi à la création des menus.

Il existe trois types différents de listes à puces : les listes non ordonnées (listes à puces), les listes ordonnées (chaque élément de la liste commence par un numéro (1, 2, 3) ou une lettre (A, B, C) et les listes de définitions (plus rares, elles permettent par exemple de créer un lexique).

### 5.1. Créer une liste à puces

Une liste **non ordonnée** se place à l'intérieur des balises `<ul></ul>`. **Ces balises doivent être situées à l'extérieur des paragraphes.** Chaque `<li></li>` représente une puce.

```
<ul>
<li> élément 1 </li>
<li> élément 2 </li>
<br> élément 2 bis </li>
<li> élément 3 </li>
</ul>
```

### 5.2. Créer une liste numérotée

Une **liste ordonnée** se construit exactement de la même manière qu'une liste non ordonnée. Seule la balise `<ul>` change : elle est remplacée par `<ol>`.

```
<ol>
<li> élément 1 </li>
<li> élément 2 </li>
<br> élément 2 bis </li>
<li> élément 3 </li>
</ol>
```

### 5.3. Changer le numéro initial

```
<ol start="4">
<li> élément 1 </li>
<li> élément 2 </li>
<br> élément 2 bis </li>
<li> élément 3 </li>
</ol>
```

*Il est possible de changer le type de numérotation ou de puce, pour ce faire nous ferons appel au CSS.*

### 5.4. Créer une liste imbriquée

Il est possible d'imbriquer une liste à puces dans une autre, il suffit d'ouvrir une nouvelle balise `<ul>` à l'intérieur d'une puce `<li>`.

```
<ol type="a">
<li> introduction
  <ol type="a">
    <li> qui sommes-nous ? </li>
    <li> historique de notre société </li>
    <li> pourquoi nous choisir ? </li>
  </ol>
</li>
<li> notre rôle</li>
```

```
<li> produits commercialisés </li>  
</ol>
```

## 5.5. Créer une citation

On utilise la balise `<dl>` pour délimiter le début et la fin d'une liste de définitions.

Une liste de définitions est une succession de mots suivis de leur définition, on alterne :

`<dt>` : indique le mot que l'on veut définir.

`<dd>` : délimite la définition de ce mot.

```
<p>Vocabulaire du parfait petit webmaster :</p>  
<dl>  
<dt>HTML</dt>  
<dd>Langage de balisage permettant la rédaction de pages web structurées.</dd>  
<dt>CSS</dt>  
<dd>Langage de description utilisé pour la mise en forme des pages web.</dd>  
</dl>
```

*Ces balises pourront être redécorées plus tard à l'aide du langage CSS.*

## 6. LES LIENS

---

Dans la pratique votre site sera certainement composé de plusieurs pages. Pour que vos visiteurs puissent naviguer entre elles, nous allons apprendre à créer des liens.

### 6.1. Créer un lien vers une de ses propres pages

La balise `<a>` pour « ancre » à laquelle s'ajoute l'attribut `href` permet d'indiquer le nom de la page qui sera appelée lorsqu'on cliquera sur ce lien.

```
<p><a href="index.html"> Retour à l'accueil </a></p>
```

Lorsque vous cliquez sur un lien, la nouvelle page s'affiche dans la même fenêtre du navigateur.

Il est d'usage de créer une liste de liens dans un `<ul>` et `<li>`, c'est comme cela que l'on crée des menus de navigation :

```
<ul>  
  <li><a href="index.html"> Retour à l'accueil </a></li>  
  <li><a href="pages/page02.html"> Page 02 </a></li>  
  <li><a href="pages/page03.html"> Page 03 </a></li>  
</ul>
```

### Des liens qui changent de couleur

Par défaut, les liens se colorent en bleu et après avoir été cliqués, se colorent en violet. C'est le navigateur qui a changé sa couleur permettant ainsi au visiteur de savoir que c'est un lien qu'il a déjà visité. Il est possible de changer ces couleurs peu élégantes à l'aide des feuilles de style CSS.

