# Formation HTML5, CSS3 JavaScript

## Introduction

### Dans cette partie :

- HTML, XHTML et CSS
- Les navigateurs Internet
- Les éditeurs XHTML/CSS

### HTML, XHTML et CSS

HTML est le langage de base permettant de définir des pages Web. Il est composé de balises (aussi appelées marqueurs ou tags) qui décrivent et mettent en forme des contenus.

XHTML est un langage assez proche d'HTML, à quelques nuances près : toutes les balises doivent avoir un parent, et le parent de tous les parents est <a href="https://example.com/structure-number 2">httml></a>. Voici la structure minimale d'une page XHTML :

```
<html>
    <head>
        ...
      </head>
      <body>
        ...
      </body>
</html>
```

### Quelques restrictions:

- pas de majuscules dans les noms et attributs des balises ;
- les balises doivent être correctement imbriquées. A titre d'exemple, seule la seconde ligne de code est correcte en XHTML :

```
<b><i>Ce texte est gras et italique</b></i><b><i>Ce texte est gras et italique</i></b>
```

Le langage HTML ou XHTML est utilisé pour définir le contenu des pages Web.

Le langage CSS est utilisé pour définir la mise en forme et la mise en page des contenus HTML. Il peut être utilisé directement dans les balises HTML/XHTML, dans l'en-tête des pages ou dans un fichier externe appelé "feuille de styles".



#### Les moteurs de rendu



Outre leur interface, les navigateurs se distinguent par leur moteur de rendu. C'est lui qui détermine comment sont interprétées puis affichées les instructions HTML, CSS et JavaScript. Pour en savoir plus à ce sujet, consultez les pages Wikipedia relatives aux navigateurs Web classés par moteur de rendu. Ces pages sont accessibles ici :

http://fr.wikipedia.org/wiki/Cat%C3%A9gorie:Navigateur web
par moteur de rendu (ou http://goo.gl/octwhx):

Ainsi, vous pouvez voir que les trois navigateurs vedettes utilisent trois moteurs de rendu différents :

Navigateur	Moteur de rendu
Mozilla Firefox	Gecko
Google Chrome et Safari	Webkit
Internet Explorer	Trident

Consultez la page <a href="http://en.wikipedia.org/wiki/Comparison of layout engines (ECMAScript">http://en.wikipedia.org/wiki/Comparison of layout engines (ECMAScript)</a> (soit <a href="http://goo.gl/CQ6SMi">http://goo.gl/CQ6SMi</a> en raccourci) pour prendre connaissance du niveau de compatibilité des moteurs de rendu avec la norme ECMAScript :



### Editeurs HTML/XHTML/CSS

La saisie du code HTML peut se faire dans un simple éditeur de texte, comme le Bloc-Notes de Windows. Mais vous lui préfèrerez un éditeur de code à coloration syntaxique.

Plusieurs éditeurs de ce type sont diffusés sous la forme de freewares. Vous utiliserez par exemple NotePad ++ ou Sublime Text 2.

### Histoire du HTML

HTML (*HyperText Markup Language*) est le langage utilisé pour représenter les pages Web. Il permet également de structurer sémantiquement et de mettre en forme le contenu des pages, d'inclure des ressources multimédia et des formulaires. Il est souvent utilisé conjointement aux langage CSS (*Cascading Style Sheet*) et JavaScript.

Le premier document portant l'extension **html** a être publié à l'attention du public l'a été en août 1991 par Tim Berners-Lee, lorsque ce dernier a annoncé la naissance du World Wide Web.

Les différentes versions du langage :

HTML 2.0 : fin 1995 : tables, expressions mathématiques

HTML 3.2 : Janvier 1997 : styles CSS, scripts

HTML 4.0 : Décembre 1997 : séparation entre structure et présentation, variantes strict (qui exclut les éléments de présentation), transitional (qui reprend les éléments de présentation de HTML 3.2) et frameset (qui permet la création de pages par l'assemblage de frames) du langage

HTML 4.01: Décembre 1999: format de transition XHTML 1.0

HTML 5.0 : Octobre 2014 : accent sur le côté sémantique du langage

### Les organismes de normalisation W3C et WHATWG

Le World Wide Web Consortium (W<sub>3</sub>C) est un organisme de normalisation à but non lucratif, fondé en Octobre 1994. Son but : promouvoir la compatibilité des technologies du Web telles que HTML<sub>5</sub>, HTML, XHTML, XML, RDF, SPARQL, CSS, XSL, PNG, SVG et SOAP.

Site officiel du W<sub>3</sub>C : <a href="http://www.w3.org/">http://www.w3.org/</a>

### Les organismes de normalisation W3C et WHATWG

En 2004, des fabricants de navigateurs Web créent le WHATWG (Web Hypertext Application Technology Working Group). Leur but : relancer le développement du format HTML et répondre aux besoins des utilisateurs. Ce groupe est né suite aux contestations quant au mode de fonctionnement du W<sub>3</sub>C, réputé trop fermé par une partie des développeurs et designers Web.

Les travaux du WHATWG ont été adoptés en mai 2007 comme point de départ d'une nouvelle spécification du langage HTML : HTML5. Les principes retenus :

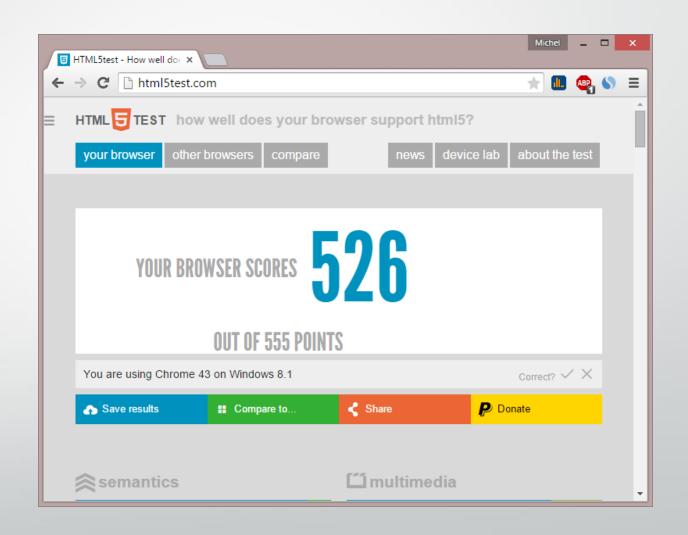
- compatibilité des futures implémentations HTML avec le contenu Web existant
- préférence des évolutions aux modifications radicales
- priorité donnée aux besoins des utilisateurs par rapport à ceux des auteurs

Site officiel du WHATWG : <a href="https://whatwg.org/">https://whatwg.org/</a>

# Compatibilité avec les navigateurs

Les nouveaux éléments propres au langage HTML5 ne sont pas encore entièrement implémentés sur les principaux navigateurs.

Pour connaître le taux de compatibilité de votre navigateur, rendez-vous à la page http://html5test.com/.



# Compatibilité avec les navigateurs

Les principaux navigateurs utilisent un préfixe pour (pré)implémenter les nouvelles propriétés CSS3 :

- <u>-moz</u> pour les navigateurs Mozilla (Firefox);
- <u>-webkit</u> pour les navigateurs Webkit (Safari, OmniWeb, Midori, etc.);
- <u>-khtml</u> pour les navigateurs Konqueror ;
- <u>-o</u> pour les navigateurs Opera;
- <u>-ms</u> pour le navigateur Internet Explorer 9 et supérieur.

Tant que la spécification du langage n'a pas atteint au moins le statut de *recommandation candidate*, vous devrez utiliser plusieurs préfixes dans les propriétés CSS3 pour assurer la plus grande compatibilité possible. À terme, ces préfixes ne devraient plus avoir cours et une seule instruction devrait être interprétée à l'identique dans tous les navigateurs.

# Différences HTML5/prédécesseurs

HTML5 est rétrocompatible

La déclaration de type de document (DTD) est simplifiée à son extrême : <!DOCTYPE html>

Le jeu de balises propre à HTML5 introduit une nouvelle logique de formulation, plus sémantique et plus intuitive : la balisage ne décrit plus le contenu du document. Il détaille sa structure et désigne le rôle de chaque section.

Enfin, les développeurs web peuvent désormais utiliser des API JavaScript gérées nativement par les navigateurs. Le but est d'obtenir des niveaux de performances comparables à ceux des applications Desktop, mais aussi de tirer parti des possibilités offertes par les périphériques mobiles, qui vont prendre une part de plus en plus importante dans le paysage Internet.

# Installation logicielle

Dans cette partie, vous allez installer :

- Notepad++ pour saisir le code
- WAMP Server pour tester le code AJAX

## Installation de NotePad++

Notepad++ est un éditeur de code à coloration syntaxique.

Téléchargez-le ici : <a href="http://notepad-plus-plus.org/fr/">http://notepad-plus-plus.org/fr/</a>
et installez-le.

