

Formation CSS3

Auteur : Cyrille Tuzi

Objectifs

- Connaître les nouveautés CSS3 et les problématiques de compatibilité
- Positionnement moderne et responsive
- Graphisme avancé, avec peu d'images
- Animations sans Flash ni JavaScript

Plan de la formation

1. Compatibilité
2. Sélecteurs et valeurs CSS3
3. Positionnement CSS3
4. Responsive design : les media-queries
5. Graphisme façon print
6. Graphisme avancé
7. Animations

Compatibilité

Standards et doctype

- Standards CSS respectés par tout le monde et par Internet Explorer depuis la version 8 incluse.
- Le seul doctype à utiliser :

`<!doctype html>`

Le cas Internet Explorer

- Internet Explorer 6 et 7 : < 1% d'usage

- <http://gs.statcounter.com/>

- Windows XP IE (6-) 8

- pays en développement, grands comptes, service public

- Windows Vista IE (7-) 9

- Windows 7 IE (8-) 11

- Windows 8 IE (10-) 11

- **Pas de vieux IE sur mobiles et tablettes**

(Rétro-)compatibilité

- Graphisme

- pas nécessaire : ça sera juste moins joli sur les anciens navigateurs

- Positionnement

- plus problématique car fonctionnel, seulement dans un environnement connu (sur mobiles et tablettes par exemple)

Nouveautés expérimentales

- Utilisables (et utilisés) en production, il faut juste faire un travail de veille pour anticiper les évolutions et utiliser les préfixes :

-moz-	Firefox (Moz illa)
-webkit-	Safari (et Chrome)
-ms-	Internet Explorer (Micro soft)

Nouveautés expérimentales

- Le système des préfixes est progressivement remplacé par une activation via les configurations avancées :

Firefox <about:config>

Chrome <chrome://flags>

Sélecteurs et valeurs CSS3

Rappels sur sélecteurs de base

p	tous les éléments
#mon-identifiant	identifiant (attribut "id")
.ma-classe	groupe (attribut "class")
h1, h2	plusieurs sélecteurs

Sélecteurs par relation

`ul li` à l'intérieur de (tous les descendants)

`ul > li` à l'intérieur de (enfants directs)

`h1 ~ p` prochains éléments

`h1 + p` élément adjacent, suivant direct

Sélecteurs par attribut

[required]

a cet attribut

[type="text"]

vaut exactement

[class~="red"]

a cette valeur dans la liste

[lang|="fr"]

vaut ou commence par « fr- »

[class^="red"]

commence par

[class\$="red"]

fini par

[class*="red"]

contient

Pseudo-classes

- Les classiques pour les liens :
 - `a:link`
 - `a:visited`
 - `element:target`
- Actions de l'utilisateur :
 - `element:hover`
 - `element:active`
 - `element:focus`

Pseudo-classes

- Etat des éléments interactifs (UI) :
 - `element:enabled`
 - `element:disabled`
 - `element:checked`
 - `element:indeterminate`
- Langue du document :
 - `:lang(fr)`

Pseudo-classes

- Etat des contrôles :
 - `element:valid`
 - `element:invalid`
 - `element:required`
 - `element:optional`

Pseudo-classes

<code>:nth-child(n)</code>	n-ième enfant
<code>:nth-last-child(n)</code>	n-ième dernier enfant
<code>:only-child</code>	seul enfant
<code>:first-child</code> <code>:last-child</code>	premier dernier enfant
 <code>:empty</code>	 pas d'enfant
 <code>:nth-of-type(n)</code>	 n-ième de ce type
<code>:nth-last-of-type(n)</code>	n-ième dernier de ce type
<code>:only-of-type</code>	seul de ce type
<code>:first-of-type</code> <code>:last-of-type</code>	premier dernier de ce type
 <code>:not(selecteur)</code>	 ne correspond pas au sélecteur

Pseudo-éléments

- Partie du contenu :
 - `::first-line`
 - `::first-letter`
- Position par rapport au contenu :
 - `::before`
 - `::after`

Génération de contenu

- Génération de contenu (à éviter) :

```
::before {  
    content: "->";  
}
```

- Récupération de la valeur d'un attribut :

```
::before {  
    content: attr(data-id);  
}
```

Valeurs dynamiques

- Calcul (espaces importants) :

