

HTML et CSS

Les bases

David Roche, lycée G Fichet (Bonneville, Haute Savoie)
version 040811

Vous êtes libres :

- de reproduire, distribuer et communiquer cette création au public
- de modifier cette création

Selon les conditions suivantes :



- **Paternité.** Vous devez citer le nom de l'auteur original de la manière indiquée par l'auteur de l'œuvre ou le titulaire des droits qui vous confère cette autorisation (mais pas d'une manière qui suggérerait qu'il vous soutient ou approuve votre utilisation de l'œuvre).



- **Pas d'Utilisation Commerciale.** Vous n'avez pas le droit d'utiliser cette création à des fins commerciales.

- À chaque réutilisation ou distribution de cette création, vous devez faire apparaître clairement au public les conditions contractuelles de sa mise à disposition.
- Chacune de ces conditions peut être levée si vous obtenez l'autorisation du titulaire des droits sur cette œuvre.
- Rien dans ce contrat ne diminue ou ne restreint le droit moral de l'auteur ou des auteurs.

Notes de l'auteur :

- Ce document est à l'origine destiné aux élèves ayant choisi de suivre la spécialité "Informatique et Sciences du Numérique" en terminale scientifique.
- Ce document n'est pas une "bible" du HTML et du CSS (il existe des ouvrages très bien faits), il a uniquement pour but de permettre aux élèves une première approche (en autonomie) de la conception des pages web. Ce document ne prétend pas se substituer au travail en classe avec l'enseignant.
- Un deuxième document "HTML et CSS pour aller plus loin...." devrait voir le jour prochainement, il abordera des sujets tels que : la mise en page avancée, le CSS3 et le HTML5.
- Pour toutes remarques et commentaires, n'hésitez pas à me contacter : informatiqueaulyce@gmail.com

David Roche

Introduction

HTML signifie HyperText Markup Language (langage de balisage hypertexte). Ce n'est pas à proprement parler un langage de programmation (il n'y a pas de condition, de boucle ou encore de variables en HTML), mais plutôt un langage description. HTML est utilisé pour "décrire" une page web, 100 % des sites internet sont "fabriqués" (en partie au moins) à l'aide du HTML. Un œil un peu exercé est capable "d'imaginer" assez facilement l'apparence d'une page web rien qu'en regardant le code HTML de cette page.

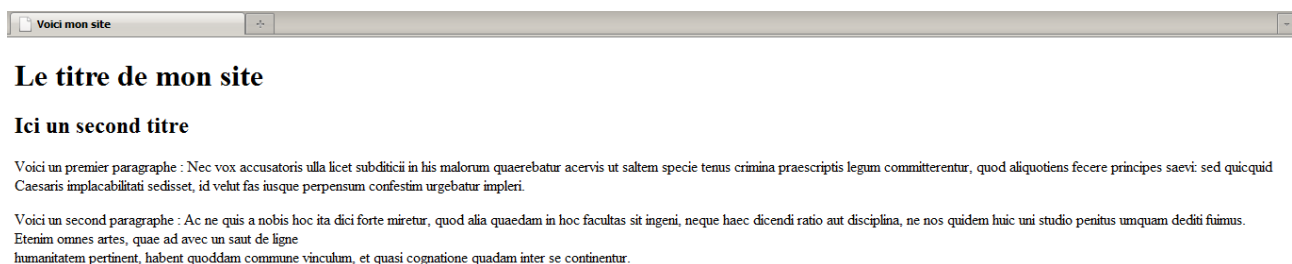
Enfin, le contenu d'une page web est donné par le code HTML; pour la mise en forme de votre page il faut utiliser un fichier CSS (Cascading Style Sheets), "feuille de style" en français (ce n'est pas une obligation, mais c'est très fortement recommandé !).

