

Techno Web

Feuilles de Style CSS

CSS : introduction

- D'un point de vue "conception" (ergonomique et programmation), HTML non satisfaisant.
 - la maintenabilité non aisée, les modifier pour en changer l'aspect complique encore plus
 - d'autant plus à cause des répercussions
- Les feuilles de style (CSS = Cascading Style Sheet) permettent :
 - Principe en IHM : séparation du contenu de la présentation :
 - Séparation du contenu et de la mise en forme.
 - Modifier l'aspect d'un page ou d'un site sans en modifier le contenu.
 - Modifier facilement l'aspect d'une page.
 - Cohésion de la présentation tout au long du site (feuilles de style externes).
 - Réduire le temps de chargement des pages.
 - Correction de certains écueils d'HTML :
 - compréhensible, simple et logique par rapport au HTML.
 - contrôle des polices, de la distance entre les lignes, des marges et des indentations (sans devoir utiliser de tableaux...)
 - moins de code, mieux structuré, et donc des sites plus faciles à maintenir et à faire.

CSS : introduction

- Utilisation de plusieurs feuilles de style (héritage)
- Syntaxe :
 - BALISE { Propriété : valeur}
 - Exemple : H2 { color : green ; } indique que tous les titres de niveau 2 sont en vert. Ici la propriété est color mais il en existe une bien plus !
- Une feuille de style CSS peut être définie de différentes façons :
 - Dans un document, il est possible de donner l'URI (url) d'un fichier externe. La syntaxe est la suivante :
<link rel="stylesheet" type="text/css" media="screen" href="fichier.css" />. La balise LINK doit être utilisée entre les balises <head> et </head>
 - @import "fichier.css", entre les balises <style>...</style> et avant la règle suivante
 - A l'intérieur du document lui-même (toujours dans la partie entête) par l'utilisation de balises STYLE : <style type="text/css"> h1 { color: green ; }</style>.
 - Par définition du style dans une balise qui supporte un ajout de style, comme par exemple <li style="color:red;"> ...

CSS : autres éléments de syntaxes

- Les commentaires
 - Entre /* */
 - <!-- et --> acceptés (commentaire HTML) mais par de commentaire permet de cacher le CSS pour un navigateur qui ne les comprend pas
- Blocs : définis par { }
- Sélecteur { }
- Déclaration de propriétés :
 - Plusieurs lignes par bloc possible : séparation par ;
 - Dernière apparition d'une propriété valide qui compte
 - exemple/declaration1.html et exemple/declaration2.html
- Non prise en compte de ce qui est non valide
 - Propriétés inconnues
 - Valeurs invalides (unité !)
 - exemple/declaration3.html

CSS : valeurs

- Les entiers et les nombres

- unités

ABBR.	UNIT	EXAMPLE	NOTES
cm	Centimeters	2.5cm	
em	Ems	3em	1 em equals the font point size.
in	Inches	1in	
mm	Millimeters	25mm	
pc	Picas	6pc	1 inch = 6 picas
px	Pixels	96px	
pt	Points	72pt	1 inch = 72 points
ex	X-heights	2ex	1 x-height usually equals the height of the letter x.

CSS : valeurs

- Les pourcentages (par rapport à quelque chose)
- URI (url) : url("http://deptinfo.unice.fr/~renevier/L2/ipw.png")
- Les compteurs : exemple/unite.html
- Les chaînes
 - entre " "
 - Caractères spéciaux : \codeascii
 - \22 ou \' ; \27 ou \'
 - "bla bla bl a 'bla' bla bla" : valide
- Angles : deg, rad ou grad (feuilles de style auditives)
- Temps : s ou ms (feuilles de style auditives)
- Fréquence : Hz ou kHz (feuilles de style auditives)

CSS : valeurs (couleurs)

KEYWORD	RGBHEX	RGB INTEGER	RGB PERCENTAGE
aqua	#00ffff	rgb(0,255,255)	rgb(0%,100%,100%)
black	#000000	rgb(0,0,0)	rgb(0%,0%,0%)
blue	#0000ff	rgb(0,0,255)	rgb(0%,0%,100%)
fuchsia	#ff00ff	rgb(255,0,255)	rgb(100%,0%,100%)
gray	#808080	rgb(128,128,128)	rgb(50%,50%,50%)
green	#008000	rgb(0,128,0)	rgb(0%,50%,0%)
lime	#00ff00	rgb(0,255,0)	rgb(0%,100%,0%)
maroon	#800000	rgb(128,0,0)	rgb(50%,0%,0%)
navy	#000080	rgb(0,0,128)	rgb(0%,0%,50%)
olive	#808000	rgb(128,128,0)	rgb(50%,50%,0%)
purple	#800080	rgb(128,0,128)	rgb(50%,0%,50%)
red	#ff0000	rgb(255,0,0)	rgb(100%,0%,0%)
silver	#c0c0c0	rgb(192,192,192)	rgb(75%,75%,75%)
teal	#008080	rgb(0,128,128)	rgb(0%,50%,50%)
white	#ffffff	rgb(255,255,255)	rgb(100%,100%,100%)
yellow	#ffff00	rgb(255,255,0)	rgb(100%,100%,0%)

- Et plus encore : http://www.w3schools.com/css/css_colornames.asp

CSS : Sélecteurs

- Pattern : élément (sélecteur de type)
 - tous les éléments « désignés » apparaissent avec le style défini entre { }
 - Ex :
 - p {color: white ; background-color: black;}
Écrit le paragraphe en blanc sur fond noir
- Pattern : , (regroupement)
 - Regroupe les déclarations
 - Ex :
 - H1, H2, H3 { color: blue;}
Les titres de niveaux 1, 2 et 3 sont écrits en bleu
- Pattern : * (sélecteur universel)
 - tous les éléments (* veut dire n'importe quel élément) apparaissent avec le style défini entre { }
 - Ex :
 - * {background-color: rgb(230,255,230);}
Effet : mets le fond en vert
- exemple/selecteurs1.html

