



Feuilles de style CSS

CSS (Cascade Style Sheets)



Histoire du positionnement en CSS

- Lorsque naît le Web en 1994, sa portée est tout d'abord extrêmement limitée et essentiellement destinée à un monde d'étudiants ou de scientifiques.
- Adaptées à des documents de type principalement textuel, où les éléments sont disposés de façon linéaire, empilés les uns sur les autres (titres, paragraphes, listes), les CSS sont dépassées dès leur apparition par un Web en pleine ébullition : gigantesque marché naissant, **Internet ne pouvait se réduire à de simples pages textuelles et frustrer ainsi tous les graphistes en quête de l'aspect le plus vendeur pour leur vitrine.**

Entre tableaux, cadres et calques

- Blocs contigus, positionnements originaux, empilements, grilles et architectures complexes, les besoins des concepteurs web deviennent de plus en plus croissant.
- Les concepteurs se sont rabattus sur des mécanismes qui apportent des réponses concrètes à leurs besoins graphiques :
 - Les tableaux de mise en page (balise HTML <table>) ;
 - Les cadres (*frames*) ;
 - Les calques (positionnement absolu).



Développement du design des sites

La plus grande **Fnac**, disponible en ligne **24h/24**

Espace Adhérents

Mon panier
Votre panier est vide

Mon compte Newsletter Fnac Aide

ACCUEIL LIVRES DISQUES DVD & VIDÉO LOGICIELS & JEUX MICRO & TÉLÉCOM IMAGE & SON VOYAGES SPECTACLES JOUETS

fnac Music Télécharger

NOUVEAU! Développement PHOTO

Rechercher : Tous Produits

Bienvenue sur Fnac.com. C'est votre 1er achat ? [cliquez ici](#)

L'APOGÉE DE LA SAGA

PRIX VERT 24,99 €
Frais de port offerts

► La Revanche des Sith en DVD

► Tout l'univers Star Wars

Nouveauté Télévision

Téléviseur LCD 16/9
Sagem Axiom HD-L32-T
1 790 €
Exclusivité Internet

[Ajouter au panier](#)
► Découvrir la gamme Sagem Axiom

Prix Goncourt des lycéens

Magnus
Sylvie Germain
-5% **16,63 €**
17,50 €

[Ajouter au panier](#)
► Voir les prix littéraires 2005

Nouveauté PDA

PDA sous Windows Mobile
HP IPAQ rx1950
299 €

[Ajouter au panier](#)
► Plus de nouveautés PDA

Jeu-concours

Johnny Hallyday
Ma Vérité

Découvrez le nouvel album événementiel de Johnny et participez à notre jeu-concours : vous gagnerez peut-être un disque platine de "Ma Vérité" dédié par l'artiste !

19,99 €

[Ajouter au panier](#)
► Gagnez un disque platine dédié

Espace adhérents

Jusqu'à 10% de remise immédiate
Remises, chèques-cadeaux offerts, facilités de paiement... Découvrez tous les avantages de la Carte Adhérent et profitez dès aujourd'hui d'une remise de bienvenue.

► [Tous les avantages Adhérents](#)
► [Devenez Adhérent](#)

Salon du cheval
Paris Expo, du 3 au 11 décembre

2,50 € de réduction

Unité centrale + clé USB 2.0

Exclusivité Sonnerie

Sonnerie Pauline Croze

Services Fnac.com

Découvrez

- Nous rachetons votre matériel **NOUVEAU**
- Offrir et utiliser un chèque-cadeau
- Réserver une nouveauté
- Conseils personnalisés
- Espace affiliés

Informez-vous

- Dossiers techniques interactifs **NOUVEAU**
- Newsletter Fnac

Magasins Fnac

- Localisation et horaires
- Événements
- Les services Fnac

Engagements

- Sécurité & Confidentialité
- Les prix Fnac

Conseils personnels

Nos conseils sont mis à jour en temps réel à partir des derniers articles que vous avez consultés.