Le contenu dans le fichier HTML et la mise en forme dans le fichier CSS : c'est une règle à ne jamais oublier quand on veut réaliser un travail correct.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="fr" lang="fr">
<head>
  <title>Voici mon site</title>
</head>
<body>
  <h1>Le titre de mon site</h1>
  <h2>Ici un second titre</h2>
  <p>Voici un premier paragraphe : Nec vox accusatoris ulla licet subditicii in his malorum
quaerebatur acervis ut saltem specie tenus crimina praescriptis legum committerentur, quod
aliquotiens fecere principes saevi: sed quicquid Caesaris implacabilitati sedisset, id
velut fas iusque perpensum confestim urgebatur impleri.</p>
  <p>Voici un second paragraphe : Ac ne quis a nobis hoc ita dici forte miretur,
quod alia quaedam in hoc facultas sit ingeni, neque haec dicendi ratio aut disciplina,
ne nos quidem huic uni studio penitus umquam dediti fuimus. Etenim omnes artes,
quae ad avec un saut de ligne <br> humanitatem pertinent, habent quoddam commune vinculum, et
quasi cognatione quadam inter se continentur.</p>
</body>
</html>
```

Pour l'instant, le but n'est pas de comprendre quoi que ce soit, juste de vous donner un exemple de fichier HTML (sans fichier de style).

Voici le résultat dans le navigateur Firefox



Nous avons une page web sans "style". Pour lui appliquer un style, nous allons écrire un autre fichier, maFeuilledeStyle.css.

Voici le fichier maFeuilledeStyle.css

```
h1
{
    text-align: center;
    background-color: red;
}
h2
{
    font-family: Verdana;
    font-style: italic;
    color: green;
}
p
{
    font-family: arial;
    font-size: 12px;
}
```

et le fichier HTML très légèrement modifié (une ligne de plus, à vous de la trouver !)

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="fr" lang="fr">
<head>
  <title>Voici mon site</title>
  <link href="maFeuilledeStyle.css" rel="stylesheet" type="text/css" media="all" />
</head>
<body>
  <h1>Le titre de mon site</h1>
  <h2>Ici un second titre</h2>
  <p>Voici un premier paragraphe : Nec vox accusatoris ulla licet subditicii in his malorum
quaerebatur acervis ut saltem specie tenus crimina praescriptis legum committerentur, quod
aliquotiens fecere principes saevi: sed quicquid Caesaris implacabilitati sedisset, id
velut fas iusque perpensum confestim urgebatur impleri.</p>
  <p>Voici un second paragraphe : Ac ne quis a nobis hoc ita dici forte miretur,
quod alia quaedam in hoc facultas sit ingeni, neque haec dicendi ratio aut disciplina,
ne nos quidem huic uni studio penitus umquam dediti fuimus. Etenim omnes artes,
quae ad avec un saut de ligne <br/> humanitatem pertinent, habent quoddam commune vinculum, et
quasi cognatione quadam inter se continentur.</p>
</body>
</html>
```

Ici aussi, pour l'instant, inutile de nous attarder sur le contenu de ces fichiers, en revanche le résultat, nous intéresse de suite :



J'espère que la différence vous saute aux yeux (pour l'instant je suis d'accord avec vous, on ne peut pas dire que ma page soit "très jolie" !)

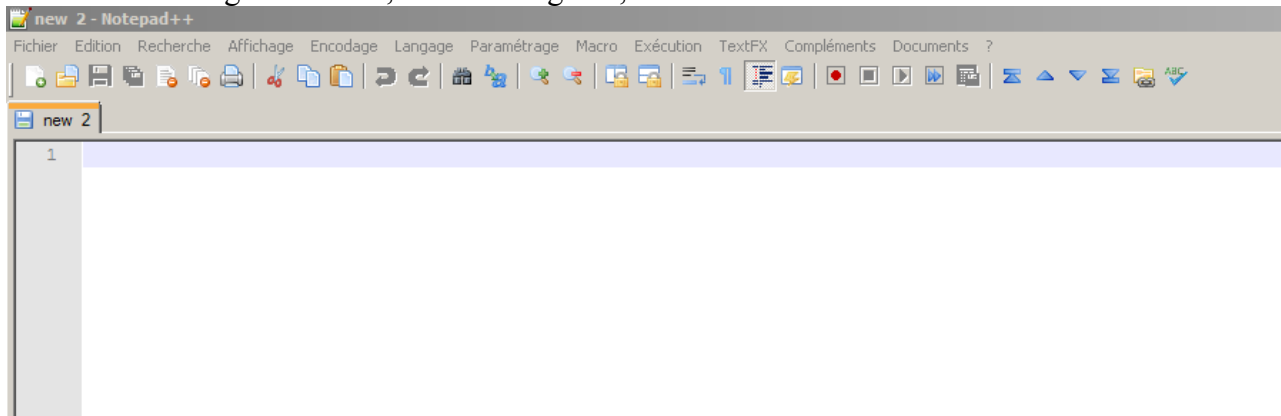
Encore une fois l'essentiel à retenir : le contenu de la page pour le fichier HTML, le style pour le fichier CSS.

