

[CSS] Les compteurs

Par Itello



www.openclassrooms.com

*Licence Creative Commons 5 2.0
Dernière mise à jour le 22/02/2012*

Sommaire

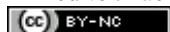
Sommaire	2
Lire aussi	1
[CSS] Les compteurs	3
Notions à avoir	3
Exemple	3
Problème à résoudre	4
Partie HTML	4
Fonctionnement d'un compteur	5
Initialiser son compteur	5
Incrémenter un nombre	6
Insertion du compteur	6
Un petit résumé	6
Résolution du problème	7
Solution du problème	7
Explication (pour ceux qui n'y sont pas arrivés)	7
Allez plus loin	7
De petites améliorations pour notre compteur	8
Les compteurs imbriqués	8
Partager	9

2 [CSS] Les compteurs



Mise à jour : 22/02/2012

Difficulté : Facile  Durée d'étude : 2 heures



Il est possible grâce aux feuilles de style (CSS) d'insérer des compteurs sur une page web. Ceux-ci permettent de numéroté chapters, paragraphes ou toute autre balise sans avoir à toucher au code HTML.

Ils vous permettront donc de réaliser une numérotation cohérente et automatisée de vos différentes pages web en modifiant simplement votre feuille de style. Vous pourrez ainsi insérer les fameuses numérotations devant vos titres (**I**, **A**), **1** - ou encore **a**).

Navigateur	Compatibilité
Mozilla Firefox	1.0 et supérieur
Internet Explorer	8.0 et supérieur
Safari	3.0 et supérieur
Opera	8.0 et supérieur

Dans ce tutoriel, je vous expliquerai le fonctionnement des compteurs et vous donnerai diverses idées d'utilisation de ceux-ci. Bonne chance ! 😊

Sommaire du tutoriel :



- [Notions à avoir](#)
- [Problème à résoudre](#)
- [Fonctionnement d'un compteur](#)
- [Résolution du problème](#)
- [Allez plus loin](#)

Notions à avoir

Pour utiliser les compteurs, il faut connaître certaines notions CSS assez basiques.

Mais pour ceux qui ne les connaissent pas (ou bien qui veulent une petite révision 😊), les voici :

- `:before` permet d'insérer des éléments avant le contenu de la balise.
C'est-à-dire qu'on peut rajouter des images, des textes, voire des compteurs.
- La propriété `content` permet d'insérer du texte avant (`:before`) ou après (`:after`) une balise.

Exemple

Code : CSS

```
/* Rajoutera "Titre :" avant la balise <h1> */  
h1:before {
```

```
    content : "Titre :";  
}
```

Problème à résoudre

Quoi de mieux qu'un bon exemple pour bien comprendre l'utilité d'un tutoriel ?

Vous avez écrit un super article mais assez long. Celui-ci contient plein de titres `<h1>` .

Vous souhaitez réaliser une présentation semblable à celle des livres en rajoutant des *Chapitre x* avant vos titres, mais ça sera très très long de le faire pour chacune de vos balises. 😞

Sauf en utilisant Super CSS ! 😊



Pourquoi utiliser les CSS alors qu'on peut faire sans ?

Pour plusieurs raisons.

Tout d'abord, vous pourrez, grâce aux multiples feuilles de style faire des présentations différentes, avec ou sans *Chapitre x*.

Mais ceci marche aussi :

Code : HTML

```
<h1><span class="chapitre">Chapitre 8 :</span> Vive le fromage</h1>
```

Code : CSS

```
span.chapitre { display : none; }
```

L'avantage premier de l'utilisation des compteurs, c'est que tout est automatique. Plus rien à faire, les compteurs s'occupent de tout.

De plus, lors d'une longue création ou même d'un site entier, rajouter du code sur toutes les balises de titre est assez inimaginable.

Une petite idée pour la route : utiliser les compteurs pour [la feuille de style d'impression](#) de la page. Cela donne un rendu structuré et pro.

Alors, convaincu ?

Partie HTML

La partie HTML vous est offerte car elle ne sera pas à modifier. En effet, nous n'utiliserons que les CSS durant tout le tutoriel.

Code : HTML

```
<h1>Qualité des pamplemousses vendus</h1>  
<p>Les pamplemousses vendus par l'Italie nordique ne sont pas juteux à souhait. Il est tout de même scandaleux qu'aujourd'hui, nous ayons encore à supporter des fruits qui ne sont pas juteux !</p>  
  
<h1>Manifestation contre les pamplemousses</h1>  
<h2>Cette semaine</h2>  
<p>Il faut taper fort et manifester dès aujourd'hui ! Nous devons lutter contre les pamplemousses qui ne sont pas assez juteux !</p>  
  
<h2>Plus tard !</h2>  
<p>Il ne faudra pas se relâcher, je prévois une manifestation par semaine pendant 150 ans !</p>
```

Essayer !

