

Introduction à css

François Médard, Jean-Marie Favreau

LASMEA (UMR 6602 UBP/CNRS), LIMOS (UMR 6158 UBP/CNRS)

1^{er} avril 2008

MCours.com

Feuilles de style en cascade

CSS pour *Cascading style sheet*

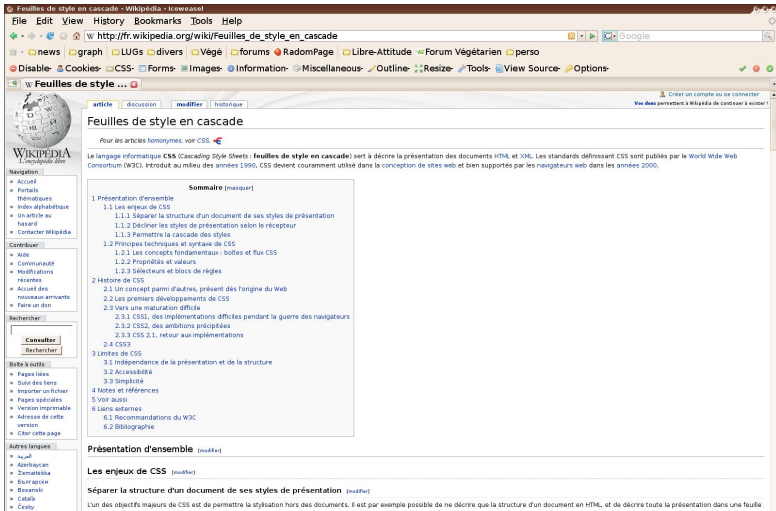
Objectifs :

- Proposer *une apparence* pour un document xhtml donné
- Permet plusieurs apparences par document
- Plusieurs documents uniformément présentés

Facilite **la maintenance** et l'**adaptabilité** de l'apparence des sites internet

Exemple : wikipédia (1)

Affichage standard de wikipédia :



Exemple : wikipédia (2)

Affichage sans css de wikipédia :

The screenshot shows a web browser window with the address bar containing 'http://fr.wikipedia.org/wiki/Feuilles_de_style_en_cascade'. The browser's menu bar includes 'File', 'Edit', 'View', 'History', 'Bookmarks', and 'Tools'. The browser toolbar shows various icons for navigation and search. The page content is displayed in a plain, unformatted font. The main heading is 'Feuilles de style en cascade', followed by a sub-heading 'Un article de Wikipédia, l'encyclopédie libre.' Below this, there is a navigation bar with links for 'Navigation', 'Rechercher', and 'Pour les articles homonymes, voir CSS'. The main text begins with 'Le langage informatique CSS (Cascading Style Sheets : feuilles de style en cascade) sert à décrire la présentation des documents HTML et XML. Les standards définissant CSS sont publiés par le World Wide Web Consortium (W3C). Introduit au milieu des années 1990, CSS devient couramment utilisé dans la conception de sites web et bien supportés par les navigateurs web dans les années 2000.' A 'Sommaire' section follows, listing various sub-topics such as 'Présentation d'ensemble', 'Histoire de CSS', 'Limites de CSS', and 'Liens externes'.

Feuilles de style en cascade - Wikipédia - Iceweasel

File Edit View History Bookmarks Tools Help

http://fr.wikipedia.org/wiki/Feuilles_de_style_en_cascade

news graph LUUGs divers Végé forums RadomPage Libre-Attitude Forum Végétarien perso

Disable Cookies CSS- Forms- Images- Information- Miscellaneous- Outline- Resize- Tools- View Source- Options-

w Feuilles de style ...

Vos dons permettent à Wikipédia de continuer à exister !

Feuilles de style en cascade

Un article de Wikipédia, l'encyclopédie libre.

Aller à : [Navigation](#), [Rechercher](#)

Pour les articles [homonymes](#), voir [CSS](#).