Quels "outils" pour écrire du HTML et du CSS ?

Pour écrire des fichiers HTML et CSS un simple éditeur de texte suffit. Pour notre part, nous allons utiliser le logiciel Notepad++ disponible (malheureusement) uniquement sous Windows. Sous linux, il existe des logiciels équivalents (Scite, gedit,.....).

Ce logiciel, libre et gratuit, est téléchargeable à l'adresse suivante : <http://notepad-plus-plus.org/>.

Une fois téléchargé et installé, lancez le logiciel, vous devriez avoir ceci :



Avant de commencer à taper du code HTML, choisissez le menu "Langage" puis la lettre "H" et enfin "HTML" (pour écrire un fichier CSS, il suffit de choisir la lettre "C" puis "CSS". Voilà, c'est prêt !

Notre première page

Avant d'entrer dans le vif du sujet, il faut quand même que vous sachiez que nous n'allons pas faire du HTML, mais plutôt du XHTML (!!!). Rassurez-vous la différence est assez subtile, tellement subtile que je continuerai à vous parler de HTML. Pour les plus curieux d'entre vous, voici un site qui explique la différence : <http://www.alsacreations.com/article/lire/44-difference-html-xhtml.html>

En HTML (ou plutôt en XHTML), vous devrez toujours partir d'une page possédant "un code minimal obligatoire" :

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="fr" lang="fr">
<head>
  <title></title>
</head>
<body>

</body>
</html>
```

Dans la plupart des cas vous n'aurez strictement rien à modifier, juste à ajouter du code entre les balises body (ouvrante et fermante).

Voilà, le mot fondamental est lâché : balise !

En HTML tout est une histoire de balise que l'on ouvre et que l'on ferme (parfois, nous le verrons dans la suite, on ouvre et on ferme les balises en même temps).

Laissons les 3 premières lignes de côté, vous n'aurez sans doute jamais l'occasion de les modifier.

La 4e ligne est constituée de la balise head (balise ouvrante). Vous pouvez constater qu'une balise ouvrante est de la forme <nom_de_la_balise>. À la 6e ligne, on trouve la balise fermante de head, les balises fermantes sont donc de la forme </nom_de_la_balise>.

En observant attentivement le code (toujours à partir de la 4e ligne), vous devriez forcément remarquer que toute balise ouverte doit être refermée à un moment ou un autre. La balise ouvrante et la balise fermante peuvent être sur la même ligne ou pas, cela n'a aucune espèce d'importance, la seule question à se poser ici est : ai-je bien refermé toutes les balises que j'ai ouvertes ?

Vous avez peut-être remarqué que la première balise ouverte est la balise html (si l'on ne tient pas compte des 2 premières lignes qui ne comportent pas de balise), c'est aussi la dernière fermée. Votre document sera donc "encadré" par la balise html.

En parlant de la balise html vous avez sans doute remarqué la longueur quelque peu étrange de la balise ouvrante :

