

Mémo CSS (versions 1 à 3, simple)

Généralités :

CSS (pour Cascading Style Sheets) permet de créer des feuilles de style en regroupant des informations de mise en forme applicables à des documents (HTML essentiellement).

HTML et CSS

Les feuilles de style peuvent être internes ou externes au document HTML. La seconde option est pratique lorsque plusieurs documents doivent avoir la même présentation ; elle rend également les mises à jour (de forme) très rapides. Cependant, le document devient dépendant d'un élément qui lui est extérieur. Séparé de sa feuille de style, il s'affichera, mais sans aucune mise en forme personnalisée.

Dans tous les cas, les balises concernant les feuilles de styles doivent s'insérer dans l'entête du document HTML entre <head> et </head> :

feuille de style complète intégrée au document :

```
<style type="text/css">
<!--
    elle sera rédigée ici
-->
</style>
```

lien vers une feuille de style externe au document :

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

Grammaire CSS

Une feuille de style est un ensemble de règles. Voici un exemple de règle CSS permettant d'afficher les titres de niveau h2 en Verdana bleu dans un corps de 14 pixels :

```
h2 {
  color: blue;
  font-family: Verdana;
  font-size : 14px;
}
```

Une règle de style est composée de 2 parties : un **sélecteur** (ici **h2**) et une ou plusieurs **déclarations** (**color: blue;**) entre **accolades**. Une **déclaration** a elle-même deux parties : une **propriété** (color) et une **valeur** (blue). Elle se termine toujours par un **point-vigule**.

1/ Sélecteurs et classes

Les règles de style peuvent s'appliquer à différents niveaux d'un document HTML :

- soit à tous les éléments de même type (dans notre exemple, la règle s'applique à tous les éléments <h2> du document),
- soit à certains d'entre eux repérés à l'intérieur du document HTML par l'attribut class.

Par exemple, **<p class="chapo">** utilise un style personnalisé (une classe) dont la règle s'écrira :

```
.chapo {
  propriété: valeur;
  propriété:valeur;
}
```

Attention à :

- placer un point devant le nom,
- utiliser ni caractères accentués, ni espaces,
- donner des noms correspondant à la structure logique du texte, (et nommés "chapo", "legende", "corps_de_texte", "titre", "sous-titre", "auteur", etc.) que liés à la mise en forme (des styles nommés "texterouge", "gras", etc.).

On peut aussi utiliser la cascade des éléments en séparant parents et descendants par une espace :
 .recette .ingredient {...}

Pour pointer un enfant direct plutôt qu'un descendant : **ul > li** {...}

L'élément qui suit un autre : **h1 + p** ou bien celui qui précède **h1 ~ p**

il existe d'autres règles pour les sélecteurs : **:first** pour le premier par exemple, ou **:not(.intertitre)** pour exclure.

On peut enfin sélectionner un élément par un de ses **attributs**, ce qui est pratique notamment pour les **formulaire** : **input[type="text"] { color: red; }**

Deux éléments HTML pratiques : et <div>

L'élément permet d'appliquer un style à l'intérieur d'une autre balise.

<p class="chapo">Ce paragraphe est en style chapo, mais ce mot en style fun.</p>

L'élément <div> permet quant à lui de regrouper d'autres éléments sous un même style.

```
<div class="fun">
<p>ce texte est en style fun, mais</p>
<table width="150" border="2" bgcolor="#FFCC33">
  <tr>
    <td>ce tableau</td>
    <td>aussi</td>
  </tr>
</table>
</div>
```

Commentaires

Pour s'y retrouver dans une feuille de style, on peut insérer des commentaires entre /* et */

2/ unités et couleurs

Unités

Elles s'expriment en système décimal (sous la forme d'un nombre algébrique entier : 2, 23, -4 ou fractionnaire 0.5, -4.7, 10.6) et doivent être suivies d'un nom d'unité :

l'**em** correspond au corps de texte hérité (si le parent est en corps 16px (html par défaut), 1em = 16px, 2em = 32px. Mais si le parent est en corps 12px, 1em=12px, etc.)

le **rem** (root em) correspond au corps de html.

ou pixel (**px**) - pris comme unité par défaut quand on ne suffixe pas l'unité, ou pourcentage (**%**), on peut même utiliser le millimètre (**mm**), centimètre (**cm**), pouce (**in**), point (**pt**), pica (**pc**) ;

Couleurs

Pour représenter une couleur, on peut :

- utiliser une des 16 couleurs prédéfinies ;
- définir précisément la couleur en indiquant ses composantes de rouge, de vert et de bleu (RVB) en hexadécimal ou en décimal.