CSS : Sélecteurs

- Pattern : élément élément (sélecteur descendant)
 - tous les éléments (en 2ième) contenus dans des éléments (en 1er) apparaissent avec le style défini entre { }
 - Ex :
 - p span {color: red ; background-color: orange;}
 - Tous les span dans des p sont écrit en rouge sur fond orange
- Pattern : élément.classname (sélecteur de classe)
 - tous les éléments « éléments » dont la class (attribut) est spécifiée à « classname » apparaissent avec le style défini entre { }
 - Ex :
 - p.special {color: white ; background-color: black;}
 - Les paragraphes de class « special » sont écrits en blanc sur fond noir
- Pattern .classname (sélecteur de classe)
 - tous les éléments dont la class est spécifiée à « classname » apparaissent avec le style défini entre { }
 - Ex :
 - .special2 {color: white ; background-color: black;}
 - Les éléments de class « special2 » sont écrits en blanc sur fond noir
- exemple/selecteurs2.html

CSS : Sélecteurs

- Pattern : #id (sélecteur d'id)
 - L'élément dont l'id (attribut, unique) est spécifiée à "id" apparaît avec le style défini entre { }
 - Ex :
 - #special {background-color:#ffff; border:blue 2px solid;}
 - L'élément défini avec id="id" apparaît sur un fond légèrement jaune et encadré en bleu
- Pattern élément > élément (sélecteur d'enfant)
 - tous les éléments (en 2ième) contenus dans des éléments (en 1er) apparaissent avec le style défini entre { }
 - Ex :
 - p > span {background-color:#ffff; border: green 2px solid;}
 - Tout span défini dans un p apparaît sur un fond légèrement violet et encadré en vert
- Pattern élément + élément (sélecteur d'enfant adjacent)
 - Chaque élément (en 2ième) immédiatement après l'élément (en 1er) apparaît avec le style défini entre { } : les deux éléments ont le même parent
 - Ex :
 - p + ul {background-color:#ffff; border-color:red; border-width:2; border-style: solid;}
 - Tout ul défini juste après un p apparaît sur un fond légèrement cyan et encadré en rouge
- exemple/selecteurs3.html

CSS : Sélecteurs

- Pattern : élément[attribut] (sélecteurs d'attribut)
 - Chaque élément p qui a un attribut id (quelque soit la valeur de l'attribut) apparaît avec le style défini entre { }
 - Ex :
 - p[id]{color: white ; background-color: black;}
 - Chaque p qui a un id est écrit en blanc sur fond noir
- Notez qu'il est possible de restreindre le style :
- Pattern : élément [att="value"]
 - le style ne vaut que quand l'attribut att a la valeur précise "value"
- Pattern : élément [att~="value"] :
 - le style ne vaut que quand la valeur de l'attribut att contient "value"
- exemple/selecteurs4.html

CSS : pseudo-classes

- élément:first-child
 - correspond au premier élément (celui-ci) enfant d'un autre élément.
 - Si l'élément est le premier fils d'un autre
- Pour une ancre (a) :link et :visited
 - généralement représentation différente des liens visités de ceux qui l'ont déjà été.
 - Equivalent CSS (déconseillé)
 - :link s'applique aux liens qui n'ont pas été visités ;
 - :visited s'applique aux liens déjà visités
- pseudo-classes dynamiques :active , :hover et :focus
 - :hover pour l'élément désigné par un pointeur (souris)
 - :active pour l'élément actif (entre le pressed et le released)
 - :focus pour l'élément qui reçoit l'attention (celui-ci acceptant les événements du clavier ou d'autres formes d'entrées de texte)
- :lang
 - Pour préciser des éléments (en association avec l'attribut lang)
- exemple/pseudo.html

CSS : pseudo-éléments

- `:first-line` : première ligne
- `:first-letter` : première lettre
- `:before` et `:after`
 - `:before` et `:after` pour à insérer un contenu généré avant ou après celui d'un élément.
 - `H1:before {content: counter(chapno, upper-roman) ". "}`
 - En combinaison avec les pseudo-éléments `:first-line` et `:first-letter`, les pseudo-éléments `:before` et `:after` s'appliquent à la première lettre ou la première ligne, y compris le texte inséré.
 - Exemple(s) :
 - `P.special:before {content: "Spécial ! "}`
 - `P.special:first-letter {color: gold}`
 - Ceci donnera une couleur or au "S" de "Spécial !".
- [exemple/pseudoElt.html](#)

Assignation, cascade et héritage

- Une fois qu'un agent utilisateur (navigateur web) a parcouru un document et en a construit une arborescence, celui-ci, pour chacun des éléments de l'arborescence, doit distribuer une valeur pour chacune des propriétés concernées du type de média visé.
- La valeur finale d'une propriété est le résultat d'un calcul en trois étapes : celle-ci est déterminée par la spécification (la valeur "spécifiée"), puis résolue dans une valeur absolue si nécessaire (la valeur "calculée") et finalement transformée en accord avec les contraintes de son contexte (la valeur "réelle").

Valeurs spécifiées

- Les agents utilisateurs (navigateur web) doivent d'abord appliquer une valeur spécifiée à une propriété selon le mécanisme suivant (dans l'ordre de priorité) :
 - 1. Si la cascade donne une valeur, utiliser celle-ci ;
 - 2. Autrement, si la propriété est héritée, utiliser la valeur de l'élément parent, qui est en général une valeur calculée ;
 - 3. Autrement, utiliser la valeur initiale de la propriété. Cette valeur initiale est définie pour chaque propriété.
- La racine de l'arborescence du document n'ayant pas de parent, et ne pouvant donc pas hériter de valeurs, c'est la valeur initiale qui est utilisée si nécessaire.

