



LE LANGAGE CSS

Module : Technologie Web

Définition et avantages



Cascading Style Sheets (**CSS**) : Feuille de style en cascade

- Langage de feuilles de style pour documents HTML.
- Syntaxe différente du HTML.
- Permet une séparation claire de la structure et du style.
- Définition de styles génériques pour les balises.

Définition et avantages



- Faciliter la modification de la présentation ou du contenu (possibilité de travailler en équipe plus efficacement).
- Coût de développement et de maintenance allégé(code plus simple).
- Code source allégé.
- Possibilité de présentations différentes selon : le profil utilisateur, la résolution d'écran, etc.
- Uniformiser la présentation sur plusieurs pages (un seul fichier CSS à modifier pour que les changements se répercutent sur l'ensemble du site).

Définition et avantages



Le code CSS peut être écrit dans une balise:

```
<p style="text-align: right; color: black">
```

Les feuilles de style permettent de gérer la mise en page des divers éléments d'une page web </p>

Pour un paragraphe :

```
<p style="text-align: right; color: black" > Les feuilles  
de style permettent de gérer la mise en page des divers  
éléments d'une page Web </p>
```

```
<p style="text-align: right; color: black" > Bien sur,  
une page Web peut être composée de nombreux paragraphes,  
comme n'importe quel texte.</p>
```

```
<p style="text-align: right; color: black" > Mais alors,  
il devient fastidieux de devoir saisir à nouveau toutes  
les informations de style ! </p>
```

Définition et avantages



❑ Mais si le site possède plusieurs pages ?

➔ créer une feuille de style externe qui sera liée à chacune des pages Web

```
<HTML>
<HEAD>
  <Meta charset="utf-8" />
  <TITLE>Le CSS</TITLE>
```

```
<link rel="stylesheet" type="text/css" href="fichier.css">
```

```
</HEAD>
<BODY>
  <H1>Introduction</H1>
  ...
</BODY>
</H1>
```

// Code CSS

Fichier.CSS

Définition et avantages



- Dans l'en-tête `<head>` du fichier HTML

```
<html>
  <head>
    <meta charset="utf-8" />
    <style>
      p
      {
        color: blue;
      }
    </style>
    <title>Premiers tests du
    CSS</title>
  </head>
  <body>
    <h1>Mon super site</h1>
    <p>Bonjour et bienvenue sur
    mon site !</p>
    <p>Pour le moment, mon site
    est un peu <em>vide</em>.
    Patientez encore un peu
    !</p>
  </body>
</html>
```

- Directement dans les balises (non recommandé)

```
<html>
  <head>
    <meta charset="utf-8" />
    <title>Premiers tests du
    CSS</title>
  </head>
  <body>
    <h1>Mon super site</h1>
    <p style="color: blue;">Bonjour
    et bienvenue sur mon site
    !</p>
    <p>Pour le moment, mon site
    est un peu <em>vide</em>.
    Patientez encore un peu
    !</p>
  </body>
</html>
```

Définition et avantages

Une règle CSS est caractérisée par deux principaux éléments :

- Un sélecteur de **balises**, permettant de préciser à quelles balises du document le style s'applique
- Une **déclaration** de style, définie entre accolades, permettant de préciser le style à appliquer aux balises sélectionnées. La déclaration est elle-même constituée :
 - d'une ou plusieurs **propriété(s)**, suivie du caractère « : » (double point),
 - d'une ou plusieurs valeur(s) associée(s) à chaque propriété, entourée de guillemets s'il s'agit de plusieurs mots ou séparés par des virgules s'il y en a plusieurs, suivie d'un point virgule

balise { propriété_de_style: valeur; propriété_de_style: valeur }

Exemple : *H1 { font-family: Arial; font-style: italic }*

Définition et avantages

Notion d'héritage :

- Si on traduit les trois mots de « Cascading Style Sheets » on obtient Feuille de style en cascade.
- Exemple:

si vous définissez une police de type "arial" sur la balise <body>, l'ensemble des autres éléments du site prendra comme police "arial", inutile de le redéfinir à chaque éléments. Sauf, si une autre police de caractère est déclarée explicitement.

Notion de classe

- Un sélecteur contextuel, ou classe, associe une règle particulière à un élément en fonction de sa situation.
- On appelle sélecteur simple les balises HTML classiques auxquelles on a attribué des caractéristiques de style.

```
.classe1 {  
    color: green;  
    font-size: 14px;  
    font-weight: bolder;  
}
```

```
.classe2 {  
    color: red;  
    font-size: 18px;  
}
```

```
<p class="classe1">  
un premier paragraphe  
</p>
```

un premier paragraphe

```
<p class="classe2">  
un deuxieme paragraphe  
</p>
```

un deuxieme paragraphe

Notion des ID

- La déclaration d'un ID en CSS est identique à la notion de class à une exception près : On ne peut déclarer une seule fois un ID

```
#once {  
  margin: 20px;  
  font-size: 11px;  
  color: blue;  
}
```

```
<p id="once">  
  est correct  
</p>
```

Mais si on rencontre dans la même page

```
<p id="once">  
  ce n'est plus correct !  
</p>
```

- Remarque : On utilise un **#** pour les id et un **."** pour les classes

Combinaison de selecteurs

- Il est possible de combiner les éléments HTML qui regroupent les mêmes caractéristiques

```
img, table {border:none;}
```

CSS : Exemple 1

- A la balise <H1> on pourra lui rattacher la police Verdana, italique, de taille 16 pixels, de couleur Rouge. Chaque fois qu'on utilisera la balise <H1> dans un document, tout le texte situé dans les limites de la balise recevra cette mise en forme.