Le **langage informatique CSS** (*Cascading Style Sheets* : **feuilles de style en cascade**) sert à décrire la présentation des documents [HTML](#) et [XML](#). Les standards définissant CSS sont publiés par le [World Wide Web Consortium](#) (W3C). Introduit au milieu des [années 1990](#), CSS devient couramment utilisé dans la [conception de sites web](#) et bien supportés par les [navigateurs web](#) dans les [années 2000](#).

Sommaire

[\[masquer\]](#)

- 1 **Présentation d'ensemble**
 - 1.1 Les enjeux de CSS
 - 1.1.1 Séparer la structure d'un document de ses styles de présentation
 - 1.1.2 Décliner les styles de présentation selon le récepteur
 - 1.1.3 Permettre la cascade des styles
 - 1.2 Principes techniques et syntaxe de CSS
 - 1.2.1 Les concepts fondamentaux : boîtes et flux CSS
 - 1.2.2 Propriétés et valeurs
 - 1.2.3 Selecteurs et blocs de règles
- 2 **Histoire de CSS**
 - 2.1 Un concept parmi d'autres, présent dès l'origine du Web
 - 2.2 Les premiers développements de CSS
 - 2.3 Vers une maturation difficile
 - 2.3.1 CSS1, des implémentations difficiles pendant la guerre des navigateurs
 - 2.3.2 CSS2, des ambitions précipitées
 - 2.3.3 CSS 2.1, retour aux implémentations
 - 2.4 CSS3
- 3 **Limites de CSS**
 - 3.1 Indépendance de la présentation et de la structure
 - 3.2 Accessibilité
 - 3.3 Simplicité
- 4 **Notes et références**
- 5 **Voir aussi**
- 6 **Liens externes**
 - 6.1 Recommandations du W3C
 - 6.2 Bibliographie

Exemple : wikipédia (2)

Affichage avec css alternative de wikipédia :

Feuilles de style en cascade - Wikipédia - Iceweasel

File Edit View History Bookmarks Tools Help

http://fr.wikipedia.org/wiki/Cascading_Style_Sheet

news graph LUUGs divers Végé forums RadomPage Libre-Attitude Forum Végétarien perso

Disable Cookies CSS- Forms- Images- Information- Miscellaneous- Outline- Resize- Tools- View Source- Options-

Feuilles de style ...

Feuilles de style en cascade

Navigation

Accueil

Portails thématiques

Index alphabétique

Un article au hasard

Contactez Wikipédia

Contribuer

Aide

Communauté

Modifications récentes

Accueil des nouveaux arrivants

Faire un don

Rechercher

Consulter

Rechercher

Boîte à outils

Pages liées

Suivre des liens

Importer un fichier

Pages spéciales

Versions antérieures

Adresse de cette version

Autres langues

suomi

Azerbaycan

Zemateleka

Български

Бразилски

Català

Česky

Cymraeg

Dansk

Deutsch

Ελληνικά

English

Española

Español

(Redigé depuis [Cascading Style Sheet](#))

Pour les articles [homonymes](#), voir [CSS](#).

Le langage informatique **CSS** (**Cascading Style Sheets** : **feuilles de style en cascade**) sert à décrire la présentation des documents **HTML** et **XML**. Les standards définissant CSS sont publiés par le **World Wide Web Consortium** (**W3C**). Introduit au milieu des **années 1990**, CSS devient couramment utilisé dans la **conception de sites web** et bien supportés par les **navigateurs web** dans les **années 2000**.