Valeurs calculées

- Les valeurs spécifiées peuvent être absolues (ex. les valeurs 'red' et '2mm' ne sont pas relatives à une autre valeur) ou relatives (ex. les valeurs 'auto', '2em' et '12%' se rapportent à une autre valeur). Une valeur absolue ne requiert aucune opération supplémentaire pour établir sa valeur calculée.
- Par contre, une valeur relative doit subir une transformation qui aboutit à sa valeur calculée : une valeur en pourcentage doit être multipliée par une valeur de référence (celle-ci est précisée pour chaque propriété), une valeur avec une unité relative (em, ex, px) doit être devenue absolue par multiplication avec les tailles de police ou de pixel appropriées, les valeurs 'auto' doivent être calculées selon les formules données pour chaque propriété, certains mots-clés ('smaller', 'bolder', 'inherit') doivent être remplacés en accord avec leurs définitions.
- Dans la plupart des cas, les éléments héritent de valeurs calculées. Cependant, certaines propriétés ont des valeurs spécifiées qui sont héritées (ex. la valeur numérique de la propriété 'line-height'). Dans les cas où les éléments enfants n'héritent pas de valeurs calculées, ceci est décrit dans les définitions des propriétés.

Héritage

- Les éléments enfants héritent de certaines valeurs de leurs éléments parents dans l'arborescence du document. Chacune des propriétés définit si elle est héritée, ou non. Supposons un élément accentué (ici EM) dans un élément H1 :
 - `<H1 style="color:blue">Le titre est important !</H1>`
 - Si aucune couleur n'est précisée pour l'élément EM, le mot accentué "est" héritera de la couleur de l'élément parent, ainsi l'élément H1 ayant une couleur bleu, EM le sera également.
- Pour appliquer une propriété de style "par défaut" à un document, un auteur peut l'appliquer à la racine de l'arborescence du document. On peut utiliser les éléments HTML ou BODY pour cet usage.
- exemple/heritage.html

Cascade

- Les feuilles de style ont trois origines différentes : l'auteur, l'utilisateur et l'agent utilisateur.
 - L'auteur : produit des feuilles de style pour un document source selon les conventions du langage de ce document. Celles-ci peuvent être incorporées dans le document ou reliées à celui-ci.
 - L'utilisateur : peut être capable d'indiquer une information de style pour un document particulier. Par exemple, celui-ci peut désigner une feuille de style contenue dans un fichier ou un agent utilisateur peut fournir l'interface pour produire une feuille de style personnelle.
 - L'agent utilisateur: l'agent utilisateur conforme doit appliquer sa feuille de style par défaut avant toutes les autres feuilles de style d'un document. Cette feuille de style devrait présenter les éléments du document de façon à satisfaire au rendu généralement admis pour ceux-ci. Par exemple, pour des navigateurs visuels, l'élément EM en HTML est rendu avec une police en italique. Noter que cette feuille de style par défaut peut varier si l'utilisateur change les réglages de son système (ex. les couleurs du système). Cependant, il peut être impossible de modifier les valeurs de celle-ci, en raison d'une implémentation réduite de l'agent utilisateur.

Cascade

- 1. Trouver toutes les déclarations qui concernent l'élément et la propriété en question, pour le type de média visé. Celles-ci s'appliquent si le sélecteur correspond à cet élément ;
- 2. Un tri primaire est effectué sur les déclarations selon leur poids et leur origine : pour une déclaration normale, les feuilles de style de l'auteur surclassent celles de l'utilisateur, ces dernières surclassant la feuille de style par défaut.
- 3. Un tri secondaire est effectué selon la spécificité des sélecteurs : les plus spécifiques surclasseront ceux plus généraux. Les pseudo-éléments et les pseudo-classes sont considérés respectivement comme des éléments et des classes normaux ;
- 4. Un tri final selon l'ordre de spécification : si deux règles ont les mêmes poids, origines et spécificités, c'est la dernière survenue qui l'emporte. Les règles issues de feuilles de style importées sont considérées comme étant survenues avant chacune de celles de la feuille de style elle-même.

Spécificité d'un sélecteur

- La spécificité d'un sélecteur est déterminée comme suit :
 - dans le sélecteur, compter le nombre d'attributs (a) ;
 - puis celui des autres attributs et des pseudo-classes (b) ;
 - et ensuite le nombre de noms des éléments (c) ;
 - ignorer les pseudo-éléments.
- Assembler les trois nombres a – b – c (dans un système de nombre avec une base étendue) pour obtenir la spécificité.
 - `* {}` /* $a=0$ $b=0$ $c=0$ → spécificité = 0 */
 - `LI {}` /* $a=0$ $b=0$ $c=1$ → spécificité = 1 */
 - `UL LI {}` /* $a=0$ $b=0$ $c=2$ → spécificité = 2 */
 - `UL OL+LI {}` /* $a=0$ $b=0$ $c=3$ → spécificité = 3 */
 - `H1 + *[REL=up] {}` /* $a=0$ $b=1$ $c=1$ → spécificité = 11 */
 - `UL OL LI.red {}` /* $a=0$ $b=1$ $c=3$ → spécificité = 13 */
 - `LI.red[title] {}` /* $a=0$ $b=2$ $c=1$ → spécificité = 21 */
 - `#x34y {}` /* $a=1$ $b=0$ $c=0$ → spécificité = 100 */
- Pour HTML, les valeurs de l'attribut "style" sont des règles de feuille de style. Ces règles n'ont pas de sélecteurs, mais dans l'optique du point 3 de l'algorithme de cascade, on considère qu'elles ont un sélecteur d'ID (spécificité : $a=1$, $b=0$, $c=0$). Et dans l'optique du point 4, on considère qu'elles surviennent après toutes les autres règles.