Livres :
 JavaScript pré et concis
David Flanagan
8,55 €

[Ajouter au panier](#)

Livres :
 Fonction publique
André de Laubadère; Yves Gaudemet
21,73 €

[Ajouter au panier](#)

Livres :
 Traité de droit administratif
André De Laubadère; Jean Venezia; Yves Gaudemet
26,07 €

[Ajouter au panier](#)

Plus de conseils personnalisés

Meilleures ventes

DVD & Vidéo

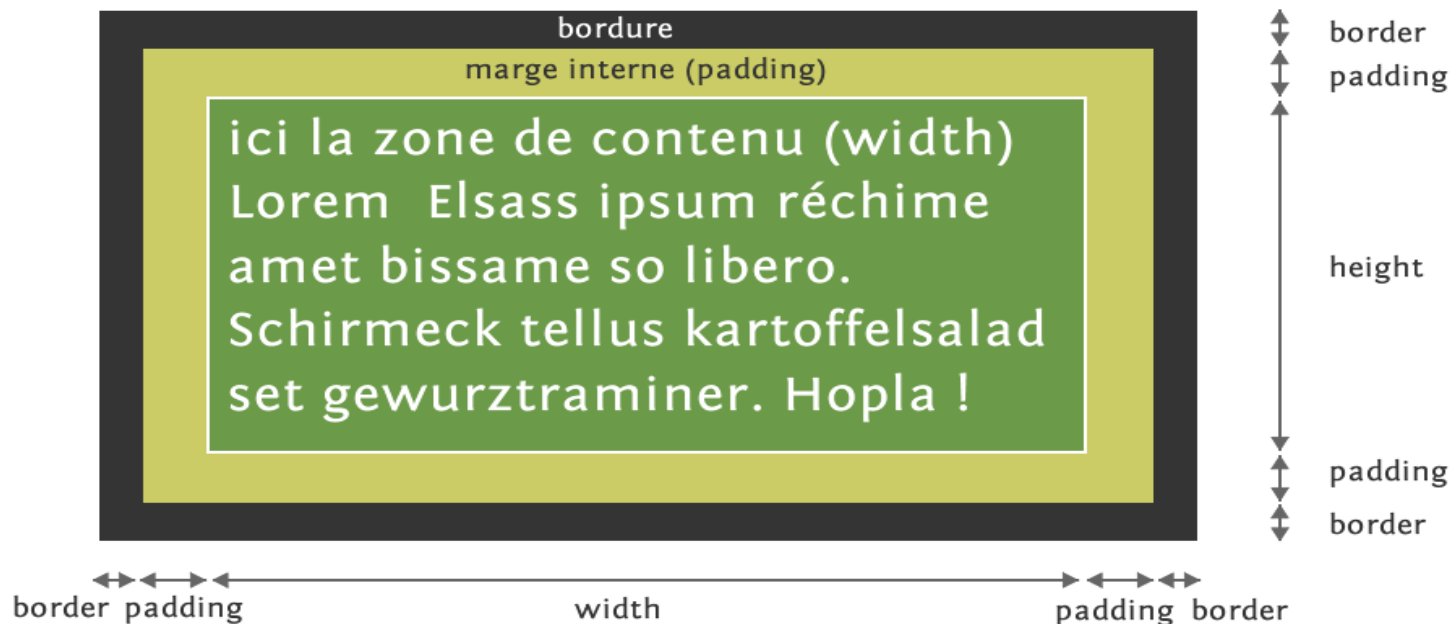
1. Kingdom of Heaven
Ridley Scott
19,99 €



Modèle de boîte

Selon les spécifications W3C, tout élément structuré par une balise HTML (ou plusieurs) se représente sous forme d'une « boîte » rectangulaire définie par diverses composantes que sont :