HTML	CSS
<pre><HTML> <BODY> <H1 style=" font-size: 16px; font-family: Verdana; font-style: italic; color:black;"> Titre de niveau 1</H1> </BODY> </HTML></pre>	<pre>h1 { font-size : 16px; font-family : Verdana; font-style: italic; color : black; }</pre>

CSS : Exemple 2

```
<style type="text/css">

body{
    margin: 0px;
    background-color:Ivory ;
    width: 100%;
    height: 100%;
    font-size:12px;
    font-family:Verdana, Geneva, sans-serif;
    font-weight:normal;
}

a{
    color: #000000;
    text-decoration:none;
}

p {
    font-size: 110%;
    font-family: Helvetica, sans-serif;
}

img {border:none;}

</style>
```


Les commentaires

- Il est possible de mettre des commentaires presque n'importe où dans une feuille de style CSS en les encadrant de `/*` et `*/`, comme dans le langage C.

```
/* style de la balise A (lient hypertexte) */  
a{  
    color: #000000;  
    text-decoration:none;  
}  
  
/* style de la balise P (paragraphe) */  
p {  
    font-size: 12px;  
    font-family: verdana, geneva, sans-serif;  
}  
  
... ..
```

Quelques propriétés

- Mise en forme d'un texte en CSS

color: Gestion de la couleur du texte
letter-spacing: Espace entre les lettres
text-align: Alignement d'un texte
text-decoration: Habilement d'un texte

text-indent: Espace avant un texte
text-transform: Transformation du texte
white-space: Controle des espaces blancs
word-spacing: Espace entre les mots

- Gestion des différentes polices

font-family: Famille de police
font-size: Taille de votre police de caratère
font-style: Gras, italique, oblique...
font-variant: Variante de police
font-weight: Taille de la police

- Les bordures en CSS

border-color: Couleur de la bordure
border-style: Apparence de la bordure
border-width: Taille de la bordure

Avantages (1/2)

- La structure du document et la présentation sont gérés dans des fichiers séparés.
- La conception d'un document se fait dans un premier temps sans se soucier de la présentation, ce qui permet d'être plus efficace.
- Dans le cas d'un site Internet, la présentation est uniformisée : Les documents (pages « html ») font référence à la (aux) même(s) feuille(s) de styles. Cette caractéristique permet de plus un relookage rapide.

Avantages (2/2)

- Un même document peut donner le choix entre plusieurs feuilles de style (impression ou lecture à l'écran par exemple). Certains navigateurs web permettent d'accéder facilement à un choix de feuilles de style.
- Le code HTML est considérablement réduit en taille et en complexité, puisqu'il ne contient plus de balises de présentation.
- Réduire le temps de chargement des pages.

Les sélecteurs (1/4)



1. Sélecteurs standards:

- *Sélecteur universel* : tous les éléments HTML

Exemple : `*{color :blue ; }`

- *Style d'une catégorie de balises:*

Exemple : `h1 { color :#FF00FF ; }`

- *Style de plusieurs catégories de balises*

Exemple : `h1, h2, h3, p { color : green ; }`

- *Style pour balises imbriquées*

Exemple : `p em {color :red ; }`

Les sélecteurs (2/4)



Fichier.HTML

```
<HTML>
  <HEAD>
    <TITLE>Le CSS</TITLE>
```

```
<link rel="stylesheet" type= "text/css" href= "fichier.css">
```

```
</HEAD>
  <BODY>

    <H1>Introduction</H1>
      ...
    </BODY>
</HTML>
```

Fichier.CSS

```
H1 {
  color : red ;
}
```

Introduction

...

Les sélecteurs (3/4)



2. Sélecteur de classes:

❑ *Sélecteur défini par l'utilisateur:* permet de dissocier un style de balises particulières

Exemple :

- `style_rouge { color:red; }`
- `style_bleu { color:blue; }`
- `p.paprag_vert { color:green; } /* applicable seulement aux paragraphes */`

➤ Utilisation dans une page Web

Il faut faire un appel explicite à la classe de style désirée.

Exemple :

`<h1 class="style_rouge">Titre de niveau 1</h1>`

Les sélecteurs(4/4)



3. Sélecteur d'identificateurs

- ❖ Un sélecteur de classe commence par un "." alors que le sélecteur d'identificateur commence par un "#".
- ❖ Le style de type classe peut être utilisé plusieurs fois dans un document html alors que le style de type identificateur ne peut être utilisé qu'une seule fois dans le document html.
- ❖ Le style de type classe s'utilise dans n'importe quelle balise avec l'attribut "class" alors que le style de type identificateur s'utilise avec l'attribut "id " .

Pseudo-format



Le CSS nous permet aussi de modifier l'apparence des éléments de façon dynamique, c'est-à-dire que des éléments peuvent changer de forme une fois que la page a été chargée

- au survol: **a:hover{ }**
- lors du clic : **a:active { }** /* Quand le visiteur clique sur le lien */
- lors du focus : **a:focus { }** /* Quand le visiteur sélectionne le lien */
- lorsqu'un lien a été consulté :
a:visited{ } /* Quand le visiteur a déjà consulté la page */

Syntaxe

Selecteur : pseudoformat

```
{ prop1: val1;  
prop2: val2;  
..}
```

Les styles de mise en forme du texte (1/6)



❑ font-family : définit <nom> ou <famille> police précise (Arial, Times, Helvetica...)