Media

- Une des fonctions primordiales des feuilles de style repose sur le fait de pouvoir adapter la représentation d'un document pour différents médias : un écran, une feuille de papier, un synthétiseur de parole, un appareil braille, etc.
- Certaines propriétés spécifiques au média
- Spécification du média :
- `@import url("loudvoice.css") aural;`
- `@media print { /* la feuille de style pour l'impression vient ici */ }`
- `@media screen, print { BODY { line-height: 1.2 } }`
- `<LINK rel="stylesheet" type="text/css" media="print, handheld" href="foo.css">`

Catégorie de média

Media Types	Media Groups			
	continuous/paged	visual/aural/tactile	grid/bitmap	interactive/static
aural	continuous	aural	N/A	both
braille	continuous	tactile	grid	both
emboss	paged	tactile	grid	both
handheld	both	visual	both	both
print	paged	visual	bitmap	static
projection	paged	visual	bitmap	static
screen	continuous	visual	bitmap	both
tty	continuous	visual	grid	both
tv	both	visual, aural	bitmap	both

Catégorie de média

- continu ou paginé (continuous/paged) ;
- visuel, auditif ou tactile (visual/aural/tactile) ;
- grille (pour les appareils avec grille de caractères) ou bitmap (grid/bitmap);
- interactif (pour les appareils qui interagissent avec l'utilisateur) ou statique (à l'inverse, pour ceux qui n'interagissent pas avec celui-ci) (interactive/static) ;
- all (comprend tous les types de médias).

Les propriétés CSS

- Color & Background
- Font
- Text
- Box Model : margin, padding, border, width, height
- Visual Effect : overflow, visibility
- Generated Content & list
- Table
- UI
- Page, Aural

Color

- color
 - Valeur : <couleur> | inherit
 - Initiale : selon l'agent utilisateur
 - S'applique à : tous les éléments
 - Héritée : oui
 - Pourcentage : sans objet
 - Médias : visuel
- Cette propriété décrit la couleur d'avant-plan du contenu de texte d'un élément. Voici plusieurs façons d'indiquer la couleur rouge
 - EM { color: red } /* nom de couleur prédéfini */
 - EM { color: rgb(255,0,0) } /* couleurs en RGB allant de 0 à 255 */

background

- Les propriétés d'arrière-plan : 'background-color', 'background-image', 'background-repeat', 'background-attachment', 'background-position' et 'background'
- background-color
 - Valeur : <couleur> | transparent | inherit
 - Initiale: transparent
 - S'applique à : tous les éléments
 - Héritée : non
 - Pourcentage : sans objet
 - Médias : visuel
- Cette propriété donne la couleur d'arrière-plan d'un élément, avec une valeur de <couleur>, ou bien avec le mot-clé 'transparent', celui-ci laissant éventuellement voir les couleurs situées plus en-dessous.

background

- background-image
 - Valeur : <uri> | none | inherit
 - Initiale : none
 - S'applique à : tous les éléments
 - Héritée : non
 - Pourcentage : sans objet
 - Médias : visuel
- Cette propriété spécifie l'image d'arrière-plan d'un élément. En même temps qu'une image, les auteurs devraient aussi spécifier une couleur d'arrière-plan, cette couleur étant employée en remplacement d'une image indisponible. Celle-ci, une fois disponible, vient se superposer sur le fond coloré. La couleur du fond étant ainsi visible au travers des zones transparentes de l'image.
- La propriété admet les valeurs d'<uri>, pointant vers une image, ou 'none', prohibant l'emploi d'image.

background

- background-repeat
 - Valeur : repeat | repeat-x | repeat-y | no-repeat | inherit
 - Initiale : repeat
 - S'applique à : tous les éléments
 - Héritée : non
 - Pourcentage : sans objet
 - Médias : visuel
- Quand on spécifie une image d'arrière-plan, cette propriété indique si l'image est répétée et la manière de la répétition. La mosaïque de fond résultante correspond aux aires de contenu, d'espacement et de bordure de la boîte de l'élément en question. Les significations des valeurs sont :
 - repeat L'image se répète à la fois horizontalement et verticalement ;
 - repeat-x L'image ne se répète qu'horizontalement ;
 - repeat-y L'image ne se répète que verticalement ;
 - no-repeat L'image ne se répète pas : un seul exemplaire de celle-ci est dessiné.

background

- background-attachment
 - Valeur : scroll | fixed | inherit
 - Initiale : scroll
 - S'applique à : tous les éléments
 - Héritée : non
 - Pourcentage : sans objet
 - Médias : visuel
- Quand on spécifie une image d'arrière-plan, cette propriété indique si l'image est fixe par rapport à l'espace de visualisation (pour la valeur 'fixed'), ou si celle-ci défile en même temps que le document (pour la valeur 'scroll').
- Noter qu'il n'existe qu'un seul espace de visualisation par document ; c.à.d., même si l'élément est doté d'un mécanisme de défilement (voir la propriété 'overflow'), un arrière-plan avec la valeur 'fixed' ne se déplace pas avec cet élément.
- Si l'image est fixe, celle-ci n'est seulement visible que quand elle se trouve dans les aires d'arrière-plan, d'espacement et de bordure de l'élément. A moins que l'image ne se répète en mosaïque ('background-repeat: repeat'), celle-ci peut ainsi ne pas apparaître.

background-position

- Valeur : [[<pourcentage> | <longueur>]{1,2} | [[top | center | bottom] | [left | center | right]] | inherit
- Initiale : 0% 0%
- S'applique à : ceux des éléments de type bloc et ceux remplacés
- Héritée : non
- Pourcentage : se rapporte à la taille de la boîte elle-même
- Médias : visuel
- Pour une image d'arrière-plan, cette propriété indique la position initiale de celle-ci.
 - <pourcentage> <pourcentage> : pour la paire de valeurs x% y%, le point de l'image correspondant est aligné avec le point de la surface d'affichage (même proportion)
 - <longueur> <longueur> : pour la paire de valeur '3cm 2cm', le coin en haut et à gauche de l'image se place sur le point, situé à 3cm vers la droite et 2cm vers le bas en partant du coin en haut et à gauche de l'aire d'affichage ;
 - top left et left top : Identique à '0% 0%' ;
 - top, top center et center top : Identique à '50% 0%' ;
 - right top et top right : Identique à '100% 0%' ;
 - left, left center et center left : Identique à '0% 50%' ;
 - center et center center : Identique à '50% 50%' ;
 - right, right center et center right : Identique à '100% 50%' ;
 - bottom left et left bottom : Identique à '0% 100%' ;
 - bottom, bottom center et center bottom : Identique à '50% 100%' ;
 - bottom right et right bottom : Identique à '100% 100%' ;
- Une seule valeur (% ou longueur) ne concerne que la position horizontale, la position verticale sera 50%. Deux valeurs (% ou longueur) la première concerne la position horizontale. Les positions négatives le sont également.