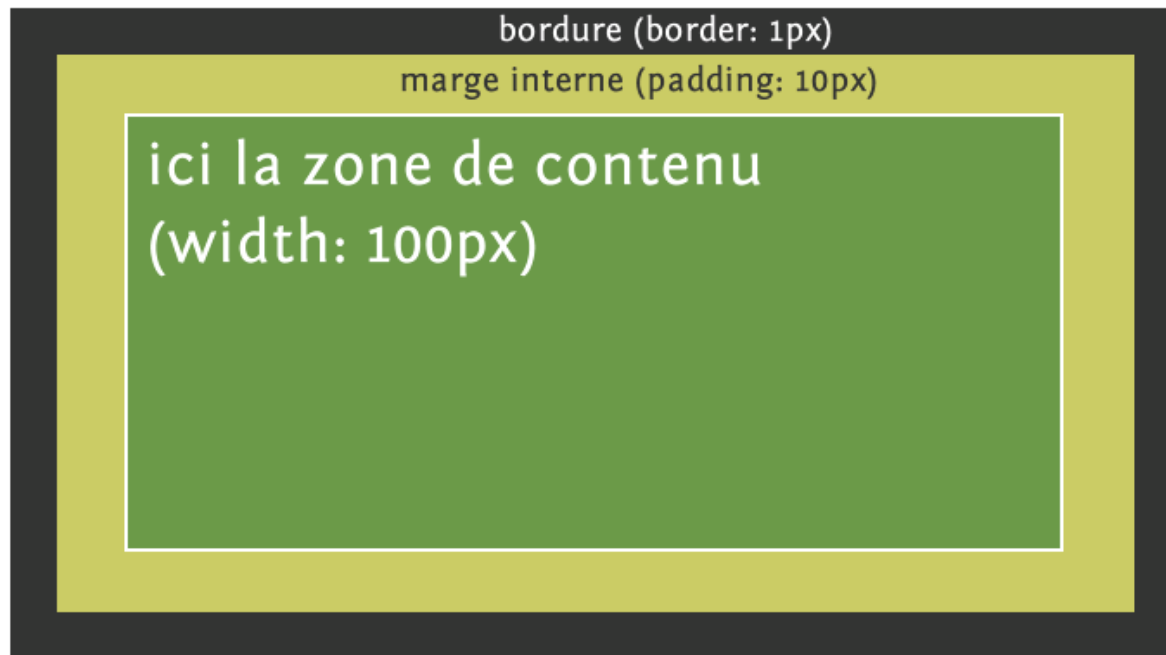
- zone de son contenu, représentée par une largeur et une hauteur : en CSS, il s'agit des propriétés **width** et **height** ;
- Espace réservé à la bordure de la boîte (propriété **border**) ;
- Marge interne à la boîte (**padding**), se situant entre la zone de contenu et la bordure.
- Marge externe (**margin**) et se situe hors de la boîte, au-delà de l'espace alloué à la bordure, elle affecte le positionnement de l'élément puisqu'elle pousse sa boîte ou ses soeurs environnantes.





Modèle de boîte

```
p {  
  width: 100px;  
  padding: 10px;  
  margin: 5px;  
  background-color: green;  
  border-style: groove;  
  border-color: black;  
  border-width: 0.5em  
}
```



1px + 10px + 100px + 10px + 1px Total = 122px



Les minima et maxima

Ces propriétés ne figent pas les dimensions de l'élément, à la différence des classiques **height** et **width**, mais elles se contentent de définir une valeur minimale ou maximale, quel que soit le contenu présent.

min-width : Largeur minimale (en pixels, pourcentage, **em**...) ;

max-width : largeur maximale ;

min-height : hauteur minimale ;

max-height : hauteur maximale.

```
body {  
    max-width: 500px;  
    min-width: 250px;  
}  
  
p {  
    max-width: 300px;  
    min-width: 200px;  
    padding: 20px;  
    margin: 5px;  
    background-color: green;  
    border-style: groove;  
    border-color: cyan;  
    border-width: 0.5em  
}
```

boîte 1
min-height:
200px

boîte 2
min-height:
200px

boîte 3
min-height:
200px

*Blocs de même hauteur minimale,
mais demeurant étirables en hauteur*



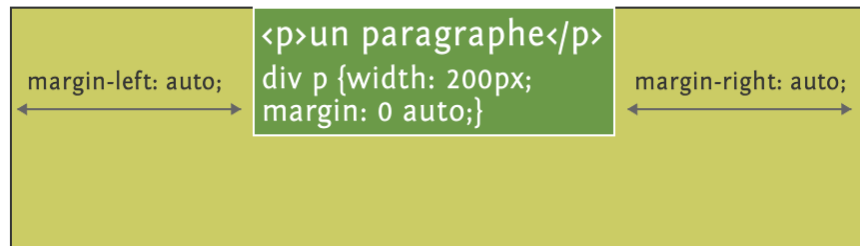
Centrage horizontal CSS

```
div p {  
  width: 200px;  
}
```

<p>un paragraphe</p> un parent <div>
div p {width: 200px}

L'alignement au centre du parent est définie à l'aide de marges externes latérales automatiques, c'est-à-dire en appliquant la valeur auto :

```
div p {  
  width: 200px;  
  margin-right: auto;  
  margin-left: auto;  
}
```



Syntaxe équivalente, mais optimisée

```
div p {  
  width: 200px;  
  margin: 0 auto;  
}
```



Fusion de marges

La fusion de marges est une particularité du modèle de boîte qui concerne les marges externes verticales (**margin-top et margin-bottom**) des éléments de type bloc positionnés dans le flux.

