

La couche présentation – HTML 5

D'après le livre :
HTML5 et CSS3
Maîtrisez les standards des applications Web
Luc VAN LANCKER

1) Présentation de l'HTML 5

HTML (*HyperText Markup Language*)

Son rôle est **de gérer et organiser le contenu**.

C'est donc en HTML que vous écrirez ce qui doit être affiché sur la page : du texte, des liens, des images... Vous direz par exemple : « Ceci est mon titre, ceci est mon menu, voici le texte principal de la page, voici une image à afficher, etc. ».

a) Les nouveautés d'HTML 5

Le Html5 est en fait un concept composé du Html5 pour la structure du document, des propriétés de style CSS pour la présentation de celui-ci et d'applications JavaScript pour des fonctionnalités avancées. Le constat pour cette seconde édition est que le Html5 a été largement adopté non seulement pour le Web classique mais aussi pour le Web mobile.

le Html5 fait table rase de tout aspect de présentation pour se consacrer exclusivement à la structure du document.

Il n'est donc plus possible en Html5 d'aligner du texte et d'autres éléments ou de dessiner des bordures aux tableaux. Terminé aussi l'ajout de couleur au texte ou arrière-plan. Le Html5 se décline désormais de façon austère, en noir et blanc !

=> Tous les éléments de présentation doivent obligatoirement être repris par les feuilles de style.

Les nouveautés introduites par le Html5 sont :

- Les nouvelles balises sémantiques qui permettent d'organiser le code source des applications afin d'en assurer une meilleure lisibilité.
- Les formulaires. Il faut bien avouer que les formulaires traditionnels comme la ligne ou la zone de texte, les boutons radio, à choix multiples ou autres boutons de commandes entraînent une lassitude certaine dans l'aspect visuel de la page. Les apports du Html5 vont révolutionner les formulaires avec des lignes de texte disposant d'une liste de suggestions, des calendriers, des compteurs numériques, des curseurs, une validation sans passer par le JavaScript et bien d'autres surprises à découvrir dans ce chapitre. Cette deuxième édition fait le point sur l'adoption, quelque peu laborieuse, de ces formulaires évolués par les navigateurs.
- Les nouvelles balises audio et vidéo permettent désormais de lire, de façon native, ces fichiers sans devoir passer par des plug-ins divers, souvent source de complications. Il importait aussi de passer en revue les différents formats audio et vidéo disponibles et en projet.

Le Html5 est aussi caractérisé par l'apparition massive d'applications JavaScript. Même si leur apprentissage risque de dérouter quelque peu les débutants, nous abordons les plus accessibles comme le dessin 2D, la géolocalisation de l'utilisateur, le stockage de données en local et les applications en mode déconnecté

b) La structure d'un document HTML

Tout document Html doit commencer par un doctype. Le Html5 propose un doctype unique et simplifié.

<!DOCTYPE html>

Pour se rendre compte de la simplification, il suffit de le comparer avec un doctype du Html 4.0.

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

Le doctype, aussi appelé DTD sert à indiquer au navigateur à quelles règles d'écriture obéit le code source de la page Html. Il utilisera celles-ci pour afficher la page selon les standards du W3C. Tous les navigateurs appliquant la même règle, vous êtes ainsi assuré d'un rendu identique entre les différents navigateurs.

Le doctype doit se situer à la première ligne du fichier Html.

S'il y a n'importe quoi d'autre, même un simple espace ou une ligne blanche, certains navigateurs considéreront que la page n'a pas de doctype

La balise `<html>` indique au navigateur qu'il s'agit d'un document Html.

La balise `<html>` est l'élément le plus haut ou l'élément racine du document. Elle prend donc place juste après la déclaration de doctype.

La balise `<html>` peut prendre comme attribut `lang="fr"` qui spécifie que le document est en langue française.

`<html lang="fr">`

Cette information est fort prisée par les moteurs de recherche comme Google.

La balise `<head> ... </head>` qui se positionne juste après le doctype et la balise `<html>` contient ce qu'on appelle l'en-tête du document.

Dans cet en-tête du document se situe toute une série d'informations relatives au document lui-même, indépendamment du contenu de celui-ci.

Ces informations peuvent être de natures très diverses :

- Le titre du document. Soit la balise `<title> ... </title>` reprise au point suivant.
- Des déclarations ou des appels de feuilles de style CSS.
- Des fonctions ou des liens vers des fichiers JavaScript.
- Des informations à l'intention des moteurs de recherche pour le référencement de la page.
- Des informations à l'intention des navigateurs.
- Des balises méta qui reprendront quant à elles l'encodage, la description de la page, les mots-clés associés à la page, le nom de l'auteur, des mentions de copyright, etc.

Dans un souci de simplification, le Html5 spécifie des valeurs par défaut pour l'attribut type des scripts, des styles et les éléments link.

