

DESIGN  
**WEB**

Préface de Christian **Heilmann**

Kaelig **Deloumeau-Prigent**

# CSS maintenables

avec Sass & Compass

2<sup>e</sup> éd.

OUTILS ET BONNES PRATIQUES POUR L'INTÉGRATEUR WEB

**EYROLLES**

# De <font> à @font-face : une problématique nouvelle

# 1

*Après des prémices chaotiques, le Web arrive peu à peu à maturité. D'un monde très artisanal, on se dirige vers une industrie aux codes de plus en plus précis : métiers, standards, organisation, homogénéisation... Retraçons l'histoire du métier de développeur web côté client, appelé intégrateur web.*

## Aux débuts : pas de CSS, ni d'intégrateurs web

Le métier d'intégrateur n'est pas né en même temps que le Web, loin s'en faut. À l'origine, le *World Wide Web* devait permettre à des chercheurs de partager de l'information scientifique entre eux ; que les pages soient ou non stylées n'entrait pas dans le champ de leurs préoccupations.

Ceux qui ont vécu cette époque se souviennent que dans les années 1990, les navigateurs étaient des logiciels rudimentaires (et parfois payants), les connexions très lentes et les écrans bien moins larges que ceux d'aujourd'hui.

En 1994, le *World Wide Web Consortium* (W3C) est fondé par un héros des temps modernes, l'inventeur du Web Tim Berners-Lee. Le W3C a pour mission de promouvoir un Web pour tous, se basant sur des technologies universelles validées par le même organisme, que l'on appelle les standards du Web.

La Toile à ses prémices peut être décrite comme un outil à l'usage d'une élite intellectuelle désireuse de partager le fruit de sa pensée, notamment à l'aide de HTML, un langage inventé pour lier entre elles différentes ressources d'Internet.

Aux balbutiements du Web, styler un document se limite grosso modo à modifier le corps du texte et à insérer une image. À cette époque, ni le grand public, ni le monde du graphisme ne s'intéressent encore au Web. Heureusement, une nouveauté technologique va venir bouleverser l'austérité ambiante : les feuilles de styles en cascade, plus connues sous le nom de CSS (*Cascading Style Sheets*).

**ANECNOTE Le premier site web : août 1991**

Le premier site web mis en ligne a été publié le 6 août 1991. Intitulé *The WWW Project*, ses pages étaient hébergées sur `info.cern.ch` par le premier serveur web, un ordinateur de marque NeXT, la compagnie fondée par Steve Jobs suite à son départ forcé d'Apple. Aujourd'hui, vous pouvez en trouver une copie sur le site du W3C.

▶ <http://www.w3.org/History/19921103-hypertext/hypertext/WWW/TheProject.html>

## L'apparition de la feuille de styles comme livrable

Au milieu des années 1990, alors que les sites commencent à proliférer sur la Toile, une question épineuse émerge : « Comment produire une charte graphique qui vaille pour l'ensemble d'un site, si ce dernier contient des dizaines et des dizaines de documents ? »

En effet, d'une page à l'autre, le créateur d'un site doit répéter les informations de présentation dans le code HTML lui-même, cette redondance posant un évident souci de maintenabilité. Les développeurs web de l'époque expriment en outre le souhait de pouvoir mieux contrôler l'apparence de leurs pages.

**PAROLE D'EXPERT HTML, la victoire de la pratique sur la règle**

Frédéric Kaplan nous livre une version courte de l'histoire du langage HTML, truffée d'anecdotes, dans laquelle il prend le recul nécessaire pour nous dévoiler les principes qui ont fait de HTML ce qu'il est aujourd'hui.

▶ <http://fkaplan.wordpress.com/2012/01/18/html-limprobable-resurrection-dune-langue-vivante/>

C'est en décembre 1996, après moult délibérations, que le W3C publie la recommandation *CSS Level 1*. C'est une révolution : les feuilles de styles permettent alors de cibler certains éléments du document et de leur appliquer des styles sans avoir recours à des informations de présentation au sein du code HTML.