Background

- La propriété 'background' est une propriété raccourcie qui sert à regrouper les propriétés individuelles 'background-color', 'background-image', 'background-repeat', 'background-attachment' et 'background-position' dans la feuille de style.
- La propriété 'background' distribue d'abord à toutes les propriétés individuelles d'arrière-plan à leur valeur initiale, puis leur applique les valeurs explicites de la déclaration.
- Ex :
 - BODY { background: red }
 - P { background: gray url("ele.jpg") 50% repeat fixed }
 - exemple/background.html

Font

- font
 - Valeur : [[[<font-style> | <font-variant> | <font-weight>]? <font-size> [/ <line-height>]? <font-family>] | caption | icon | menu | message-box | small-caption | status-bar | inherit
 - Initiale : voir les propriétés individuelles
 - S'applique à : tous les éléments
 - Héritée : oui
 - Pourcentages: permis sur 'font-size' et 'line-height'
 - Médias : visuel
- Les valeurs suivantes concernent les polices de système :
 - caption : La police employée pour légender les contrôles (ex. boutons, menus déroulants, etc.) ;
 - icon : La police employée pour légender les icônes ;
 - menu : La police employée dans les menus (ex. les menus déroulants et les listes de menus) ;
 - message-box : La police employée dans les boîtes de dialogue ;
 - small-caption : La police employée pour étiqueter les contrôles de petite taille ;
 - status-bar : La police employée dans les barres de statut de la fenêtre.

font-style

- Le style de police précise l'aspect dans lequel doit être rendu le texte, normal, italique ou oblique.
- font-style
 - Valeur : normal | italic | oblique | inherit
 - Initiale : normal
 - S'applique à : tous les éléments
 - Héritée : oui
 - Pourcentage : sans objet
 - Médias : visuel

font-variant

- La variante de police indique si le texte doit être rendu avec des glyphes normaux ou avec des glyphes en petites capitales, pour les caractères en minuscule.
- font-variant
 - Valeur : normal | small-caps | inherit
 - Initiale : normal
 - S'applique à : tous les éléments
 - Héritée : oui
 - Pourcentage : sans objet
 - Médias : visuel

font-weight

- Le poids de police se réfère à la graisse, plus ou moins grasse, des glyphes utilisés pour rendre le texte, en fonction des autres membres de la même famille de polices.
- font-weight
 - Valeur : normal | bold | bolder | lighter | 100 | 200 | 300 | 400 | 500 | 600 | 700 | 800 | 900 | inherit
 - Initiale : normal
 - S'applique à : tous les éléments
 - Héritée : oui
 - Pourcentage : sans objet
 - Médias : visuel
- Les significations des valeurs sont :
 - 100 à 900 : Ces valeurs forment une séquence ordonnée, où chacun des nombres indique, pour la police, une graisse au moins aussi grasse que celle du nombre précédent ;
 - normal Équivaut à '400' ;
 - bold Équivaut à '700' ;
 - bolder : Spécifie la graisse supérieure à celle assignée à une police, cette graisse étant plus grasse que celle héritée par la police. S'il n'y en a pas, la valeur de graisse de la propriété prend simplement la valeur numérique supérieure (l'aspect de la police ne changeant pas), à moins que la valeur héritée ne soit déjà égale à '900', auquel cas la valeur résultante devient également '900' ;
 - lighter : Spécifie la graisse inférieure à celle assignée à une police, cette graisse étant moins grasse que celle héritée par la police. S'il n'y en a pas, la valeur de graisse de la propriété prend simplement la valeur numérique inférieure (l'aspect de la police ne changeant pas), à moins que la valeur héritée ne soit déjà égale à '100', auquel cas la valeur résultante devient également '100'.

font-size

- font-size
 - Valeur : <taille-absolue> | <taille-relative> | <longueur> | <pourcentage> | inherit
 - Initiale : medium
 - S'applique à : tous les éléments
 - Héritée : oui, la valeur calculée également
 - Pourcentages : se rapporte à la taille de la police du parent de l'élément
 - Media : visuel
- Cette propriété décrit la taille d'une police spécifiée explicitement. Celle-ci correspond au carré em, un concept issu de la typographie. Noter que certains glyphes peuvent déborder de leur carré em. Les significations des valeurs sont :
 - <taille-absolue> Un mot-clé <taille-absolue> se réfère à une entrée de la table des tailles de police, celle-ci étant dressée et mise en œuvre par l'agent utilisateur. Les valeurs possibles :
 - [xx-small | x-small | small | medium | large | x-large | xx-large]
 - facteur d'échelle de 1.2 entre les valeurs consécutives de la table ; si la valeur 'medium' correspond à 12pt, la valeur 'large' devrait correspondre à 14.4pt.
 - Les facteurs d'échelle peuvent différer selon les médias considérés.
 - <taille-relative> Un mot-clé <taille-relative> s'entend par rapport à la table des tailles de police et par rapport à la taille de la police de l'élément parent. Les valeurs possibles :
 - [larger | smaller]
 - <longueur> Une valeur de longueur spécifie une taille de police absolue (indépendante de la table des tailles de police de l'agent utilisateur). Les valeurs négatives ne sont pas admises ;
 - <pourcentage> Une valeur en pourcentage spécifie une taille de police absolue par rapport à celle de l'élément parent. Leur emploi, de même pour les valeurs exprimées en 'em', conduit à des feuilles de style plus fiables, en plein accord avec le principe de la cascade.

