

Chapitre 1

HTML et CSS

Les langages informatique HTML et CSS sont les langages des pages Web. Ce sont deux langages complémentaires compris par les navigateurs. Ce ne sont pas des langages de programmation mais des langages de description.



Source : Wikipédia, W3C

I. Le langage HTML

Le format HTML (de l'anglais *HyperText Markup Language*, ou « langage de balisage hypertexte ») est un format textuel qui permet de **décrire le contenu et la structure d'une page Web**. Plusieurs versions du langage HTML se sont succédées depuis la première en 1992, celle utilisée actuellement est HTML5.

L'inventeur du HTML (et du Web)



Source : Paul Clarke

Tim Berners-Lee est considéré comme *l'inventeur du HTML en 1992*. C'est un informaticien britannique, qui a aussi inventé le Web (World Wide Web ou WWW) en 1989 lors de ses travaux au CERN. En juillet 2004, il est fait chevalier par la reine Élisabeth II pour ce travail et son nom officiel devient Sir Timothy John Berners-Lee. Depuis 1994, il préside le World Wide Web Consortium (W3C), organisme qu'il a fondé et qui est en charge de promouvoir la compatibilité des technologies du Web. Il est également lauréat du prix Turing 2016.

A. Un langage structuré par des *balises*

La description du contenu et de la structure d'une page Cette description se fait en utilisant ce qu'on appelle des *balises*. Ce sont ces balises qui vont indiquer si telle partie du document est un titre, ou bien un paragraphe, ou bien une image, ...

A faire **Activité 1 : Découverte de certaines balises HTML**

Quelques précisions sur les balises

Les balises sont délimitées par < et > (exemple <html> ou </p>).

Les balises ne contenant pas de slash (« / ») comme second caractère sont appelées des balises *ouvrantes* (exemple : <body>, <h2>), celles en contenant sont des balises *fermantes* (exemple : </h1>, </head>).

Une paire de balises ouvrante et fermante, ainsi que le contenu situé entre les deux, est appelée un *élément*.

Les commentaires sont délimités par les caractères <!-- et -->, ils ne sont pas interprétés par le navigateur et servent à l'auteur pour écrire des notes (destinées à lui-même ou à d'autres personnes).

B. Contenu d'un fichier HTML

La structure d'un fichier HTML est toujours identique.

```
<!DOCTYPE html>
<html>
  <head>
    <!-- les en-têtes de la page sont à écrire ici -->
  </head>
  <body>
    <!-- Le contenu de la page est à écrire ici -->
  </body>
</html>
```

On retrouve toujours :

- Une première ligne `<!DOCTYPE html>` qui indique que le fichier texte qui suit est écrit dans le langage HTML
- **La balise `<html>`** qui est la *balise principale du code*, elle englobe tout le contenu écrit en HTML du fichier. Dans cette balise `<html>` on distingue deux parties différentes :
 - **La balise `<head>`** qui est la balise d'*en-tête* du document et qui donne des informations générales sur la page Web : on peut y indiquer son titre, l'encodage des caractères, les mots-clés, son auteur, etc.
 - **La balise `<body>`** qui est la balise du *corps de la page* : elle encadre tout le contenu que le navigateur doit afficher à l'écran.

Il est conseillé d'indenter le code HTML pour s'y retrouver au milieu de toutes les balises, même si ce n'est pas une obligation comme ça l'est avec le langage Python.

C. Balises classiques

Outre les balises vues dans l'activité 1, d'autres sont importantes et énormément utilisées dans les pages Web. Il s'agit des balises permettant l'insertion d'un lien hypertexte (balise `<a>`), d'une image (balise ``) ou encore l'intégration d'une vidéo (balise `<iframe>`) dans une page Web. Leur utilisation est expliquée dans l'activité 2 qui va vous guider pour écrire votre première page Web (de l'année).

A faire **Activité 2 : Première page HTML**

Les balises classiques les plus importantes sont celles qui ont été vues dans les activités 1 et 2. Ce sont celles qui doivent être connues. Il en existe bien sûr beaucoup d'autres, vous en trouverez une partie dans le document fourni en annexe.

Identifiants et classes

Chaque balise peut recevoir deux attributs : **id** et **class**. Ces attributs permettent de cibler des éléments dans le document HTML. L'attribut `id` représente un identifiant, il doit être unique, autrement dit il n'est pas possible que deux éléments aient un attribut `id` ayant la même valeur dans le document. L'attribut `class` est similaire mais plusieurs éléments du fichier peuvent avoir la même valeur pour cet attribut : il sert donc à étiqueter plusieurs éléments du document. Ces étiquettes seront utilisées dans la

définition du style graphique des éléments en CSS. Sans style CSS associé, les informations de classe et d'identifiant n'ont aucun impact sur l'affichage de la page Web.

Balises neutres : <div> et

Il existe deux balises neutres (ou universelles) : <div> et . La première, <div>, est une balise de structure qui se comporte comme la balise <p> car elle représente un bloc de texte. La seconde, , est une balise de texte qui se comporte comme la balise ou <i>. Ces deux balises n'ont pas de rôle particulier en HTML : le contenu situé entre ces balises sera affiché comme si les balises n'étaient pas présentes. En revanche, vous pourrez appliquer à ces éléments le style graphique que vous souhaitez avec le langage CSS.