⇒ **H3 {font-family: Arial}**

❑ font-style : définit le style de l'écriture normal ou italique ou oblique

⇒ **H3 {font-style: italic}**

❑ font-weight : définit l'épaisseur de la police normal ou bold ou bolder ou lighter ou valeur numérique soit (100 | 200 | 300 | 400 | 500 | 600 | 700 | 800 | 900)

⇒ **P {font-weight: bold}**

Les styles de mise en forme du texte (2/6)



❑ **font-size** : définit la taille de la police en points (pt), inches (in), centimètres (cm), pixels (px) ou pourcentage (%)

⇒ *P {font-size: 12pt}*

❑ **font-variant** : définit une variante par rapport à la normale normal ou small-caps

⇒ *P {font-variant: small-caps}*

Les styles de mise en forme du texte (3/6)



□ **text-align** : définit l'alignement du texte left ou center ou right

⇒ *H1 {text-align: center}*

□ **text-indent** : définit un retrait dans la première ligne d'un bloc de texte souvent utilisé avec <P>spécifié en inches (in) ou en centimètres (cm) ou en pixels (px)

⇒ *P {text-indent: 1cm}*

□ **text-decoration** : définit une décoration du texte, soit barré, clignotant, etc. blink ou underline ou line-through ou overline ou none

⇒ *A:visited {text-decoration: blink }*

Les styles de mise en forme du texte (4/6)



❑ **text-transform** : définit la casse du texte (majuscule, minuscule)

- ❑ uppercase (met les caractères en majuscules) ou
- ❑ lowercase (met les caractères en minuscules) ou
- ❑ capitalize (met le premier caractère en majuscule)

⇒ *P {text-transform: uppercase}*

❑ **Color** : définit la couleur du texte par exemple en hexadécimal

⇒ *H3 {color: #000080}*

❑ **word-spacing** : définit l'espace entre les mots en points (pt), inches (in), centimètres (cm), pixels (px) ou pourcentage (%)

⇒ *P {word-spacing: 5pt}*

Les styles de mise en forme du texte (5/6)



❑ **letter-spacing** : définit l'espace entre les lettres spécifié en points (pt), inches (in), centimètres (cm), pixels (px) ou pourcentage (%)

$\Rightarrow P \{letter-spacing: 2pt\}$

❑ **line-height** : définit l'interligne soit l'espace entre les lignes du texte en points (pt), inches (in), centimètres (cm), pixels (px) ou pourcentage (%)

$\Rightarrow P \{line-height: 10pt\}$

Les styles de mise en forme du texte (6/6)



❑ **Width** : détermine la longueur d'un élément de texte ou d'une image en points (pt), inches (in), centimètres (cm), pixels (px) ou pourcentage (%)

⇒ ***H1 {width: 200px}***

❑ **height** : détermine la hauteur d'un élément de texte ou d'une image en points (pt), inches (in), centimètres (cm), pixels (px) ou pourcentage (%)

⇒ ***H1 {height: 100px}***

❑ **white-space** : espace ou blanc normal ou pre ou nowrap PRE

{white-space: pre}

Les couleurs (1/2)

□ Color

la propriété `color` permet de fixer la couleur du texte.

Comment manipuler les couleurs ?

- liste de couleur standard
- Un code hexadécimal long `#00FF45`
- Un code hexadécimal court `#0F3`
- Un code rgb, de 0 à 255 `rgb(0,0,100)`
- un code rgb, en pourcentage `rgb(10%,10%,30%)`

Liste des couleurs standard :

**aqua, black, blue, fuchsia, gray, green, lime, maroon, navy, olive,
purple, red, silver, teal, white, yellow**

Les couleurs (2/2)



❑ background-color

la propriété background-color permet de fixer la couleur de fond de la page.

```
body
{
background-color : rgb(0,0,0) ;
color : white
}
```

FLOAT – CLEAR (1/2)



- ❑ **Float** : Une boîte flottante est **retirée du flux normal** placée le plus à droite (float: right) ou le plus à gauche (float: left) possible dans son conteneur.

Exemple : Boîte flottante simple

Code HTML:

```
<p class="jaune">Une boîte jaune flottant</p> <p  
class="verte">Une boîte verte doté d'un contenu plus long...</p>
```

Code CSS:

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

FLOAT – CLEAR (1/2)



- ❑ **Clear:** définit si un élément peut être à côté des éléments flottants qui le précèdent ou s'il doit être placé en dessous d'eux (se libérer).

Les valeurs

Left: L'élément est déplacé vers le bas pour se libérer des éléments flottants à gauche.

Right : L'élément est déplacé vers le bas pour se libérer des éléments flottants à droite.