Pour colorer le texte des cellules d'un tableau, on n'écrit donc plus :

```
<table>
  <tr>
    <td><font color="red">Une cellule</font></td>
    <td><font color="red">Encore une</font></td>
    <td><font color="red">Et une dernière pour la route...</font></td>
  </tr>
</table>
```

Mais plutôt :

**CSS pour styler les cellules de tableau**

```
td { color: red; }
```

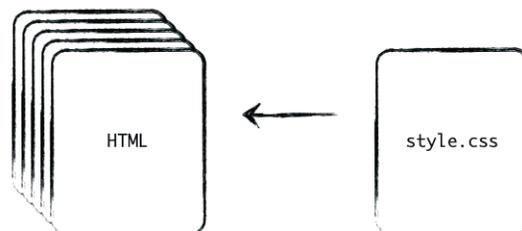
**Code HTML du tableau**

```
<table>
  <tr>
    <td>Une cellule</td>
    <td>Encore une</td>
    <td>Et une dernière pour la route...</td>
  </tr>
</table>
```

Et surtout, les CSS s'appliquent à un site entier, alors que chaque page devait être stylée indépendamment des autres auparavant. On imagine le gain de temps pour effectuer la maintenance et les évolutions du design !

**Figure 1-1**

Comme le dirait ce cher J. R. R. Tolkien :  
« une feuille de styles pour les gouverner tous ».



## CSS aujourd'hui : standards et diversité des navigateurs

Hélas, il faut attendre que l'an 2000 pointe son nez pour que cette spécification soit implémentée correctement par les éditeurs de navigateurs. C'est Internet Explorer 5 Mac qui obtient le premier un taux de conformité frôlant les 100 %. Une belle prouesse... qui survient avec quatre ans de retard si l'on se fie au calendrier du W3C, avec des bogues persistants et pas d'implémentation des CSS niveau 2, dont la recommandation par le W3C date de 1998.

L'industrie du butineur est alors à la traîne et freine l'adoption des dernières avancées du Web par les professionnels d'Internet : pour prendre en charge les navigateurs de générations précédentes, les développeurs conservent leurs bonnes vieilles habitudes de codage en tableaux, aboutissant à des sites peu sémantiques, truffés d'informations de présentation. En raison du lourd retard technologique accusé par les navigateurs, CSS a donc du mal à s'imposer. Certains développeurs voient même dans ce nouvel apprentissage une perte de temps, considérant qu'un standard aussi mal implémenté est, de fait, voué à l'échec.

Pour des raisons de productivité, la tendance s'oriente vers une pratique qui sera fort dommageable au Web des années 2000 : développer ses sites pour un seul navigateur à la fois, celui qui domine le marché, d'abord Netscape (plus de 80 % de parts de marché à son apogée), puis Internet Explorer. Cette pratique donne lieu à un désastre général : des sites entiers cessent de fonctionner lorsqu'ils sont visités sur une plate-forme différente de celle pour laquelle ils ont été développés. Qui n'a pas en mémoire la mention apposée sur de nombreux sites web de l'époque : « Site optimisé pour Internet Explorer 6 » ? Devant une telle débâcle, les designers désireux d'exprimer leur talent graphique se tournent alors vers des technologies propriétaires comme Flash...

Malgré toutes ces difficultés, les recommandations du W3C s'imposent petit à petit comme la voie à suivre par les fabricants de navigateurs.

Les développeurs ne sont plus les seuls à toucher au HTML et certains designers commencent à mettre la main à la pâte. Le 7 mai 2003, Dave Shea dévoile CSS Zen Garden, un projet personnel lancé dans le but de montrer au monde des concepteurs web encore sceptiques que l'on peut réaliser des choses fantastiques à l'aide du langage CSS. L'idée est matérialisée sous la forme d'une page web codée en XHTML valide, associée aux feuilles de styles de talentueux designers. Le principe est simple : chaque designer peut composer la présentation d'un document (dont la structure est imposée) grâce aux feuilles de styles.

Avec son « jardin zen », Dave ouvre la porte des standards du Web aux artistes. Il pave la route du futur en incitant les techniciens à faire fi des pratiques du passé pour se diriger vers un monde excitant et en pleine ébullition.

Figure 1-2  
<http://www.csszengarden.com/>



Le succès de ce site participatif est immédiat. On recense à ce jour 210 designs acceptés, sans compter les 1 000 « rejetés ». Excellent laboratoire de techniques et savoirs variés, CSS Zen Garden reste en 2012 une source inspiration pour quiconque souhaite appréhender sereinement le langage CSS.