II. Le langage CSS

Le format CSS (de l'anglais *Cascading Style Sheets*, ou « feuilles de styles en cascade ») est un format textuel qui permet de **décrire la mise en forme des documents HTML (et XML)**. La première version du langage CSS date de 1996. Celui-ci est pris en charge par tous les navigateurs depuis les années 2000. La version utilisée actuellement est CSS3.



Source : Wikipédia, Rudloff

Un langage de style

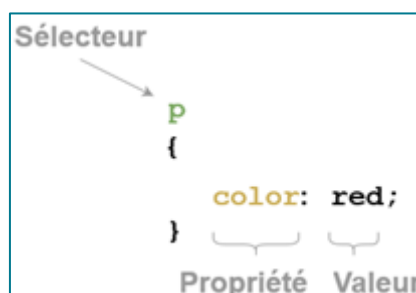
Le langage CSS permet de définir les propriétés graphiques des éléments HTML constituant une page Web : les polices utilisées, leurs tailles, les couleurs utilisées, la position des éléments dans la page, etc. Ce langage permet de séparer le contenu (décrit en HTML) de la forme (décrit en CSS) d'une page Web. En pratique, le fichier CSS sera associé au fichier HTML et le navigateur pourra alors savoir comment il doit mettre en forme le contenu de la page Web.

A. Contenu d'un fichier CSS

Un fichier CSS contient les propriétés graphiques des éléments HTML. On y indique successivement pour chaque sélecteur, les règles CSS que l'on souhaite lui appliquer.

Anatomie d'une règle CSS

L'anatomie d'une règle CSS est la suivante :



Source : developer.mozilla.org

Ici `p` est le sélecteur, et on déclare entre deux accolades `{` et `}` toutes les règles que l'on souhaite appliquer au sélecteur. La déclaration d'une règle est toujours de la forme `propriété: valeur;` (ne pas oublier de terminer par un point-virgule). Dans cet exemple, on indique que le contenu des balises `<p>` du fichier HTML devra être écrit en rouge.

Il est possible de déclarer plusieurs propriétés pour un sélecteur, comme dans l'exemple ci-dessous.

```
h1
{
    text-align: center;
    color: white;
    background-color: rgb(0,121,143); /*couleur stylée*/
}

p
{
    color: red;
    font-family: Arial;
}
```

Les commentaires en CSS sont à écrire entre /* et */, ils ne sont pas interprétés par le navigateur et servent à l'auteur à écrire des notes.

Associer un fichier CSS à un fichier HTML

Il est très simple d'associer une feuille de style CSS à un fichier HTML : pour cela, il suffit de rajouter dans l'en-tête du fichier HTML (dans les balises <head>...</head>) une balise <link> de la façon suivante :

```
<link href="style.css" rel="stylesheet" type="text/css"/>
```

L'attribut href permet, comme pour une balise <a>, de donner l'emplacement du fichier CSS à utiliser.

Vous allez avoir un aperçu de certaines des règles CSS grâce à l'activité 3.

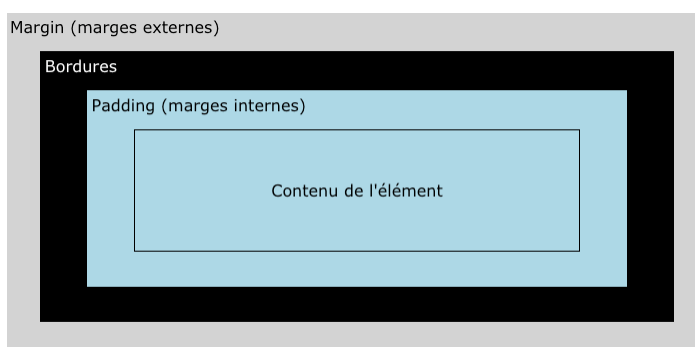
A faire **Activité 3 : Découverte du langage CSS**

B. Quelques propriétés CSS

Il existe bien entendu énormément de propriétés CSS, il est impossible ici de toutes les citer. Celles vues dans l'activité 3 permettent déjà de faire beaucoup de choses. Le document donné en annexe regroupe quelques-unes des propriétés les plus importantes. Donnons ci-dessous quelques généralités sur celles qui ont été vues dans l'activité 3.

Modèle de boîte

Tout élément HTML possède une boîte rectangulaire qui délimite son contenu. Lorsque le navigateur affiche ce dernier, il définit une boîte rectangulaire possédant plusieurs zones distinctes.



Source : Pierre Giraud (<https://www.pierre-giraud.com/html-css-apprendre-coder-cours/modele-boite/>)

Les propriétés `width` et `height` permettent de définir la largeur et la hauteur de la boîte « contenu ».

La propriété `padding` permet de définir la taille des marges internes.

La propriété `border` permet de définir des bordures pour notre élément.

La propriété `margin` permet de définir la taille des marges externes.