Sommaire (masquer)

1 Présentation d'ensemble

1.1 Les enjeux de CSS

1.1.1 Séparer la structure d'un document de ses styles de présentation

1.1.2 Définir les styles de présentation selon le récepteur

1.1.3 Permettre la cascade des styles

1.2 Principes techniques et syntaxe de CSS

1.2.1 Les concepts fondamentaux : boîtes et flux CSS

1.2.2 Propriétés et valeurs

1.2.3 Sélecteurs et blocs de règles

2 Histoire de CSS

2.1 Un concept parmi d'autres, présent dès l'origine du Web

2.2 Les premiers développements de CSS

2.3 Vers une maturation difficile

2.3.1 CSS1, des implémentations difficiles pendant la guerre des navigateurs

2.3.2 CSS2, des ambitions précipitées

2.3.3 CSS 2.1, retour aux implémentations

2.4 CSS3

3 Limites de CSS

3.1 Indépendance de la présentation et de la structure

3.2 Accessibilité

3.3 Simplicité

4 Notes et références

5 Voir aussi

6 Liens externes

6.1 Recommandations du W3C

6.2 Bibliographie

Présentation d'ensemble modifier

Les enjeux de CSS modifier

- 1 Introduction
- 2 Fonctionnement de css
 - Description générale
 - Syntaxe
 - Fonctionnement général
- 3 Les propriétés disponibles
 - Valeurs de mesure et de couleur
 - Principales propriétés
- 4 Structuration de la page
 - Structure classique
- 5 Conclusion

- 1 Introduction
- 2 Fonctionnement de css
 - Description générale
 - Syntaxe
 - Fonctionnement général
- 3 Les propriétés disponibles
 - Valeurs de mesure et de couleur
 - Principales propriétés
- 4 Structuration de la page
 - Structure classique
- 5 Conclusion

xhtml et css

Une *feuille de style css* :

- **Contient** une description de l'apparence
- Est **associé à un document xhtml** par une balise dans <head>

Dans le xhtml :

```
<link rel="stylesheet" href="style.css" type="text/css" media="screen" />
```

MCours.com

- 1 Introduction
- 2 **Fonctionnement de css**
 - Description générale
 - **Syntaxe**
 - Fonctionnement général
- 3 Les propriétés disponibles
 - Valeurs de mesure et de couleur
 - Principales propriétés
- 4 Structuration de la page
 - Structure classique
- 5 Conclusion

La syntaxe

Document constitué d'une suite de **règles**, chacune composée d'un **sélecteur**, et de **propriétés** auxquelles on associe des **valeurs**.

Dans le xhtml :

```
selecteur {  
    propriete: valeur, valeur;  
    propriete: valeur, valeur;  
    propriete: valeur, valeur;  
}
```

La syntaxe (exemple)

- **Sélecteur** : décrit les balises affectées
- **Propriété** : caractéristique de l'apparence
- **Valeur** : apparence choisie

Structure :

```
h1 {  
  color: red;  
}
```

Mon titre

Première partie

Fusce vulputate lacus at *ipsum*. Nam condimentum. Nulla

Suite

Du contenu...

Curabitur adipiscing, auris non dictum aliquam.

La syntaxe (exemple 2)

- **Sélecteur** : décrit les balises affectées
- **Propriété** : caractéristique de l'apparence
- **Valeur** : apparence choisie

Structure :

```
h1, h2 {  
  color: red;  
}
```

Mon titre

Première partie

Fusce vulputate lacus at *ipsum*. Nam condimentum. N

Suite

Du contenu...

Curabitur adipiscing, auris non dictum aliquam.

- 1 Introduction
- 2 **Fonctionnement de css**
 - Description générale
 - Syntaxe
 - **Fonctionnement général**
- 3 Les propriétés disponibles
 - Valeurs de mesure et de couleur
 - Principales propriétés
- 4 Structuration de la page
 - Structure classique
- 5 Conclusion

L'imbrication

Possibilité de sélectionner une balise en **fonction de son imbrication**

- Sélecteur décrivant une imbrication
- Sélectionne uniquement les balises ayant les ancêtres correspondants

Exemple : tous les `` contenu dans des `<h1>` en vert

L'imbrication (exemple)

xhtml :

```
<body>
  <h1>Mon <em>titre</em></h1>
  <h2>Première partie</h2>
  <p><strong>Fusce</strong>
vulputate lacus at
<em>ipsum</em>.Nam
condimentum.Nulla ut
mauris.</p>
  <h2>Suite</h2>
  <h3>Du contenu...</h3>
  <p><em>C</em>urabitur
adipiscing, auris
non dictum aliquam.
  </p>
</body>
```

CSS :

```
h1 em {
  color: green;
}
```

Mon *titre*

Première partie

Fusce vulputate lacus at *ipsum*.Nam condimentum.Nu

Suite

Du contenu...