Mais revenons à 2001 et à la guerre des navigateurs qui fait rage, Internet Explorer grignotant des parts de marché à Netscape qui peine à conserver son avance. Le navigateur de Microsoft doit alors faire face à un nouveau challenger : Firefox (qui a porté d'autres noms avant celui-ci : Phoenix puis Firebird). Le navigateur au panda roux, développé par la fondation Mozilla, devient de plus en plus populaire auprès des professionnels du Web jusqu'à toucher le grand public et atteindre de fortes parts de marché dans certains pays.

Figure 1-3  
Le fameux panda roux,  
emblème de Firefox



À l'époque, Firefox est le premier navigateur moderne capable de rivaliser avec Internet Explorer. Libre et gratuit, couramment mis à jour, plus rapide, multi-plates-formes et proposant un système de modules, Firefox présente de nombreux avantages comparé au navigateur au « e » bleu. Petit à petit, l'oiseau fait son nid : ce navigateur alternatif gagne la confiance des internautes. Le poids d'Internet devient tel dans les foyers que Microsoft doit réagir pour ne pas se faire devancer par ses concurrents.

Les ingénieurs de Redmond dégainent un peu tard et sortent en 2011 la première version d'Internet Explorer qui soit respectueuse des standards, même si cette version 9 du navigateur ne reconnaît que partiellement les CSS3.

Pendant ce temps, Google lui aussi est entré dans la course avec Chrome, un navigateur basé sur le moteur de rendu WebKit, plus connu pour être à la base de Safari, le navigateur par défaut des ordinateurs Apple et périphériques iOS (iPad, iPhone, iPod touch).

Mi-2013, Google fait un pas de côté en créant le moteur de rendu Blink à partir de WebKit pour l'intégrer dans Chrome. Opera saute dans le train en marche et abandonne Presto au profit de Blink en devenant aussi contributeur. Cette séparation du code entre WebKit et Blink permet à Google et Opera d'innover avec une vélocité accrue...

La course est désormais à qui de Chrome, Firefox ou Internet Explorer proposera les avancées technologiques les plus spectaculaires. La cadence de mise à jour s'accélère pour suivre le train de l'innovation : Chrome, dont la première version a vu le jour en 2008, en est déjà à sa version 30 au moment où ce livre est publié dans sa seconde édition. Mozilla a aussi revu sa feuille de route à partir de sa version 4 (2011) pour délivrer des mises à jour majeures quasi tous les mois. Internet Explorer 10 voit le jour en 2012, après un cycle court, comparé aux nombreuses années qui ont séparé IE6 (2001) de IE7 (2006). La version 11 est publiée en octobre 2013 et Microsoft clame que son navigateur est le plus rapide à cette date dans le test Sunspider, qui évalue la rapidité d'exécution de code JavaScript. Ce rythme soutenu dénote un fort désir d'avancer et trahit l'importance de l'innovation sur ce terrain.

Bien que certains modules de CSS3 ne soient pas encore passés au stade de recommandation par le W3C, les navigateurs implémentent déjà des propositions de nouveautés comme les animations, les transitions, les coins arrondis, les dégradés de couleurs... C'est avec les contributions du moteur de rendu WebKit que de nombreuses nouveautés de CSS3 sont apparues.

On peut dire que les navigateurs dits « modernes » ont devancé le W3C sur la plupart des questions techniques liées au HTML5 (CSS3, JavaScript, HTML), et que celui-ci devient en quelque sorte l'arbitre du match que se livrent les fabricants de navigateurs à coups de propositions innovatrices.

Si tout n'est pas rose dans le monde des navigateurs, on apprécie de voir une industrie entière se diriger vers une harmonisation technologique globale. C'est une belle victoire pour les standards, qui profite tout d'abord à l'utilisateur, le réel bénéficiaire d'un Web où tous les contenus sont accessibles quel que soit son terminal.

## Le futur des CSS

L'avenir d'Internet est tout tracé : des utilisateurs qui se connectent à de multiples services, sites et applications via une interface web. Pour satisfaire des besoins croissants en interactivité et rapidité, les navigateurs se perfectionnent, deviennent de plus en plus agiles au profit d'une expérience utilisateur accrue. Firefox OS est un exemple parfait de système se basant sur ces standards ouverts : l'interface du système d'exploitation mobile développé par Mozilla est codée en HTML et habillée en CSS.