- `calc(100% - 4px)`

- Bientôt des variables

```
:root {  
    --main-color: rgb(10,10,10);  
}  
.color-main {  
    color: var(--main-color);  
}
```

Positionnement en CSS3

Rappels : inline et block

- **block** : `div`, `h1`, `p`, `ul`, `article`, `header`, ...
 - Conteneurs sémantiques
 - Affichage à la ligne par défaut (vertical)
 - Dimension, marges et bordure
- **inline** : `span`, `strong`, `em`, `a`, le texte, ...
 - Contenu
 - Affichage en ligne (horizontal)
 - Pas de dimension
 - Espaces conservés

inline-block

- inline-block : img, video, ...
 - Affichage en ligne (horizontal)
 - Dimension, marges et bordures
 - Espaces conservés
- Attention au mélange des concepts

Calcul des dimensions

- Les dimensions d'un bloc sont celles de son contenu : il faut donc retirer la taille des bordures et de l'espacement interne (padding).

`box-sizing: content-box | border-box;`

Clearfix

- Astuce pour stopper les **float** :

```
.clearfix:before, .clearfix:after {  
    content: " ";  
    display: table;  
}  
.clearfix:after {  
    clear: both;  
}
```

Flexbox

`display: flex;`

- Axe d'affichage (horizontal ou vertical) :

`flex-direction: row | row-reverse | column | column-reverse;`

- Retours à la ligne :

`flex-wrap: nowrap | wrap | wrap-reverse;`

- Raccourci : `flex-flow`

Alignement des flexbox

- Alignement des blocs sur l'axe principal :
`justify-content:` `flex-start` | `flex-end` | `center` |
`space-between` | `space-around`;
- Alignement des blocs sur l'axe perpendiculaire :
`align-items:` `stretch` | `flex-start` | `flex-end` | `center` |
`baseline`;
- En cas de lignes multiples : `align-content`

Flexibilité

- Ordre d'affichage :

`order: 0 | 1, 2,... | -1,...;`

- Dimension des blocs :

`flex: 0 1 200px;`

- Proportion d'agrandissement
- Proportion de rétrécissement
- Taille de préférence

Flexibilité

- Cas courants :

`flex: 0 auto;`

Non agrandit mais réduit si nécessaire

`flex: auto;`

Complètement flexible

`flex: none;`

Complètement inflexible

`flex: 2;`

Proportions flexibles

Responsive design

Rappels sur les médias

@media print {}

- Les différents médias :
 - all (défaut)
 - screen (navigateur visuel classique)
 - print (pour l'impression)
 - handled (pour les mobiles classiques)
 - projection
 - tv

Media-queries

@media all and (option: value) {

- (min-|max-)(device-)width
- (min-|max-)(device-)height
- (min-|max-)(device-)aspect-ratio
- (min-|max-)resolution (en dpi ou dpcm)
- orientation: landscape | portrait
- color | monochrome | (min-|max-)color-index

Viewport

- Les mobiles ont de plus en plus grandes résolutions d'affichage, mais leur écran est évidemment bien plus petit : le contenu apparaît donc très petit, comme si la page était dézoomée.
- Le viewport, c'est la taille d'affichage utilisée. On peut la faire correspondre avec la taille physique de l'écran :

```
<meta name="viewport" content="width=device-width,  
initial-scale=1">
```

Options pour le viewport

- D'autres valeurs existent pour les dimensions :
 - `width`
 - `height`
- Pour le zoom :
 - `initial-scale` (`initial-scale=1 ~ width=device-width`)
 - `maximum-scale`
 - `user-scalable=no`

Graphisme
façon print

Colonne

column-width: auto | 100px;

column-count: auto | 3;

- Raccourci : columns

column-gap: normal | 10px;

column-rule-width: medium | 2px;

column-rule-style: none | solid | dotted | dashed | double;

column-rule-color: currentColor | red;

- Raccourci : column-rule

Options pour les colonnes

- Contenu transversal à toutes les colonnes :

`column-fill:` `balance` | `auto`;

- Contenu transversal à toutes les colonnes :

`column-span:` `none` | `all`;

Régions

- Répartir un contenu dans plusieurs blocs :

- zones cibles :

flow-from: none | nomdelaregion;

- zone qui contient le contenu :

flow-into: none | nomdelaregion;