### 6.2. Insérer un lien vers une autre page web

```
<p><a href="http://www.google.be"> Vers le site de Google </a>
```

Le navigateur affichera le lien Vers le site de Google et tout lecteur pourra cliquer sur ce lien pour atteindre la page, le protocole est obligatoire !

### 6.3. Ouvrir un lien dans une nouvelle fenêtre

```
<a href="http://www.google.be" target="_blank"> Vers le site de Google </a>
```

Si vous souhaitez utiliser cet attribut, pour que votre site soit valide, le doctype utilisé dans votre page doit impérativement être HTML5.

### 6.4. Ajout d'une infobulle

Cette bulle est un petit message qui s'affiche lorsque vous pointez avec la souris sur le lien.

Pour créer une bulle d'aide sur un lien, ajoutez à la balise `<a>` un attribut `title`.

```
<a href="livreor.html" title="Signez mon livre !"> Livre d'or </a>
```

Le texte de la bulle d'aide doit être assez court

## 6.5. Créer un lien au sein d'une même page Web

### 1. Nommer la partie de page à atteindre

Pour créer une ancre, il suffit de rajouter un attribut `id` à n'importe quelle balise, on mémorise ainsi un emplacement sur la page en cours.

```
<h2 id="titre01">Titre 01</h2>  
texte texte texte texte texte (...)  
<h2 id="titre02">Titre 02</h2>  
texte texte texte texte texte (...)
```

Attention : Les identifiants ne peuvent contenir que des lettres ou des chiffres (ne débutez jamais par un chiffre) et ils doivent être uniques (il ne peut pas y avoir deux ancres avec le même nom dans une page).

Pour que le lien vers l'ancre fonctionne, il faut que votre page web contienne suffisamment de texte.

### 2. Créer un lien vers cette partie de page

Lorsque l'ancre est créée, utilisez la balise `<a>` avec l'attribut `href="#nomid"`. C'est-à-dire le nom de l'ancre précédé d'un dièse (#).

```
<a href="#titre01">étape 1 : préparation </a>  
<a href="#titre02">étape 2 : fabrication </a>
```

### Pour créer un lien vers une zone d'une autre page Web

```
<a href="page.html#partie1">allez consulter la partie 1</a>
```

## 6.6. Créer un lien vers une messagerie électronique

```
<a href="mailto:marcdifo@musicclassic.fr">faites-le moi savoir par courrier électronique</a>
```

## 6.7. Créer un lien vers un fichier

```
<a href="testpp.ppt">cliquez ici pour découvrir power point </a>
```

## 6.8. L'attribut `download` (HTML5)

L'attribut `download` de HTML5 permet de forcer le téléchargement d'une ressource au lieu de la faire s'afficher par le navigateur, avant l'apparition de cet attribut il était nécessaire de passer par une solution PHP ou `.htaccess` pour forcer le téléchargement d'une ressource (ou autre solution serveur).

```
<a href="logo-societe.png" download="logo">Télécharger l'image</a>
```

Cet attribut permet de spécifier au navigateur qu'il ne doit pas se diriger vers la ressource ciblée, mais la télécharger.

## 6.9. Créer un lien vers un site FTP

```
<a href="ftp://ftp.loc.gov">site ftp</a>
```

## 6.10. Les liens relatifs et absolus

Dans le cas d'un **lien relatif**, on ne connaît pas l'adresse complète de la page vers laquelle on amène, on cible la page par rapport à celle qui contient le lien.