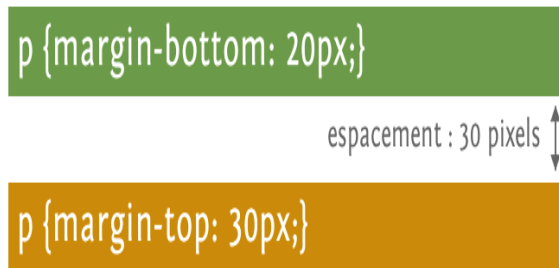
Lorsque deux ou plusieurs éléments sont adjacents, qu'ils soient frères ou imbriqués (parent-enfant), leurs marges verticales se combinent pour n'en former qu'une seule : la plus grande des deux.

Partie HTML

```
<p>La marge inférieure de ce  
paragraphe fusionne...  
</p>  
<p>... avec la marge supérieure de  
celui-ci  
</p>
```

Partie CSS

```
p {  
  background-color: green;  
  margin-bottom: 20px;  
}  
p + p {  
  margin-top: 30px; /* une marge haute  
sur le second paragraphe */  
}
```





- Tous les éléments d'une page web sont définis par la norme HTML en termes d'imbrications tolérées,
- Ils sont également dans leurs spécifications CSS au travers de la propriété intrinsèque **display**, qui leur confère un certain nombre de caractéristiques :
- Un rendu général sous forme de "boîte" ;
- Une disposition par défaut (les uns sous les autres, à côté, en haut à gauche du parent ou sous le frère précédent) ;
- D'éventuelles marges internes ou externes ;
- La possibilité d'être dimensionnés, etc.

La propriété **display accepte 17 valeurs depuis CSS 2.1 et quelques valeurs supplémentaires en CSS 3.**

Les plus couramment employées sont **block, inline et none**, mais le tableau ci-après vous dévoile quelques autres valeurs intéressantes :



Rendu des éléments

Display (défaut)	Exemples	Spécificités
block	<code><div></code> , <code><p></code> , <code></code> , <code><h1></code>	<p>Les éléments de rendu CSS display: block se placent <i>toujours l'un en dessous de l'autre par défaut</i> (comme un retour chariot).</p> <p>Par exemple : <code><p></code>, <code><div></code>, <code><section></code>, <code><article></code>, ...</p> <p>Par ailleurs, un élément bloc occupe automatiquement, par défaut, toute la largeur disponible dans son conteneur et peut être dimensionné à l'aide des propriétés telles que width, height, min-width, ou min-height...</p>
inline		<p>Les éléments de rendu display:inline <i>se placent toujours l'un à côté de l'autre afin de rester dans le texte</i>. leur taille va être déterminée par le texte ou les éléments qu'ils contiennent. (img, span, ...)</p> <p>Certaines propriétés de marges peuvent être appliquées aux éléments, comme les marges latérales (margin-left et margin-right).</p>
none	<code><head></code>	<p>L'élément est complètement retiré du flux, il n'est pas restitué par les agents utilisateurs.</p>
list-item	<code></code>	<p>Les éléments de rendu display:list-item ont un rendu de type block mais peuvent bénéficier des propriétés CSS liées aux puces (list-style...)</p>
inline-block	<code><input></code> , <code><select></code>	<p>Les éléments de rendu display:inline-block conservent les mêmes caractéristiques que les "inline", mais peuvent être dimensionnés, par exemple l'élément <code><input></code> d'un formulaire</p>



Le flux courant dans une page Web

- L'ordre dans lequel apparaissent les balises dans le code HTML est celui dans lequel les boîtes sont affichées et s'empilent dans le document.
- Ce schéma de positionnement par défaut se nomme **le flux courant**. La mise en place des différents éléments de la page s'organise par défaut selon le flux courant.
- Les règles de positionnement dans le flux courant sont relativement simples et intuitives.
- Chaque élément :
 - Est situé sur le même plan que les autres éléments dans le flux ;
 - se place le plus haut possible et le plus à gauche possible au sein de son parent ;
 - est dépendant de l'élément frère précédent :
- Deux éléments de type **block** s'empilent verticalement, l'un sous l'autre,
- Deux éléments de type **inline** se suivent sur la même ligne si la place le permet.



`<h1>`Titre de niveau 1

`<p>`Paragraphe

`<a>`lien

`<a>`lien

``image

``liste non ordonnée

``élément de liste

``élément de liste



Positionnement absolu

Une boîte en positionnement absolu peut être placée n'importe-où dans le code HTML et s'afficher à l'endroit de votre choix.