```
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="fr" lang="fr">
```

Cette structure est assez courante dans les balises ouvrantes : xmlns, xml:lang et lang sont des attributs. Chaque attribut possède une valeur (la valeur est obligatoirement entre guillemets); par exemple, la valeur de l'attribut lang est "fr". Au passage remarquez l'intérêt d'utiliser un éditeur de texte "évolué" comme notepad++ qui applique ce que l'on appelle "la coloration syntaxique" (le nom de la balise est en bleu, l'attribut est en rouge (le xml:lang est un cas particulier d'où la couleur noire) et la valeur de l'attribut en mauve. Nous aurons l'occasion de revenir sur ces attributs plus tard.

Il faut savoir qu'en XHTML les balises sont obligatoirement écrites en minuscules (en HTML on fait ce que l'on veut !).

Enfin pour terminer avec les généralités sur les balises, il est important de savoir qu'une structure du type :

```
<balise1>  
<balise2>  
</balise1>  
</balise2>
```

est interdite, la balise2 a été ouverte après la balise1, elle devra donc être refermée avant la balise1. L'enchaînement suivant est correct :

```
<balise1>  
<balise2>  
</balise2>  
</balise1>
```

Vous pouvez vérifier sur notre page que cette règle est bien appliquée.

Poursuivons maintenant l'inspection de nos balises :

La balise head contient ici la balise title mais peut contenir d'autres balises. La balise title permet de donner un titre à votre page. Dans les navigateurs de dernière génération, le contenu de la balise head est affiché sur l'onglet (voir le premier exemple de ce document, avec "voici mon site").

Nous arrivons enfin à l'essentiel, la balise body, tout votre site se trouvera entre <body> et </body>.

Premières balises, retour sur le premier exemple

```
<h1>Le titre de mon site</h1>  
<h2>Ici un second titre</h2>
```

Les balises h1 et h2 vous permettent d'afficher des titres sur votre page, h1 pour un titre "principal" et h2 pour les titres "secondaires" (la différence se fait surtout au niveau de la taille des caractères). Les titres sont aussi mis en gras "automatiquement".

Le titre de mon site

Ici un second titre

Un paragraphe doit être encadré par une balise p. Le changement de paragraphe entraîne un retour à la ligne et un saut de ligne.

Voici un premier paragraphe : Nec vox accusatoris ulla licet subditi in his malorum quaere
Caesaris implacabilitati sedisset, id velut fas iusque perpensum confestim urgebatur impleri.

Voici un second paragraphe : Ac ne quis a nobis hoc ita dici forte miretur, quod alia quaedam
omnes artes, quae ad avec un saut de ligne

Il existe aussi des balises qui sont à la fois ouvrantes et fermantes : <balise /> (noter la position du /)

Un premier exemple avec la balise qui permet un saut de ligne :

Il faut en effet bien comprendre qu'un retour à la ligne dans le code HTML n'entraîne pas de retour à la ligne sur votre page web; pour ce faire, il faut donc utiliser la balise
.

quod alia quaedam in hoc facultas sit ingeni, neque
e nos quidem huic uni studio penitus umquam dediti
nae ad avec un saut de ligne
 humanitatem per
donne

omnes artes, quae ad avec un saut de ligne
humanitatem pertinent, habent quoddam commune

Donner un style à sa page : les fichiers CSS

Pour ajouter un style à la page, vous devez ajouter une ligne à votre code HTML qui permettra à ce dernier de pointer sur un fichier CSS :

```
<link href="maFeuilledeStyle.css" rel="stylesheet" type="text/css" media="all" />
```

Cette ligne est à ajouter dans la balise head. Attention pour que cette ligne ait un effet, il faut que votre fichier de style (CSS) se nomme "mafeuilledeStyle.css" et que ce fichier soit placé dans le même dossier que votre fichier HTML. Vous pouvez très bien choisir un autre nom et un autre dossier, il faudra alors modifier la valeur de l'attribut href de la balise link en conséquence.

Pour votre information, il existe d'autres méthodes qui permettent d'appliquer un style à une page, notamment en insérant le code CSS directement dans la page HTML, mais ces méthodes sont déconseillées et je n'en parlerai pas dans ce document.

Reprenons ici le fichier CSS vu un peu au-dessus :