Ainsi, en Html 4.0, on devait écrire :

```
<script type="text/javascript">
...
</script>
```

Et

```
<script type="text/javascript" src="fichier.js"></script>
```

En Html5, on peut se contenter de :

```
<script>
...
</script>
```

La balise <title>, incluse dans l'en-tête du document, est la seule balise obligatoire de celui-ci

La notion du jeu de caractères utilisé (**charset**) est une notion essentielle dans le développement de pages Web.

Cette notion est pourtant parsemée d'embûches, surtout en termes **d'interopérabilité**.

C'est à cause de ces problématiques que vous pouvez recevoir ce type de message :
vÃ©rification ?

ASCII	Jeu de caractères basique mais relativement ancien qui a permis l'échange de textes en anglais à un niveau mondial. Il a été repris à l'origine de la toile mais comme l'anglais ne comporte pas d'accents, il fallait encoder les caractères accentués et spéciaux par des entités du genre ´ (pour le é) ou € (pour l'€).
ISO-8859-1	Extension de l'ASCII, l'ISO-8859-1, souvent appelée Latin-1 ou Europe occidentale, est le codage fréquemment utilisé dans notre région du monde. Il ajoute de nombreux caractères latins ainsi que quelques autres signes de ponctuation ou symboles. Aujourd'hui, son utilisation tend à décroître au profit de l'Unicode.
ISO-8859-15	Version actualisée de l'ISO-8859-1 qui comporte entre autres le signe €.
UTF-8	UTF-8 est un format de codage de caractères (charset) conçu pour coder l'ensemble des caractères internationaux d'Unicode de plusieurs langues, contrairement aux formats plus vieux comme l'iso-8859-1 qui requiert une forme d'encodage afin d'accepter, par exemple, des lettres accentuées ou des caractères spéciaux. Son principal avantage pour le web est qu'il permet de représenter les milliers de caractères des différentes langues et permet donc de créer des sites Web qui puissent être multilingues (français, chinois ou arabe,...) sans avoir à créer de multiples versions du site dans l'encodage du pays concerné.

Depuis fin 2011, le WC3, suivant en cela l'IETF (Internet Engineering Task Force) qui édite les standards Internet, recommande **le charset UTF-8 pour l'édition de pages Web en grande partie pour son universalité**.

. Il est ainsi important de définir le charset utilisé dans l'en-tête du document.

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

Les balises <body> ... </body> déterminent ce que l'on appelle le corps du document.

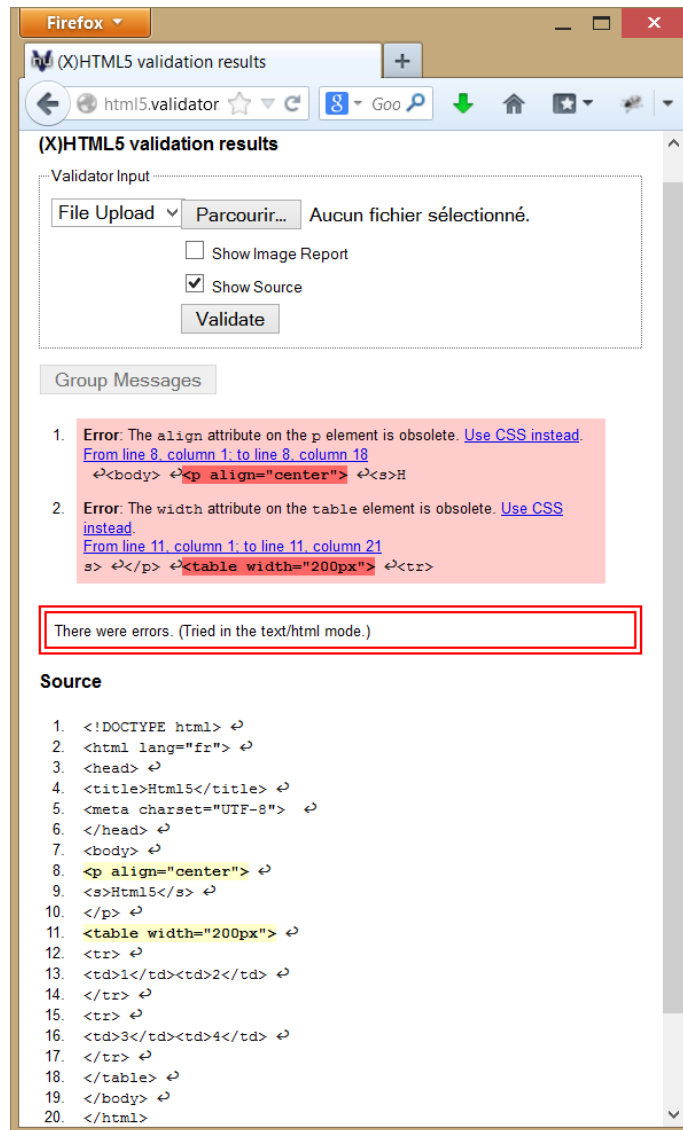
En Html5, les attributs dédiés à la présentation du corps du document de la balise <body> sont maintenant abandonnés pour laisser place à l'utilisation des feuilles de style.