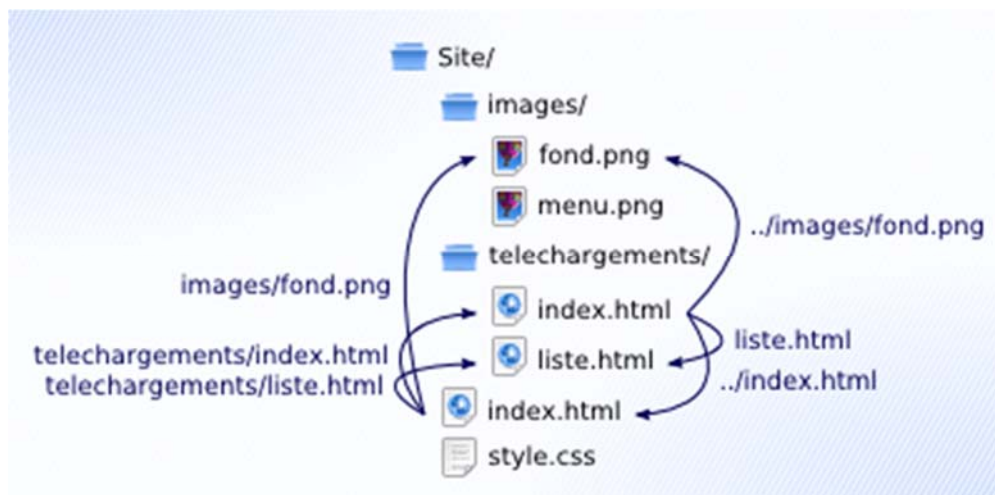
**S'il y a plusieurs dossiers imbriqués :**

```
<a href="autrepage.html">Lien</a>  lorsque les 2 pages se trouvent dans le même dossier  
<a href="dossier/page2.html">Lien</a>  lorsque la page se trouve dans un sous-dossier  
<a href="dossier1/dossier2/page2.html">Lien</a>  lorsque la page se trouve dans un sous-  
sous-dossier
```

En revanche, si votre fichier `page2.html` se trouve dans le dossier situé avant `page1.html` dans l'arborescence (et donc sortir du dossier courant pour remonter d'un niveau), il vous faudra l'indiquer grâce à `../`

```
<a href=" ../page2.html">Lien</a>
```

**Exemple :**



Ce schéma montre une hiérarchie de fichiers dans un dossier racine de site : seule la page d'accueil du site est dans le dossier racine. Les pages de la rubrique téléchargements du site sont rassemblées dans un dossier (deux page pour la rubrique de téléchargements dans cet exemple).

- Le schéma montre que pour faire un lien vers une page ou une image à partir de la page d'accueil `Site/index.html`, il suffit de nommer le ou les dossiers contenant le fichier, séparés par des slashes (/), puis d'indiquer le nom du fichier. Par contre, pour faire un lien à partir de la page `Site/telechargements/index.html`, il est parfois nécessaire de remonter d'un dossier avant de donner l'adresse relative d'un fichier, ce qui est fait avec `../`.
- Pour faire un lien vers l'image `fond.png`, il est nécessaire de remonter dans le dossier racine (donc `../`), puis de descendre dans le dossier `images`, et enfin trouver le fichier `fond.png`.
- Le fichier `liste.html` étant dans le même dossier que `index.html`, il n'y a aucun dossier à mentionner dans l'adresse relative.

### Attention !

Évitez les noms de fichiers complexes, donnez à vos fichiers des noms en minuscules, sans espace ni accent.

Dans le cas d'un **lien absolu**, on précise l'adresse complète :

```
<a href="http://www.csszengarden.com">Allez voir le Site de ZenGarden ! </a>
```

Ce type de lien est le seul utilisable pour faire un lien vers un autre site que le sien.

Si tous les liens de votre site sont en absolu et que celui-ci change d'adresse, il faudra modifier tous ces liens pour les adapter à la nouvelle adresse. Les liens relatifs ne contenant pas le début de l'adresse, ils se révèlent plus pratiques.



## 7. INSERER DES IMAGES

---

Au-delà de l'aspect purement décoratif, les images permettent de donner une identité visuelle au site (avec le logo du site) et peuvent même constituer le centre de l'information.

### 7.1. Choisissez bien le nom de votre image

Prenez l'habitude d'enregistrer vos fichiers avec des noms en minuscules, sans espaces ni accents. Par exemple : monimage.png. Vous pouvez remplacer les espaces par un tiret afin d'améliorer la lisibilité du nom du fichier<sup>11</sup>.

### 7.2 Les formats d'image du Web

Si les pages web sont parfois longues à charger, c'est souvent à cause des images. L'objectif est d'avoir des fichiers images les plus légers possible, d'où l'importance de choisir le format d'image approprié.