line-height

- **line-height**
 - Valeur : normal | <nombre> | <longueur> | <pourcentage> | inherit
 - Initiale : normal
 - S'applique à : tous les éléments
 - Héritée : oui
 - Pourcentages: se rapporte à la taille de la police de l'élément lui-même
 - Médias : visuel
- Hauteur minimale pour les in-line inclus ou hauteur exacte pour un in-line
- Les valeurs signifient :
 - normal : Indique aux agents utilisateurs une valeur calculée "raisonnable", basée sur la taille de la police de l'élément. Cette valeur revêt la même signification que <number>. On recommande pour 'normal' une valeur calculée entre 1.0 et 1.2 ;
 - <longueur> : Cette longueur est appliquée à la hauteur de la boîte. Les valeurs négatives ne sont pas admises ;
 - <nombre> : La valeur calculée de la propriété est obtenue en multipliant ce nombre par la taille de la police héritée de l'élément. Les valeurs négatives ne sont pas admises.
 - <pourcentage> : La valeur calculée de la propriété correspond à ce pourcentage multipliée par la valeur calculée pour la taille de la police. Les valeurs négatives ne sont pas admises.

font-family

- **font-family**
 - Valeur : [[<famille-nom> | <famille-générique>],]* [<famille-nom> | <famille-générique>] | inherit
 - Initiale : selon l'agent utilisateur
 - S'applique à : tous les éléments
 - Héritée : oui
 - Pourcentage : sans objet
 - Médias : visuel
- Cette propriété donne une liste, par ordre de priorité, de noms de familles de polices et/ou de noms de familles génériques.
- Ex :
 - BODY { font-family: Baskerville, "Heisi Mincho W3", Symbol, serif }
 - La police "Baskerville" va fournir les glyphes pour les caractères latins, la police "Heisi Mincho W3", ceux pour les caractères japonais, la police "Symbol", ceux pour les symboles mathématiques, et la famille de polices générique 'serif', ceux éventuels d'autres caractères.
 - La famille de polices générique sera utilisée si une, ou plusieurs, des polices d'un jeu de polices n'étaient pas disponibles. Bien que certaines polices fournissent un glyphe de substitution pour indiquer un "caractère manquant", celui-ci prenant typiquement l'aspect d'un carré, on ne devrait pas considérer ce glyphe comme une correspondance valide.

Exemple de police

- **serif**
 - des terminaisons au bout de leurs traits, des bouts évasés ou en pointes, ou plus simplement des terminaisons avec des empattements
 - Polices latines : Times New Roman, Polices grecques : Bitstream Cyberbit, Polices cyrilliques : Adobe Minion Cyrillic, Polices hébraïques : New Peninim, Polices japonaises : Ryumin Light-KL, Polices arabes : Bitstream Cyberbit
- **sans-serif**
 - des traits pleins, sans évasement, ni recouvrements ou autres ornements
 - Polices latines MS Arial, MS Verdana, Helvetica, Polices grecques : Helvetica Greek, Polices cyrilliques Helvetica Cyrillic, Polices hébraïques : Arial Hebrew, Polices japonaises Shin Go
- **cursive**
 - des traits qui se recoupent ou bien un aspect manuscrit plus affirmé
 - Polices latines : Caffisch Script, Polices cyrilliques : ER Architekt, Polices hébraïques : Corsiva, Polices arabes : DecoType Naskh, Monotype Urdu 507
- **fantasy**
 - Critter
- **monospace**
 - Courier

font-size-adjust

- Valeur : <nombre> | none | inherit
- Initiale : none
- S'applique à : tous les éléments
- Héritée : oui
- Pourcentage : sans objet
- Médias : visuel
- Cette propriété permet de spécifier une valeur d'aspect pour un élément, ce qui préserve la valeur hauteur-x de la première police à choisir dans la liste des polices de substitution. Les significations des valeurs sont :
 - none : ne préserve pas la valeur hauteur-x de la police ;
 - <nombre> : Spécifie la valeur d'aspect. Ce nombre concerne la valeur d'aspect de la première police à choisir. Le facteur d'échelle pour les polices disponibles se détermine selon cette formule :
 - $y(a/a') = c$
 - ce qui correspond à :
 - y = la valeur de 'font-size' de la première police du choix ;
 - a = la valeur d'aspect de la première police du choix ;
 - a' = la valeur d'aspect de la police disponible ;
 - c = la valeur de 'font-size' appliquée à la police disponible.
 - Exemple :
 - Par exemple, la police Verdana (valeur d'aspect de 0.58) avec une taille de police 14px spécifiée n'étant pas disponible, une police de substitution avec une valeur d'aspect de 0.46 étant disponible, la taille de police de celle-ci serait : $14 * (0.58/0.46) = 17.65px$.

font-stretch

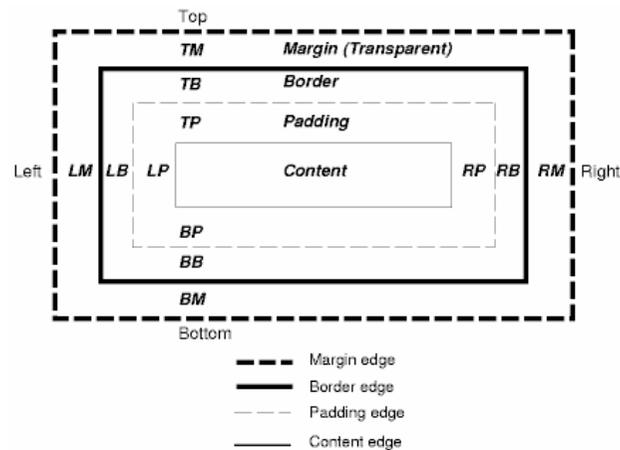
- font-stretch
 - Valeur : normal | wider | narrower | ultra-condensed | extra-condensed | condensed | semi-condensed | semi-expanded | expanded | extra-expanded | ultra-expanded | inherit
 - Initiale : normal
 - S'applique à : tous les éléments
 - Héritée : oui
 - Pourcentage : sans objet
 - Médias : visuel
- La propriété 'font-stretch' sélectionne les dessins normal, compacté ou élargi dans une famille de polices.
- exemple/font.html