```
h1
{
    text-align: center;
    background-color: red;
}
h2
{
    font-family: Verdana;
    font-style: italic;
    color: green;
}
p
{
    font-family: arial;
    font-size: 12px;
}
```

Vous avez sans doute compris que h1 correspond à la balise h1 du fichier HTML. Dans le fichier CSS, h1 est suivi d'instructions entre accolades :

```
h1
{
    text-align: center;
    background-color: red;
}
```

Ces instructions (on parle plus exactement de propriétés) : text-align et background-color (nous détaillerons un peu plus loin leurs significations) s'appliqueront à tous les titres h1 de la page.

Il en est exactement de même pour h2 (instruction font-family, font-style et color) et pour p (font-family et background-color), tous les paragraphes de notre page se verront appliqués la valeur arial pour la propriété font-family et la valeur 12px pour la propriété font-size.

Remarquez le point virgule à la fin de chaque ligne, sans lui, votre ligne ne sera pas prise en compte par le navigateur.

Mais comment faire pour attribuer une propriété donnée à notre premier paragraphe et une autre à notre deuxième paragraphe ?

C'est ici qu'entrent en jeu les notions de classe (class) et d'identifiant (id).

Nous devons différencier le premier paragraphe et le second (dans le code HTML) pour cela, nous allons ajouter un attribut "id" à nos balises ouvrantes <p> ayant pour valeur "para1" pour le 1er paragraphe et "para2" pour le second paragraphe (j'ai choisi ces noms au hasard, vous pouvez les changer sans aucun souci).

```
<h2>Ici un second titre</h2>
<p id="para1">Voici un premi
quaerebatur acervis ut salte
aliquotiens fecere principes
velut fas iusque perpensum o
<p id="para2">Voici un secon
quod alia quaedam in hoc fac
```

Modifions maintenant notre fichier CSS :

```
h1
{
    text-align: center;
    background-color: red;
}
h2
{
    font-family: Verdana;
    font-style: italic;
    color: green;
}
#para1
{
    font-family: arial;
    font-size: 19px;
}
#para2
{
    font-family: arial;
    font-size: 12px;
}
```

Voici le résultat de notre modification



Je pense qu'en analysant le code HTML, le code CSS et le résultat, vous n'aurez aucune difficulté à comprendre de quoi il en retourne.

Pour le 1er paragraphe, la police aura une taille de 19 pixels le second paragraphe aura une taille de police de 12 pixels. Je pense que vous aviez tous compris que la propriété font-size vous permet de choisir la taille des caractères (il existe d'autres unités que px (pixel), mais nous n'aborderons pas ce sujet ici).

Le code CSS peut-être amélioré, en effet, la propriété font-family (qui vous permet de choisir le type de police de caractères) est identique pour para1 et para2, on peut donc écrire :

```
p
{
    font-family: arial;
}
#para1
{
    font-size: 19px;
}
```

```
#para2
{
    font-size: 12px;
}
```

Ce code signifie :

- pour tous les paragraphes de la page : **font-family: arial;**
- pour la balise qui a pour identifiant para1 : **font-size: 19px;**
- pour la balise qui a pour identifiant para2 : **font-size: 12px;**

J'espère que vous aurez compris que le résultat restera le même.

Un peu au-dessus je vous ai aussi parlé de la notion de classe, on peut donc aussi écrire :

dans le fichier HTML :

```
<p class="para1">Voici .....
```

dans le fichier CSS :