# Installation de WAMP Server

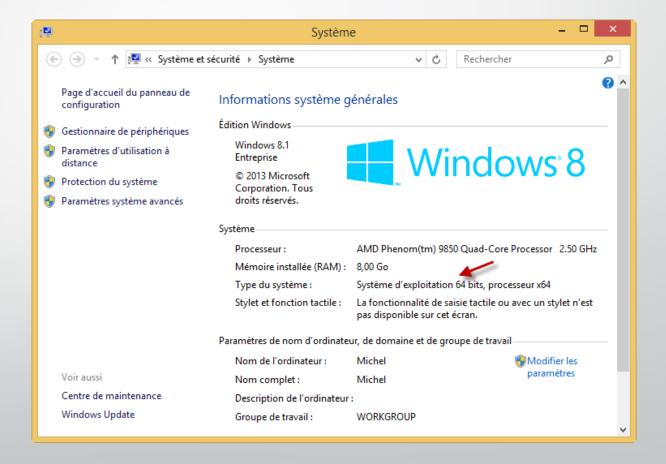
Pour tester le code serveur (en particulier le code AJAX), vous devez au préalable installer :

- 1) Un serveur Web Apache. C'est lui qui sera chargé de délivrer les pages Web aux visiteurs. Par défaut, Apache ne peut gérer que des sites statiques, constitués de pages HTML. Pour programmer en PHP, vous devez installer le support des langages PHP et MySQL dans ce serveur.
- **2)** Le plugin PHP pour Apache. Ainsi, Apache sera en mesure de traiter des pages écrites en PHP.
- 3) Le logiciel de gestion de bases de données MySQL. Vous pourrez ainsi créer des bases de données et les interroger pour créer vos pages Web dynamiquement.

Plusieurs paquetages incluant Apache, PHP et MySQL sont disponibles. Par exemple :

- WAMP Server sous Windows;
- MAMP sous Mac OS X;
- XAMPP sous Linux.

Avant d'installer un paquetage, vérifiez le type de votre système d'exploitation : **32** ou **64 bits**. Pour cela, appuyez simultanément sur les touches *Windows* et *Pause*. La fenêtre **Système** s'affiche. L'information apparaît en face du libellé **Taille du système** :

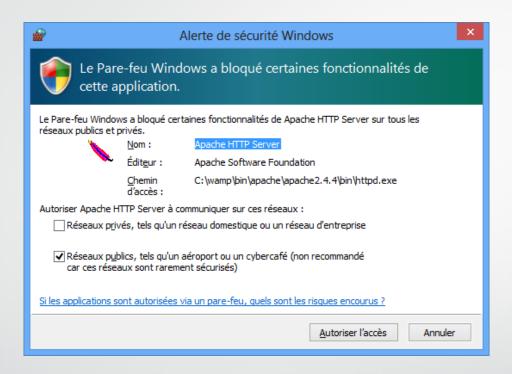


Rendez-vous sur la page <a href="http://sourceforge.net/projects/wampserver/">http://sourceforge.net/projects/wampserver/</a> et installez le logiciel WAMP Server, en version 32 ou 64 bits selon le type de votre système d'exploitation. Pour cela, sélectionnez l'onglet Files, cliquez sur Wamp Server 2, sur la dernière version du logiciel (2.5 alors que j'écris ces lignes), puis sur la version 32 ou 64 bits en fonction de votre système.

A titre d'information, la version 32 bits est moins "lourde" que la version 64 bits. Il est donc facile de la reconnaître même si son nom n'est pas totalement affiché sur l'écran. Vous pouvez aussi pointer les liens proposés pour afficher le nom complet du fichier correspondant. Si vous voyez "32b" dans le nom du fichier, il s'agit d'une version 32 bits. Si vous voyez "64b", il s'agit d'une version 64 bits :

39.9 MB)			
Name   MampServer 2 / Wampserver 2.5	lodified \$	Down	oloads / Week
↑ Parent folder			
wampserver2.5-Apache-2.4.9-Mysql 20	014-05-01	39.9 MB	34 035
wampserver2.5-Apache-2.4.9-Mysql 20	014-05-01	43.5 MB	38 737 🕍 📵
Totals: 2 Items		83.4 MB	72 772

Installez Wamp Server en conservant les options par défaut. A la fin de l'installation, le pare-feu de Windows se manifeste :



Cliquez sur **Autoriser l'accès** pour autoriser Apache à communiquer sur votre réseau.

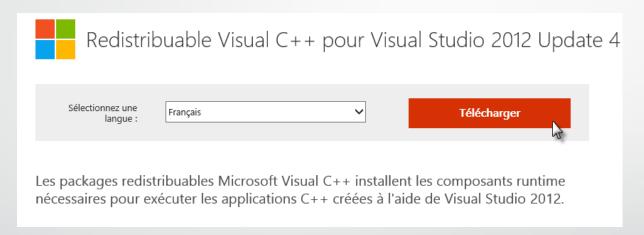
Acceptez toutes les options par défaut jusqu'à la fin de l'installation.

### **Erreur d'installation**

En cas d'erreur 0xc000007b sur la DLL msvcr100.dll lors de l'installation de WAMP Server, téléchargez et installez le logiciel Visual C++ Redistributable for Visual Studio 2012. Pour cela, rendez-vous sur la page suivante :

http://www.microsoft.com/en-us/download/details.aspx?id=30679

Choisissez Français dans la liste déroulante et cliquez sur Télécharger :



Choisissez la version **32 bits** ou **64 bits** en fonction de votre système d'exploitation et installez-la sur votre ordinateur :

Choisissez le téléchargement souhaité	
Nom du fichier	Taille
VSU4\vcredist_arm.exe	1.4 MB
VSU4\vcredist_x64.exe	6.9 MB
VSU4\vcredist_x86.exe	6.3 MB

Désinstallez WAMP Server, redémarrez l'ordinateur et réinstallez-le.

### WAMP Server s'est-il bien installé?

Une icône représentant WAMP Server devrait se trouver dans la Zone de notification. Si ce n'est pas le cas, tapez wamp dans le menu Démarrer (Windows XP/7) ou dans la page d'accueil (Windows 8) et cliquez sur **Start WampServer**. Quelques secondes plus tard, une icône représentant Wamp Server est disponible dans la Zone de notification :



Si l'icône de WAMP Server reste orange dans la zone de notification, il se peut que le service Apache ou MySQL ne fonctionne plus. Dans ce cas :

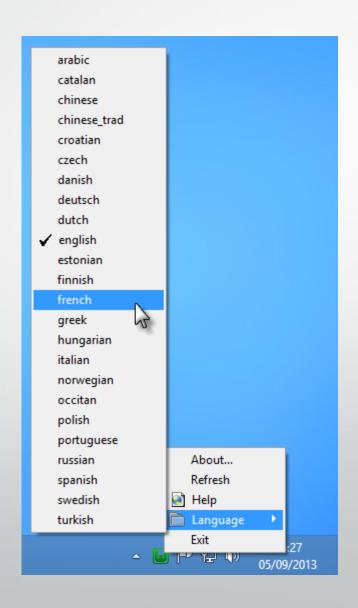
- cliquez sur l'icône de WAMP Server, puis sur Apache, Service et Installer le service.
- cliquez sur l'icône de WAMP Server, puis sur MySQL, Service et Installer le service.

Il se peut aussi que le **port 80** soit utilisé par un autre service que Apache. Cliquez sur l'icône de **WAMP Server**, puis sur **Apache**, **Service** et **Tester le port 80**.

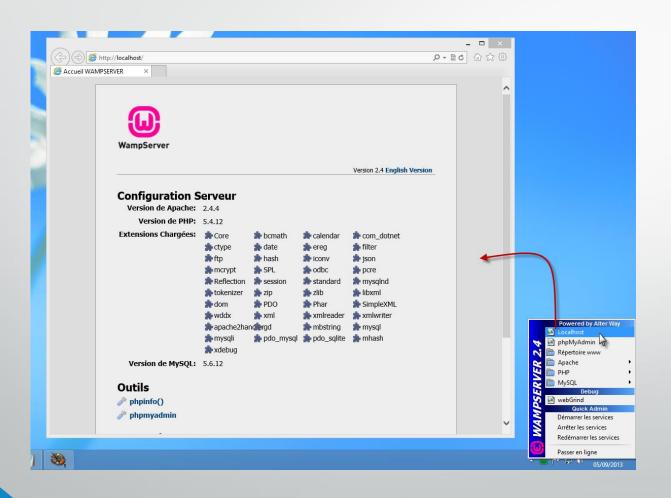
Si le port 80 n'est pas utilisé par Apach, :

- cliquez sur l'icône de **WAMP Server**, sur **Apache** puis sur **httpd.conf**.
- recherchez le port **80** dans ce fichier et remplacez-le par **81**.
- fermez puis redémarrez **Wamp Server** pour prendre en compte la nouvelle configuration.

Par défaut, WAMP s'installe en anglais. Pour lui faire parler la langue de Molière, cliquez du bouton droit sur son icône, pointez **Language** et cliquez sur **French**.