Comment procéder pour créer une page HTML ?

- **utiliser un éditeur de texte (Sublime text, notepad++)**
- **retenez "UTF sans BOM" ou "UTF-8 without BOM".**
- **Enregistrer sous l'extension .html ou .htm (qui semble se généraliser)**
- **Ouvrir le fichier dans un navigateur pour observer le résultat**

Vous pouvez utiliser des validateurs HTML5 proposé par la W3C (Consortium international qui vérifie les normes HTML) par exemple.

Exemple du validateur du site validator.nu :



Ecrire le code d'une page html dont le titre est « Html 5 » et qui a pour contenu « Un premier document html 5 ». Vous suivrez les recommandations d'html5.

2) Le texte

Tout le contenu de votre page Web se place dans le corps du document Html5, soit entre les balises <body> ... </body>.

En Html, les retours chariot, ou passages à la ligne dans le code, sont ignorés et assimilés à un espace.

Pour mettre du texte en gras (bold en anglais), il suffit d'inclure celui-ci entre les balises

Pour mettre le texte en italique (italic), il faut l'inclure entre les balises <i> ... </i>.

La balise ... permet de marquer un élément de texte comme supprimé (delete) ou périmé, par exemple lors d'une mise à jour de prix dans un site commercial. Le texte apparaît alors barré à l'écran.

La balise <bdo> ... </bdo> indique le sens d'affichage du texte (de la gauche vers la droite ou de la droite vers la gauche).

Les attributs sont :

dir="ltr" (left to right) pour le sens de lecture de la gauche vers la droite (défaut).

dir="rtl" (right to left) pour le sens de la droite vers la gauche.

Il est parfois utile de commenter le code Html (comme pour tout code de programmation d'ailleurs), pour en faciliter la compréhension lors d'une mise à jour.

En Html5, tout comme en Html 4.0, les commentaires doivent être précédés de la balise <!-- et être fermés par la balise -->.

Le formatage du texte avec les feuilles de style

Le Html5 se préoccupe principalement de la structure du document

Il faut donc utiliser plus généralement le css.

Pour mettre en gras :

```
<!DOCTYPE html>
<html lang="fr">
<head>
<title>Html5</title>
<meta charset="UTF-8">
<style type="text/css">
```

```
span { font-weight: bold;}
```

```
</style>
</head>
<body>
```

```
Texte en <span class="gras">gras</span>
```

```
</body>
</html>
```

La taille des caractères

Avec la suppression de la balise `` et de son attribut `size`, la taille des caractères est uniquement définie par la propriété de style CSS `font-size`.

```
<!DOCTYPE html>
<html lang="fr">
<head>
<title>Html5</title>
<meta charset="UTF-8">
<style type="text/css">

#grand { font-size: 36px;}

</style>
</head>
<body>

<div id="grand">Texte en grand</div>

</body>
</html>
```

La taille des caractères (`font-size`) est fixée à 36 pixels (36px). Elle ne s'appliquera qu'à la seule balise `<div>` identifiée par `id="grand"`.

3) La structuration du document

Les titres

Le langage Html5 propose six grandeurs de caractères pour les titres. Quand on sait que titre se dit en anglais heading, l'élaboration de la balise de titre devient évidente ; `<h>` pour heading suivi du numéro 1 (pour le plus grand), 2, 3, 4, 5, 6 (pour le plus petit) et le `>` de fermeture. Par la suite, la notation `<hx> ... </hx>` sera utilisée où x est un niveau de 1 à 6.

Les paragraphes

Un contenu textuel gagne en lisibilité et compréhension lorsqu'il est divisé en différents paragraphes.

Pour définir un paragraphe, la balise `<p> ... </p>` est utilisée.

Le trait horizontal peut se révéler fort utile pour définir un changement dans le contenu. La balise `<hr>` réalise cette fonction.

Les listes, bien connues des utilisateurs de traitement de texte, sont une façon efficace pour structurer un contenu textuel.

Une liste ordonnée est mise en place par la balise ` ... `.

On détermine ensuite, à l'intérieur de ces balises, les éléments ou les items de la liste, soit les balises ` ... `.

Le nouvel attribut Html5 `reversed` permet une numérotation descendante

Les listes non ordonnées, aussi appelées listes à puces, sont introduites par la balise :

` ... `.

On détermine ensuite, à l'intérieur de cette balise, les éléments ou les items de celle-ci, soit les balises ` ... `.

4) Les liens

Les liens constituent l'essence du langage Html et des pages Web. La richesse de la fonction hypertexte permet de tisser cette toile gigantesque qu'est le Web.