### EN SAVOIR PLUS **Firefox OS**

Mozilla tente de révolutionner le monde du mobile avec un système d'exploitation pour smartphone appelé Firefox OS. Le système lui-même et les applications sont conçus à l'aide de HTML, CSS et JavaScript, ce qui rend le développement sur cette plate-forme très aisé et démocratique (fidèlement à l'éthique de Mozilla).

▶ <http://www.mozilla.org/firefox/os/>

Pour relever les défis de demain, notamment l'utilisation sur différents terminaux (TV, mobile, tablette), il faut exploiter à bon escient les toutes dernières innovations en termes de conception web.

Jusqu'ici, on s'est contenté de la propriété `float` pour concevoir des mises en page à l'aide de blocs flottants... Mais après toutes ces années, il est vraiment temps que CSS offre un terrain de jeu bien plus adapté à ce type d'exercice.

## Un Web adaptatif

Le Web se dirige clairement vers une consommation des contenus très orientée vers la mobilité. Pour consulter des informations personnalisées sur tous nos périphériques high-tech, que l'on soit au bureau devant son écran d'ordinateur, dans le bus avec son mobile ou dans son salon avec sa tablette sur les genoux, les feuilles de styles permettront bientôt d'adapter encore mieux la mise en page à chaque usage en particulier. Certaines techniques sont disponibles comme les *Media Queries*. De gros efforts sont fournis sur ce module, tant par le W3C que par les fabricants de navigateurs.

La communauté se pose actuellement des questions relatives aux images adaptatives : comment servir une image de petite taille à un mobile sans pénaliser le rendu et les performances sur le même site visité depuis un ordinateur de bureau ? Actuellement, plusieurs solutions existent, à base de JavaScript et/ou de détection d'agent utilisateur, mais aucune d'entre elles n'est viable. Le W3C a lancé un appel à la communauté pour discuter de ce sujet entre professionnels du Web.

Cette discussion peut être suivie sur :  
▶ <http://www.w3.org/community/respimg/>

**Figure 1-4**

Un site, plusieurs apparences.  
Ici, le même blog sous  
différents périphériques  
<http://blog.kaelig.fr/>



## Les régions CSS

Le futur de CSS est aussi du côté de la mise en page : nous verrons naître des innovations et de sensibles améliorations sur les modules concernés, notamment **Layout**, **Regions** et **Flexbox** (boîtes flexibles), qui sont trois des nouveautés CSS3 les plus prometteuses pour les éditeurs de contenus.

### **EN SAVOIR PLUS** Positionnement avancé en CSS

Pour en savoir plus sur les nouvelles façons de pratiquer le positionnement des blocs avec CSS3, lisez le livre *CSS avancées* de Raphaël Goetter. Le chapitre 5 y est entièrement dédié.

📖 Raphaël Goetter, *CSS avancées*, Eyrolles, 2012

À propos du module **Regions**, c'est en mai 2011 qu'Adobe a soumis au W3C une proposition de module dédié au « zoning » de page. Nous avons déjà à notre disposition des outils de mise en page, dont le positionnement absolu, les éléments flottants et les tables, alors pourquoi chercher plus loin ?

Passé au statut de brouillon (*Editor's Draft*) le 6 avril 2012, le module **Regions** CSS est un moyen avancé d'organiser un document selon un flux de contenu non conventionnel, proche des possibilités de mise en page du monde du papier.

Le principe des régions est simple et se base sur un constat bien réel : aujourd'hui, nous savons étaler un texte sur plusieurs zones ou colonnes, mais pour cela, nous devons mettre en œuvre un découpage HTML complexe et quasi impossible à maintenir (il faut notamment calculer le nombre de caractères pouvant tenir dans chaque zone...). Avec les régions CSS, le document comporte une mise en page prédéfinie, à laquelle on fournit du contenu de manière brute. En s'adaptant au flux de texte délivré, les régions incorporent le contenu.