Pour savoir si WAMP s'est bien installé, cliquez sur son icône et choisissez **Localhost** dans le menu. Au bout de quelques instants, la page d'accueil de WAMP Server s'affiche dans votre navigateur par défaut. Apache est donc opérationnel :



Si cette page refuse de s'afficher, relancez Apache. Pour cela, cliquez sur l'icône de WAMP Server, puis sur Redémarrer les services.

Si cela n'a toujours aucun effet, désinstallez puis réinstallez WAMP Server.

# Les bases du HTML5/CSS3

### Dans cette partie :

- Déclaration de type de document (DTD)
- Syntaxes HTML et XHTML
- Jeux de caractères
- Définir la langue dans un document HTML5
- Structure d'un document HTML5
- Valider du code HTML5
- Styles CSS dans les balises
- Feuille de styles interne
- Feuille de styles externe
- Éléments, sélecteurs, propriétés et valeurs CSS
- Sélecteurs CSS 2.1
- Sélecteurs CSS<sub>3</sub>
- id ou class?
- Pseudo-classes et pseudo-éléments
- Unités CSS

# Déclaration de type de document

La balise <!DOCTYPE> est utile pour que les navigateurs adaptent le rendu d'un document HTML.

La balise <!DOCTYPE> des documents écrits en HTML5 est réduite à sa plus simple expression :

<!DOCTYPE html>

En HTML4.01, les trois variantes généralement utilisées étaient les suivantes :

• HTML 4.01 strict (mode rigoureux):

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd">
```

• HTML 4.01 transitionnel (langage XHTML utilisé):

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
```

• HTML 4.01 frameset (la page contient des frames) :

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Frameset//EN"
"http://www.w3.org/TR/html4/frameset.dtd">
```

## DTD

Voici le squelette d'un document écrit en HTML5:

## Jeu de caractères

Les *charsets* (jeux de caractères) sont associés à la notion de claviers nationaux. Pour indiquer aux navigateurs dans quel jeu de caractères vous travaillez, vous pouvez insérer une balise dans l'en-tête de votre document :

<meta charset="jeu-à-utiliser">

### Vous utiliserez le jeu de caractères :

- **ISO-8859-1** pour accéder directement à la majorité des caractères des langues occidentales (français, anglais, allemand, espagnol, etc.).
- **UTF-8** pour afficher sur une même page des caractères issus de plusieurs langues (français et japonais par exemple). Consultez la page **www.columbia.edu/kermit/utf8.html** pour vous rendre compte des immenses possibilités du jeu de caractères UTF-8.

## Squelette standard HTML5

```
<!DOCTYPE html>
<html lang="fr">
  <head>
    <meta ...>
    <meta ...>
    <title>...</title>
    1 ink ...>
    <style>...</style>
  </head>
  <body>
  </body>
</html>
```

Les éléments <u>meta</u>, s'ils sont présents, permettent, entre autres, d'associer un jeu de caractères, une description et un nom d'auteur au document. Par exemple :

```
<meta charset="UTF-8" />
<meta name="description" content="Description de
la page en quelques phrases." />
<meta name="author" content="Michel Martin" />
```

<u>lie le document courant à un autre document (HTML, CSS, image, XML)</u>. Par exemple, pour définir une feuille de styles externe :

```
<link rel="stylesheet" href="style.css" />
```

## Valider le code HTML/XHTML

Avant de tester un code HTML/XHTML dans votre navigateur, je vous conseille de vous assurer de sa conformité. Pour cela, rendez-vous à la page <a href="http://validator.w3.org/">http://validator.w3.org/</a>.

Pour faire apparaître du texte dans une page Web, il suffit de le taper tel qu'il doit apparaître, entre les balises <body> et </body> :

```
Texte sur la ligne 1 <br/>
Texte sur la ligne 2
```

Vous pouvez aussi définir des paragraphes avec la balise :

```
Ceci est un paragraphe
Ceci est un autre paragraphe
```

Quelle est la différence entre <br/> et ?

### Des commentaires dans le texte

Si vous concevez des pages longues et/ou complexes, vous voudrez peut-être y insérer des commentaires pour faciliter leur maintenance. Pour cela, vous utiliserez le marqueur <!>:

```
<!-- Commentaires -->
```

Le texte placé dans le marqueur n'est pas affiché dans le navigateur : il est uniquement visible dans le code source.

## Titres et soustitres

Vous pouvez affecter jusqu'à six niveaux de titre à vos documents HTML. Pour cela, vous utiliserez les marqueurs  $\leq hx >$  et  $\leq /hx >$ , où  $\underline{x}$  est un chiffre compris entre 1 et 6.

### Exemple:

Ce petit exemple montre comment utiliser les marqueurs <a href="https://www.ncmarqueurs.com/hx">hx</a>. Remarquez qu'un marqueur de fin de titre (<a href="https://www.ncmarqueurs.com/hx">hx</a>) provoque un passage à la ligne. Il est donc inutile de faire appel aux marqueurs <a href="https://www.ncmarqueurs.com/hx">https://www.ncmarqueurs.com/hx</a>) ou <a href="https://www.ncmarqueurs.com/hx">https://www.ncmarqueurs.com/hx</a>) ou <a href="https://www.ncmarqueurs.com/hx">https://www.ncmarqueurs.com/hx</a>).

```
<html>
<head>
<title>Les balises de titre</title>
</head>
<body>
<h1>titre de niveau 1</h1>
<h2>titre de niveau 2</h2>
<h3>titre de niveau 3</h3>
<h4>titre de niveau 4</h4>
<h5>titre de niveau 5</h5>
<h6>titre de niveau 6</h6>
texte normal
</body>
</head>
</html>
```

# Liens hypertexte

Une grande part de la magie inhérente au Web est liée à la présence de liens hypertexte dans les pages HTML : en cliquant sur certains mots ou sur certaines images, on peut avoir accès à d'autres parties du document en cours de consultation, à d'autres pages du même site ou d'un autre site, ou encore à d'autres services Internet (mail, ftp, gopher, telnet).

Avant de passer à la pratique, vous devez savoir que l'intérêt d'un site tient à la qualité et à la quantité de ses liens.

Au niveau de la qualité, il est nécessaire de placer les liens à des endroits logiques et de bien spécifier vers quoi ils pointent. N'oubliez pas que les informations qui se trouvent dans vos pages sont consultées on-line. La rapidité d'accès aux données recherchées est donc essentielle.

Au niveau quantité, veillez à ne pas submerger l'utilisateur sous une profusion de liens. Un document aéré est toujours plus facile à lire. Sauf cas particuliers (tableau, sommaire, etc.), limitez-vous à quelques liens par page et révisez-les fréquemment pour que votre site soit toujours proche de l'actualité (c'est un des intérêts majeurs d'Internet).

# Lien dans la même page

Si une de vos pages HTML vous semble un peu longue, vous pouvez définir quelques liens pour en faciliter la lecture.

Pour chaque lien, vous procéderez en deux étapes :

- 1. Définition d'un signet pour marquer la partie à accéder.
- 2. Définition d'un lien hypertexte vers ce signet.

Pour placer un signet dans le document courant, vous utiliserez une variante de la balise  $\leq a \geq 1$ :

```
<a name="Signet">
  Texte
</a>
```

où <u>Signet</u> est le nom du marqueur et <u>Texte</u> le texte vers lequel vous désirez pointer.

Pour créer un lien hypertexte vers le signet que vous venez de définir, vous utiliserez une autre variante de la balise  $\leq a >$ :

```
<a href = "#Signet">
Texte
</a>
```

où Signet est le nom du marqueur à pointer et Texte le texte qui servira de lien.

#### Exercice

Définissez un document qui contient plusieurs paragraphes de texte. Ce texte pourra être pris sur le site <a href="http://fr.lipsum.com/">http://fr.lipsum.com/</a>.

Placez ce texte dans des paragraphes. Définissez des signets pour chacun d'entre eux et des liens hypertextes pour accéder aux différents paragraphes.