La balise `<a> ... ` introduit un lien. Sa syntaxe de base est :

`Texte de lien`

- La destination du lien peut être :
- Un endroit de la page en cours.
- Une autre page du site.
- Un endroit dans une autre page du site.
- Une page d'un autre site situé sur le Web.
- Une adresse électronique.
- Un fichier à télécharger.

Quelques attributs :

href

L'attribut href définit l'adresse (url) de la cible du lien.

target

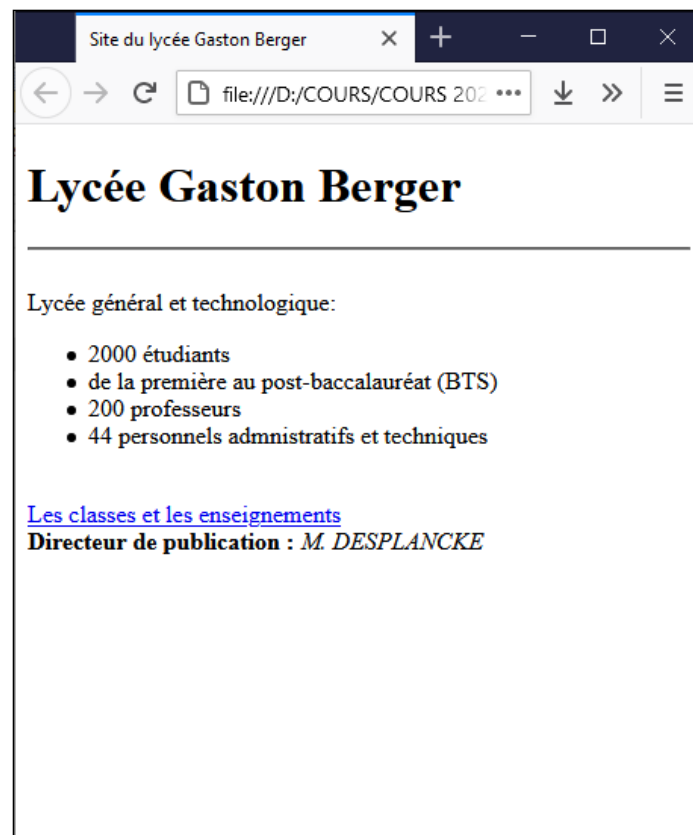
Spécifie au navigateur la façon d'afficher la cible du lien. Cela peut être dans une nouvelle instance du navigateur ou un nouvel onglet du navigateur (`target="_blank"`), dans la même fenêtre que la page de départ du lien (`target="_self"`) et dans la même fenêtre mais en occupant la totalité de la fenêtre du navigateur.

title

L'attribut `title="texte"` (qui n'est pas spécifique à la balise de lien) permet de proposer une infobulle explicative au survol du lien par le curseur de la souris.

Les premiers liens lors de l'apprentissage de la publication Web, consistent à pointer vers une autre page située dans le site (adressage relatif).

Exercice :
Réaliser la page suivante :



Le lien pointe sur le fichier classe.htm qui se trouve dans le sous-dossier « Annexe » de la page courante.
N'oubliez pas d'indiquer des commentaires indiquant au développeur que le nom du proviseur peut changer

5) Les tableaux

Les tableaux sont exclusivement réservés à la présentation de données. Il n'est plus question dans une application Web moderne d'utiliser les tableaux pour réaliser la disposition de la page. Ces tableaux de présentation, souvent imbriqués dans d'autres tableaux, étaient le défaut de la conception de pages Web dans ce que l'on peut appeler a posteriori le Web 1.0.

Ces multiples tableaux alourdissaient le code source et en rendaient la lecture quasi impossible. La disposition des éléments de la page est maintenant assurée par des balises `<div>`.

Un tableau (`<table>`) comporte des lignes `<tr>`. Ces lignes contiennent des cellules `<td>`.

La largeur d'un tableau est défini en css :

```
<style>
table { width: 250px;}
</style>
```

Il est également possible de déterminer la hauteur du tableau par la propriété de style height.

```
<style>
table { height: 150px;}
</style>
```

Les bordures d'un tableau et d'autres éléments Html sont introduits par la propriété de style CSS border (pour bordure).

```
<style>
table { border: 1px solid black;
width: 250px;}
</style>
```

Il faut également demander d'ajouter une bordure aux cellules `<td>` qui constituent le tableau (td).

```
td { border: 1px solid black;
width: 33%;}
```

Pour fusionner des colonnes, le Html5 dispose de l'attribut de cellule `colspan="x"` où x correspond au nombre de colonnes que l'on souhaite fusionner horizontalement.

Pour fusionner des lignes, le Html5 dispose de l'attribut de cellule `rowspan="x"` où x correspond au nombre de lignes que l'on souhaite fusionner verticalement.

Les cellules d'en-tête de colonnes ou de lignes sont définies par la balise `<th> ... </th>`. Elles fournissent, en quelque sorte, un titre aux données de la colonne ou de la ligne.