```
.para1
{
    font-size: 19px;
}
```

Remarquez le point (.) qui remplace le dièse (#) dans le fichier CSS

La différence entre "id" et "class" n'est pas très importante :

L'attribut "id" sert à donner un nom unique dans toute la page.

L'attribut "class" permet de donner le même nom à plusieurs reprises dans une même page.

Si nous avions eu un 3e paragraphe, nous aurions pu avoir : **<p class="para1">Voici un 3e paragraphe,** mais nous n'aurions pas pu avoir : **<p id="para1"> Voici un 3e paragraphe,** car le nom para1 a déjà été utilisé pour le 1er paragraphe.

Terminons l'étude de notre fichier CSS en donnant la signification des autres propriétés :

- text-align permet d'aligner le texte soit au centre, soit à gauche, soit à droite. Il permet aussi de justifier un texte (comme pour un traitement de texte), cette propriété est bien sûr utilisable avec d'autres balises (comme la balise p par exemple).
- background-color vous permet de colorier le fond. Elle fonctionne aussi avec d'autres balises, essayez cette propriété dans un paragraphe pour voir le résultat.
- font-style vous permet de choisir entre italique, gras ou normal (valeur par défaut) pour votre texte (encore une fois comme pour un traitement de texte).
- color vous permet de choisir la couleur de votre police de caractère.

Il existe beaucoup (mais alors beaucoup) d'autres propriétés en CSS (il y a des livres de plusieurs centaines de pages uniquement consacrées à CSS !), ne pouvant pas être exhaustif, je préfère m'arrêter là en ce qui concerne les fichiers CSS (nous reviendrons sur le CSS dans le deuxième document consacré au couple HTML-CSS : "HTML et CSS pour aller plus loin...."). En revanche, je vous encourage très vivement à télécharger un formidable document écrit par Eric Bellot "Mémento CSS 2.1 Bases et mises en forme" (<http://www.eric-bellot.fr/node/32>) (ce document est publié en Creative Common). Je suis persuadé qu'il vous sera d'une grande aide lors de vos développements futurs.

Quelques autres balises HTML

Sans chercher à être exhaustif, voici quelques balises HTML qui pourront vous être utiles :

La balise strong

`Ceci est un texte important`

La balise strong permet de mettre en évidence une partie importante d'un texte. Son principal effet est de mettre le texte concerné en gras.

La balise a

`Cliquez ici pour vous rendre sur mon autre page`

La balise a permet de créer des liens hypertextes, ce sont ces liens hypertextes qui vous permettent de "voyager" entre les pages d'un site ou entre les sites. Les liens hypertextes sont par défaut soulignés et de couleur bleue (modifiable grâce au CSS). La balise a possède un attribut href qui a pour valeur le chemin du fichier que l'on cherche à atteindre ou l'adresse du site cible (exemple : `Cliquez ici pour vous rendre sur google.fr`). Entre la balise ouvrante et fermante, on trouve le texte qui s'affichera à l'écran (c'est ce texte qui est souligné et de couleur bleue). La balise peut sans problème se trouver en plein milieu d'un paragraphe.

`<p id="para1">Voici un premier paragraphe : Nec vox accusatoris ulla licet subditicii in his malorum quaerebatur acervis ut saltem specie tenus crimina praescriptis legum committerentur, quod aliquotiens fecere principes saevi: sed quicquid CaesarisCliquez ici velut fas iusque perpensum confestim urgebatur impleri.</p>`

`<p id="para2">Voici un second paragraphe : Ac ne quis a nobis hoc ita dici forte miretur, quod alia quaedam in hoc facultas sit ingeni, neque haec`

Donnera ceci

Ici un second titre

Voici un premier paragraphe : Nec vox accusatoris ulla licet subditicii in his malorum quaerebatur acervis ut saltem specie tenus crimina praescriptis legum committerentur, quod aliquotiens fecere principes saevi: sed quicquid CaesarisCliquez ici velut fas iusque

La balise image

Comme vous devez déjà vous en douter, la balise image sert à insérer des.....images :

``

la balise img est à la fois ouvrante et fermante comme la balise br. Elle possède 2 attributs :

- src qui doit être égal au nom du fichier image (ou au chemin si le fichier image se trouve dans un autre dossier).
- alt qui doit être égal à une description de votre image (cet attribut est utilisé notamment par les systèmes de description des pages web utilisés par les non voyants), il faut donc systématiquement renseigner cet attribut.

`<body>
<h1>Le titre de mon site</h1>
<h2>Ici un second titre</h2>
<p id="para1">Voici un premier paragraphe : Nec vox accusatoris ulla licet subditicii in his malorum quaerebatur acervis ut saltem specie tenus crimina praescriptis legum committerentur, quod aliquotiens fecere principes saevi: sed quicquid CaesarisCliquez ici velut fas iusque perpensum confestim urgebatur impleri.</p>

Hello world !
<p id="para2">Voici un second paragraphe : Ac ne quis a nobis hoc ita dici forte miretur, quod alia quaedam in hoc facultas sit ingeni, neque haec dicendi ratio aut disciplina, ne nos quidem huic uni studio penitus umquam dediti fuimus. Etenim omnes artes,`

A large commercial airplane is shown from a low angle, flying upwards and away from the viewer. The plane is white with dark markings on the tail. It has two large engines mounted under the wings. The sky is a deep blue with several large, fluffy white clouds. The perspective makes the plane appear to be ascending steeply.

Je vous conseille d'utiliser des images de type jpeg, même si d'autres formats sont aussi acceptés. Il existe aussi d'autres attributs de la balise img qui vous permettent de donner une taille à votre image, cependant, je vous déconseille de les utiliser. Donnez directement les dimensions voulues à votre image à l'aide d'un logiciel de retouche d'image comme the gimp (libre et gratuit) : <http://www.gimp.org/>

13

Mise en page : les balises de type "block" et "inline"

Il faut bien comprendre que l'ordre des balises dans le code HTML a une grande importance. Pour vous en convaincre, inversez la balise h1 et h2 ou bien encore la balise img et la deuxième balise p. Alors, convaincus ?

Les balises sont donc affichées les unes après les autres, on parle du flux normal de la page.

C'est ici qu'entrent en jeu les balises de type "block" et les balises de type "inline".

Les contenus des balises de type "block" (p, div, h1, ...) se placent sur la page web les uns en dessous des autres.

Les contenus des balises de type "inline" (strong, img, span, a) se placent sur la page web les uns à côté des autres.

Nous avons vu dans l'exemple précédent que le second paragraphe se place en dessous de l'image.

En revanche, dans l'exemple suivant, nous avons 2 balises "inline" à la suite (a et strong)