Both : L'élément est déplacé en dessous des éléments flottants de gauche et de droite

Modèles d'affichage en C.S.S (1/3)



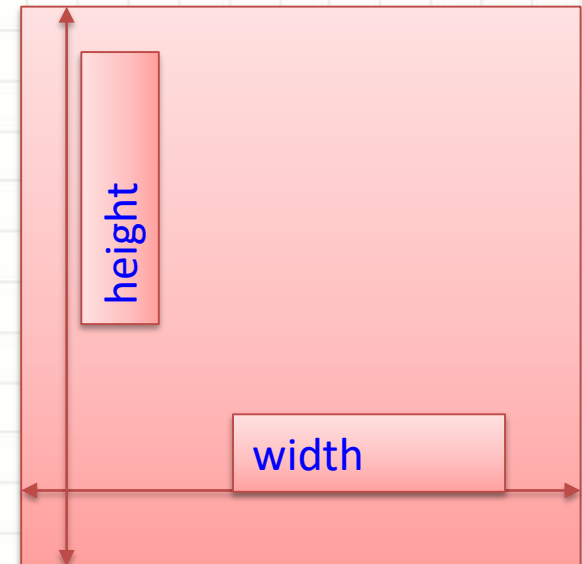
- **Les éléments « bloc »** : une balise de type block sur votre page web crée automatiquement un retour à la ligne avant et après
 - Les éléments blocs s'empilent les uns sur les autres
 - Un élément bloc peut contenir d'autres éléments blocs
 - Un élément bloc peut contenir des éléments en ligne*exemples* : p, h1, h2, h3, h4, h5, h6, div

- **Les éléments « en ligne »** : une balise de type inline se trouve obligatoirement à l'intérieur d'une balise block. Une balise inline ne crée pas de retour à la ligne
 - Les éléments en ligne se placent les uns à côté des autres
 - Un élément en ligne peut contenir d'autres éléments en ligne*exemples* : em, strong, b, i, a, span

Modèles d'affichage en C.S.S (2/3)



- *width*
 - Permet de spécifier la largeur d'un bloc
- *height*
 - Permet de spécifier la hauteur d'un bloc
- *position*
 - **Absolute** Le bloc est placé de façon absolue
 - **Relative**: Le bloc est placé relativement à la position qu'il aurait du occuper



Modèles d'affichage en C.S.S (3/3)



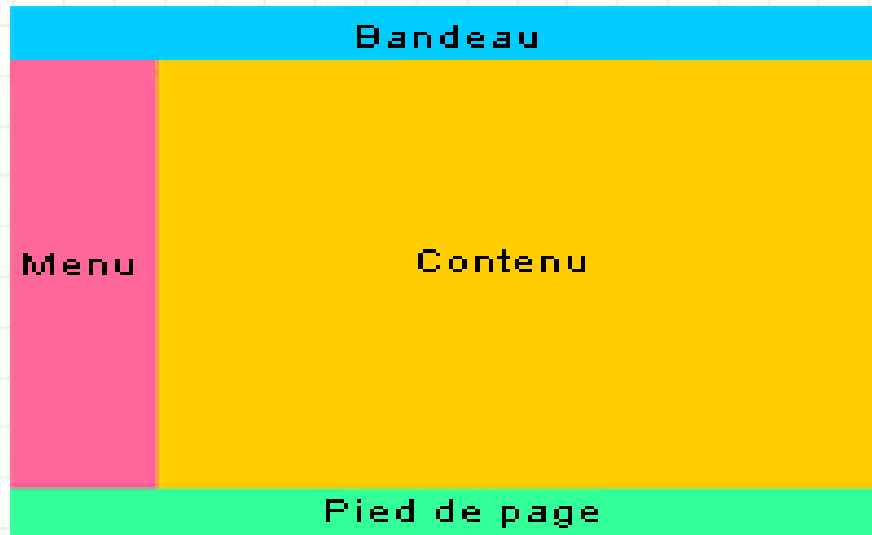
- Containers génériques.

`<div>` permet de structurer le document

`` permet de changer la mise en page de quelques mots dans un texte

`<div>` et `` s'utilisent toujours avec les classes et les ID.

Exemple 1



Exemple d'utilisation de DIV



```
<html>
<head>
<meta http-equiv="Content-Type"
content="text/html; charset=iso-8859-1">
<link rel="stylesheet" type="text/css"
href="styles.css">
<title>Positionnement CSS</title>
</head>
<body>
<div id="bandeau">
Ceci est le bandeau</div>
<div id="menu">
Ceci est le menu</div>
<div id="contenu">
Ceci est le contenu</div>
<div id="pied_page">Ceci est le pied de
page</div>
</body>
</html>
```

```
div {
text-align:center;
}
div#bandeau {
width:600px;
height:50px;

background-color:#00CCFF;
}
div#menu {
float:left;
width:100px;
height:400px;

background-color:#FF6699;
}
div#contenu {
float:left;
width:500px;
height:400px;

background-color:#FFCC00;
}
div#pied_page {
clear:both;
width:600px;
height:50px;

background-color:#33FF99;
}
```