Ce mode permet de retirer totalement le contenu concerné du flux

- Placer les menus de navigation en fin de page, pour améliorer l'accessibilité de votre site en donnant un accès immédiat à son contenu dans les navigateurs textes, tout en les faisant apparaître en haut de page ou encore dans une colonne pour les navigateurs graphiques ;
- Créer plusieurs colonnes au positionnement indépendant de l'ordre dans lequel elles se trouvent en HTML.

Déclaration

position: absolute

```
.gauche {  
    position: absolute;  
    top: 1em;  
    left: 1%;  
    width: 18%;  
}
```



Positionnement absolu

```
#info {  
position: absolute;  
  
top: 0;  
left: 0; /* Coin supérieur gauche du bloc #info se place dans le coin supérieur  
         gauche de son référent */  
  
top: 0;  
right: 0; /* Coin supérieur droit du bloc #info se place dans le coin supérieur droit du  
          référent */  
left: 0;  
bottom: 0; /* dans le coin inférieur gauche du référent */  
  
top: 20px;  
left: 20px; /* à 20 pixels du coin supérieur gauche du référent */  
  
top: 10%;  
right: 10%; /* à 20 % du coin supérieur droit du référent */  
  
top: 50%;  
left: 50%; /* Coin supérieur gauche du bloc #info se place au centre de son référent */  
}
```

Référent : Premier ancêtre positionné



Positionnement absolu

```
16  .droite {
17      position: absolute;
18      top: 0;
19      right: 1%;
20      width: 18%;
21      color: red;
22  }
23
24  .gauche {
25      position: absolute;
26      top: 0;
27      left: 1%;
28      width: 20%;
29  }
30
31  .centre {
32      position: absolute;
33      top: 0;
34      left: 50%;
35      right: 50%;
36      width: 20%;
37  }
```



Positionnement absolu

```
<body>
<div class="centre">
  <h3>Liens de navigation horizontal</h3>
  <p>la propriété inline-block permet de disposer les éléments d'une liste horizontalement</p>
</div>
<nav>
  <ul>
    <li><a href="#Accueil">Accueil</a></li>
    <li><a href="#Clients">Clients</a></li>
    <li><a href="#Fournisseurs">Fournisseurs</a></li>
    <li><a href="#contact">Contact</a></li>
  </ul>
</nav>
<div class="droite">
  <p>Ce texte est positionné en absolute
</p>
</div>
<div class="gauche">
  
</div>
</body>
</html>
```



Liens de navigation horizontal

Ce texte est positionné en absolute

la propriété inline-block permet de disposer les
éléments d'une liste horizontalement

[Accueil](#) [Clients](#) [Fournisseurs](#) [Contact](#)



Positionnement fixe

- Le positionnement fixe (**position:fixed**) s'apparente au positionnement absolu, à l'exception des points suivants:
 - Lorsque le positionnement est précisé (**top, right, ...**), l'élément est toujours positionné par rapport à la fenêtre du navigateur.
 - L'élément est fixé à un endroit et ne pourra se mouvoir, même lors de la présence d'une barre de défilement.

En d'autres termes, la position initiale est fixée au chargement de la page, le fait qu'une éventuelle scrollbar puisse être utilisée n'a aucune influence sur le positionnement de l'élément : ***il ne bouge plus de la position initialement définie.***

```
div.fixed {  
    position: fixed;  
    bottom: 0;  
    right: 0;  
    width: 300px;  
    border: 3px solid #73AD21;  
}
```



Positionnement relatif

- La position relative (**position:relative**) permet de décaler un élément par rapport à une position de référence: celle qu'il avait dans le flux.
- Les éléments qui le suivent et le précèdent ne sont pas influencés par ce décalage puisqu'ils considèrent que l'élément est toujours dans le flux à sa position initiale.
- En pratique, ce comportement est rarement recherché bien qu'il puisse s'avérer utile dans certains cas.
- Attribuer à un élément une position relative peut par contre être pratique, voire indispensable, dans d'autres situations dont les plus courantes sont sans nul doute les suivantes:
- Servir de référent à un élément enfant positionné en absolu (rappelons qu'un élément positionné absolument grâce aux propriétés top, left, ... le fera par rapport à la fenêtre du navigateur à défaut d'avoir un parent lui-même positionné).