```
<!DOCTYPE html>
<html>
  <head>
   <meta charset=UTF-8" />
   <title>Liens et signets</title>
 </head>
 <body>
    <a href="#par1">Paragraphe 1</a> <a href="#par2">Paragraphe 2</a> <a href="#par3">Paragraphe
3</a> <a href="#par4">Paragraphe 4</a> <a href="#par5">Paragraphe 5</a>
<a name="par1">Paragraphe 1 : Lorem ipsum dolor sit amet, consectetur adipiscing elit...
<a name="par2">Paragraphe 2 : Lorem ipsum dolor sit amet, consectetur adipiscing elit...
<a name="par3">Paragraphe 3 : Lorem ipsum dolor sit amet, consectetur adipiscing elit...
<a name="par4">Paragraphe 4 : Lorem ipsum dolor sit amet, consectetur adipiscing elit...
<a name="par5">Paragraphe 5 : Lorem ipsum dolor sit amet, consectetur adipiscing elit...
 </body>
</html>
```

# Lien vers un document local

Comme nous venons de le voir, les signets facilitent l'accès à un passage déterminé dans un même document. Lorsqu'un lien doit donner accès à des informations qui ne se trouvent pas dans le document courant mais dans un autre document du site, les paramètres de la balise <a> doivent être légèrement différents :

```
<a href = "fichier">
Texte
</a>
```

où <u>Fichier</u> est le nom du fichier HTML auquel accéder et <u>Texte</u> le texte qui servira de lien.

# Lien vers une partie d'un document local

Vous savez maintenant définir un lien hypertexte vers un signet du même document et vers un autre document HTML. En combinant les deux syntaxes, il est possible d'accéder à un signet d'un autre document :

```
<a href = "fichier#signet">
Texte
</a>
```

où Fichier est le nom du fichier HTML auquel accéder, Signet le nom du signet dans le fichier HTML, et Texte le texte qui servira de lien.

### Lien vers un document distant

Jusqu'à présent, vous avez appris à utiliser un lien hypertexte pour pointer vers un passage du même document, un autre document du site ou un passage d'un autre document du site. En utilisant une syntaxe légèrement différente, il est possible d'accéder à un document d'un autre site :

```
<a href = "http://adresse_url">
   Texte
</a>
```

où <a href="http://Adresse URL">http://Adresse URL</a> est l'adresse URL de la page HTML et Texte le texte qui servira de lien.

Un URL (*Uniform Ressource Locator*) est une adresse permettant d'identifier de manière unique un objet Internet. Sa syntaxe "façon UNIX" détermine le nom du serveur, le chemin d'accès dans le serveur et le nom du fichier HTML à accéder : http://serveur/chemin/document#signet

### Lien vers un fichier

Pour donner accès à un fichier, il suffit de préciser son nom dans le chemin URL. Par exemple, pour accéder au document ebook.pdf dans le dossier documents du serveur <u>www.monsite.com</u>, vous utiliserez ce lien :

<a href="http://www.monsite.com/documents/ebook.pdf">Cliquez ici</a>

# Accéder à un service Internet

Dans les lignes précédentes, vous avez appris à accéder à une page Web située sur un site distant. Comme le montre ce tableau, il suffit de modifier la première partie de l'adresse pour accéder tout aussi simplement à un autre service Internet.

Service	Marqueur
Site ftp	<a href="ftp://adresse ">Texte</a>
Serveur e-mail	<a href="mailto:adresse ">Texte</a>
Gopher	<a href="gopher://adresse ">Texte</a>
Telnet	<a href="telnet://adresse ">Texte</a>

Dans chacun des marqueurs précédents, adresse désigne l'adresse URL du service auquel accéder et texte le texte utilisé pour réaliser le lien.

Pour renseigner le visiteur de votre site, vous pouvez ajouter des informations concernant une image ou un lien via une infobulle.

Voici la syntaxe à utiliser pour un lien :

<a href="adresse" title="Texte de l'infobulle">Texte du lien</a>

Essayez ce code

### **Tableaux**

Certaines pages Web contiennent des tableaux. Ceux-ci ont été réalisés à l'aide de balises .

La syntaxe du marqueur <a href="table"><a href

<u>Bordure</u> est un entier qui définit l'épaisseur de la bordure autour du tableau. Si <u>BORDER</u> n'apparaît pas dans le marqueur ou si <u>Bordure</u> vaut <u>0</u>, le tableau n'a pas de bordure.

<u>Espace</u> définit l'espace entre le contenu des cellules et la bordure gauche de la cellule, en pixels.

<u>Epaisseur</u> définit l'épaisseur du trait entre les cellules, en pixels.

<u>Largeur</u> force la largeur du tableau, en pixels. Si ce paramètre n'est pas spécifié, la largeur du tableau est calculée automatiquement.

<u>Hauteur</u> force la hauteur du tableau, en pixels. Si ce paramètre n'est pas spécifié, la hauteur du tableau est calculée automatiquement.

Vous utiliserez le marqueur <a href="tr>
pour créer chacune des lignes du tableau":</a>

où <u>alignement</u> définit l'alignement des données sur la ligne. Il peut prendre la valeur <u>left</u>, <u>center</u> ou <u>right</u> selon l'effet recherché. Si ce paramètre n'est pas spécifié, les données sont alignées à gauche.

Enfin, pour définir les caractéristiques d'une cellule, vous utiliserez le marqueur <a href="tel://td>
</ra> dont voici la syntaxe :

```
elément
```

<u>NbCol</u> est le nombre de colonnes occupées par la cellule.

<u>NbLig</u> est le nombre de lignes occupées par la cellule.

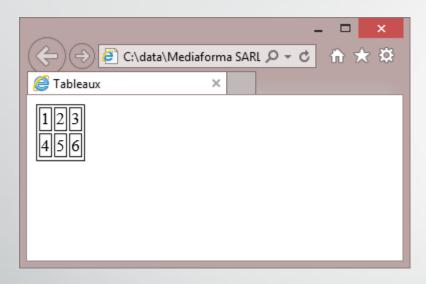
<u>AlignHoriz</u> indique l'alignement horizontal du texte dans la cellule. Ce paramètre peut prendre la valeur <u>LEFT</u>, <u>CENTER</u> ou <u>RIGHT</u>. S'il n'est pas spécifié, les données sont alignées à gauche.

<u>AlignVert</u> indique l'alignement vertical du texte dans la cellule. Ce paramètre peut prendre la valeur <u>TOP</u>, <u>MIDDLE</u> ou <u>BOTTOM</u>. S'il n'est pas spécifié, les données sont alignées sur la partie supérieure de la cellule.

<u>Largeur</u> indique la largeur de la cellule, en pixels.

<u>Elément</u> décrit le contenu de la cellule. Il peut s'agir d'un texte ou d'une image (marqueur <u>IMG</u>).

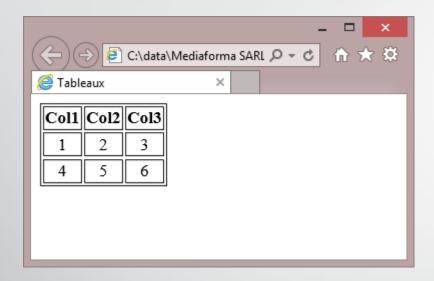
# Exercice 1 Définissez le tableau suivant :



#### Solution

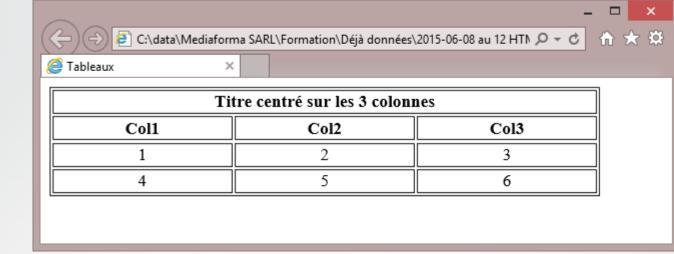
```
<!DOCTYPE html>
<html>
 <head>
  <meta charset=UTF-8" />
  <title>Tableaux</title>
 </head>
 <body>
  13
   \langle t.r \rangle
    456
   </body>
</html>
```

### Exercice 2 Définissez le tableau suivant :

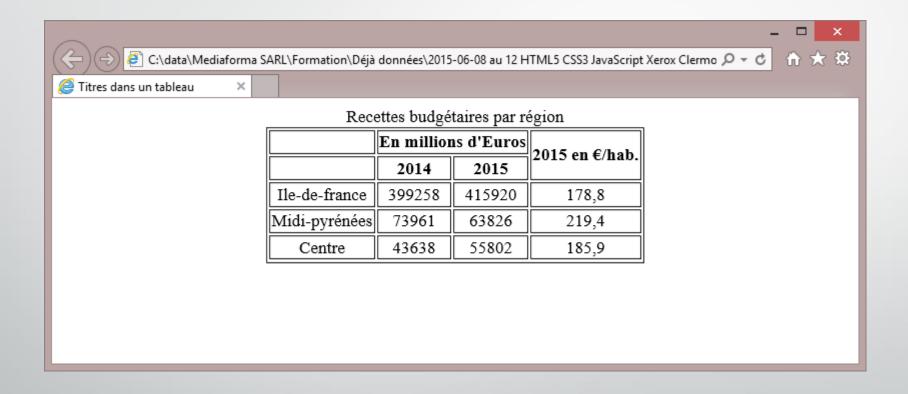


```
<!DOCTYPE html>
<html>
<head>
<meta charset=UTF-8"/>
<title>Tableaux</title>
</head>
<body>
Col1Col2Col3
 12<td
align="center">3
 45<td
align="center">6
 </body>
</html>
```

### Exercice 3 Définissez le tableau suivant :



Exercice : Définissez le tableau suivant