Exercice :
Réaliser la page classe.htm

Firefox

Site du lycée Gaston Berger

file:///C:/Users/Julien/Desktop/Exo3.htm

Google

Le BTS SIO

	Classes	Effectifs
1ère année SIO	1SIO1	33
	1SIO2	34
2nde année SIO	2SISR	26
	2SLAM	29

6) Les images

Pour réduire au maximum le temps de téléchargement, le Web réclame des formats de fichier particuliers ; le format GIF, JPEG et PNG. Ces trois types de format comportent tous un algorithme de compression.

Le format GIF

Le format GIF (pour Graphics Interchange Format) a connu son heure de gloire aux débuts du Web. Il était alors parfaitement adapté aux cartes graphiques de l'époque (souvent maximum 256 couleurs) et aux faibles capacités des lignes de téléphone analogiques (taux de compression élevé).

Caractéristiques

- Le format GIF ne permet de coder les images qu'en maximum 256 couleurs. Ce qui n'est plus très réaliste avec les développements et les possibilités des images numériques.
- C'est un format de compression efficace et rapide. Les images GIF sont d'un poids en octet réduit.
- La compression s'effectue sans perte de données et sans perte de qualité.
- La fonction dite entrelacée permet un affichage rapide d'images qui deviennent progressivement de plus en plus nettes.
- La possibilité, sous la spécification GIF89a, de définir une des 256 couleurs comme étant transparente. À l'affichage apparaissent alors les éléments qui sont placés sous le graphique aux endroits où l'image est transparente.
- Les images sont facilement éditables et modifiables vu le nombre limité de couleurs.
- Seul format qui permette de construire des animations, sortes de petits dessins animés, avec ce qu'on appelle les GIFS animés.

Conclusion

En raison de sa limitation à 256 couleurs, le format GIF n'est pas du tout approprié aux graphiques à haute résolution comme les photos, les images nuancées et les dégradés. Par contre, il excelle encore pour les petits logos, boutons, puces, barres, symboles, et autres cliparts fréquents pour certains graphiques du Web qui ne nécessitent pas une palette de couleurs très étendue. Il est cependant de plus en plus supplanté par le format PNG.

Le format JPEG

Le format JPEG figure pour Joint Photographic Expert Group, donc non pas pour le nom du format graphique mais pour le nom de la corporation qui a développé ce format. C'est le type de format le plus répandu dans l'univers de la photo numérique.

Caractéristiques

- Le format JPEG permet de sauvegarder jusqu'à 16,7 millions de couleurs par image.
- Son taux de compression est efficace en ce qui concerne la taille résultante.
- Il permet de faire varier le taux de compression selon les besoins du concepteur. Une compression de 15 à 20 % fournira une image de qualité satisfaisante pour une taille de fichier réduite.
- Revers de la médaille, il y a une perte de qualité en fonction du taux de compression. Plus le facteur de compression est élevé, plus la qualité de l'image est mauvaise. C'est donc une compression destructive.
- Des déformations (artifacts ou artefacts) apparaissent souvent dans les passages de couleurs tranchés, donc sur les coins et les bords des objets.
- Depuis quelques années, la fonction entrelacée est implémentée, ce qui permet un affichage progressif et plus rapide sur le Web.
- Le format JPEG n'offre pas de possibilité de transparence ni d'animations.

Conclusion

Le JPEG est un excellent format pour les photos, spécialement celles où il y a des tonalités diverses et des dégradés de couleurs. Mais il n'est pas exempt de défauts avec ses pertes de données et ses déformations.

À cause des pertes de données et des déformations, il est impératif de respecter la règle de sauvegarder l'image originale et de travailler uniquement sur des copies de celle-ci.

Le format PNG

Le format PNG, pour Portable Network Graphic, est présenté comme le format du futur. Il s'agit d'un format graphique conçu spécialement pour la mise en œuvre sur le Web. La recommandation du W3C concernant le PNG remonte à 1996. Le PNG doit théoriquement rassembler tous les avantages de GIF et du JPEG. Il commence à être largement retenu par les concepteurs d'applications Web.

Caractéristiques

- Le format PNG soutient 16,7 millions de couleurs comme le format JPEG.
- Le format PNG est une spécification conçue spécialement pour Internet et est l'objet d'une recommandation du W3C.
- Sa compression est performante. Le taux de compression du format PNG est de 5 % à 25 % supérieur à celui du format GIF.
- Cette compression s'effectue sans pertes de données et de qualité.
- Il permet la transparence jusqu'à 256 couleurs.
- Le format PNG est un format ouvert et n'est pas breveté.
- Sa fonction entrelacée permet un affichage progressif.
- Il ne peut générer des images animées.

Conclusion

Le format PNG est appelé à être de plus en plus utilisé à l'avenir car il représente un excellent compromis entre le format GIF et JPEG. Les professionnels l'adoptent largement en lieu et place du GIF et pour les petites photos.