Curabitur adipiscing, auris non dictum aliquam.

Attributs class et id (description)

Par défaut, **toutes les balises d'un type** sont affectées.

Pour une sélection plus fine :

- Utilisation de **classes** (peut correspondre à **plusieurs balises**)
- Utilisation d'**identifiants uniques**

Attributs class et id (utilisation)

S'appliquent à **n'importe quelle balise** Les classes :

- Dans le xhtml, **attribut** `class="..."`
- **Plusieurs classes** possibles par balise :
`class="premiere deuxieme"`
- **Plusieurs balises** peuvent avoir la même classe
- Dans le css, préfixée par un **point** : `.premiere { ... }`

Les identifiants :

- Dans le xhtml, **attribut** `id="..."`
- **Une seule balise** par identifiant
- **Un seul identifiant** par balise
- Dans le css, préfixée par un **#** : `#premiere { ... }`

Attributs class et id (exemple)

xhtml :

```
<body>
  <h1>Mon titre</h1>
  <h2 id="premier">Première partie</h2>
  <p><strong class="rouge">Fusce</strong>
vulputate lacus at
<em>ipsum</em>.Nam
condimentum.Nulla ut
mauris.</p>
<h2>Suite</h2>
<h3 class="rouge">Du contenu...</h3>
<p><em>C</em>urabitur
adipiscing, auris
non dictum aliquam.
</p>
</body>
```

CSS :

```
.rouge {
  color: red;
}
#premier {
  color: blue;
}
```

Mon titre

Première partie

Fusce vulputate lacus at ipsum.Nam condimentum.Nulla

Suite

Du contenu...

Curabitur adipiscing, auris non dictum aliquam.

Priorité des règles

Il arrive que plusieurs sélecteurs correspondent à une balise.

- Les règles sont **appliquées en cascade** (depuis la première jusqu'à la dernière)
- Avec priorité à **la plus spécifique**

Priorité des règles (exemple 1)

xhtml :

```
<body>
  <h1>Mon <em>titre</em></h1>
  <h2>Première partie</h2>
  <p><strong>Fusce</strong>
vulputate lacus at
<em class="ici">ipsum</em>.Nam
condimentum.Nulla ut
mauris.</p>
  <h2>Suite</h2>
  <h3>Du contenu...</h3>
  <p><em>C</em>urabitur
adipiscing, auris
non dictum aliquam.
  </p>
</body>
```

CSS :

```
em.ici {
  font-size: 200%;
}

em {
  color: green;
}
```

Mon **titre**

Première partie

Fusce vulputate lacus at **ipsum**.Nam condi

Suite

Du contenu...

Curabitur adipiscing, auris non dictum aliquam.

Priorité des règles (exemple 2)

xhtml :

```
<body>
  <h1>Mon <em>titre</em></h1>
  <h2>Première partie</h2>
  <p><strong>Fusce</strong>
vulputate lacus at
<em>ipsum</em>.Nam
condimentum.Nulla ut
mauris.</p>
  <h2>Suite</h2>
  <h3>Du contenu...</h3>
  <p><em>C</em>urabitur
adipiscing, auris
non dictum aliquam.
  </p>
</body>
```

CSS :

```
h1 em {
  color: green;
}
em {
  color: blue;
  font-size: 300%;
}
```

Mon **titre**

Première partie

Fusce vulputate lacus at *ipsum*.Nam condimentum.Nulla ut

Suite

Du contenu...