```
<!DOCTYPE html>
<html>
<head>
 <title>Tableaux</title>
</head>
<body>
 <center>
 Recettes budgétaires par région
 En millions d'Euros
  2015 en €/hab.
  2014
   2015
  Ile-de-france399258415920178,8
  Midi-pyrénées7396163826219,4
  Centre4363855802185,9
  </body>
</html>
```

# Insertion d'une image dans un document HTML

L'insertion d'une image dans un document HTML s'effectue en trois étapes :

- 1. Définition ou choix de l'image.
- 2. Sauvegarde au format GIF (.GIF), JPEG (.JPG) ou PNG (.PNG) et stockage dans le même dossier que le fichier HTML.
- 3. Insertion d'un marqueur <u><img></u> dans le document HTML.

# Avantages et inconvénients de ces trois types d'images

GIF : limité à 256 couleurs, peut posséder une couleur de transparence, non compressé

JPG : 16 millions de couleurs, pas de couleur de transparence, compressé

PNG : 16 millions de couleurs, pet posséder une couleur de transparence, non compressé

Téléchargez l'application PhotoFiltre 7 sur <a href="http://www.photofiltre-studio.com/news.htm">http://www.photofiltre-studio.com/news.htm</a>

Entrainez-vous à créer des images au format GIF, JPG (avec plusieurs formats de compression) et PNG et comparez leurs tailles en octets

Lorsque l'image a été sauvegardée au bon format, insérez un marqueur <<u>IMG></u> dans la page HTML, à l'endroit où vous désirez l'afficher. La syntaxe du marqueur est la suivante :

```
<img
src = "nom de l'image"

[align = {top|bottom|middle|left|right}]

[border = epaisseur]

[alt = "texte"]>
```

- 1. <u>Nom de l'image</u> est le nom complet de l'image (nom du fichier et extension).
- 2. <u>ALIGN</u> définit l'alignement de l'image par rapport au texte : <u>TOP</u>, <u>BOTTOM</u> et <u>MIDDLE</u> alignent le texte respectivement par rapport à la partie supérieure, à la partie inférieure et au centre de l'image ; <u>LEFT</u> et <u>RIGHT</u> alignent l'image respectivement à gauche et à droite du texte.
- 3. Epaisseur définit l'épaisseur du cadre autour de l'image, en pixels.
- 4. Texte est un court texte qui apparaît à la place de l'image lorsque l'affichage des images a été désactivé.

Entrainez-vous à insérer les images GIF, JPG et PNG définies précédemment et comparez leur rendu.

# Rendre une image cliquable

Pour insérer un lien sur une image, il suffit de l'entourer d'une balise <a></a> :

```
<a href="http://www.site.com"><img src="image.jpg"></a>
```

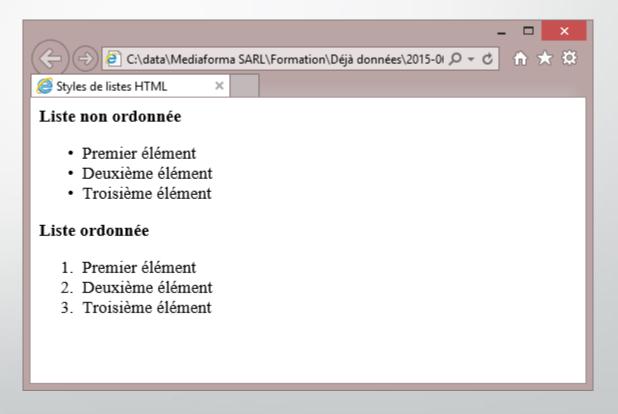
# Listes

Le langage HTML compte trois types de listes :

- non ordonnées ou listes à puces : éléments <u>ul</u> et <u>li</u> ;
- ordonnées : éléments <u>ol</u> et <u>li</u> ;

### Cet exemple illustre ces deux types de listes :

```
<!DOCTYPE html">
<html>
 <head>
   <meta charset=UTF-8" />
   <title>Styles de listes HTML</title>
 </head>
 <body>
   <b>Liste non ordonnée</b>
   <l
    Premier élément
    Deuxième élément
    Troisième élément
   <b>Liste ordonnée</b>
   <01>
    Premier élément
    Deuxième élément
    Troisième élément
   </body>
</html>
```



### **Formulaires**

Si la plupart des pages Web se contentent de diffuser des informations, elles sont également en mesure d'en recueillir par l'intermédiaire de formulaires. Les données sont stockées sur le serveur. Elles peuvent être traitées à l'aide d'un programme spécifique écrit par vos soins, fourni par le prestataire ou tout simplement expédiées sans traitement dans votre boîte aux lettres.

Pour définir un formulaire, vous utiliserez le marqueur <a href="form"> </form</a>> dont voici la syntaxe :

```
<form
  action = "traitement"
  method = "{get|post}">
</form>
```

- 1. <u>traitement</u> est le nom d'un programme de traitement, par exemple <a href="http://www.site.com/traitement.php">http://www.site.com/traitement.php</a>, ou encore l'adresse de votre boîte aux lettres, précédée de mailto. Par exemple. <a href="mailto:pierre.durand@Bertold.fr">mailto:pierre.durand@Bertold.fr</a>.
- 2. <u>method</u> définit la méthode à utiliser pour prendre en charge les données du formulaire. La méthode <u>GET</u> étant limitée quant à la quantité de données transmissibles, la méthode <u>POST</u> est presque toujours utilisée.

Entre les balises <u><form></u> et <u></form></u>, vous utiliserez des balises <u><input></u> pour définir les données à saisir :

```
<input
  [type = {"text|password|checkbox|radio|submit|reset|hidden|image"}]
  name = "nom"
  [value = "valeur"]
  [size = "taille1[,taille2]"]
  [maxlength = "longueur"]
  [checked]>
```

Le paramètre optionnel <u>TYPE</u> définit le type de l'objet :

Valeur Effet

TEXT Champ de saisie

PASSWORD Mot de passe

CHECKBOX Case à cocher

RADIO Bouton radio

SUBMIT Bouton d'envoi, pour envoyer les données du formulaire au serveur

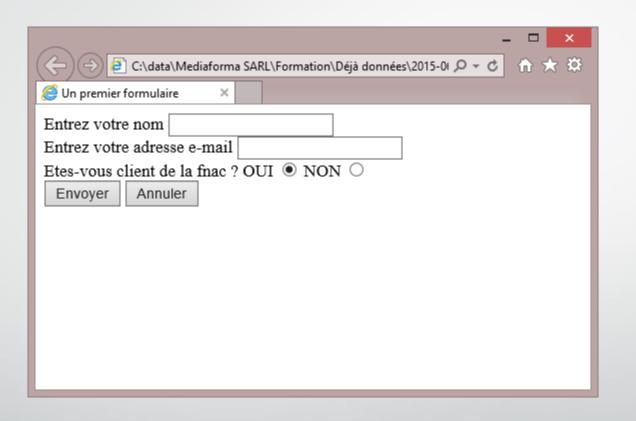
RESET Bouton de réinitialisation du formulaire

HIDDEN Champ caché qui n'apparaît pas sur le formulaire mais qui est envoyé lors

de l'appui sur le bouton SUBMIT ou sur une image cliquable

#### Exercice:

Définissez un formulaire pour obtenir le résultat suivant :

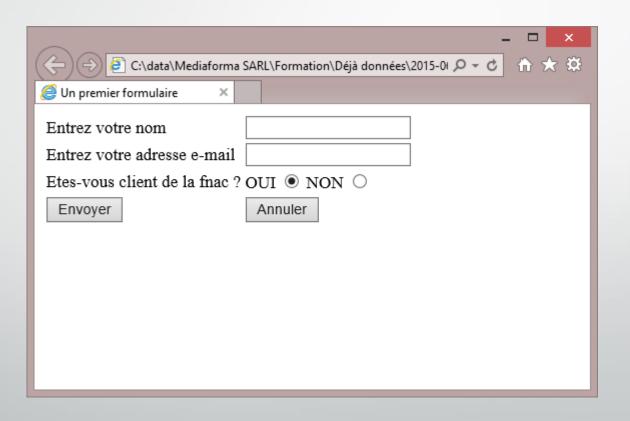


#### Solution

```
<!DOCTYPE html">
<ht.ml>
 <head>
   <title>Un premier formulaire</title>
 </head>
 <body>
   <form action = "mailto:bertrand.oscar@oracle.fr" method = "post">
      Entrez votre nom <input type = "text" name = nom><br>
     Entrez votre adresse e-mail <input type = "text" name = nom><br>
     Etes-vous client de la fnac ?
     OUI <input type = "radio" name = "client" value = "oui" checked>
     NON <input type = "radio" name = "client" value = "non" > <br>
     <input type = "submit" value = "Envoyer">
      <input type = "reset" value = "Annuler">
   </form>
 </body>
</html>
```

### Exercice

Utilisez un tableau pour améliorer le formulaire comme ceci :



#### Solution