**Les principaux formats que l'on utilise sur le Web sont :**

#### JPEG

Les images au format JPEG (Joint Photographic Expert Group) sont très répandues. Elles sont en particulier très adaptées pour les photos, c'est-à-dire les images comportant beaucoup de couleurs différentes.

Le format JPEG possède la particularité de proposer différents niveaux de compression, c'est à vous de trouver le meilleur compromis entre une image de très bonne qualité mais de poids plus important ou de qualité moindre mais de poids très faible.

#### PNG

Le format d'image PNG (Portable Network Graphics) est assez récent. Contrairement au JPEG, il compresse l'image sans en détériorer la qualité. On peut enregistrer indifféremment des images avec peu ou beaucoup de couleurs, bien que le format JPEG reste le plus indiqué pour les photos.

Le format PNG est libre de droit, contrairement au GIF qui est soumis à un brevet (il a d'ailleurs justement été créé pour offrir une alternative à ce format).

On distingue les PNG 8 bits qui peuvent stocker jusqu'à 256 couleurs différentes (adapté pour les images possédant très peu de couleurs) et les PNG 24 bits supportant plusieurs millions de couleurs (16 777 216 pour être précis). Notez toutefois qu'on recommande l'utilisation du JPEG pour les photos car la compression est plus efficace pour ce type d'image.

#### GIF

Le format GIF (Graphics Interchange Format) est très répandu sur le Web (plus que le PNG) car c'est un format qui existe depuis plus longtemps. Contrairement aux PNG, les GIF ne peuvent pas contenir plus de 256 couleurs ce qui les rend inutilisables pour les photos.

Ils peuvent être rendus transparents mais ne proposent pas autant d'options de transparence que les PNG. Il est donc aujourd'hui recommandé d'utiliser les PNG dans la mesure du possible.

Les GIF, contrairement aux png peuvent être animés. Notez que les images animées peuvent distraire les visiteurs, voire les agacer. Par conséquent, évitez l'utilisation de GIF animés à moins que cela ne soit vraiment nécessaire.

#### SVG

Le format d'image SVG est un format d'image vectoriel qui est maintenant supporté par tous les navigateurs modernes. Il est très adapté pour la création d'icônes ou de petites illustrations.

Dans une image JPG ou PNG on va sauvegarder la couleur des différents pixels qui la compose. Du coup si on augmente la taille d'une image, on a l'impression qu'elle est floutée.

Le format vectoriel va sauvegarder les informations différemment un utilisant des formes géométriques et des courbes.

---

<sup>11</sup> <http://blog.internet-formation.fr/2016/06/google-fonctionne-aussi-bien-avec-des-underscores-ou-tirets-dans-les-url/>

L'avantage est alors que l'on peut redimensionner l'image à volonté, il suffit d'adapter les valeurs des formes en fonctions du niveau de zoom voulu. Attention cependant, il n'est pas possible d'utiliser le format SVG pour sauvegarder une photo par exemple, car nous ne pouvons pas représenter l'image avec des formes simples.

## Comparaison des formats d'image

<https://www.obogo.com/blog/optimiser-images-web-format-jpg-png-resolution-nom-fichier-export-photoshop>

### 7.3. Insérer une image dans une page Web

La balise `<img>`, qui est auto-fermante, **doit être insérée dans un paragraphe**.

Elle requiert **obligatoirement** deux attributs : **src** précisant le chemin vers l'image que vous voulez afficher et **alt** permettant d'indiquer un texte de remplacement.

```
<p>Voici une photo de mes dernières vacances en Espagne à Moraira :</p>
<p></p>
```

### Cibler l'image

```
 lorsque l'image se trouve dans le même dossier que la page
 lorsque l'image se trouve dans un sous-dossier (images)
 l'image se trouve dans le répertoire parent de la page web
```

### 7.4. Prévoir un texte de remplacement si l'image ne s'affiche pas

```

```

L'attribut **alt** est particulièrement utile aux personnes qui ne peuvent pas voir les images, comme c'est le cas des non-voyants. Grâce à ce texte alternatif, ces personnes peuvent avoir une idée de l'image qui se trouve là. De plus, le texte alternatif est affiché par le navigateur à la place de l'image si celle-ci ne peut être chargée (parce que le fichier est introuvable, parce que l'utilisateur a désactivé le chargement des images, etc.). Notez que ce texte est également référencé par les moteurs de recherche.