## HTML

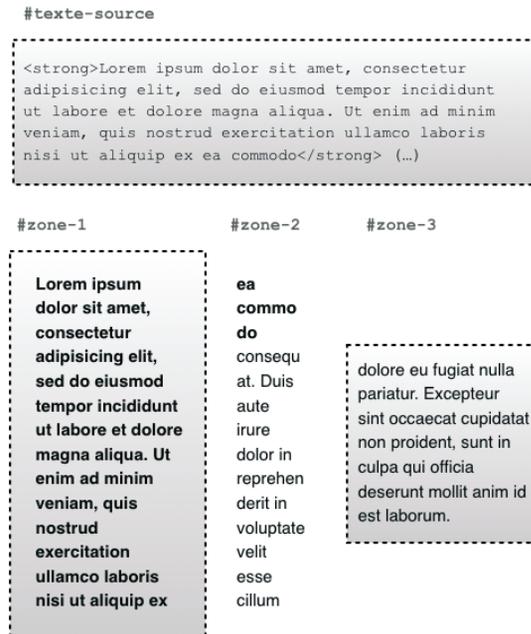
```
<div id="texte-source">
  <strong>Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed
  do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim
  ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut
  aliquip ex ea commodo</strong> consequat. Duis aute irure dolor in
  reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla
  pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa
  qui officia deserunt mollit anim id est laborum.
</div>
<div class="css-region" id="zone-1"></div>
<div class="css-region" id="zone-2"></div>
<div class="css-region" id="zone-3"></div>
```

## CSS

```
#texte-source {
  flow-into: contenu-principal;
}
.css-region {
  flow-from: contenu-principal;
}
```

Figure 1-5

Des colonnes de tailles différentes avec les régions CSS



#### NORMALISATION Propositions d'Adobe au W3C sur les régions CSS

Adobe publie sur son site le détail de sa proposition au W3C ainsi qu'un descriptif du fonctionnement des régions. Vous trouverez aussi une documentation étendue et à jour sur le site du W3C.

- « *CSS regions Module Level 3 Editor's Draft* » (Vincent Hardy, Alex Mogilevsky)  
▶ <http://dev.w3.org/csswg/css3-regions/>
- « *CSS3 regions: Rich page layout with HTML and CSS3* » (Arno Gourdol)  
▶ <http://www.adobe.com/devnet/html5/articles/css3-regions.html>
- « *CSS regions: Build complex, magazine-like CSS layouts using web standards* »  
▶ <http://labs.adobe.com/technologies/cssregions/>

## Les exclusions CSS3

Aujourd'hui, en CSS, nous n'avons pas la possibilité de commander au texte d'épouser les contours d'une image centrée dans un paragraphe. Nous sommes limités aux positionnements flottants à droite ou à gauche (`float: left;` et `float: right;`).

Les exclusions CSS3 viennent combler un manque dans le paysage des positionnements flottants : positionner un élément de nombreuses manières dans le flux du

contenu. Il faudra encore attendre quelques mois avant de pouvoir en profiter. De plus, il n'est pas dit que des *polyfills* (équivalents pour navigateurs qui ne proposent pas cette fonctionnalité) existeront pour ce type de besoin. Quoiqu'il en soit, cette nouveauté est plutôt excitante : imaginez ce qu'elle peut apporter aux éditeurs dans la mise en page de leurs contenus !

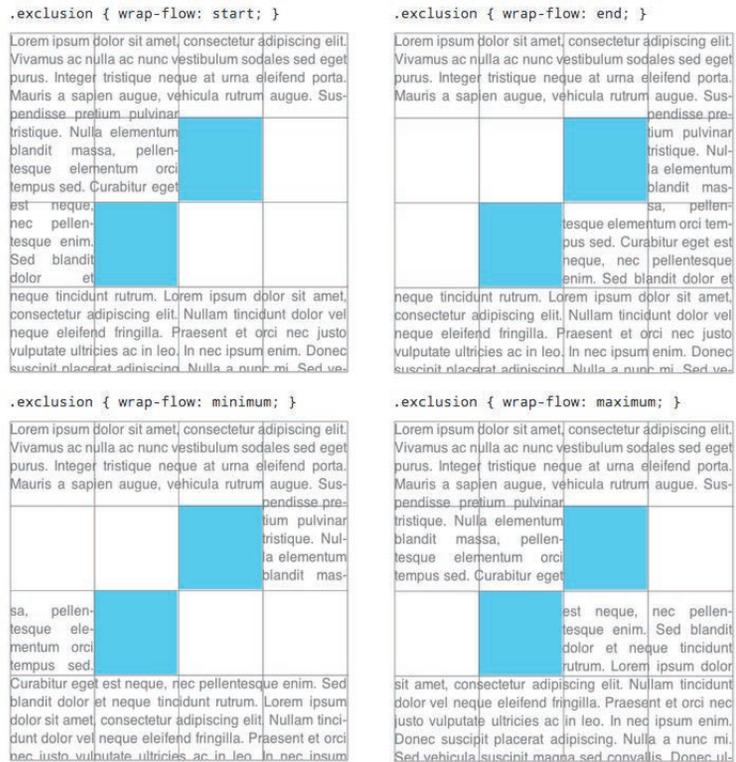
**NORMALISATION Brouillon Adobe et Microsoft sur les exclusions**