```
<!DOCTYPE html">
<ht.ml>
 <head>
   <title>Un premier formulaire</title>
 </head>
 <body>
   <form action = "mailto:bertrand.oscar@oracle.fr" method = "post">
    Entrez votre nom<input type = "text" name = nom>
      Entrez votre adresse e-mail<input type = "text" name = nom>
      Etes-vous client de la fnac ?
      OUI <input type = "radio" name = "client" value = "oui" checked>
      NON <input type = "radio" name = "client" value = "non" >
      <input type = "submit" value = "Envoyer">
      <input type = "reset" value = "Annuler">
    </form>
 </body>
</html>
```

### Une liste déroulante dans un formulaire

Pour définir une liste déroulante, vous utiliserez la balise <select>. Chacun des éléments de la liste sera placé dans une balise enfant <option>

Dans la balise **<select>**, vous pouvez définir l'attribut **size** pour indiquer combien d'éléments sont visibles dans la liste (par défaut, **size** vaut 1).

Dans l'une des balises **<option>** vous pouvez définir l'attribut **selected**, sans lui affecter aucune valeur. L'élément correspondant est alors sélectionné par défaut dans la liste.

#### Outre les boutons de formulaires submit et reset :

```
<input type="submit">
<input type="reset">
```

Vous pouvez utiliser de simples boutons à l'intérieur ou à l'extérieur d'un formulaire avec la balise **<button>** :

```
<button>Texte</putton>
```

Partie 2

Mise en forme en CSS

Vous utiliserez le langage CSS pour mettre en forme les balises d'un ou de plusieurs documents, à travers des attributs ou des feuilles de styles.

Les prochaines diapositives vont vous montrer comment utiliser ce langage.

## Styles CSS dans les balises

style=""

#### où:

- <u>balise</u> est un nom de balise :  $\leq p > ou \leq h1 > par$  exemple.
- <u>propriété1</u> sont des propriétés de style de la balise.
- <u>valeur1</u> sont les valeurs affectées aux propriétés.

Cette technique n'est pas optimale, car limitée au seul élément qui l'utilise. Dans la mesure du possible, privilégiez l'utilisation d'une feuille de styles externe, d'extension .css et reliée au document à l'aide d'un élément <u>link</u>. Cette même feuille de styles pourra être utilisée dans tous les documents apparentés pour uniformiser leur apparence.

## Styles CSS dans les balises

À titre d'exemple, pour affecter la couleur jaune à l'arrière-plan d'un élément p, vous utiliserez le code suivant :

```
Ce texte a un arrière-plan jaune
```

Ou encore, pour utiliser la police Verdana corps 18 dans un titre <u>h2</u>, vous utiliserez le code suivant :

```
<h2 style="font-family:Verdana; font-size: 18px;">
Ce titre est en Verdana corps 18
</h2>
```

# Feuille de styles interne

Pour étendre la portée des définitions CSS, vous pouvez les regrouper dans l'en-tête du document HTML.

Voici la syntaxe permettant d'affecter des propriétés à une balise :

```
élément {propriétél:valeurl; ... propriétéN:valeurN}
```

#### où:

- <u>élément</u> est un nom d'élément : <u>p</u> ou <u>h1</u> par exemple.
- <u>propriété1</u> sont des propriétés de style de l'élément.
- <u>valeur1</u> sont les valeurs affectées aux propriétés.

Ces lignes de code doivent être insérées entre les balises <u><style></u> et <u></style></u>, à l'intérieur de l'élément <u>head</u>.

### Exercice

Ecrivez une feuille de styles interne pour :

- Définir dans tout le document un arrière-plan de couleur jaune pour les éléments <u>p</u>
- Utiliser la police Verdana corps 18 dans tous les titres <u>h2</u>

## Solution

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8" />
    <title>Une feuille de styles interne</title>
    <style>
      p {background-color:yellow;}
      h2 {font-family: Verdana; font-size: 18px;>
    </style>
  </head>
  <body>
  </body>
</html>
```

# Feuille de styles externe

Pour augmenter encore le champ d'action des styles définis dans l'en-tête d'un document, vous pouvez les stocker dans un fichier CSS. Dans ce cas, plusieurs documents HTML peuvent référencer cette "feuille de styles" (le fichier CSS) pour utiliser les styles qui y sont définis.

L'opération se fait avec un élément <u>link</u> (ici, la feuille de styles a pour nom *moncss.css*) :

```
<link rel="stylesheet" href="moncss.css" />
```

### Exercice

Définissez une feuille de styles externe qui reprend les règles définies dans l'exercice précédent et faites référence à cette feuille de styles dans un document HTML.

#### Solution

p {background-color:yellow;}
h2 {font-family:Verdana; font-size: 18px;>

La feuille de styles moncss.css

Le document HTML

# Sélecteurs, propriétés et valeurs CSS

La syntaxe à utiliser est la suivante :

```
sélecteur {propriété1: valeur1; ... propriétéN: valeurN;} élément {propriété1: valeur1; ... propriétéN: valeurN;}
```

Pour définir le style d'un élément, il suffit d'entrer le nom de l'élément, d'ouvrir des accolades, d'énumérer des propriétés et de leur affecter les valeurs souhaitées.

#### Exercice

Définissez un document HTML5 qui contient une liste à puces.

Définissez une feuille de styles internet qui donne les caractéristiques de la liste à puces :

• Couleur : blue

• Arrière-plan : #FF4040

• Marges: 14px

• Style des puces : square

## Solution

codeoo1.htm

```
<!DOCTYPE html>
<html>
<head>
 <meta charset=UTF-8"/>
 <title>Style des li</title>
 <style>
  color: blue;
  background: #FF4040;
  margin: 14px 14px 14px 14px;
  list-style: square;
 </style>
</head>
<body>
 Une liste à puces :
 Premier
  Deuxième
  Troisième
  Quatrième
 </body>
</html>
```

### Sélecteurs CSS 2.1

Le sélecteur universel étoile (\*) s'adresse à tous les éléments. Vous l'utiliserez essentiellement pour modifier le style de tous les éléments de la page. Par exemple, la police.

```
* { font-family: Verdana; }
```

Le sélecteur de type permet de changer le style d'un élément. Ici, par exemple, le style de tous les éléments <u>h1</u>:

```
h1 {font-size: 120%;}
```

Le sélecteur de classe est représenté par un point (<u>.</u>). Il cible les éléments en fonction de la valeur de leur attribut <u>class</u>. Par exemple, le style ci-après s'applique à tous les éléments <u>p</u> dont l'attribut <u>class</u> vaut <u>info</u>.

```
p.info {backgroung: yellow;}
```

Si aucun élément ne restreint le sélecteur de classe, tous les éléments dont l'attribut <u>class</u> a la valeur spécifiée sont concernés. Ici, par exemple, il s'agira de tous les éléments dont l'attribut <u>class</u> vaut <u>info</u> :

```
.info {backgroung: yellow;}
```

## Sélecteurs CSS 2.1

Il est possible d'affecter plusieurs classes à un élément (l'attribut <u>class</u> peut recevoir plusieurs valeurs séparées par des espaces). On parle alors de "sélecteur de classes multiples". Ici par exemple, seuls les éléments <u>p</u> dont l'attribut <u>class</u> vaut <u>info</u> et <u>premier</u> sont affectés :

```
p.info.premier {backgroung: yellow;}
```

Voici un exemple d'élément p concerné par cette règle CSS:

```
texte
```

### Sélecteurs CSS 2.1

Le sélecteur d'identificateur est représenté par le caractère dièse (<u>#</u>). Il cible l'élément dont l'attribut <u>id</u> a la valeur spécifiée. Ici, seul l'élément dont la propriété <u>id</u> vaut <u>corne</u> est concerné :

```
#corne {font-family: Serif;}
```

Vous pouvez restreindre le ciblage en précisant un élément devant le sélecteur d'<u>id</u>. Ici, par exemple, seul l'élément <u>p</u> dont le marqueur <u>id</u> vaut <u>corne</u> est concerné :

```
p#corne {font-family: Serif;}
```

Un sélecteur descendant est construit avec deux sélecteurs traditionnels (ou plus) séparés par un espace. Il cible les enfants du premier sélecteur. Par exemple, pour affecter un arrière-plan jaune à tous les éléments <u>li</u> enfants d'un élément <u>div</u>, vous utiliserez le sélecteur descendant ci-après :

```
div li {color: yellow;}
```

#### Exercice

- 1) Affectez la couleur rouge aux balises p de classe rouge
- 2) Affectez la police Courrier aux balises div de classe code ou exemple
- 3) Affectez la couleur d'arrière-plan jaune aux balises enfants de la balise

#### Voici le code HTML sur lequel appliquer ces règles :

```
<body>
  Ceci est un paragraphe de classe rouge
  Ceci est un paragraphe de classe vert
  Ceci est un paragraphe de classe noir
  <div class="code">Ce texte est affiché dans une balise div de classe code</div>
  <diy>Ce texte est affiché dans une balise div sans classe</div>
  <div class="exemple">Ce texte est affiché dans une balise div de classe
exemple</div>
  Une liste UL
  Premier
  Deuxième
  Troisième
  Une liste OL
  <0|>
  Premier
   Deuxième
  Troisième
  </body>
```

#### Exercice

#### Et voici le résultat à obtenir :

#### Ceci est un paragraphe de classe rouge

Ceci est un paragraphe de classe vert

Ceci est un paragraphe de classe noir

Ce texte est affiché dans une balise div de classe code Ce texte est affiché dans une balise div sans classe Ce texte est affiché dans une balise div de classe exemple Une liste UL

- Premier
- Deuxième
- · Troisième

#### Une liste OL

- 1. Premier
- 2. Deuxième
- Troisième

## Solution

codeoo2.htm

#### id ou class?

#### L'attribut id est généralement utilisé :

- Sur les éléments qui structurent le document et qui sont désignés comme uniques : par exemple, les éléments <u>header</u>, <u>container</u>, <u>content</u>, <u>nav</u>, <u>footer</u>, etc. Une fois ces éléments identifiés avec l'attribut <u>id</u>, ils peuvent être stylés avec des propriétés CSS.
- Sur d'autres éléments qui doivent être accessibles en JavaScript.
- Pour accéder à une ancre.