Curabitur adipiscing, auris non dictum aliquam.

Balises universelles

Les balises universelles

- Par défaut, elles sont **silencieuses**
- Permettent de **définir une apparence spécifique** à une partie du document
- Sont de deux types :
 - *inline* : `` (dans le fil du texte)
 - *block* : `<div>` (boîtes englobantes)

Balises universelles (exemple)

xhtml :

```
<body>
  <h1>Mon <em>titre</em></h1>
  <h2>Première partie</h2>
  <p><strong>Fusce</strong>
vulputate lacus at
<span class="rouge">ipsum</span>.Nam
condimentum.Nulla ut
mauris.</p>
  <h2>Suite</h2>
  <h3>Du contenu...</h3>
  <p><em>C</em>urabitur
adipiscing, auris
non dictum aliquam.
  </p>
</body>
```

CSS :

```
.rouge {
  color: red;
}
```

Mon *titre*

Première partie

Fusce vulputate lacus at **ipsum**.Nam cond

Suite

Du contenu...

Curabitur adipiscing, auris non dictum aliq

- 1 Introduction
- 2 Fonctionnement de css
 - Description générale
 - Syntaxe
 - Fonctionnement général
- 3 Les propriétés disponibles**
 - **Valeurs de mesure et de couleur**
 - Principales propriétés
- 4 Structuration de la page
 - Structure classique
- 5 Conclusion

Unités de mesure

Nécessité de définir des tailles :

- Taille des caractères
- Taille des boîtes
- Taille des bords, etc

Plusieurs unités :

- % : relatif à la taille courante
- em : largeur d'un *m* minuscule
- px : taille d'un pixel
- pt : taille d'un point (dépend de la configuration *dpi*)

Couleurs

Pour définir les couleurs :

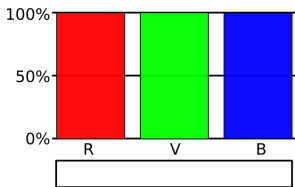
- Quelques noms en anglais (red, blue, green, etc)
- Codage *Rouge*, *Vert*, *Bleu* en hexadécimal :
 - **Petite précision** : chaque composante codée entre 0 et F
 - **Grande précision** : chaque composante codée entre 00 et FF

Rappel : l'hexadécimal signifie base 16 :

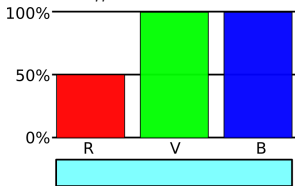
- 0, ... 9, A, B, ..., F
- 10, ... 19, 1A, 1B, ..., 1F
- 20, ... 29, 2A, 2B, ..., 2F
- ...
- F0, ... F9, FA, FB, ..., FF

Valeurs de mesure et de couleur

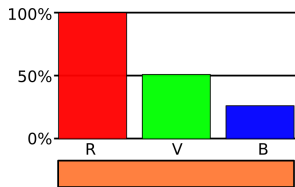
Couleurs (exemples)



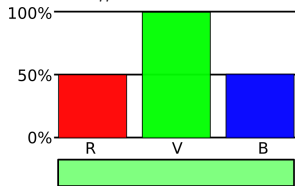
#FFFFFF



#7FFFFFFF



#FF7F3F



#8F8

Principales propriétés

- 1 Introduction
- 2 Fonctionnement de css
 - Description générale
 - Syntaxe
 - Fonctionnement général
- 3 Les propriétés disponibles**
 - Valeurs de mesure et de couleur
 - Principales propriétés**
- 4 Structuration de la page
 - Structure classique
- 5 Conclusion

Mise en forme du texte

Pour modifier le texte :