Text

Property	Description	Values
color	Sets the color of a text	color
direction	Sets the text direction	ltr rtl
letter-spacing	Increase or decrease the space between characters	normal length
text-align	Aligns the text in an element	left right center justify
text-decoration	Adds decoration to text	none underline overline line-through blink
text-indent	Indents the first line of text in an element	length %
text-transform	Controls the letters in an element	none capitalize uppercase lowercase

**Nb : en suspend
text-shadow**

Organisation visuel



Border

- border-color
 - 'border-top-color', 'border-right-color', 'border-bottom-color', 'border-left-color'
 - 1, 2, 3 ou 4 valeurs (répétées par côtés opposés)
- border-width
 - 'border-top-width', 'border-right-width', 'border-bottom-width', 'border-left-width', and 'border-width'
 - Thin : a thin border (fin) ; medium : a medium border (moyen) ; thick : a thick border (épais) ou une longueur (pixel, cm, etc.) ; inherit
 - 1, 2, 3 ou 4 valeurs (répétées par côtés opposés)
- border-style
 - 'border-top-style', 'border-right-style', 'border-bottom-style', 'border-left-style', and 'border-style'
 - none : aucun, force 'border-width' à '0' ; hidden : idem aucun sauf dans les tables ; dotted : pointillé ; dashed : tirêts ; solid : une simple ligne ; double : deux lignes espacées de border-width - 2 ; groove : "sculté" ; ridge : "qui ressort" ; inset : "incrusté" ; outset : "qui sort" ; inherit
 - 1, 2, 3 ou 4 valeurs (répétées par côtés opposés)
- border : couleur épaisseur style
- exemple/border.html

Margin

- margin
 - 'margin-top', 'margin-right', 'margin-bottom', 'margin-left'
 - une longueur ; un pourcentage (de la largeur de la boîte) ; auto (calcul) ; inherit
 - 1, 2, 3 ou 4 valeurs (répétées par côtés opposés)
- Exemple :
- BODY { margin: 2em } /* all margins set to 2em */
- BODY { margin: 1em 2em } /* top & bottom = 1em, right & left = 2em */
- BODY { margin: 1em 2em 3em } /* top=1em, right=2em, bottom=3em, left=2em */
- La dernière ligne est équivalente à :
- BODY {
 - margin-top: 1em;
 - margin-right: 2em;
 - margin-bottom: 3em;
 - margin-left: 2em; /* copied from opposite side (right) */
- }

Padding

- padding
 - 'padding-top', 'padding-right', 'padding-bottom', 'padding-left'
 - longueur ; pourcentage (de la largeur de la boîte); inherit
 - 1, 2, 3 ou 4 valeurs (répétées par côtés opposés)
- exemple/marginpadding.html

Display

- display
 - inline | block | list-item |
 - Comme les éléments html correspondant
 - run-in | compact |
 - Block ou inline selon le contexte
 - marker |
 - Élément généré
 - table | inline-table | table-row-group | table-header-group | table-footer-group | table-row | table-column-group | table-column | table-cell | table-caption |
 - Comme des éléments d'un tableau
 - none |
 - n'apparaît pas
 - inherit
 - valeur initiale : inline
- exemple/display.html

position

- position
 - static (normale) | relative (position calculée par rapport à la position normale) | absolute (position absolue dans le document, n'influence plus la mise en page) | fixed (fixe par rapport à la fenêtre – scroll) | inherit
 - Position précisée (si non static) avec top, right, bottom et left
- top, right, bottom et left
 - <length> | <percentage> (largeur ou hauteur de la boîte) | auto (calcul automatique, dépend des cas) | inherit
 - Besoin de 2, sinon étirement
- exemple/position.html

float

- float
 - Texte flottant
 - Value: left | right | none | [inherit](#)
 - Initial: none
- clear
 - Pour indiquer quel côté ne peut pas être adjacent avec des éléments flottants précédents
 - left ; right ; both ; none
- exemple/float.html
- z-index : pour préciser la position : ceux de plus haut niveau apparaissent au dessus, pour les éléments positionnés (top...)
 - Sinon, ordre inverse d'apparition
- exemple/z.html

width

- 'width'
 - <length> | <percentage> | auto | inherit
 - Initial: auto
 - Applies to: all elements but non-replaced inline elements, table rows, and row groups
 - Inherited: no
 - Percentages: refer to width of containing block
 - Media: visual
- 'min-width', 'max-width'
 - Value: <length> | <percentage> | inherit (| none pour max-width)
 - Initial: User Agent dependent
 - Applies to: all elements except non-replaced inline elements and table elements
 - Inherited: no
 - Percentages: refer to width of containing block
 - Media: visual
- exemple/width.html

height

- 'height'
 - <length> | <percentage> | auto | inherit
 - Initial: auto
 - Applies to: all elements but non-replaced inline elements, table rows, and row groups
 - Inherited: no
 - Percentages: refer to width of containing block
 - Media: visual
- 'height -width', 'height -width'
 - Value: <length> | <percentage> | inherit (| none pour max-width)
 - Initial: User Agent dependent
 - Applies to: all elements except non-replaced inline elements and table elements
 - Inherited: no
 - Percentages: refer to width of containing block
 - Media: visual
- exemple/height.html

Hauteur de ligne

- 'line-height'
 - Value: normal | [<number>](#) | [<length>](#) | [<percentage>](#) | [inherit](#)
 - Nombre : coefficient multiplicatif (par rapport à la plus grande font)
 - Initial: normal ; Applies to: all elements ; Inherited: yes ; Percentages: refer to the font size of the element itself ; Media: [visual](#)
- 'vertical-align'
 - Value: baseline | sub | super | top | text-top | middle | bottom | text-bottom | [<percentage>](#) | [<length>](#) | [inherit](#)
 - Pourcentage ou longueur : augmente (> 0) ou diminue (< 0) la position verticale
 - Initial: baseline ; Applies to: inline-level and 'table-cell' elements ; Inherited: no ; Percentages: refer to the 'line-height' of the element itself ; Media: [visual](#)
- exemple/h.html