Répartition

- Gestion manuelle des changements de colonnes ou de régions :

break-before: auto | avoid-column | column |
avoid-region | region;

break-after: auto | avoid-column | column |
avoid-region | region;

break-inside: auto | avoid-column | avoid-region;

Exclusions

- Répartir un contenu autour d'autres blocs :

- sur l'élément à éviter :

`wrap-flow: auto | start | end | minimum | maximum |
clear;`

Graphisme avancé

Rappels sur les polices

font-style: normal | italic;

font-weight: normal | bold;

font-size: 2em;

line-height: 1.5;

font-family: Tahoma, Arial, sans-serif;

- Raccourci : font

- Nouveauté :

font-stretch: normal | condensed | expanded;

Familles de polices

- Il existe peu de polices de caractères disponibles partout :
 - <http://www.codestyle.org/css/font-family/sampler-CombinedResultsFull.shtml>
- Les familles génériques :
 - `sans-serif` (le plus courant)
 - `serif`
 - `cursive` (manuscrit)
 - `fantasy`
 - `monospace` (machine à écrire, largeur identique)

Polices personnalisées

```
@font-face {  
    font-family: 'Fely';  
    src: local('Fely'), url(fonts/fely.woff) format('woff');  
}
```

```
.ma-classe {  
    font-family: 'Fely', Arial, sans-serif;  
}
```

Les formats de polices

- Il existe plusieurs formats de police :
 - `svg` .svg
 - `woff` .woff **Web Open Font Format**
 - `embedded-opentype` .eot (IE8-)
 - `opentype` .otf
 - `truetype` .ttf
- Polices open-source :
 - <http://www.google.com/webfonts>

Unités

- Fixes (à éviter) :

- `px` (pixels)
- `pt` (points)
- `pc` (pica = 12pt)
- `cm`, `mm`

- Relatives :

- `%` par rapport à la taille parente
- `em` multiplicateur par rapport à la taille parente
- `rem` multiplicateur par rapport à la taille globale
- `vw`, `vh` pourcentage par rapport aux dimensions du viewport

Rappels sur les textes

text-align: left | center | right | justify;
text-decoration: none | underline | line-through;
text-indent: 10px;
text-transform: none | capitalize | uppercase | lowercase;

color: red;

letter-spacing: normal | 2em;
word-spacing: normal | 2em;
white-space: normal | nowrap | pre
| pre-wrap | pre-line; (nouveau)
word-wrap: normal | break-word;

Nouveautés pour les textes

tab-size: 8;

hyphens: manual | auto | none;

text-align-last: left | center | right | justify;

Rappels sur les couleurs

- Il existe plusieurs notations pour les couleurs :
 - `red` couleurs simples
 - `#00FF00` hexadécimal complet
 - `#0F0` hexadécimal raccourci
 - `rgb(0,255,0)` *red, green, blue*
- Une couleur est un mélange de rouge, vert et bleu, à un degré allant de 0 à 255. L'hexadécimal correspond au même concept en base 16 (FF = 255).
 - Noir absence `rgb(0,0,0)` `#000`
 - Blanc mélange `rgb(255 255 255)` `#FFF`

Nouveaux formats de couleurs

- Nouveautés :

- `rgba(0, 255, 0, 0.5)` *alpha channel* (opacité)
- `hsl(0, 50%, 50%)` *hue, saturation, lightness*
- `hsla(0, 50%, 50%, 0.5)`
- `currentColor`

- Hue = teinte sur la roue des couleurs, de 0 à 360

- Rouge 0 ou 360
- Vert 120
- Bleu 240

Ombres

`text-shadow: none | 2px 2px grey;`

`box-shadow: none | 2px 2px 2px grey |
inset 2px 2px 2px grey;`

Ce qui dépasse

- Déjà existant :

`overflow:` `visible` | `hidden` | `scroll`;

- Nouveauté si on masque ce qui dépasse :

`text-overflow:` `clip` | `ellipsis`;

Rappels sur les fonds

`background-image:` `url(image.png);`
`background-position:` `horizontal vertical;` (ex: `left top`)
`background-repeat:` `repeat | repeat-x | repeat-y | no-repeat;`
`background-color:` `red;`

- Raccourci : `background`
- La répétition est utile pour les performances.