- La couleur : propriété `color`:
- La décoration : propriété `text-decoration`: (`none`, `underline`, `overline`)
- L'ajustement : propriété `text-align`: (`left`, `right`, `center`, `justify`)
- Le style : propriété `font-style`: (`normal`, `italic`)
- La graisse : propriété `font-weight`: (`normal`, `bold`)
- La taille : propriété `font-size`:
- La police : propriété `font-family`: (`sans-serif`, `serif`, `"New Roman"`, etc)

Mise en forme du texte (exemple)

xhtml :

```
<body>
  <p><span class="t1">Essai</span>
  de texte
  <span class="t2">avec du style</span>
</p>
</body>
```

CSS :

```
.t1 {
  color: green;
  font-weight: bold;
  font-size: 110%;
}

.t2 {
  text-decoration: underline;
  font-style: italic;
  color: red;
}
```

Essai de texte avec du style

Fond

Fond d'une balise : préfixe de propriété background

- La couleur : propriété background-color :
- L'image : propriété background-image: url(...) avec adresse de l'image
- La position de l'image : propriété background-position: (top, bottom, left, right, middle, center, ou *mesure*)
- La répétition : background-repeat: (no-repeat, repeat-x, repeat-y, repeat)

On peut tout décrire en une propriété : background:

Fond (exemple)

xhtml :

```
<body>
  <p>Fusce
    vulputate lacus at
    ipsum.Nam
    condimentum.Nulla ut
    mauris. Curabitur
    adipiscing, auris
    non dictum aliquam.</p>
</body>
```



CSS :

```
body {
  background: url("ubp.jpg") repeat;
}
```

Fusce vulputate lacus at ipsum.Nam c

Bords

Bord d'une balise : ligne sur chaque côté

Préfixe :

- Préfixe de propriété `border`
- Préfixes de propriété `border-left`, `border-right`, `border-top`, `border-bottom`

Suffixe :

- Suffixe de propriété `color`
- Suffixe de propriété `style` (`solid`, `dashed`, `dotted`, `none`, etc)
- Suffixe de propriété `width` (en unité de mesure)

Exemples :

- `border-bottom: 1px solid red;`
- `border-top-style: dashed;`

Bords (exemple)

xhtml :

```
<body>
  <div>
    <h1>Titre</h1>
    <p>Ici est le texte.</p>
  </div>
</body>
```

CSS :

```
div {
  border: 1px solid black;
}
h1 {
  border-bottom: 1px dashed #ccc;
}
```



Titre

Ici est le texte.

Taille de l'élément

Seul la **taille des éléments block** peut être modifiée (<div>, , <table>, etc.)

- Propriété **width**: pour décrire la **largeur**
- Propriété **height**: pour décrire la **hauteur**
- Différentes valeurs possibles (**taille** en unité de mesure, en **pourcentage** du conteneur, ou **auto**)

Taille de l'élément (exemple)

xhtml :

```
<body>
  <div id="menu">
    <ul>
      <li>Menu 1</li>
      <li>Menu 2</li>
      <li>Menu 3</li>
    </ul>
  </div>
  <div id="contenu">
    <p>Fusce
    vulputate lacus at
    ipsum.Nam
    condimentum.Nulla ut
    mauris. Curabitur
    adipiscing, auris
    non dictum aliquam.</p>
```

CSS :

```
div {
  border: 1px solid gray;
}
#menu {
  width: 200px;
  background-color: yellow;
}
```

- Menu 1
- Menu 2
- Menu 3

Fusce vulputate lacus at ipsum.Nam condimentum.Nulla ut mauris. Curabitur adipiscing, auris non dictum aliquam.