L'attribut <u>class</u> est utilisé dans tous les autres cas.

```
<div class="perso">
```

# Sélecteurs CSS3

Tout ce qui a été dit sur les sélecteurs à la section précédente reste valable en CSS3. Cependant, de nouveaux sélecteurs sont maintenant disponibles, qui permettent de cibler encore plus précisément les éléments auxquels vous vous adressez.

# Sélecteurs CSS3

Syntaxe	Signification
nom_élément[attr^="valeur"]	Tout élément <u>nom_élément</u> dont la valeur de l'attribut <u>attr</u> commence exactement par la chaîne <u>valeur</u>
nom_élément[attr\$="valeur"]	Tout élément <u>nom_élément</u> dont la valeur de l'attribut <u>attr</u> finit exactement par la chaîne <u>valeur</u>
nom_élément[attr*="valeur"]	Tout élément <u>nom_élément</u> dont la valeur de l'attribut <u>attr</u> contient la sous- chaîne <u>valeur</u>
:root	Un élément <u>nom_élément</u> , racine du document
nom_élément:nth-child(n)	Un élément <u>nom_élément</u> qui est le n-ième enfant de son parent
nom_élément:nth-last-child(n)	Un élément <u>nom_élément</u> qui est le n-ième enfant de son parent en comptant depuis le dernier enfant
nom_élément:nth-of-type(n)	Un élément <u>nom_élément</u> qui est le n-ième enfant de son parent et de ce type
nom_élément:nth-last-of-type(n)	Un élément <u>nom_élément</u> qui est le nième enfant de son parent et de ce type en comptant depuis le dernier enfant
nom_élément:last-child	Un élément <u>nom_élément</u> , dernier enfant de son parent
nom_élément:first-of-type	Un élément <u>nom_élément</u> , premier enfant de son type

# Sélecteurs CSS3

Syntaxe	Signification
nom_élément:last-of-type	Un élément <u>nom_élément</u> , dernier enfant de son type
nom_élément:only-child	Un élément <u>nom_élément</u> , seul enfant de son parent
nom_élément:only-of-type	Un élément <u>nom_élément</u> , seul enfant de son type
nom_élément:empty	Un élément <u>nom_élément</u> qui n'a aucun enfant
<u>nom_élément:target</u>	Un élément <u>nom_élément</u> qui est la cible de l'URL
	d'origine
nom_élément:enabled	Un élément d'interface utilisateur <u>nom_élément</u> qui est
	actif ou inactif
nom_élément:checked	Un élément d'interface utilisateur <u>nom_élément</u> qui est
	coché ou dont l'état est indéterminé (bouton radio ou
	case à cocher par exemple)
<u>nom_élément:contains("attr")</u>	Un élément <u>nom_élément</u> dont le contenu textuel
	concaténé contient la sous-chaîne <u>attr</u>
nom_élément:selection	La partie d'un élément <u>nom_élément</u> qui est
	actuellement sélectionnée par l'utilisateur
<u>nom_élément:not(sel)</u>	Un élément <u>nom_élément</u> qui n'est pas représenté par
	le sélecteur simple <u>sel</u>
<u>nom_élément ~ F</u>	Un élément <u>F</u> précédé par un élément <u>nom_élément</u>

# Pseudo-classes, pseudo-éléments

Le sélecteur de pseudo-classe est représenté par le caractère deux-points (<u>:</u>).

Vous ferez appel aux pseudo-classes pour cibler des éléments en fonction de caractéristiques inaccessibles aux sélecteurs traditionnels : premier enfant ou focus par exemple.

La pseudo-classe <u>:first-child</u> permet de cibler le premier enfant d'un élément. Par exemple, pour mettre en gras le premier élément <u>p</u> enfant de l'élément <u>div</u>, vous utiliserez le sélecteur ci-après :

```
div p:first-child {font-weight: bold;}
```

Les pseudo-classes <u>:link</u> et <u>:visited</u> ciblent les éléments <u>a</u> dont (respectivement) le lien n'a pas été visité/a été visité. Les deux lignes suivantes définissent la couleur des liens :

```
:link {color: fushia}
:visited {color: navy}
```

# Pseudo-classes, pseudo-éléments

La pseudo-classe <u>:focus</u> cible les éléments qui ont le focus (par défaut, les liens et les éléments de formulaires). Elle permet très simplement de modifier la couleur d'arrière-plan (ou un autre style) d'un élément. Ici, par exemple, nous affectons un arrière-plan rouge à l'élément <u>input</u> de type <u>text</u> qui a le focus :

```
input[type=text]:focus {background: red;}
```

La pseudo-classe <u>:hover</u> cible les éléments dont le contenu est survolé. Cela permet par exemple de changer la bordure d'une image lorsqu'elle est pointée par la souris :

```
img:hover {border-style: dotted;}
```

# Pseudo-classes, pseudo-éléments

#### CSS<sub>3</sub> s'enrichit de plusieurs pseudo-classes :

Pseudo-classe	Signification
:nth-child(position)	Éléments repérés par leur position dans le parent
:nth-last-child(position)	Dernier élément du parent
type:nth-of-type(position)	Éléments du type spécifié repérés par leur position dans le parent
type:nth-last-of-type(N)	Dernier élément du type spécifié du parent
:last-child	Dernier élément du parent
type:first-of-type	Premier élément du type spécifié
:last-of-type	Dernier élément du type spécifié
:only-child	Élément qui est l'enfant unique du parent
type:only-of-type	Élément qui est l'enfant unique du type spécifié du parent
:root	Éléments à la racine du document
:empty	Éléments sans enfant
:target	Élément dont l'identifiant est l'ancre spécifiée dans l'URL de la page
<u>:enabled</u>	Éléments validés (enabled)
:disabled	Éléments dévalidés (disabled)
:checked	Éléments (cases à cocher ou boutons radio) cochés
:not(Sélecteur)	Éléments qui sont différents du sélecteur

#### Exercice

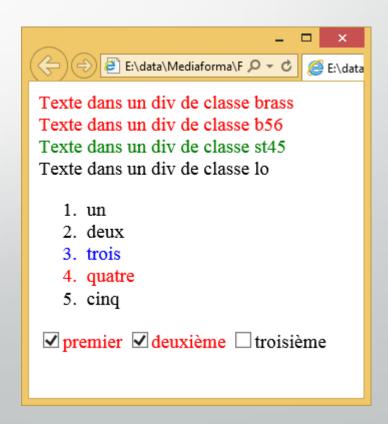
Quelle syntaxe utiliser pour s'adresser :

- 1) Aux éléments div dont l'attribut class commence par "br"
- 2) Aux éléments div dont l'attribut class contient la lettre "t"
- 3) L'avant-dernier élément li enfant de ol
- 4) Le troisième enfant li de l'élément ol de classe "t"
- 5) Les éléments checkbox qui sont cochés

### Solution

- div[class^="br"]
- 2) div[class\*="t"]
- 3) ol li:nth-last-child(2)
- 4) ol li:nth-child(3)
- 5) input[type=checkbox]:checked+label

Définissez le code HTML5 et CSS3 nécessaire pour obtenir ce résultat :



#### Solution

codeoo3.htm

```
<!DOCTYPE html>
<html lang="fr">
<head>
 <meta charset=UTF-8"/>
 <style>
  div[class^="br"] {color: red;}
  div[class*="t"] {color: green;}
  ol li:nth-last-child(2) {color: red;}
  ol li:nth-child(3) {color: blue;}
  input[type=checkbox]:checked+label {color:red; }
 </style>
</head>
<body>
 <div class="brass">Texte dans un div de classe brass</div>
 <div class="br56">Texte dans un div de classe b56</div>
 <div class="st45">Texte dans un div de classe st45</div>
 <div class="lo">Texte dans un div de classe lo</div>
 <0|>
  un
  deux
  trois
  quatre
  cinq
 <form>
  <input type="checkbox" checked><label>premier</label>
  <input type="checkbox" checked><label>deuxième</label>
  <input type="checkbox"><label>troisième</label>
 </form>
</body>
</html>
```

## Unités CSS

Voici la liste des unités exprimées de façon absolue :

```
<u>cm</u>: centimètre;
<u>in</u>: pouce (2,54 cm);
<u>mm</u>: millimètre;
<u>pt</u>: point (1/62 inch);
<u>pc</u>: pica (12 points).
```

À titre d'information, 1 in = 2,54 cm = 25,4 mm = 72 pt = 6 pc.