Le format WebP

Le WebP est un nouveau format d'image qui risque de prendre de l'importance dans les prochaines années. Ce format est développé par Google à partir d'un des formats de la nouvelle balise <video> du Html5 (le WebM). Selon les premiers essais, le WebP réduirait la taille des fichiers de 39 % par rapport aux formats JPEG, PNG et GIF, sans perte de qualité perceptible.

Le poids des photos

Il semble illusoire de charger directement une photo de votre appareil photo numérique sur un site. Certaines "pèsent" jusqu'à 4 Mo, ce qui, même avec une connexion rapide, prendra un temps certain de chargement.

On peut conseiller :

- De recadrer les images pour ne recueillir que ce qui est utile.
- De redimensionner cette image recadrée.
- D'adopter un taux de compression d'environ 15 à 20 % pour les images JPEG. La perte de qualité n'est quasiment pas perceptible à l'œil normal.
- De diminuer le nombre de couleurs de l'image, si l'opération s'y prête.

Ces différentes manipulations devraient ramener la taille du fichier à une dimension beaucoup plus raisonnable et améliorer sensiblement le temps de téléchargement.

L'insertion d'une image est obtenue par la balise :

```

```

Réaliser un lien à partir d'une image :

Les liens à partir d'une image se réalisent simplement en entourant celle-ci par la balise de lien <a>

Pour enlever cette bordure, il faut passer en Html5 par une déclaration de style. Dans l'attente de l'étude détaillée des feuilles de style, il suffit d'ajouter à la balise la déclaration style="border: none;".

```
<a href="globe.htm">  
  
</a>
```

7) Les balises sémantiques et d'organisation

Ces balises sémantiques décrivent la nature de certains éléments du contenu d'une page. Ces derniers pourront ainsi être plus facilement récupérés et réutilisés par des programmes automates, soit des moteurs de recherche.

Exemple

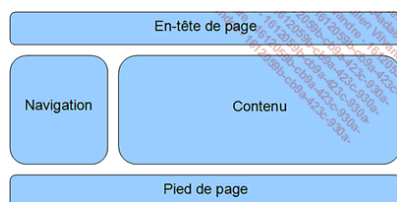
La nouvelle balise sémantique ou d'organisation de la page `<nav> ... </nav>` indique simplement que son contenu est un menu de navigation.

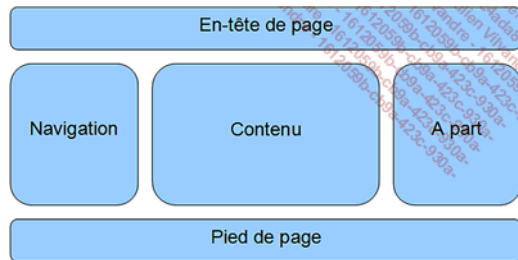
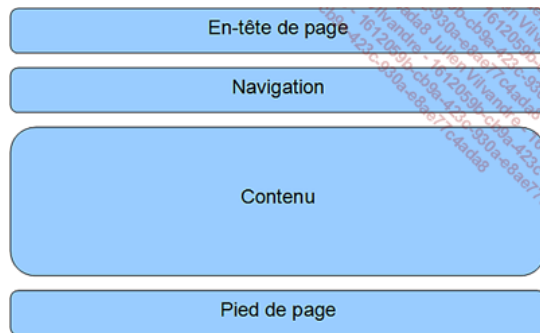
```
<nav>
<a href="#">Accueil</a> |
<a href="#.htm">Item 1</a> |
<a href="#.htm">Item 2</a> |
<a href="#.htm">Item 3</a> |
<a href="#.htm">Contact</a>
</nav>
```

Les balises `<header>`, `<nav>`, `<footer>` et `<aside>`

Si l'on effectue une synthèse des pages existantes sur la toile, on peut constater qu'elles comportent la plupart du temps, en totalité ou en partie, les éléments suivants :

- Un en-tête de page avec, par exemple, un logo, une bannière, le nom du site, un slogan ou un champ de recherche.
- Les outils de navigation, précieux et indispensables pour la consultation des différentes parties du site.
- Une partie consacrée au contenu.
- Une zone annexe qui permet d'apporter des éléments accessoires au contenu proprement dit, comme une publicité.
- Un pied de page avec par exemple la mention d'un copyright, le plan du site, les mentions légales, les règles d'accessibilité, etc.

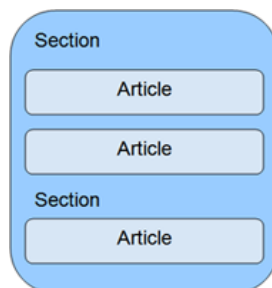




Les balises <section> et <article>

Le contenu principal peut à son tour être organisé en diverses parties :

- La balise <section> détermine une partie du contenu de la page se rapportant à un thème déterminé.
- La balise <article> définit un contenu indépendant du document qui possède une identité à part entière dans la page comme l'article d'un blog, un post dans un forum ou un produit dans un site commercial.