Positionnement

Positionnement, par propriété `position`:

- Position statique (`static`) : par défaut, dans le fil du contenu
- Position fixe (`fixed`) : position fixe dans la fenêtre

Positionnement à l'intérieur du conteneur, par propriété `float`:

- Position à droite ou à gauche (`left`, `right`)
- Remis à zéro par propriété `clear`: (`left`, `right`)

Positionnement (exemple)

xhtml :

```

<body>
  <div id="menu">
    <ul>
      <li>Menu 1</li>
      <li>Menu 2</li>
      <li>Menu 3</li>
    </ul>
  </div>
  <div id="contenu">
    <p>Fusce
    vulputate lacus at
    ipsum.Nam
    condimentum.Nulla ut
    mauris. Curabitur
    adipiscing, auris
    non dictum aliquam.</p>
  </div>

```

CSS :

```

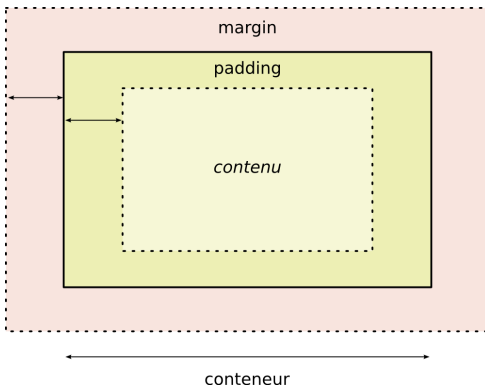
div {
  border: 1px solid gray;
}
#menu {
  width: 200px;
  float: left;
  position: fixed;
  background-color: yellow;
}

```

- Menu 1
- Menu 2
- Menu 3

Marges (1)

Il existe deux types de bord : `margin` et `padding` :



Le vilain petit carnard : Internet explorer `margin` contenus dans la taille...

Marges (2)

Marges :

- Propriété `margin`: (**taille** en unité de mesure, en **pourcentage** du conteneur, ou `auto`)
- Ou propriétés `margin-left`:, `margin-top`:, etc.
- Possibilité de réunir tout sur une ligne :
 - haut/bas puis gauche/droite. Exemple : `margin: 5px 20px`;
 - haut, droite, bas, gauche. Exemple : `margin: 5px 20px 2px 50px`;

Identique avec les **marges de remplissage**, avec le préfixe `padding`

Marges (exemple)

xhtml :

```
<body>
  <div id="menu">
    <ul>
      <li>Menu 1</li>
      <li>Menu 2</li>
      <li>Menu 3</li>
    </ul>
  </div>
  <div id="contenu">
    <p>Fusce
    vulputate lacus at
    ipsum.Nam
    condimentum.Nulla ut
    mauris. Curabitur
    adipiscing, auris
    non dictum aliquam.</p>
```

CSS :

```
...
#contenu {
  margin-left: 220px;
}
```

- Menu 1
- Menu 2
- Menu 3

Fusce
vulputate
lacus at
ipsum.Nam
condimentum.
ut mauris.
Curabitur
adipiscing,
auris non
dictum
aliquam.

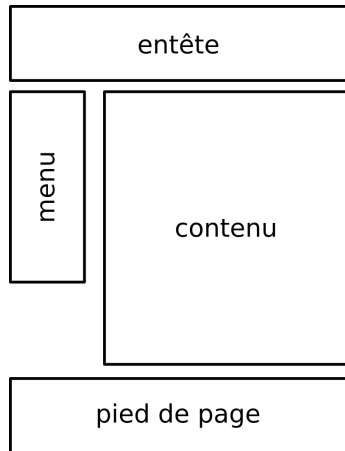
Structure classique

- 1 Introduction
- 2 Fonctionnement de css
 - Description générale
 - Syntaxe
 - Fonctionnement général
- 3 Les propriétés disponibles
 - Valeurs de mesure et de couleur
 - Principales propriétés
- 4 Structuration de la page
 - Structure classique
- 5 Conclusion

Structure classique d'une page

La structure classique d'une classe, décrite par `<div>` :

- Une entête (accès rapide, nom du site, logo, etc.)
- Un menu (navigation à l'intérieur du site)
- Du contenu
- Un pied de page (auteur, date, copyright, etc.)