Exemple d'utilisation de DIV

```
<body>
<div id="header">
  <h1>Header Text</h1>
</div>
<div id="navcol">
  <h4>Nav Column</h4>

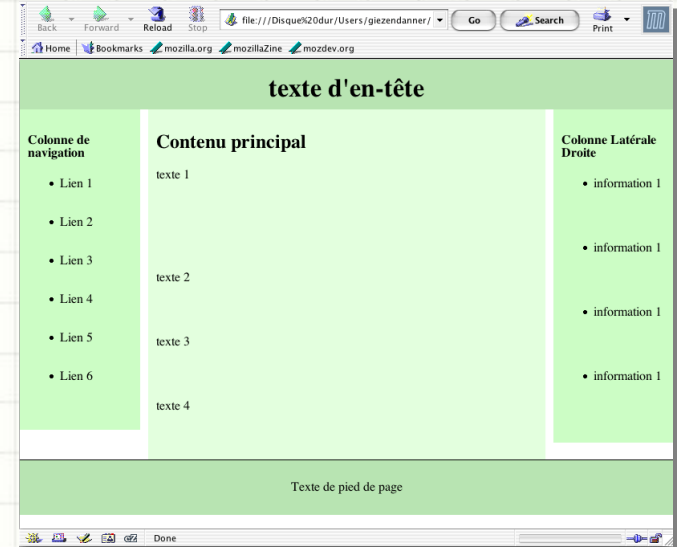
  <ul>
    <li>Let me not
    <li>Admit imped
    <li>Which alter
    <li>Or bends wi
    <li>Oh, no, it

    <li>Let me not
  </ul>
</div>
<div id="sidecol">
  <h4>Starboard Side Colu
  <ul>
    <li>Let me not
    <li>Admit imped

    <li>Which alter
    <li>Or bends wi
  </ul>
</div>
<div id="main">
  <h2>Main Content</h2>
  <p> That looks on tempe

  <p>That looks on tempes
  <p>If this be error and
  <p>That looks on tempes
</div>
<div id="foot">
  <p>Footer text goes her
</div>
</body>
```

Exemple 2



```

1 body {
2     margin: 0px;
3     padding: 0px;
4 }
5 div#header {
6     text-align: center;
7     background-color: #B8E4B2;
8     height: 60px;
9     margin: 0px;
10    padding: 1px;
11 }
12 div#navcol {
13     padding: 10px;
14     width: 130px;
15     float: left;
16     background-color: #CCFDC5;
17 }
18 div#main {
19     padding: 10px;
20     margin-left: 160px;
21     margin-right: 160px;
22     background-color: #E3FFDE;
23 }
24 div#sidecol {
25     padding: 10px;
26     width: 130px;
27     float: right;
28     background-color: #CCFDC5;
29 }
30 div#foot {
31     border-top: solid #000 1px;
32     background-color: #B8E4B2;
33     padding: 10px;
34     text-align: center;
35     clear: both;
36 }

```

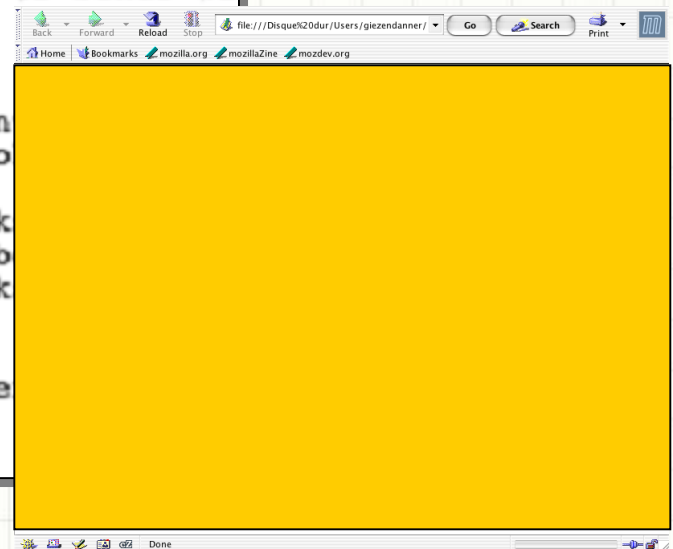
```

<body>
<div id="header">
    <h1>Header Text</h1>
</div>
<div id="navcol">
    <h4>Nav Column</h4>

    <ul>
        <li>Let me not
        <li>Admit imped
        <li>Which alter
        <li>Or bends wi
        <li>Oh, no, it
        <li>Let me not
    </ul>
</div>
<div id="sidecol">
    <h4>Starboard Side Colu
    <ul>
        <li>Let me not
        <li>Admit imped
        <li>Which alter
        <li>Or bends wi
    </ul>
</div>
<div id="main">
    <h2>Main Con
    <p> That loo

    <p>That look
    <p>If this b
    <p>That look
</div>
<div id="foot">
    <p>Footer te
</div>
</body>

```



```
1 body {  
2     margin: 0px;  
3     padding: 0px;  
4 }
```

```
5 div#header {  
6     text-align: center;  
7     background-color: #B8E4B2;  
8     height: 60px;  
9     margin: 0px;  
10    padding: 1px;  
11 }
```