On peut spécifier séparément les tailles à gauche (`left`), à droite (`right`), en haut (`top`) et en bas (`bottom`) de ces trois dernières propriétés : par exemple, `margin-top` est la propriété définissant la marge externe gauche de la boîte.

Longueurs

On peut définir des longueurs dans beaucoup de propriétés CSS (taille d'une police, d'une bordure, d'une marge, ...). Les valeurs acceptées sont exprimées en pixel (`px`), en point (`pt`) et même en pourcentage (%) de la taille de l'élément contenant.

Couleurs

Il est possible d'utiliser leur nom (`red`, `blue`, `pink`, ...), d'utiliser la fonction `rgb(r, v, b)` où `r`, `v` et `b` sont les valeurs décimales comprises entre 0 et 255 des composantes RVB du pixel ou d'utiliser la notation hexadécimale `#rrvvbb` où `rr`, `vv` et `bb` sont les valeurs hexadécimales des composantes RVB du pixel. Par exemple, `rgb(255, 0, 255)` donne la couleur magenta que l'on peut aussi écrire `#ff00ff` puisque `ff` est l'écriture hexadécimale de 255 et `00` celle de 0.

Cas particuliers des attributs `id` et `class`

Si on veut attribuer un style à la balise étiquetée par l'attribut `id="ressources"`, il suffit de faire précéder le nom de l'identifiant par le symbole `#`.

```
#ressources
{
    /*déclaration des règles ici*/
}
```

Si on veut attribuer un style aux balises étiquetées par l'attribut `class="formations_diplomes"`, il suffit de faire précéder le nom de la classe par un point.

```
.formations_diplomes
{
    /*déclaration des règles ici*/
}
```

Retenez que si vous souhaitez *cibler un élément* particulier il faut utiliser l'attribut `div`, alors que si vous souhaitez en *cibler plusieurs* il faut utiliser l'attribut `class`.

Créer des apparences dynamiques

Il est possible de modifier l'apparence d'éléments grâce à CSS. On utilise pour cela des pseudo-formats (ce nom n'a pas beaucoup d'importance).

Le pseudo-format sans doute le plus utilisé s'appelle `:hover` qui signifie « survoler ».

```
a:hover
{
    color: white;
    background-color: orange;
}
```

Avec cette règle CSS, si on survole les liens de la page Web avec la souris, ceux-ci apparaîtront en blanc sur fond orange.

C. Compléments sur les règles CSS

Si vous êtes ici, vous avez déjà vu beaucoup de choses sur le langage CSS. Elles vous suffiront pour créer des pages Web d'apparence plutôt jolie mais ne vous permettront pas de tout faire. Lisez la fin de ce document pour savoir que ces fonctionnalités existent et pouvoir éventuellement y revenir si besoin (dans le cadre d'un travail, d'un projet). L'activité 4, optionnelle, présente l'utilisation des balises neutres `<div>` et ``.

Sélectionner plusieurs éléments

Il est possible de sélectionner plusieurs éléments en les séparant par des virgules. Dans cet exemple, on colore en bleu les titres `<h1>` et les paragraphes `<p>`.

```
h1, p
{
    color: blue;
}
```

Imbrication de sélecteurs

Il est aussi possible de cibler précisément des sélecteurs imbriqués dans d'autres en les séparant par des espaces (mais sans virgule !). Dans cet exemple, seuls les liens (balise `<a>`) situés à l'intérieur des balises `` seront ciblés (et pas les autres liens).

```
li a
{
    color: pink;
    background: yellow;
}
```

A faire Activité 4 : Pour aller plus loin en CSS

Utilisation des balises neutres

Les balises neutres `<div>` et `` sont des conteneurs génériques, c'est-à-dire qu'elles ne servent pas à préciser la nature du contenu. La balise `<div>` sert de conteneur à plusieurs éléments (on peut regrouper par exemple un paragraphe, une image puis un autre paragraphe entre les balises `<div>` et `</div>`) alors que la balise `` sert de conteneur interne à un élément (on peut encadrer par exemple une partie seulement d'un paragraphe entre les balises `` et ``). Pour appliquer un style CSS à des éléments `div` ou `span`, on leur ajoutera le plus souvent un attribut `id` ou `span` pour lesquels on définira leur style CSS.

A retenir

Le langage **HTML** est un langage de description des pages Web. Ces dernières sont des documents structurés par des **balises** suivant un ensemble de règles. Après avoir défini la structure d'une page, on peut définir sa charte graphique au moyen du langage **CSS** qui permet d'ajuster finement toutes les caractéristiques graphiques d'une page Web.

Ressources :

- *Numérique et Sciences Informatiques*, T. BALABONSKI, S. CONCHON, J.-C. FILLIATRE, K. NGUYEN, Ellipses.
- *Réalisez votre site web avec HTML 5 et CSS 3*, Mathieu Nebra, OpenClassrooms, EYROLLES ;
- Cours OpenClassrooms : <https://openclassrooms.com/fr/courses/1603881-apprenez-a-creer-votre-site-web-avec-html5-et-css3>
- Pierre Giraud : <https://www.pierre-giraud.com/html-css-apprendre-coder-cours/>