1 - Les noms des 16 couleurs prédéfinies sont : aqua, black, blue, fuchsia, gray, green, lime, maroon, navy, olive, purple, red, silver, yellow, teal, white, yellow.

2 – Indiquer les valeurs RVB en hexadécimal s'écrit #rrvvbb où rr correspond à la valeur hexadécimale du rouge, vv à celle du vert et b à celle du bleu.

L'écran émet de la lumière, et la couleur, dans l'espace de la lumière, s'obtient par « synthèse additive », le blanc correspond à l'intensité maximale de rouge, de vert et de bleu, et le noir par simple absence de lumière. En bref, c'est le contraire de ce qui se passe avec des encres.

La machine comptant sur des octets (huit valeurs binaires, donc 2^8 , soit 256), il faut utiliser une base un peu spéciale pour compter. En hexadécimal, on compte des octets, et donc jusqu'à 255 (256 en tenant compte du zéro). Pour cela, il faut « deux mains de seize doigts. » C'est pourquoi on a ajouté au système décimal des lettres figurant les 11^e à 16^e « doigts. » Ainsi 0=0, 1=1, 2=2, 3=3 ... 9=9, et pour les unités supérieures à 10, 10=A, 11=B, 12=C, 13=D, et ce jusqu'à 15 (F). Deux X valent une couleur primaire Rouge vert ou bleu (prenant donc une valeur possible de 0 à 255). Et on obtient une couleur complexe en additionnant trois couleurs primaires (c'est la synthèse additive). C'est pourquoi les couleurs sont codées sur 3 paires de nombres hexadécimaux.

Par exemple : #4D708F veut dire que la couleur est composée de 4D ($77/256^e$) de rouge, 70 ($112/256^e$) de vert et 8F ($148/256^e$) de bleu.

3 – On peut aussi rester dans notre bon vieux système décimal. Alors, il faut indiquer les valeurs RVB en décimal ainsi : rgb(r,v,b) où r correspond à la valeur décimale du rouge (un chiffre entre 0 et 255 ou un pourcentage), v à celle du vert et b à celle du bleu.

avec éventuellement l'opacité entre 0 et 1 : rouge 50% = rgba(255,0,0,0.5);

Voici 4 façons d'obtenir du rouge dans la syntaxe CSS :

color : red;

color : #ff0000;

color: rgb(255,0,0);

color: rgb(100%, 0%, 0%);

color: rgb(255,0,0,1);

3/ quelques propriétés et valeurs des feuilles de style

Propriété	Valeurs(s)	Description	Exemple
Display			
display	block, inline, inline-block, none, flex, table, etc.	type d'affichage pour l'élément.	display: block;
Positions			
position	static, relative, absolute, fixed	position du bloc	position: absolute; (nécessite un parent en position: relative ou body)
top	valeur	position du bloc à partir du haut de la fenêtre	top: 10px;
right	valeur	position du bloc à partir de la droite	right: 20%;
bottom	valeur	position du bloc à partir du bas	bottom: 0;
left	valeur	position du bloc à partir de la gauche	left: 1em;
width	valeur, auto	Largeur du bloc	width: 100%;
height	valeur, auto	hauteur du bloc	height: auto;
z-index	valeur entière positive ou négative	superposition des blocs	z-index: 1;
Blocs, boîtes			
margin-top (ou right, bottom, left)	valeur ou pourcentage (%)	marges	margin-right: 10px;
padding-top (ou right, bottom, left)	valeur ou pourcentage (%)	remplissage du bloc	padding-left: 10px;
border-top-width (ou right, bottom, left)	medium, thin, thick ou valeur ou pourcentage (%)	taille de la bordure	border-top-width: 10px;
border-style	none, dotted, dashed, solid, double, groove, ridge, inset, outset	style de la bordure	border-style: ridge;
border-color	nom, valeur hexadécimale ou valeur RVB de la couleur	couleur de la bordure	border-color: red;
border-radius	valeur, %	rayon de l'angle pour un	border-radius: 5px;