8) Les formulaires

Les formulaires occupent une place prépondérante dans la conception et l'exploitation d'une application ou un site Web. C'est en effet la seule façon de recevoir en retour des informations provenant directement de l'utilisateur final, et en outre, structurées selon les besoins du concepteur.

La balise `<form> ... </form>` a comme unique fonction de déclarer au navigateur qu'il doit mettre en place un formulaire. Elle englobera les éléments ou champs de formulaires comme une ligne de texte, des cases à cocher, des listes déroulantes, etc.

Les formulaires

Les attributs usuels de la balise `<form>` sont :

name

Pour attribuer un nom (name) au formulaire.

action

Lorsque vous donnez l'ordre au navigateur de transmettre les données du formulaire, il a besoin de connaître l'action qu'il doit effectuer.

Cette action sera :

- soit l'adresse d'un programme de traitement des données, situé sur le serveur, en CGI, Perl, PHP, ASP... Par exemple, `action = "http://www.serveur/traitement.php"`.
- soit une adresse de courrier électronique pour récupérer simplement les données. Le protocole `mailto` est alors utilisé. Par exemple, `action="mailto:mon_email@serveur"`.
- soit, lorsque les données d'un formulaire sont traitées en interne (côté-client) par du JavaScript, l'attribut `action` reste vide. Par exemple `action=""`.

Si vous utilisez le validateur du W3C (<http://validator.w3.org>), la présence de l'attribut `action` est obligatoire.

Method

La transmission des données d'un formulaire s'effectue selon la méthode GET ou la méthode POST. La méthode GET effectue le transfert en caractères ASCII et les données ne peuvent excéder 100 caractères. La méthode POST gère les caractères non-ASCII et une quantité de caractères illimitée. Dans la pratique, la méthode POST s'est imposée pour des raisons de facilité et d'efficacité.

La ligne de texte

Cette ligne de texte permet d'accueillir l'encodage de données tant alphabétiques que numériques.

Nom : `<input type="text">`

Les attributs possibles sont :

name

Définit un nom (name) unique pour cet élément. Cet attribut est utilisé pour collecter le contenu de la ligne de texte lors de la soumission du formulaire.

size

Définit le nombre de caractères visibles de la ligne de texte et, par conséquent, la largeur de la zone de texte. L'utilisateur peut néanmoins encoder autant de caractères qu'il souhaite même s'ils débordent du champ de la zone visible. La valeur par défaut de `size` est de 20.

maxlength

Détermine le nombre maximal de caractères que l'utilisateur peut encoder dans la ligne de texte. Cet attribut est particulièrement utile pour des données avec un nombre défini de caractères comme, par exemple, cinq chiffres pour le code postal français.

value

Définit la valeur par défaut de la ligne de texte. Celle-ci apparaît dans la ligne de texte au chargement de la page. Par exemple, `<input type="text" value="Votre nom ici !">`.

required

Rend l'encodage de l'élément obligatoire par l'utilisateur pour la soumission et le traitement du formulaire. Très utile pour des éléments essentiels comme le nom ou l'adresse e-mail de l'utilisateur.

La liste déroulante

La liste déroulante propose différentes options à l'utilisateur.

```
<select>
<option value="1">Internet Explorer</option>
<option value="2">Firefox</option>
<option value="3">Safari</option>
<option value="4">Opera</option>
<option value="5">Google Chrome</option>
</select>
```

name

Définit un nom pour la liste déroulante en vue d'un traitement ultérieur.

Size

Par défaut, l'attribut size de la balise `<select>` est égal à 1, ce qui est assez pratique car cela permet un gain de place appréciable dans la disposition de la page. Cependant par l'attribut `size="x"`, vous pouvez définir le nombre d'éléments du menu qui sera visible. Votre menu n'aura alors plus rien de déroulant !

```
<body>
<form action="">
<select size="4">
<option>Moins de 20 ans</option>
<option>De 20 à 40 ans</option>
<option>De 40 à 60 ans</option>
<option>Plus de 60 ans</option>
</select>
</form>
</body>
```

multiple

Par défaut, l'utilisateur ne peut choisir qu'un élément du menu déroulant. Avec l'attribut multiple de la balise `<select>`, plusieurs choix peuvent être effectués. Pour ce faire, l'utilisateur doit maintenir la touche [Ctrl] du clavier et cliquer sur les éléments avec la souris. Il est alors préférable de rappeler dans la page cette façon de procéder peu commune pour l'internaute moyen.