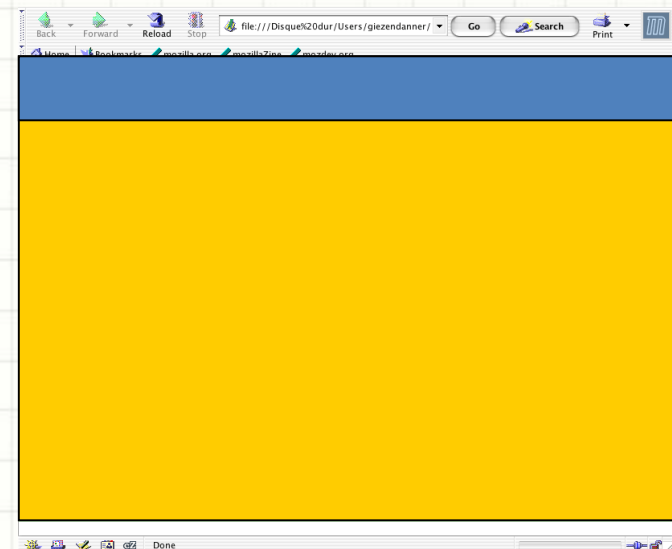
```
<div id="header">  
    <h1>Header Text</h1>  
</div>
```

```
12 div#navcol {  
13     padding: 10px;  
14     width: 130px;  
15     float: left;  
16     background-color: #CCFDC5;  
17 }
```

```
18 div#main {  
19     padding: 10px;  
20     margin-left: 160px;  
21     margin-right: 160px;  
22     background-color: #E3FFDE;  
23 }
```

```
24 div#sidecol {  
25     padding: 10px;  
26     width: 130px;  
27     float: right;  
28     background-color: #CCFDC5;  
29 }
```

```
30 div#foot {  
31     border-top: solid #000 1px;  
32     background-color: #B8E4B2;  
33     padding: 10px;  
34     text-align: center;  
35     clear: both;  
36 }
```



```

1  body {
2      margin: 0px;
3      padding: 0px;
4  }

```

```

5  div#header {
6      text-align: center;
7      background-color: #B8E4B2;
8      height: 60px;
9      margin: 0px;
10     padding: 1px;
11 }

```

```

12 div#navcol {
13     padding: 10px;
14     width: 130px;
15     float: left;
16     background-color: #CCFDC5;
17 }

```

```

18 div#main {
19     padding: 10px;
20     margin-left: 160px;
21     margin-right: 160px;
22     background-color: #E3FFDE;
23 }

```

```

24 div#sidecol {
25     padding: 10px;
26     width: 130px;
27     float: right;
28     background-color: #CCFDC5;
29 }

```

```

30 div#foot {
31     border-top: solid #000 1px;
32     background-color: #B8E4B2;
33     padding: 10px;
34     text-align: center;
35     clear: both;
36 }

```

```

<div id="navcol">
    <h4>Nav Column</h4>

    <ul>

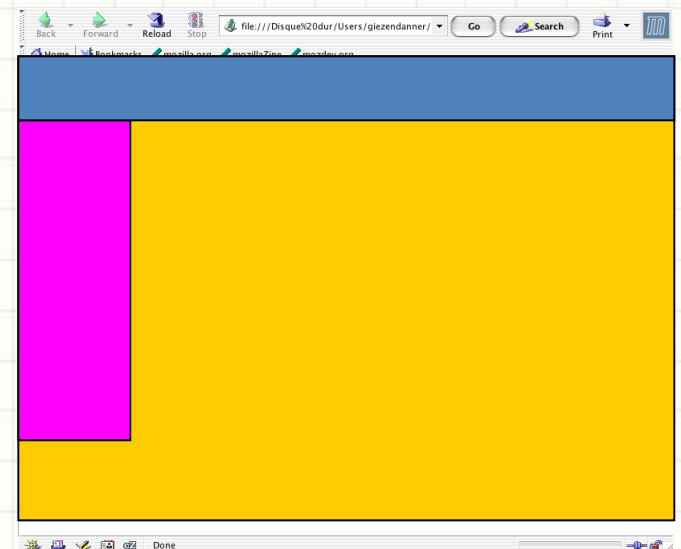
        <li>Let me not
        <li>Admit imped
        <li>Which alter
        <li>Or bends w
        <li>Oh, no, it

        <li>Let me not

    </ul>

</div>

```




```

1  body {
2      margin: 0px;
3      padding: 0px;
4  }

```

```

5  div#header {
6      text-align: center;
7      background-color: #B8E4B2;
8      height: 60px;
9      margin: 0px;
10     padding: 1px;
11 }

```

```

12 div#navcol {
13     padding: 10px;
14     width: 130px;
15     float: left;
16     background-color: #CCFDC5;
17 }

```

```

18 div#main {
19     padding: 10px;
20     margin-left: 160px;
21     margin-right: 160px;
22     background-color: #E3FFDE;
23 }

```

```

24 div#sidecol {
25     padding: 10px;
26     width: 130px;
27     float: right;
28     background-color: #CCFDC5;
29 }

```

```

30 div#foot {
31     border-top: solid #000 1px;
32     background-color: #B8E4B2;
33     padding: 10px;
34     text-align: center;
35     clear: both;
36 }

```

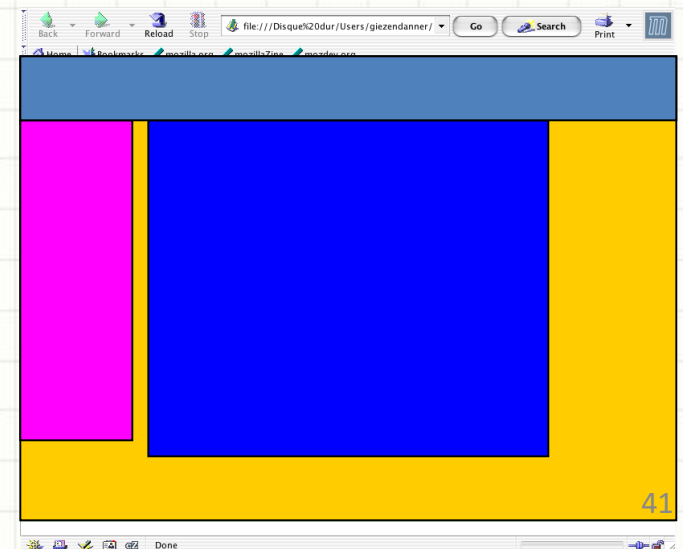
```

<div id="main">
    <h2>Main Content</h2>
    <p> That looks on tempests and is never sha

    <p>That looks on tempests and is never sha
    <p>If this be error and upon me proved, wi
    <p>That looks on tempests and is never sha

</div>