```
<p id="para1">Voici un premier paragraphe : Nec vox accusatoris ulla licet subditicii in his malorum  
quaerebatur acervis ut saltem specie tenus crimina praescriptis legum committerentur, quod  
aliquotiens fecere principes saevi: sed quicquid Caesaris<a href="http://www.google.fr">Cliquez ici</a>  
velut fas iusque perpensum confestim urgebatur impleri.</p>  
<a href="http://www.airbus.com/"></a>  
<strong>Ceci est un avion</strong>  
<p id="para2">Voici un second paragraphe : Ac ne quis a nobis hoc ita dici forte miretur,  
quod alia quaedam in hoc facultas sit ingeni, neque haec dicendi ratio aut disciplina,  
ne nos quidem huic uni studio penitus umquam dediti fuimus. Etenim omnes artes,  
quae ad avec un saut de ligne <br/> humanitatem pertinent, habent quoddam commune vinculum, et  
quasi cognatione quadam inter se continentur.</p>
```

committerentur, quod aliquotiens fecere principes saevi



Ceci est un avion

Voici un second paragraphe : Ac ne quis a nobis hoc ita dici forte miretur, quod alia q
quae ad avec un saut de ligne

L'image de l'avion et le contenu de la balise strong se sont bien placés l'un à côté de l'autre.

Tout ceci est assez complexe et il y aurait beaucoup d'autres choses à dire. Pour creuser la question, je vous donne rendez-vous dans le deuxième document "HTML et CSS pour aller plus loin".

Faire des listes en HTML

Il existe 2 types de liste en HTML, les listes ordonnées et les listes non ordonnées :

Une liste ordonnée se place à l'intérieur des balises et . Ensuite chaque élément de la liste se place à l'intérieur des balises et , voici un exemple