Bon, c'est parti, on va commencer !

Fonctionnement d'un compteur

Pour pouvoir résoudre notre problème, il faut savoir utiliser les compteurs. C'est drôle comme coïncidence, mais c'est ce sur quoi va porter la suite de ce tutoriel. 🤖

Les compteurs permettent de compter (logique ?) le nombre de balises identiques contenues dans notre page web.

Un compteur a besoins d'être :

- **initialisé** avant sa première utilisation ;
- **incrémenté** ;
- **inséré** sur la page web.

Initialiser son compteur

Il faut d'abord donner un nom au compteur. Dans notre cas précis, on l'appellera « chapitre ».

Avant utilisation, on doit **obligatoirement** initialiser notre compteur. Pour ceci, nous utilisons la propriété **counter-reset** :

Code : CSS

```
body {  
  counter-reset: chapitre; /* Nous initialisons le compteur  
  chapitre */  
}
```

Le compteur a été initialisé sur la balise **<body>** .

C'est-à-dire que celui-ci est déclenché à partir de la balise **<body>** et commencera à compter après celle-ci.

Par défaut, le compteur est initialisé à 0 mais on peut changer ça :

Code : CSS

```
body { counter-reset: section -1 } /* Nous initialisons le compteur  
section à -1 */  
body { counter-reset: prout 99 } /* Nous initialisons le compteur  
prout à 99 */
```



Opera et Konqueror fonctionnent correctement sans que le compteur soit initialisé.
Mais ces deux navigateurs sont ainsi en contradiction avec les recommandations du W3C.

Sur Mozilla Firefox, un compteur doit être initialisé sur une balise.

Code : CSS



```
body:before {  
  counter-reset: chapitre;  
}
```

Dans le cas suivant, le compteur n'est pas initialisé sur la balise **<body>** mais avant celle-ci.

Cela ne fonctionnera donc pas correctement.

Incrémenter un nombre

Un code sera plus compréhensible qu'un long discours :

Code : CSS

```
h1:before {  
  counter-increment: chapitre; /* Ajoute 1 au compteur chapitre */  
}
```

La propriété **counter-increment** ajoute 1 au compteur par défaut.

Ça aussi, on peut le changer :

Code : CSS

```
h1:before {  
  counter-increment: chapitre + 4; /* Ajoute 4 au compteur chapitre  
  */  
}
```

Mais l'utilité n'est pas visible du premier coup d'œil. 😊

Insertion du compteur

En fait, nous n'avons encore rien changé au niveau de l'apparence du site.

Le plus important reste à faire : l'insertion de notre compteur sur la page web.

Pour ceci, on utilisera la propriété **content** :

Code : CSS

```
h1:before {  
  content: counter(chapitre) ".";  
}
```

Vous ne devriez pas avoir de mal à comprendre ce code.

Counter() sert à afficher un compteur avec **content**, c'est comme **url()** qui affiche une image.

Les guillemets doubles **"..."** permettent quant à elles de signaler qu'on affiche du texte normal. Dans notre cas, on affiche un point après le nombre pour réaliser cet effet :

1 . Vive le fromage

Ce n'est pas compliqué... 😊

Un petit résumé

Un petit résumé pour bien tout se remettre en mémoire ?

Code : CSS

```
body {  
  counter-reset: chapitre; /* Nous initialisons le compteur  
  chapitre */  
}  
h1:before {  
  counter-increment: chapitre; /* Ajoute 1 au compteur chapitre */  
}
```

```

    content: counter(chapitre) ". "; /* Affiche notre compteur */
}

```

Ce sera tout pour le fonctionnement d'un compteur.
Relisez cette partie plusieurs fois si vous en avez besoin.
Nous allons maintenant essayer de résoudre notre problème.

Résolution du problème

Maintenant que les compteurs n'ont plus le moindre secret pour vous, on peut essayer de résoudre notre petit problème. Avec le code HTML donné précédemment, voici [le résultat que nous souhaitons obtenir](#). C'est vrai que ça peut paraître dur au premier abord, mais avec un peu de réflexion, vous y arriverez. N'hésitez évidemment pas à relire la partie "Fonctionnement d'un compteur".