Si vous ne voyez pas de texte alternatif pertinent pour votre image (ex : images décoratives), laissez l'attribut **alt** vide, comme ceci : ``

**Il faut toujours mettre cet attribut alt, car il est obligatoire.**

### 7.5. Que faire si l'image ne s'affiche pas ?

Si l'image refuse de s'afficher, c'est que vous avez mal indiqué son adresse : vérifiez si l'image se trouve bien dans le bon dossier, si le nom du fichier est bien correct, renommez le fichier de préférence afin que tout soit en **minuscules, sans accents et sans espaces**.

### 7.6. Ajouter une infobulle

Il est possible d'ajouter une infobulle qui s'affiche lorsqu'on pointe sur l'image, cela permet de donner au visiteur une information supplémentaire sur l'image qu'il est en train de consulter.

```
<p>

</p>
```

### 7.7. Fixer une taille d'image

```
<p></p>
```

### 7.8. Habiller un texte autour d'une image

C'est le langage CSS qui permettra de placer correctement plusieurs images sur une page. En attendant, indiquez la mention suivante :

```

```

L'image se placera à gauche de la page et le texte longera l'image, à sa droite.

```
<br clear="left">
```

Ce texte arrêtera de longer l'image et se placera en dessous de celle-ci.

```

```

L'image se placera à droite de la page et le texte longera l'image, à sa gauche.

```
<br clear="right">
```

Ce texte arrêtera de longer l'image et se placera en dessous de celle-ci.

## 7.9. Ajouter un trait horizontal

```
<h1> les tigres ...</h1>
```

```
<hr/>
```

La ligne horizontale se tracera là où est insérée la balise. Notez toutefois que l'utilisation des bordures en CSS est souvent préférable dans la mesure du possible plutôt que les lignes horizontales.

## 7.10. La balise Figure

En HTML5, on dispose de la balise `<figure>`. Voici comment on pourrait l'utiliser :

```
<figure>
```

```

```

```
</figure>
```

**Si vous faites de votre image une figure, l'image peut être située en-dehors d'un paragraphe.**

Une figure est le plus souvent accompagnée d'une légende. Pour ajouter une légende, utilisez la balise `<figcaption>` à l'intérieur de la balise `<figure>`, comme ceci :

```
<figure>
```

```

```

```
<figcaption>La représentation de la Joconde</figcaption>
```

```
</figure>
```

### Quand insérer une image dans une balise `<figure>` plutôt que dans un paragraphe ?

*Si elle n'apporte aucune information et constitue seulement une illustration pour décorer :* placez l'image dans un paragraphe.

*Si elle apporte une information importante pour la bonne compréhension du texte :* placez l'image dans une figure.

**La balise `<figure>` a un rôle avant tout sémantique.** Elle indique à l'ordinateur que l'image a du sens et peut permettre à un programme de récupérer toutes les figures du texte afin de les référencer (dans une table des figures par exemple).

### Une figure peut très bien comporter plusieurs images :

```
<figure>
```

```

```

```

```

```

```

```
<figcaption>Logos des différents navigateurs</figcaption>
```

```
</figure>
```

## 7.11. Créer une image cliquable

```
<a href="http://www.mondetigres.com">  </a>
```

## 7.12. Supprimer la bordure d'un lien graphique

Lorsque vous créez une image cliquable, le navigateur crée automatiquement un cadre (bleu ou violet) autour de l'image pour montrer qu'il s'agit d'un lien. C'est le langage CSS qui permettra de supprimer ce type de décoration sur plusieurs images. En attendant, indiquez la mention suivante :

```
<a href="http://www.mondetigres.com">  </a>
```

## 7.13. Les images Maps

La balise <MAP> (=carte en français) sert à définir des zones réactives (zones déterminées comme étant cliquables) sur une seule image afin d'insérer plusieurs liens sur une même surface. Cette technique impose de calculer les coordonnées de chacun des coins de chaque map, ce qui n'est pas chose évidente.