Positionnement relatif

```
strong {  
    position: relative;  
    top: 10px;  
    left: 10px;  
    background: #abc;  
}
```

Lorem Elsass ipsum réchime amet
bissame so **kiwi**. Leo Richard
Schirmeck tellus kartoffelsalad set

← élément #kiwi en
flux classique

Lorem Elsass ipsum réchime amet
bissame so **kiwi** Leo Richard
Schirmeck tellus kartoffelsalad set

← élément #kiwi
décalé avec
position: relative;
top: 10px;
left: 10px;



Positionnement flottant

La propriété **float** apparaît dès CSS1, classée dans les « modèles de mise en forme », tels que **inline** et **block**.

La propriété **float** indique qu'un élément doit être retiré du flux normal et doit être placé sur le côté droit ou sur le côté gauche de son parent.

Le texte et les autres éléments en ligne (**inline**) entoureront alors l'élément flottant.

Un élément flottant est un élément pour lequel la valeur calculée de **float** est différente de none.

Avant flottement (deux blocs en flux)

<p> Paragraphe en flux

<p> Autre paragraphe en flux. Lorem Elsass ipsum réchime amet bissame so libero. Schirmeck tellus kartoffelsalad set. Hopla !

bloc parent

Après flottement (premier bloc flottant à gauche)

<p> Paragraphe flottant <p> Autre paragraphe en flux. Lorem Elsass ipsum réchime amet bissame so libero. Schirmeck tellus kartoffelsalad set. Hopla !

bloc parent



Positionnement flottant

```
.jaune {  
  background-color: #ffff00;  
  float: right;  
  width: 100px;  
  margin: 0;  
}  
.verte {  
  background-color: #00ff00;  
}
```

Une boîte verte dotée d'un contenu plus long...

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Claritas est etiam processus dynamicus, qui sequitur mutationem consuetudium lectorum. Typi non habent claritatem insitam; est usus legentis in iis qui facit eorum claritatem.

Une boîte
jaune flottante

```
.content {  
  float: left;  
  width: 70%;  
}  
.menu {  
  margin-left: 80%;  
  border: 1px solid #000000;  
  padding: 1em;  
}
```

Duis autem vel eum
iriure dolor in hendrerit
in vulputate velit esse
molestie consequat,
vel illum dolore eu
feugiat nulla facilisis at
vero eros et accumsan
et iusto odio dignissim
qui blandit praesent
luptatum zzril delenit
augue dui dolore te
feugait nulla facilisi.
Investigationes
demonstraverunt
lectores legere me lius
quod ii legunt saepius.

lien 1
lien 2
lien 3



Réaliser la page suivante :

[Accueil](#)

[Clients](#)

[Fournisseurs](#)

[Contact](#)



Le paragraphe 3 teste l'utilisation de la propriété float avec plusieurs autres paragraphes Test Test Test



Le paragraphe 2 teste l'utilisation de la propriété float avec plusieurs autres paragraphes Test Test Test



Le paragraphe 1 teste l'utilisation de la propriété float avec plusieurs autres paragraphes Test Test Test



La propriété CSS *overflow* définit comment un bloc doit se comporter lorsqu'un contenu dépasse de ce bloc.

Les valeurs possibles sont de cette propriété sont :

- *visible* : Le contenu qui dépasse reste visible),
- *hidden* : Le contenu qui dépasse est masqué),
- *scroll* : On affiche systématiquement des barres de défilement pour le bloc
- *auto* : On affiche des barres de défilement si un contenu dépasse.

Cette propriété a aussi pour caractéristique, lorsqu'on utilise les valeurs autres que visible, de créer ce qu'on appelle un contexte de formatage de bloc. Et une des propriétés des contextes de formatage de bloc, c'est d'empêcher le dépassement des flottants.

Voici comment on peut utiliser *overflow* pour éviter que les éléments flottants:

```
#conteneur {  
  overflow: hidden; /* Empêche le dépassement des flottants. */  
}
```



La propriété CSS *clear* permet d'indiquer qu'un élément doit venir se placer en dessous des éléments flottants qui le précèdent.

Les valeurs possibles sont :

- *none* (par défaut),
- *left* (l'élément passe en dessous des éléments flottant à gauche)
- *right* (l'élément passe en dessous des éléments flottant à droite)
- *both* (il passe en dessous de tout les flottants).

Pour bloquer le dépassement des flottants, on peut placer un élément vide juste avant la balise fermante du conteneur, et on lui applique un `clear:both`;

Attention, *clear* n'affecte que les éléments en *display:block*;

En reprenant notre exemple, nous devons modifier le code HTML ainsi: