

Leçon 1.2 – HTML et CSS

s1 -----

s2 -----

Dans cette partie, nous allons introduire le langage CSS. Nous allons voir comment l'utiliser au sein d'une page web, découvrir comment écrire des règles CSS : la notion de sélecteur qui permet de cibler une ou plusieurs parties du document HTML, la notion de propriété et la notion de valeur. Nous ferons un focus particulier sur la propriété qui permet de positionner des éléments les uns par rapport aux autres.

CSS est l'acronyme de Cascading Style Sheets : feuilles de style en cascade). C'est un langage informatique qui sert notamment à décrire la présentation des documents HTML, tant du point de vue des polices, tailles, couleurs... que du positionnement des différents éléments constitutifs de la page HTML.

CSS3 est le dernier standard. Une certaine prudence est de mise, car son implémentation dépend des navigateurs. Il est conseillé de vérifier l'aspect de la page que l'on vient de créer sur plusieurs navigateurs.

s3 -----

On peut utiliser CSS principalement de deux façons :

soit en écrivant les instructions CSS décrivant les styles directement dans la page HTML à l'intérieur d'une balise `<style>`, comme le montre la première illustration. Ici on définit la couleur de police du document ainsi que la couleur de fond

soit dans un simple fichier texte externe (possédant l'extension `.css`) et relié à la page HTML par la balise `<link>` comme le montre la seconde illustration. Ici, en l'occurrence un fichier appelé `styles.css`.

Notons que la balise `link` est autofermante, c'est-à-dire qu'il n'y pas de balise fermante `/link` et que toutes les informations sont contenues dans les attributs.

Notons qu'en HTML 5, il y a deux manières d'écrire les balises autofermantes.

soit par un simple caractère supérieur à, soit en le faisant précéder par un slash. Cette dernière écriture est héritée du format XHTML.

s4 -----

Une feuille de style externe introduite par la balise `<link>` ou une liste d'instructions CSS introduites par la balise `<style>` représente un ensemble de règles. Une règle est en trois parties D'abord le sélecteur qui indique au navigateur quelles parties du document sont affectées par la règle (dans l'exemple présenté, c'est l'élément "body")

Ensuite, la propriété, qui spécifie quel aspect de l'affichage est paramétré (dans l'exemple présenté, il y a deux règles qui s'appliquent à l'élément `body` et qui concernent respectivement les propriétés

'color' (c'est-à-dire le texte du corps du document) et 'background-color' (c'est-à-dire la couleur de fond du corps du document)

Enfin, la valeur de cette propriété (ici 'purple' pour la propriété body et '#d8da3d' c'est-à-dire le code hexadécimal d'une couleur verdâtre pour la propriété background-color).

À titre d'exemple vous pouvez d'ores et déjà reproduire les instructions présentées ici.

s5 -----

Le sélecteur peut désigner simplement une balise. Mais on peut cibler plus finement certaines classes d'éléments, certains ensembles d'éléments de la page HTML. L'écriture du sélecteur peut alors être un peu plus complexe que précédemment. Voici ci-dessous quelques exemples.

Détaillons trois exemples parmi ceux présentés :

Premier exemple. À l'intérieur d'une balise, par exemple ici p, on peut introduire un attribut class (voir la troisième colonne de cette ligne). Cet attribut indique que cet élément fait partie d'une classe particulière, ici la classe introduction. Il devient alors possible d'appliquer le même style à tous les éléments de cette classe introduction, quelque soit le type d'élément, pour peu que le style s'applique effectivement en définissant un sélecteur .introduction ou (c'est l'exemple 2) seulement à tous les éléments de type p en définissant un sélecteur p.introduction

Troisième exemple. À l'intérieur d'une balise, par exemple ici un conteneur div et un titre h1, on peut introduire un attribut id permettant de le repérer. Il devient alors possible d'appliquer un style à cet élément unique en définissant un sélecteur #titre, ou à l'élément de type p portant cet identifiant.

Nous allons mettre en oeuvre ces notions dans l'exemple qui va suivre. Mais avant, traitons du positionnement des éléments.

s6 -----

Considérons les boîtes de type "block" : principalement l'élément div, les titres h1, h2, h3, h4, h5, h6, le paragraphe p. Les éléments de type block peuvent contenir à la fois d'autres éléments de type block, des éléments de type inline ou du texte.

Par défaut, les navigateurs affichent les éléments de type block du document HTML dans l'ordre dans lequel il les lit et il aligne la marge gauche de chaque élément sur la marge gauche de l'élément conteneur, ici le document lui-même.

Considérons les boîtes de type inline : principalement span, a, img, code. Les éléments de type inline ne peuvent contenir que d'autres éléments de type inline ou du texte.

Par défaut, le navigateur affiche les éléments de type en-ligne du document HTML dans l'ordre dans lequel il les lit, les uns à la suite des autres sur une même ligne. Et il aligne la marge gauche du premier élément sur la marge gauche de l'élément conteneur, ici le document lui-même.

À vous d'essayer ce petit exemple.

s7 -----

Examinons à présent trois types de positionnement des éléments, les uns par rapport aux autres.

Le positionnement relatif permet d'inscrire un contenu dans le flux normal (c'est à dire dans son positionnement par défaut bloc ou en ligne), puis de le décaler horizontalement ou verticalement. Le contenu suivant n'est pas affecté par ce déplacement, ce qui peut donc entraîner des chevauchements comme sur l'exemple présenté.

À vous d'essayer ce petit exemple.

s8 -----

Le positionnement flottant retire une boîte du flux normal (c'est à dire de son positionnement par défaut bloc ou en ligne) pour la placer le plus à droite possible (float: right) ou le plus à gauche possible (float: left) possible à l'intérieur de son conteneur. Dans cet exemple, c'est la boîte jaune qui est flottante. Le navigateur doit la placer en premier. Nous avons introduit un peu plus de texte dans la boîte verte pour que vous puissiez voir comment le navigateur se débrouille pour éviter que le texte de la boîte verte soit masqué par celui de la boîte jaune.

À vous d'essayer ce petit exemple.

s9 -----

Le positionnement absolu « retire » totalement le contenu concerné du flux normal (c'est à dire de son positionnement par défaut bloc ou en ligne) : sa position est déterminée par référence aux limites de l'objet qui le contient. Celui-ci peut-être :

- une autre boîte elle-même positionnée (position relative ou absolue)
- par défaut le bloc conteneur initial c'est-à-dire la fenêtre du navigateur.

Comme dans l'exemple présenté ici.

Au passage, nous découvrons la propriété width qui donne une largeur de 100 pixels à tous les objets de classe jaune.

À vous d'essayer ce petit exemple.

s10 -----

CSS est vaste. Ce module ne prétend qu'à un simple tour d'horizon. Pour une véritable formation à ce langage, je vous renvoie aux liens proposés tout à la fin de ce chapitre.

Sachez enfin, comme pour HTML, que des outils de validation existent, tels que celui proposé ici.

s11 -----

Plutôt que de se lancer dans une leçon fastidieuse sur la syntaxe de CSS, je vous propose de découvrir un peu mieux ce langage à travers la création d'une page et de créer les fichiers HTML et CSS pour l'exemple ci-dessous en respectant toutes les indications qui sont données.

Pour cela, il vous faudra chercher par vous-même des éléments concernant les propriétés border, margin et padding

Évidemment il y a plusieurs façons de développer ce jeu. Y compris avec d'autres propriétés que celles présentées dans les dernières planches.

Le cheminement est le suivant

- créer tout d'abord la page HTML avec autant de conteneurs (de type block ou inline, à vous de voir)
- enrichir une feuille de style progressivement : la première chose à régler étant le positionnement de la carte et de la liste de gauche.
- pensez à vérifier le rendu de votre page web avec différents navigateurs et à le valider grâce aux liens donnés précédemment.

s11 -----

Bon travail. Je vous donne rendez-vous dans la prochaine partie qui traitera du langage JavaScript.