L'attribut size doit être spécifié et égal au nombre des balises `<option>`.

selected

Par défaut, le premier élément de la liste est retenu. Mais il est possible de présélectionner un autre élément par l'attribut selected de la balise <option>.

```
<body>
<form action="">
<p>Votre navigateur préféré :
<select>
<option value="IE">Internet Explorer</option>
<option value="FF">Firefox</option>
<option value="S" selected>Safari</option>
<option value="O">Opera</option>
<option value="GC">Google Chrome</option>
</select>
</p>
</form>
</body>
```

value

En principe, c'est le texte de l'élément choisi placé derrière <option> qui est transmis lors de l'envoi du formulaire. Vous pouvez toutefois spécifier qu'une autre valeur (généralement numérique) soit transmise avec l'attribut value="valeur".

Les boutons de choix unique (radio)

Les boutons de choix unique, aussi appelés boutons radio, ont comme particularité qu'une seule option à la fois peut être activée (le "ou" exclusif).

```
<input type="radio" name="sexe">Masculin
```

name

Ici l'attribut name est obligatoire. En outre, dans le cas de boutons radio, le nom doit être identique pour tous les boutons.

checked

Permet de présélectionner un bouton radio.

value

En vue d'un traitement ultérieur, on attribue une valeur à chaque bouton radio par l'attribut value="valeur".

Les boutons de choix multiples (checkbox)

La mise en œuvre de ces boutons de choix multiples, aussi appelés boutons checkbox, est proche des boutons à choix unique mais, dans le cas présent, plusieurs choix simultanés peuvent être réalisés.

Votre pizza avec :


```
<input type="checkbox" name="n1"> Mozzarella<br>
<input type="checkbox" name="n2"> Jambon<br>
```

Le bouton d'envoi

`<input type="submit">`

Le bouton d'annulation

`<!DOCTYPE html>`

`<html lang="fr">`

`<head>`

`<input type="reset">`

L'organisation des champs de formulaires

Dans le cas de formulaires longs et complexes, il est parfois utile de regrouper graphiquement certains éléments pour organiser la page de façon logique. Les balises `<fieldset>` et `<legend>` permettent d'améliorer sensiblement l'ergonomie et l'usabilité des formulaires.

La balise `<fieldset> ... </fieldset>` englobe les champs de formulaires que vous déterminez. Ces champs sont alors visualisés à l'écran par une bordure.

La balise `<legend> ... </legend>`, qui se place directement derrière la balise `<fieldset>`, ajoute une légende qui vient s'insérer dans la bordure dessinée par le `<fieldset>` (voir capture d'écran suivante).

L'étiquetage des champs de formulaires

La balise `<label>` associe explicitement l'intitulé à un champ de formulaire particulier. Un peu comme si on collait une étiquette (label) en face d'un élément de formulaire.

Dans un premier temps, la balise `<label>` améliore grandement l'ergonomie des formulaires en permettant d'activer un élément de celui-ci, par exemple un bouton radio, en cliquant sur le bouton radio lui-même ou sur l'intitulé de celui-ci.

Dans un premier temps, le texte assigné à un élément de formulaire doit être placé entre les balises `<label> ... </label>`.

La ligne de texte d'adresse e-mail

Le Html5 répond à ce besoin avec la balise `<input type="email">`.

Réaliser le formulaire suivant :

The screenshot shows a web browser window with the address bar displaying 'file:///C:/Users/JV/Desktop/FORM.HTML'. The form is styled with a simple border and contains two main sections: 'Client' and 'Commande'. The 'Client' section has four input fields: 'Nom et prénom', 'Adresse', 'Code Postal', and 'Pays' (which is pre-filled with 'France'). The 'Commande' section has four input elements: 'Modèle' (pre-filled with 'Road66'), 'Taille' (a slider control), 'S M L XL' (radio buttons), and 'Quantité' (a spinner control set to '1'). At the bottom of the form are two buttons: 'Commander' and 'Recommencer'.

```
<!DOCTYPE html>
<html lang="fr">
<head>
<title>Html5</title>
<meta charset="iso-8859-1">
<style>
fieldset { border : 1px solid black;}
</style>
</head>
<body>
<form action="mailto:votre_adresse_email" method="post"
enctype="text/plain">
<fieldset>
<legend><i>Client</i></legend>
Nom et prénom : <input type="text" name="Nom" required
placeholder="Nom et prénom"><br>
Adresse: <input type="text" name="Adresse" required><br>
Code Postal : <input type="text" name="CP" size="4" maxlength="5"
required pattern="[0-9]"><br>
Pays : <input type="text" readonly name="Pays" value="France">
</fieldset>
<br>
<fieldset>
<legend><i>Commande</i></legend>
Modèle : <input type="text" readonly name="Modele" value="Road
```

```
66"><br>
Taille : <br>
<input type="range" name="Taille" min="1" max="4" step="1"
value="2">
<div>
<span>S</span>
<span>M</span>
<span>L</span>
<span>XL</span>
</div>
<br>
Quantité : <input type="number" name="Quantite" min="1" max="6"
step="1" value="1"><br>
</fieldset>
<br>
<input type="submit" value="Commander">
<input type="reset" value="Recommencer">
</form>
</body>
</html>
```