```
<h2>Voici une liste ordonnée</h2>
```

```
<ol>
```

```
  <li>lundi</li>
```

```
  <li>mardi</li>
```

```
  <li>vendredi</li>
```

```
</ol>
```

voici le résultat sur la page

Voici une liste ordonnée

1. lundi
2. mardi
3. vendredi

Pour une liste non ordonnée, il suffit de remplacer la balise ol par la balise ul

```
<h2>Voici une liste non ordonnée</h2>
```

```
<ul>
```

```
  <li>lundi</li>
```

```
  <li>mardi</li>
```

```
  <li>vendredi</li>
```

```
</ul>
```

Voici une liste non ordonnée

- lundi
- mardi
- vendredi

Évidemment, l'aspect des listes (les puces ou la numérotation) est modifiable grâce à CSS (voir le document d'Eric Bellot).

Pourquoi vous parler des listes ?

Tout simplement parce que la combinaison des liens hypertextes et des listes vont nous permettre de "fabriquer" des menus.

Faire des tableaux en HTML

Les tableaux vont vous permettre de synthétiser des données, ils ont donc une importance certaine. La création d'un tableau débute avec la balise table. Cette balise possède 2 attributs : border (qui donne l'épaisseur des bordures du tableau) et cellspacing (qui donne l'espace entre les cellules du tableau, un cellspacing="0" vous permet d'avoir une bordure "simple" toutes les autres valeurs vous donneront une bordure "double").

Ensuite, les balises tr vous permettent de créer les lignes de votre tableau. Les balises td vous permettant de créer des colonnes. Tout cela est un peu confus, prenons un exemple pour éclaircir la situation :

```
<table border="2" cellspacing="0">
  <tr>
    <td>L1C1</td>
    <td>L1C2</td>
    <td>L1C3</td>
  </tr>
  <tr>
    <td>L2C1</td>
    <td>L2C2</td>
    <td>L2C3</td>
  </tr>
</table>
```

Ce qui donne ceci :

L1C1	L1C2	L1C3
L2C1	L2C2	L2C3

Je pense qu'avec un peu de réflexion vous comprendrez assez facilement comment cela fonctionne.

Il est aussi possible de faire fusionner des cellules :

- horizontalement grâce à l'attribut colspan de la balise td (la valeur donne le nombre de cellules à faire fusionner)
- verticalement grâce à l'attribut rowspan toujours de la balise td (la valeur donne le nombre de cellules à faire fusionner)

un exemple avec colspan

```
<table border="2" cellspacing="0">
  <tr>
    <td>L1C1</td>
    <td>L1C2</td>
    <td>L1C3</td>
  </tr>
  <tr>
    <td>L2C1</td>
    <td colspan="2">L2C2</td>
  </tr>
</table>
```


donne :

L1C1	L1C2	L1C3
L2C1	L2C2	

NB : il est bien sûr possible de centrer L2C2 dans la cellule grâce au CSS (text-align: center pour td)

un exemple pour rowspan

```
<table border="2" cellspacing="0">
  <tr>
    <td>L1C1</td>
    <td>L1C2</td>
    <td rowspan="2">L1C3</td>
  </tr>
  <tr>
    <td>L2C1</td>
    <td>L2C2</td>
  </tr>
</table>
```

donne :

L1C1	L1C2	L1C3
L2C1	L2C2	

Enfin, pour terminer, nous allons donner un titre à une colonne et à une ligne en remplaçant la balise td par la balise th

```
<table border="2" cellspacing="0">
  <tr>
    <th></th>
    <th>Titre C1</th>
    <th>Titre C2</th>
    <th>Titre C3</th>
  </tr>
  <tr>
    <th>Titre L1</th>
    <td>L1C1</td>
    <td>L1C2</td>
    <td>L1C3</td>
  </tr>
  <tr>
    <th>Titre L2</th>
    <td>L2C1</td>
    <td>L2C2</td>
    <td>L2C3</td>
  </tr>
</table>
```

résultat

	Titre C1	Titre C2	Titre C3
Titre L1	L1C1	L1C2	L1C3
Titre L2	L2C1	L2C2	L2C3

Voilà je vous laisse méditer sur ce dernier exemple