Largeur fixe

Largeur définie en pixels :

- On choisi la **résolution minimale supportée** (par exemple 800x600)
- On **retranche 20 pixels** à la largeur (pour l'ascenseur)
- On créé un premier `<div>` de cette largeur
- Tout le site inclu dedans



Structure classique

Largeur fixe (exemple)

CSS :

```
body {
    text-align: center;
}
div {
    border: 1px solid gray;
}
#main {
    width: 780px;
    margin: auto;
    text-align: justify;
}
#head {
    padding: 5px;
}
#menu {
    width: 200px;
    float: left;
    background-color: yellow;
}
#contenu {
    margin-left: 220px;
    padding: 5px;
}
#footer {
    clear: both;
}
```



Largeur dynamique

Largeur définie dynamiquement :

- On souhaite adapter la largeur au document
- On ne définit **aucune largeur en pixels** (sauf exception)

Structure classique

Largeur dynamique (exemple)

CSS :

```
body {
    text-align: center;
}
div {
    border: 1px solid gray;
}
#main {
    margin: 0 30px;
    text-align: justify;
}
#head {
    padding: 5px;
}
#menu {
    width: 200px;
    float: left;
    background-color: yellow;
}
#contenu {
    margin-left: 220px;
    padding: 5px;
}
#footer {
    clear: both;
}
```



Exemple de décoration



xhtml :

```
<body>
  <div id="d1">
    <div id="d2">
      <h1>Titre</h1>
    </div>
  </div>
</body>
```

CSS :

```
body {
  background: #f5f4ef url("haut.png") repeat-x;
  padding: 0;
  margin: 0;
}

#d1 {
  background: url("haut-gauche.jpg") no-repeat top left;
  padding: 0;
  margin: 0;
  height: 130px;
}

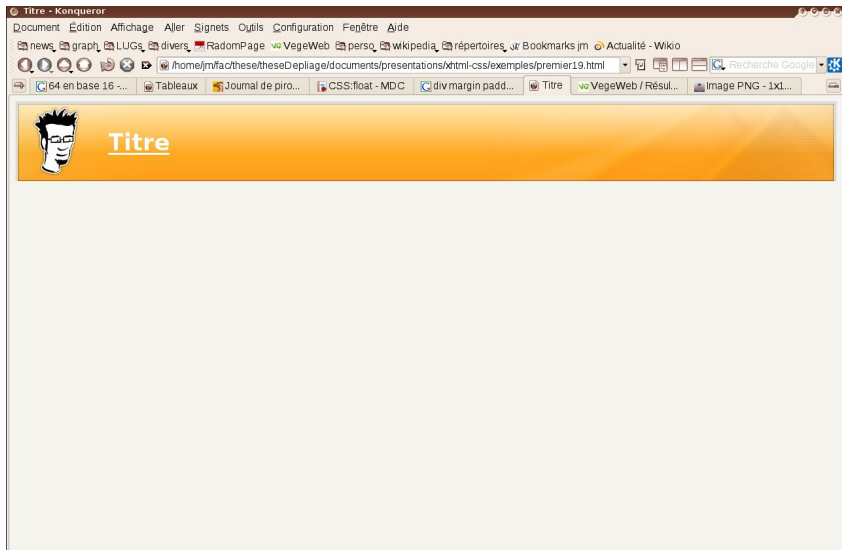
#d2 {
  background: url("haut-droite.jpg") no-repeat top right;
  padding: 0;
  margin: 0;
  height: 130px;
}

h1 {
  padding: 50px 150px;
  margin: 0;
  color: #fff;
  text-decoration: underline;
}
```



Structure classique

Exemple de décoration (résultat)



Conclusion

- Langage css **très complet**
- Allié au xhtml : **possibilités infinies**
- Mais quelques **limitations dues aux navigateurs**
- **Rapide à découvrir**, long à maîtriser

Bibliographie

- <http://www.csszengarden.com/tr/francais/>
- <http://openweb.eu.org/css/>
- <http://developer.mozilla.org/fr/docs/CSS>

MCours.com