Rendu visuel

- overflow
 - Pour ce qui dépasse
 - Value: visible | hidden | scroll | auto | [inherit](#)
 - Initial: visible ; Applies to: block-level and replaced elements
 - Inherited: no ; Percentages: N/A ; Media: [visual](#)
- clip (ne fonctionne pas)
 - Pour définir ce qui est visible ou non
 - Value: [<shape>](#) | auto | [inherit](#)
 - rect (<top> <right> <bottom> <left>)
- visibility
 - Value: visible | hidden | collapse | [inherit](#)
 - Hidden : espace pris, mais rien n'est visible
 - Collapse pour table (ligne et colonne), sinon équivalent à hidden
 - Initial: inherit ; Applies to: all elements ; Inherited: no ; Percentages: N/A ; Media: [visual](#)
- exemple/rendu.html

Génération automatique

- Pseudo-éléments :before :after
 - Impact de compact et run-in
- Content
 - Value: [<string> | <uri> | <counter> | attr(X) | open-quote | close-quote | no-open-quote | no-close-quote]+ | [inherit](#)
 - Initial: empty string ; Applies to: :before and :after pseudo-elements ; inherited: no ; Percentages: N/A ; Media: all
 - <string> : texte entre " "
 - <uri> : ressource externe
 - <counter> : compteur, deux formes : 'counter(name)' or 'counter(name, style)'
 - open-quote et close-quote : par exemple caractères " "
 - no-open-quote et no-close-quote : un rien qui remplace une " ou une "
 - attr(X) : This function returns as a string the value of attribute X for the subject of the selector. The string is not parsed by the CSS processor. If the subject of the selector doesn't have an attribute X, an empty string is returned. The case-sensitivity of attribute names depends on the document language. Note. In CSS2, it is not possible to refer to attribute values for other elements of the selector.
- 'quotes' : Value: [[<string>](#) [<string>](#)]+ | none | [inherit](#)
 - Définit une paire de guillemets

compteurs

- 'counter-reset'
 - Value: [[<identifiant>](#) [<integer>](#)?]+ | none | [inherit](#)
 - Initial: none ; Applies to: all elements ; Inherited: no ; Percentages: N/A ; ; Media: [all](#)
 - Valeur de reset par défaut : 0 ; valeur négative possible
- 'counter-increment'
 - Value: [[<identifiant>](#) [<integer>](#)?]+ | none | [inherit](#)
 - Initial: none ; Applies to: all elements ; Inherited: no ; Percentages: N/A ; Media: [all](#)
 - Valeur d'incrément par défaut : 1 ; valeur négative possible

listes

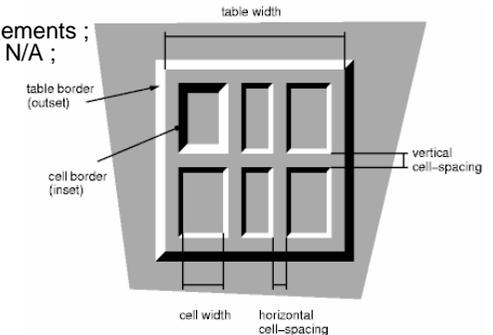
- 'list-style-type'
 - Value: disc | circle | square | decimal | decimal-leading-zero | lower-roman | upper-roman | lower-greek | lower-alpha | lower-latin | upper-alpha | upper-latin | hebrew | armenian | georgian | cjk-ideographic | hiragana | katakana | hiragana-iroha | katakana-iroha | none | inherit
 - Initial : disc ; Applies to : elements with 'display: list-item' ; Inherited : yes ; Percentages: N/A ; Media : visual
- 'list-style-image' : url ou none ou inherit
- 'list-style-position' : inside ou outside ou [inherit](#)
- 'list-style' : [[<'list-style-type'>](#) || [<'list-style-position'>](#) || [<'list-style-image'>](#)] | [inherit](#)
- exemple/listesetnum.html

table

- 'caption-side'
 - Value: top | bottom | left | right | [inherit](#)
 - Initial: top ; Applies to: 'table-caption' elements
- 'table-layout'
 - Value: auto | fixed | [inherit](#)
 - Auto : calcul automatique
 - Fixed : fait référence aux éléments tels que width : rendu par colonne (selon s'il y a un ou plusieurs éléments fixés, sinon auto)
 - Initial: auto ; Applies to: 'table' elements ; Inherited: no ; Percentages: N/A ; Media: [visual](#)
- Alignement vertical : 'vertical-align'
- Alignement horizontal : 'text-align'

table

- 'border-collapse'
 - Value: collapse | separate | [inherit](#)
 - Initial: collapse ; Applies to: 'table' elements ; Inherited: yes ; Percentages: N/A ; Media: [visual](#)
- 'border-spacing' (espace entre les bords des cellules)
 - Value: [<length> <length>?](#) | [inherit](#)
 - Horizontal (puis vertical)
 - Initial: 0 ; Applies to: 'table' elements ; Inherited: yes ; Percentages: N/A ; Media: [visual](#)
- 'empty-cells'
 - Value: show | hide | [inherit](#)
- 'speak-header' (média aural)
 - Value: once | always | [inherit](#)
- exemple/table.html



User interface

- 'cursor'
 - Value: [[[<uri>](#) ,]* [auto | crosshair | default | pointer | move | e-resize | ne-resize | nw-resize | n-resize | se-resize | sw-resize | s-resize | w-resize | text | wait | help]] | [inherit](#)
- Preference
 - Des valeurs pour faire référence au valeur de l'utilisateur
 - www.w3.org/TR/REC-CSS2/ui.html

media

- Attribut de la balise link
- `@import url("loudvoice.css") aural;`
- `@media print { /* style sheet for print goes here */ }`
- exemple/media.html