```



```

1  body {
2      margin: 0px;
3      padding: 0px;
4  }

```

```

5  div#header {
6      text-align: center;
7      background-color: #B8E4B2;
8      height: 60px;
9      margin: 0px;
10     padding: 1px;
11 }

```

```

12 div#navcol {
13     padding: 10px;
14     width: 130px;
15     float: left;
16     background-color: #CCFDC5;
17 }

```

```

18 div#main {
19     padding: 10px;
20     margin-left: 160px;
21     margin-right: 160px;
22     background-color: #E3FFDE;
23 }

```

```

24 div#sidecol {
25     padding: 10px;
26     width: 130px;
27     float: right;
28     background-color: #CCFDC5;
29 }

```

```

30 div#foot {
31     border-top: solid #000 1px;
32     background-color: #B8E4B2;
33     padding: 10px;
34     text-align: center;
35     clear: both;
36 }

```

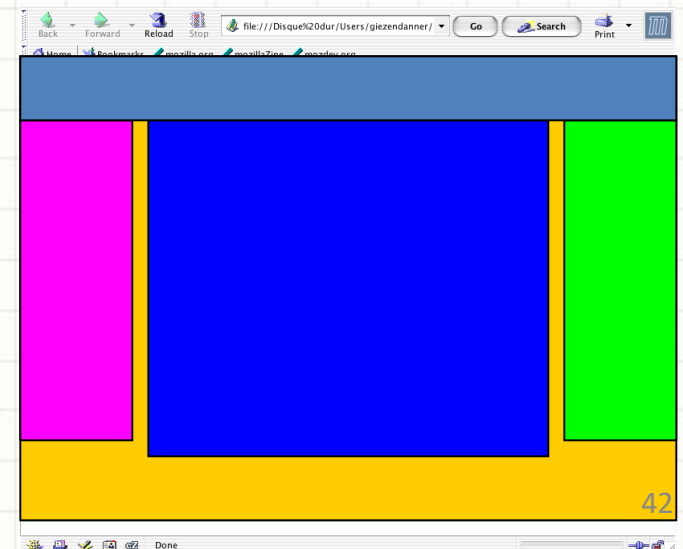
```

<div id="sidecol">
    <h4>Starboard Side Column</h4>
    <ul>
        <li>Let me not to the ma
        <li>Admit impediments; l

        <li>Which alters when it
        <li>Or bends with the re

    </ul>
</div>

```



```

1  body {
2      margin: 0px;
3      padding: 0px;
4  }

```

```

5  div#header {
6      text-align: center;
7      background-color: #B8E4B2;
8      height: 60px;
9      margin: 0px;
10     padding: 1px;
11 }

```

```

12 div#navcol {
13     padding: 10px;
14     width: 130px;
15     float: left;
16     background-color: #CCFDC5;
17 }

```

```

18 div#main {
19     padding: 10px;
20     margin-left: 160px;
21     margin-right: 160px;
22     background-color: #E3FFDE;
23 }

```

```

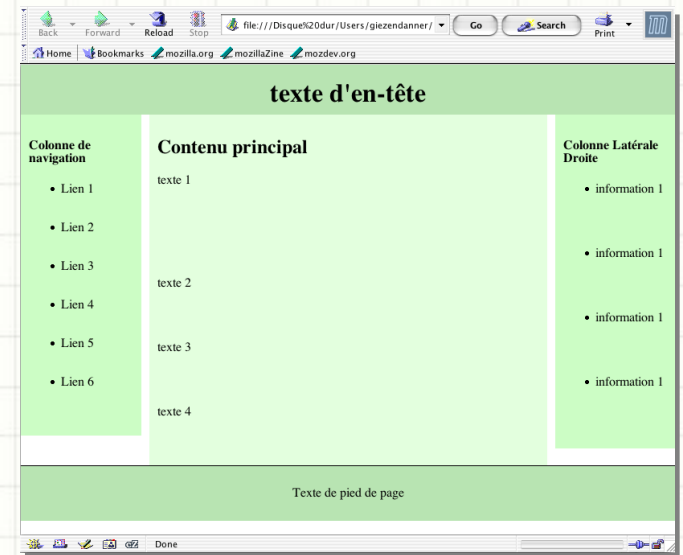
24 div#sidecol {
25     padding: 10px;
26     width: 130px;
27     float: right;
28     background-color: #CCFDC5;
29 }