Le CSS Working Group a publié un brouillon à l'initiative de deux ingénieurs de chez Adobe et Microsoft, Vincent Hardy et Rossen Atanassov.

▶ <http://dev.w3.org/csswg/css3-exclusions/>

**Figure 1–6**

Une mise en page vivante grâce aux exclusions CSS <http://dev.w3.org/csswg/css3-exclusions/>



## La liberté typographique

L'industrie de l'impression papier a toujours possédé une avance considérable sur son concurrent numérique pour ce qui est de l'affichage typographique. Entre les polices crénelées (les fameux effets d'escaliers qui tendent à disparaître grâce aux écrans à

haute densité de type Retina depuis l'iPhone 4), la mauvaise gestion des polices et le rendu disparate d'une plate-forme à l'autre, on peut dire que l'informatique a encore des efforts à fournir avant de rattraper le papier.

Toutefois, le contrôle de l'affichage typographique est de mieux en mieux géré par les différents périphériques connectés. Nous avons déjà de magnifiques exemples sur la Toile conçus par des designers expérimentés dans le domaine de la typographie.

**Figure 1-7**

Un usage intensif de @font-face sur le site de l'agence Yebocreative  
<http://www.yebocreative.com>



Les nouveautés relatives à la typographie qui profiteront à la créativité sur le Web sont :

- @font-face : dites adieu à la domination d'Arial, Georgia et Verdana sur tous les sites web. Importez les polices qui vous plaisent pour aller plus loin dans le design ;
- la césure automatique (*hyphenation* en anglais) : pour une justification et un colonnage du texte toujours lisibles ;
- les ligatures entre caractères : l'élégance typographique à l'état brut.

Il ne s'agit pas tant d'attendre des avancées du côté du W3C, mais plutôt que ces fonctionnalités soient implémentées dans les navigateurs.

**Figure 1–8**

Le site de la conférence  
Ampersand et ses invités  
de marque  
<http://2012.ampersandconf.com/>



**Figure 1–9**

Les ligatures, marques d'une  
typographie sophistiquée



**CONSEIL DE PRO** La typographie sous Windows

Quand vous implémentez de telles nouveautés, n'oubliez pas de tester leur rendu dans les navigateurs et systèmes d'exploitation les plus courants. En effet, Windows propose différents modes de rendu typographique en apposant un traitement différent d'une machine à l'autre selon que la technologie de lissage ClearType est activée ou non. Faites attention aussi à vérifier l'apparence du texte sous Windows 7, qui lui-même apporte un moteur de rendu complètement revu (notamment sous Internet Explorer 9).

## Nouvelles structures de langage

Les préprocesseurs CSS (pour en savoir plus sur ce sujet, consultez le chapitre 7) influencent fortement la qualité et la rapidité de développement, tout en ajoutant une couche de complexité pour le novice. CSS veut être un langage démocratique, accessible à tous. Toutefois, il a été proposé au W3C d'intégrer des fonctionnalités présentes dans Sass (et WebKit pourrait les incorporer très bientôt de son côté) :

- les mixins, sortes de fonctions réutilisables (abordés dans le chapitre 7) ;
- les variables (déjà à l'état de brouillon W3C au moment de la publication de ce livre : <http://dev.w3.org/csswg/css-variables/>) ;
- les sélecteurs imbriqués (abordés dans le chapitre 7) ;
- les opérateurs : addition, soustraction, multiplication, division, trigonométrie...