Les balises img et map étant en ligne, elles doivent être contenues dans un paragraphe

```
<p>  

```

La balise correspondant au nom spécifié dans l'attribut USEMAP contient toutes les infos concernant les zones réactives (sensibles à la souris et pointant vers une destination interne ou externe) :

```
<map name="imageMap" id="imageMapId">
```

**name=le nom de la map, il est obligatoire et doit être unique** / id=id de la map (unique aussi), celui-ci n'a pas vraiment d'utilité, mais est obligatoire pour respecter les normes W3C.

Le conteneur <MAP> contient autant de balises <AREA> qu'il y a de zones réactives.

```
<area shape="circle" alt="site de ..." title="" coords="1248,232,158"  
href="http://www.lesite.be" target="_blank" />  
<area shape="rect" alt="site de ..." title="" coords="362,244,736,502"  
href="http://www.lesite.be" target="_blank" />  
</map>  
</p>
```

A une zone réactive est associée une balise <AREA> dont les attributs sont **SHAPE** : le type de zone (cercle, rectangle ou polygone), **COORDS** : les coordonnées de la zone (séparées par des virgules), **HREF** : l'adresse destination, **ALT** : le commentaire qui apparaîtra lorsque le curseur de la souris passera sur la zone et **TITLE** : le titre du lien.

Lecture web :

<http://www.siteduzero.com/informatique/tutoriels/images-mappees-et-liens-hypergraphiques/votre-premiere-image-mappee>

Créer ses maps en ligne :

<http://www.image-maps.com>, <http://www.maschek.hu/imagemap/imagmap>

## 7.14. Le format SVG

JPG, PNG et GIF sont des formats d'images bitmaps ou image matricielle (composés de pixels, elle a une résolution fixe) tandis que SVG<sup>12</sup> (Scalable Vector Graphics) est un format vectoriel (les traits sont définis par des calculs mathématiques qui s'adaptent à la taille de l'image), construit à partir de formes géométriques.

Les avantages de ce format sont la légèreté (lorsqu'il s'agit de représenter des formes simples) et elles sont étirables sans perte de qualité.

Concrètement, les informations décrivant ces formes sont stockées (coordonnées, couleurs, effets) contrairement aux images bitmap (JPG, PNG, GIF) qui doivent mémoriser le contenu pixel par pixel.

SVG est dans la lignée de HTML : spécifié par le W3C, ouvert, libre, simple d'utilisation. Il interagit avec les différents langages qui composent nos pages web.

### Intégrer SVG à sa page web

Il y a différentes possibilités plus ou moins pratiques et plus ou moins supportées, afin d'intégrer du SVG dans une page web, en voici deux :

#### La balise <svg>

HTML5 introduit un nouvel élément afin d'embarquer du contenu SVG dans une page web.

```
<svg width="200" height="100">
```

```
...
```

```
</svg>
```

#### La balise <img>

Approprié pour une utilisation basique. Accompagnée d'une utilisation de Javascript cette méthode est à éviter car cette balise n'a pas été faite pour accueillir des comportements scriptés.

```

```

#### Par CSS

On peut inclure ce format dans une propriété CSS comme un background.

```
.fonddepage {  
    background : url("fond.svg") no-repeat left left ;  
}
```

#### Lecture web :

<http://www.alsacreations.com/tuto/lire/1421-svg-initiation-syntaxe-outils.html>

<http://letrainde13h37.fr/5/svg-ami-du-responsive-webdesign/>

## 8. VALIDER SON CODE HTML

La validation HTML permet aux webmasters de vérifier la syntaxe de leur code HTML. Le validateur du W3C est accessible à l'adresse : <http://validator.w3.org/>

Avoir des pages valides aux yeux du validateur du W3C permet une certitude de l'absence de faute grammaticale dans le HTML de nos pages et une garantie d'une structure valide. Pour qu'il soit efficace, il convient d'indiquer dans notre code le Doctype correspondant à notre page Web.

Il faudra aussi faire valider les CSS.

<sup>12</sup> Élaboré à partir de 1998 par un groupe de travail comprenant entre autre IBM, Apple, Microsoft, Xerox.