Solution du problème

J'arrête de vous faire languir 😊 et je vous donne la solution que vous devriez avoir :

Secret (cliquez pour afficher)

Code : CSS

```

body {
    counter-reset: chapitre; /* Nous mettons à zéro le compteur
    chapitre */
}
h1:before {
    counter-increment: chapitre; /* Ajoute 1 au compteur chapitre
    */
    content: "Chapitre " counter(chapitre) " : ";
}
h1 {
    counter-reset: section;
    /* On ré-initialise le compteur section à chaque fois qu'on
    tombe sur la
    balise <h1> */
}
h2:before {
    counter-increment: section; /* Ajoute 1 au compteur section */
    content: counter(section) ". ";
}

```

Essayer !

Explication (pour ceux qui n'y sont pas arrivés)

- Nous initialisons le compteur **chapitre** (à zéro) dès **<body>** (**body**) grâce à **counter-reset**.
- Avant chaque titre **<h1>** (**h1:before**), nous rajoutons 1 au compteur **chapitre** (**counter-increment**) puis nous affichons "Chapitre x :". La première fois, celui-ci affichera donc 1 (le compteur est à zéro et on rajoute 1).
- Sur chaque titre **<h1>**, nous mettons à zéro le compteur **section** qui sert à numérotter les titres **<h2>**. On veut compter les titres **<h2>** après un titre **<h1>**.
- Avant chaque titre **<h2>** (**h2:before**), nous rajoutons 1 au compteur **section** (**counter-increment**) puis nous affichons "x.". La première fois, celui-ci affichera donc 1 (le compteur est à zéro et on rajoute 1).

Allez plus loin

De petites améliorations pour notre compteur

counter-reset

La propriété **counter-reset** permet d'initialiser plusieurs compteurs à la fois.

Code : CSS

```
h1 { counter-reset: section -1 imagenum 99; }
```

Dans le cas suivant, nous initialisons le compteur section à -1 et le compteur imagenum à 99 sur la balise **<h1>**.

counter()

Vous pouvez personnaliser le style du numéro pour votre compteur

Code : CSS

```
h1 { content : counter(chapitre, upper-roman); }  
/* Compteur sous la forme I-II-III */
```

Et pour savoir quoi mettre à la place de **upper-roman** :

disc, **circle**, **square**, **none** sont possibles mais il est tout de même plus utile d'afficher des listes numérotées :

Valeurs possibles pour la propriété counter()

Valeur possible	Action	Exemple
Compatible Internet Explorer :		
decimal	Nombres décimaux.	1, 2, 3, etc.
lower-roman	Chiffres romains minuscules.	i, ii, iii, etc.
upper-roman	Chiffres romains majuscules.	I, II, III, etc.
lower-latin ou lower-alpha	Numérotation alphabétique minuscule.	a, b, c, etc.
upper-latin ou upper-alpha	Numérotation alphabétique majuscule.	A, B, C, etc.
Incompatible Internet Explorer :		
decimal-leading-zero	Nombre sur deux décimales.	01, 02, 03, etc.
lower-greek	Numérotation avec l'alphabet grec.	...
Rarement compatible :		
hebrew	Numérotation hébraïque.	...
georgian	Numérotation géorgienne.	an, ban, gan, etc.
armenian	Numérotation arménienne.	...
cjk-ideographic	Numérotation en cjk-ideographic.	non-disponible
hiragana	Numérotation en hiragana.	a, i, u, etc.
katakana	Numérotation en katakana.	A, I, U, etc.
hiragana-iroha	Numérotation en hiragana iroha.	i, ro, ha, etc.

Source : <http://www.laltruiste.com/courscss/listestyle.html>

Les compteurs imbriqués



Mais que nous apportent de plus les compteurs imbriqués ?

```
1 item
2 item
  2.1 item
  2.2 item
  2.3 item
    2.3.1 item
    2.3.2 item
    2.3.1 item
    2.3.2 item
    2.3.3 item
  2.4 item
3 item
4 item

1 item
2 item
```

Les conteurs imbriqués seront utiles pour les listes. Vous pouvez voir un exemple d'emploi de ceux-ci à gauche de l'écran.

Eh bien, c'est fini ! 😊

Je souhaite remercier les personnes suivantes :

- **les créateurs** du premier MC sur le HTML / CSS qui m'ont fait découvrir cette fonction en recherchant sur la Toile ;
- **Laurent Jouanneau** pour son article « [Précisions sur les compteurs CSS](#) » qui m'a appris qu'il fallait initialiser son compteur ;
- et **vous**, pour avoir lu ce tuto jusqu'au bout.

N'hésitez pas à lire l'article de **Laurent Denis** « [Faire des statistiques à l'aide des compteurs CSS](#) ». Il y donne une idée originale concernant l'utilisation des compteurs.

Partager