Nouveautés pour les fonds

background-repeat: round | space;
background-clip: border-box | padding-box | content-box;
background-origin: border-box | padding-box | content-box;
background-size: 150px 100px | contain | cover;
background-attachment: scroll | local;

background: url(top.png) top no-repeat,
url(bottom.png) bottom no-repeat;

box-decoration-break: slice | clone;

Dégradés

`linear-gradient(45deg, blue, white 20%, red);`

`radial-gradient(10px circle at top left, blue, red);`

- Un dégradé s'applique à `background-image`
- Variantes :
 - `repeating-linear-gradient`
 - `repeating-radial-gradient`

Rappels sur les bordures

`border-width: 1px;`

`border-style: none | solid | dotted | dashed | double;`

`border-color: black;`

- Raccourci : `border`
- Côtés différenciés :
 - `border-top`
 - `border-right`
 - `border-bottom`
 - `border-left`

Bordures arrondies

`border-radius: 5px 4px 3px 2px / 9px 8px 7px 6px;`

- Angles différenciés :
 - `border-top-left-radius`
 - `border-top-right-radius`
 - `border-bottom-left-radius`
 - `border-bottom-right-radius`

Bordures en images

```
border-image-source: url(image.png);  
border-image-width: 10;  
border-image-repeat: stretch | repeat | round;
```

- Raccourci : `border-image`

Bordures externes (contours)

- Les contours ne prennent pas d'espace, ils sont extérieurs au bloc.

`outline-width: 1px;`

`outline-style: none | solid | dotted | dashed | double;`

`outline-color: black;`

- Raccourci : `outline`

`outline-offset: 5px;`

Filtres

- Nouvelle propriété **filter**:

- `grayscale(100%);`
- `sepia(100%);`
- `saturate(20%);`
- `hue-rotate(90deg);`
- `invert(100%);`
- `brightness(150%);`
- `contrast(150%);`
- `blur(10px);`
- `opacity(50%);`
- `drop-shadow(2px 2px 2px grey);`

Animations

Transformations 2D

transform:

- `rotate(45deg);`
- `translate(10px, 5px);`
- `translateX(5px); translateY(5px);`
- `scale(2, 1);`
- `scaleX(2); scaleY(1);`
- `skewX(5deg); skewY(5deg);`

Préliminaires pour la 3D

- Le conteneur doit créer un environnement 3D :
 - `position: relative;`
 - `perspective: none | 500px;`
 - Valeur élevée : objet éloigné / perspective légère
 - Valeur faible : objet proche / perspective forte
- Les blocs 3D doivent permettre la superposition :
 - `position: absolute;`
 - `transform-style: flat | preserve-3d;`

Transformations 3D

transform:

- `translateZ(100px);`
- `translate3d(0, 0, 100px);`
- `scaleZ(2);`
- `scale3d(1, 1, 2);`

Rotations 3D

transform:

- `rotateX(45deg); / rotate3d(1, 0, 0, 45deg);`
 - `rotateY(45deg); / rotate3d(0, 1, 0, 45deg);`
 - `rotateZ(45deg); / rotate3d(0, 0, 1, 45deg);`
-
- Il s'agit du mouvement **autour** des axes :
 - x : penché en avant / arrière
 - y : penché sur les côtés
 - z : tourné sur les côtés

Options pour les transformations

- Changer l'origine d'une transformation 2D :
 - `transform-origin: 50% 50% | top left;`
- Changer la position du point de fuite en 3D :
 - `perspective-origin: 50% 50% | top left;`
- Pour masquer le côté opposé d'un objet 3D :
 - `backface-visibility: hidden;`

Transitions

transition-property: all | none | propriete1, etc;
transition-duration: 3s;
transition-timing-function: ease | linear;
transition-delay: 0s | 2s;

- Raccourci : **transition**
- Une transition n'a de sens que s'il y a un changement de style.

Animations

`animation-name:` none | nomdelanimation;
`animation-duration:` 3s;
`animation-timing-function:` ease | linear;
`animation-iteration-count:` 1 | infinite;
`animation-direction:` normal | reverse | alternate;
`animation-delay:` 0s | 2s;
`animation-fill-mode:` none | forwards | backwards |
both;

- Raccourci : `animation`
- `animation-play-state:` running | paused;

Keyframes

```
@keyframes nomdelanimation {  
  from {  
    width: 200px;  
  }  
  50% {  
    width: 300px;  
  }  
  to {  
    width: 500px;  
  }  
}
```