```

```

30 div#foot {
31     border-top: solid #000 1px;
32     background-color: #B8E4B2;
33     padding: 10px;
34     text-align: center;
35     clear: both;
36 }

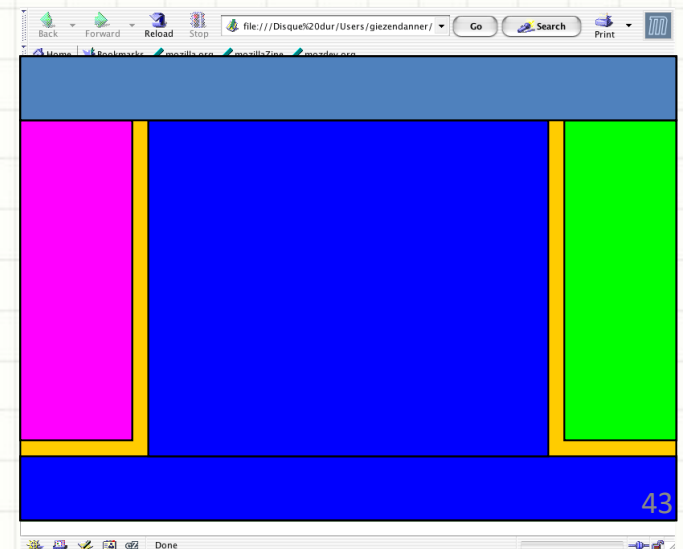
```



```

<div id="foot">
    <p>Footer text goes here.
</div>

```



Les Bordures



❑ Couleur de bordure : propriété **border-color**

➤ Possibilité d'attribuer différentes couleurs pour chaque bordure (**border-top-color, border-bottom-color, border-right-color, border-left-color**)

➤ Exemples : `p{border-top-color: green;};` `img{bordercolor:#00FF00;}`

❑ Définition globale de bordure : propriété **border**

➤ Une écriture simplifiée des différentes propriétés de bordure

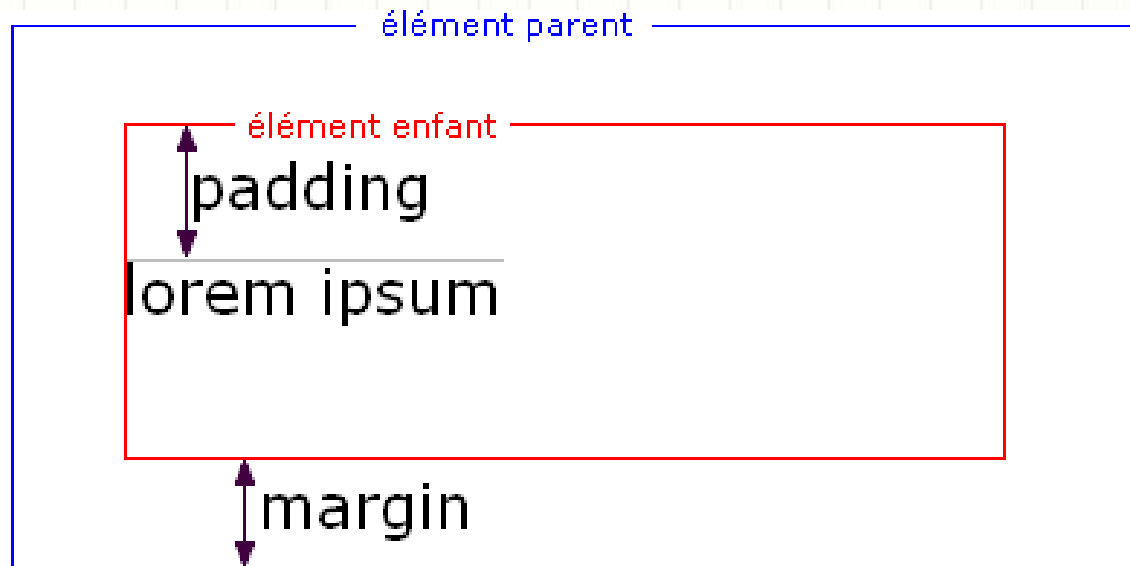
➤ Exemple : `h1{border: 5px double blue;}`

➤ Une écriture simplifiée pour chaque arête de bordure

(**border-top, border-bottom, border-right, border-left**)

➤ Exemple : `p{border-bottom: #00FFFF;}`

Les Marges - Padding





Les Marges

- Les marges peuvent être négatives ; dans ce cas la boîte d'un élément sort de celle de son parent
- Si la largeur de la marge est donnée en pourcentage, elle est calculée par rapport à celle du bloc parent.
- Possibilité de définir de une à quatre marges dans le sens des aiguilles d'une montre (haut, droit, bas, gauche).
- Pour réaliser un design centré quand on ne connaît pas la résolution du visiteur, on utilise **margin : auto**

- Exemples :

```
h1 { margin: 20px 10px 15px 10px; }
```

```
p { margin: 10%; }
```

```
b { margin-left: 3em; }
```

```
.block_centre { margin: auto; width:200px; }
```


- }

La création de menu horizontal



```
<ul>
<li><a href="p1.html">lien1</a>
</li>
<li><a
href="p2.html">lien2</a></li>
<li><a
href="p3.html">lien3</a></li>
</ul>
```

```
ul li{
list-style: none;
float: left;
padding:5px; }
```

```
ul li a{
text-decoration: none;
color: red;
background-color: blue; }
```

```
ul li a:hover{
color: green;
font-weight: bold;
background-color: orange; }
```



MERCI POUR VOTRE ATTENTION