		arrondi	
float	auto, left, right	autorise l'habillage d'un bloc	float: right;
clear	none, left, right, both	Autorise ou interdit l'alignement d'un bloc par rapport à un autre	clear: left;
box-shadow	valeurs, couleur	ombre du bloc, décalage, taille, couleur	box-shadow: -1px -1px 0 #555;
Polices de caractères			
font-family	serif, sans-serif, cursive, fantasy, monospace, ou nom de la police (Arial, Verdana, "Stone Sans", etc...)	type de police	font-family: Arial, Helvetica, sans-serif;
font-size	xx-small, x-small, small, medium, large, x-large, xx-large, larger, smaller ou taille en pixels (px), cm (cm), mm (mm), points (pt), pourcentage (%)	taille de la police	font-size: 12px;
font-weight	normal, bold, bolder, lighter ou valeur de 100 à 900 (par tranche de 100)	poids de la police	font-weight: bold;
font-style	normal, italic, oblique	style de la police	font-weight: italic;
font-variant	normal, small-caps	variante de la police	font-variant: small-caps;
+ règle @font-face	règle pour importer une police (woff, truetype, svg) des formats différents pour les différents navigateurs <pre>@font-face { font-family: 'OpenSans'; src: url('OpenSans-Regular-webfont.eot'); src: url('OpenSans-Regular-webfont.eot?#iefix') format('embedded-opentype'), url('OpenSans-Regular-webfont.woff') format('woff'), url('OpenSans-Regular-webfont.ttf') format('truetype'), url('OpenSans-Regular-webfont.svg#OpenSansRegular') format('svg'); font-weight: normal; font-style: normal; }</pre>		
Texte			
line-height	valeur en pixels (px) ou pourcentage (%)	hauteur de la ligne	line-height: 15px;
vertical-align	baseline, sub, super, top, text-top, middle, bottom, text-bottom ou valeur en pourcentage (%)	alignement vertical	vertical-align: top;
text-align	left, right, center, justify	alignement du texte	text-align: left;

text-indent	valeur, %	retrait de la première ligne	text-indent: 10px;
text-decoration	none, underline, overline, blink, line-through	décoration du texte	text-decoration: overline;
text-transform	none, capitalize, uppercase, lowercase	casse du texte	text-transform: none;
text-shadow	None ou nom, valeur hexadécimale ou valeur RVB de la couleur	ombrage	text-shadow: red;
word-spacing	valeur	espace entre les mots	word-spacing: 10px;
letter-spacing	valeur	espace entre les lettres	letter-spacing: 2px;
text-shadow	valeurs, couleur	ombre du texte, décalage, taille, couleur	text-shadow: -1px -1px 0 #555;
columns	nombre	nombre de colonnes	columns: 2
Couleurs et arrières plans			
color	nom, valeur hexadécimale ou valeur RVB de la couleur	couleur du texte	color: green;
background-color	nom, valeur hexadécimale ou valeur RVB de la couleur	couleur du fonds	background-color: yellow;
background-image	URL du fichier	image d'arrière-plan	background-image: url('images/portrait.gif');
background-repeat	repeat, no-repeat, repeat-x, repeat-y	répétition de l'arrière-plan	background-repeat: no-repeat;
background-attachment	scroll, fixed	effet de filigrane	background-attachment: fixed;
background-position	top, center, bottom left, center, right	position de l'arrière-plan (verticale et horizontale)	background-position: center center;
background-size	largeur hauteur, auto, 100%, cover, contain	taille de l'arrière plan	background-size: cover;
Listes, classification			
list-style-type	disk, circle, square decimal (1, 2, ...) lower-roman (i, ii, ...) upper-roman (I, II, ...) lower-alpha (a, b, ...) upper-alpha (A, B, ...)	format de la puce	list-style-type: decimal;
list-style-image	url du fichier image sous la forme url()	utilisation d'une image comme puce	list-style-image: url(puce.gif);
list-style-position	inside : puce alignée sur la marge gauche du texte outside : puce en marge du texte	position de la puce	list-style-position: inside;

Et beaucoup d'autres choses palpitantes comme par exemple :

les dégradés (gradients)

les animations (transition)...

les règles @media pour l'affichage « responsive » (@media only screen and (max-width: 768px) { })