### Unités CSS

Il existe également des unités exprimées de façon relative, qui dépendent de leur élément parent :

- <u>em</u> : cadratin. Un em correspond grossièrement à la hauteur de la lettre "Ç" de l'élément parent (taille du texte définie dans la propriété <u>font-size</u>).
- <u>ex</u>: fraction de la hauteur des caractères. Dans les polices occidentales, ex correspond à la hauteur de la lettre minuscule "x".
- <u>px</u> : pixel. Cette unité dépend de la résolution du périphérique d'affichage.
- <u>%</u> : pourcentage de la taille de l'élément ou de son parent.

L'unité <u>em</u> n'est pas à exclure. Elle permet en effet aux utilisateurs finaux de redimensionner le texte proportionnellement à la taille de police, tout en conservant la mise en page intacte. C'est la raison pour laquelle je ne saurais trop vous recommander de l'utiliser!

## Ossature d'un document HTML

#### Dans cette partie :

- Nouvelle organisation des documents HTML
- Sections et niveaux de titre
- address

# Organisation des documents HTML5

Bien souvent, les pages web sont constituées d'un en-tête et d'un pied de page, d'un menu, d'une zone réservée au contenu (article, images, vidéos, etc.) et, éventuellement, d'une zone annexe, n'ayant aucun rapport direct avec le contenu de la page.

Le langage HTML5 a de nouvelles balises pour répondre à ces besoins.

```
<html>
   <head>
    <meta charset=UTF-8" />
    <title>Un document HTML5 avec en-tête, bas de page, menu, zone annexe et Zone de contenu</title>
    <link rel="stylesheet" href="styles004.css">
    <script src="Creation-elements-HTML5.js"></script>
  </head>
  <body>
    <header>
      <h1>En-tête du document</h1>
      Texte de l'en-tête
    </header>
    <nav>
      < h2 > Menu < / h2 >
      <111>
        <a href="page1.htm">Page 1</a>
        <a href="page2.htm">Page 2</a>
        <a href="page3.htm">Page 3</a>
      </111>
      <br><br><br><br></pr>
    </nav>
    <aside>
      Texte affiché dans la partie droite de la page avec la balise <aside&gt;<br>
    </aside>
    <article>
      <h2>Premier article</h2>
        Texte du premier article
    </article>
    <article>
      <h2>Deuxième article</h2>
        Texte du deuxième article
    </article>
    <footer>
      Copyright et e-mail du Webmaster
    </footer>
```

Les balises <u><div></u> n'ont pas disparu du langage HTML5. Elles sont simplement moins souvent utilisées, car plusieurs nouvelles balises plus spécialisées viennent lui prêter main-forte.

HTML<sub>5</sub>

<!DOCTYPE html>

</body> </html>

Saisissez ce code et sauvegardez-le sous le nom codeoo4.htm

# Saisissez ce code et sauvegardez-le sous le nom stylesoo4.css

#### CSS<sub>3</sub>

```
Le rendu <u>block</u> (<u>display: block;</u>) est affecté aux nouveaux
header, nav, article, section, footer, aside
                                                                            éléments HTML5 pour assurer le support de ces
                                                                            nouveaux éléments dans les navigateurs peu ou pas
  display: block;
                                                                            compatibles.
header
                                                                     header défini l'en-tête d'une section ou d'une page web.
  background-color: red;
nav
                                                                     nav est destiné à contenir des liens de navigation dans le site.
  float:left;
  width:20%;
  background-color:yellow;
                                                                     article correspond à une entité autonome du document : par
article
                                                                     exemple, un article (ou un ensemble d'articles) extrait d'un blog ou
                                                                     d'un forum.
  background-color: #66FFFF;
  width: 80%;
aside
                                                                     aside définit un contenu qui n'a pas de rapport direct avec la page. Il
                                                                     peut être utilisé pour définir un menu ou pour donner accès aux
                                                                     archives du site.
  float:right;
  width:20%;
  background-color:yellow;
                                                                     <u>footer</u> définit le pied de page d'une section ou d'une page web.
footer
  clear:both;
  background-color: #99FF66;
```

Les navigateurs Internet Explorer 8 et inférieur ne sont pas en mesure d'affecter un style aux nouveaux éléments HTML5. En effet, puisqu'ils ne les connaissent pas, ils doivent les intégrer au DOM *via* la méthode <u>createElement</u>. Pour ce faire, quelques lignes de JavaScript sont nécessaires :

```
document.createElement("footer");
document.createElement("header");
document.createElement("article");
document.createElement("nav");
document.createElement("aside");
```

Saisissez ce code et sauvegardez-le sous le nom Creation-elements-HTML5.js

Exécutez le code HTML. Voici ce que vous devriez obtenir :



Les éléments <u>header</u>, <u>nav</u>, <u>aside</u>, <u>article</u> et <u>footer</u> ne sont pas les seuls à faire leur apparition dans le langage HTML5. Vous pouvez également utiliser l'élément <u>section</u> pour définir des sections dans un document.

Il sert généralement pour segmenter une page ou un article en plusieurs sujets.

Il peut être employé conjointement aux éléments de titre <u>h1</u> à <u>h6</u> pour définir la structure du document.

Il correspond plus ou moins à l'ancien élément <u>div</u> du HTML 4.0x/XHTML 1.x, à ceci près qu'il y ajoute une signification sémantique. En effet, il regroupe des éléments qui traitent d'un même sujet.

Ici, par exemple, nous utilisons un élément <u>section</u> pour englober plusieurs articles au sein du document :

```
<section>
  <article>
    Contenu du premier article
  </article>
   <article>
    Contenu du dernier article
  </article>
</section>
```

Vous pouvez également inclure plusieurs contenus dans un article en utilisant plusieurs sections.

```
<article>
 <section>
    Contenu de la première section
 </section>
   <section>
    Contenu de la dernière section
  </section>
</article>
```

Prenons l'exemple d'un blog, constitué d'un empilement de plusieurs articles (ou *posts*), non nécessairement liés les uns aux autres. Ces composants pourraient être insérés dans des éléments <u>article</u>, eux-mêmes insérés dans un élément <u>section</u>:

```
<section>
    <article>
        <h3>Titre de la première section</h3>
        Texte ou autre contenu qui présente la section
</article>
        <article>
            <h3>Titre de la dernière section</h3>
            Texte ou autre contenu qui présente la section
</article>
</section>
```

Si vous avez du mal à faire la différence entre <u><section></u> et <u><article></u>, dites-vous qu'une section peut être composée d'un ou de plusieurs articles qui traitent d'un même sujet.

Par exemple, si une page web contient plusieurs sujets sur les baleines blanches et plusieurs autres sur les arbres fruitiers, les sujets sur les baleines blanches seront regroupés dans une même section et décomposés en autant d'articles que nécessaire.

De même en ce qui concerne les sujets sur les arbres fruitiers.

```
<section>
 <h1>Articles sur les baleines blanches</h1>
 <article>
   <h2>Les baleines blanches, habitat</h2
   texte de l'article
 </article>
 <article>
   <h2>Les baleines blanches, population</h2
   texte de l'article
 </article>
 <article>
   <h2>Les baleines blanches, une espèce menacée</h2
   texte de l'article
 </article>
</section>
<section>
 <h1>Articles sur les arbres fruitiers</h1>
 <article>
   <h2>Les arbres fruitiers, catégories</h2
   texte de l'article
 </article>
 <article>
   <h2>Les arbres fruitiers, taille</h2
   texte de l'article
 </article>
 <article>
   <h2>Les arbres fruitiers, récolte</h2
   texte de l'article
 </article>
</section>
```

Un ou plusieurs niveaux de titre  $\leq h1 \geq$  à  $\leq h6 \geq$  peuvent être utilisés à l'intérieur d'un élément <u>section</u>, mais aussi <u>article</u>, <u>aside</u> ou <u>nav</u>. Dans le code suivant, des niveaux de titre  $\leq h2 \geq$  sont utilisés dans les sections et un niveau de titre  $\leq h1 \geq$  dans l'article qui englobe les sections :

#### address

L'élément <u>address</u> permet de définir des informations permettant aux visiteurs d'une page d'entrer en contact avec son initiateur. Vous pouvez ainsi renvoyer vers les contributeurs d'une page, les adresses e-mail en rapport avec une page de produit ou encore l'adresse postale de la société qui est à l'origine du site.

Par exemple, pour renvoyer vers les pages web des personnes qui assurent la maintenance d'un site, vous pourriez utiliser le code suivant (ce code pourrait être placé dans l'élément <u>footnote</u>) :

### address

Il est possible de combiner plusieurs types d'informations dans un élément <u>address</u> :

```
<address>
   Ecrit par Michel Martin <br />
   <a href="mailto:lem.korp@free.fr">Ecrivez moi</a><br />
   Address: 5, rue des Formateurs, 75005 Paris<br />
   Téléphone: 01 23 45 67 89
</address>
```

# Mise en forme d'un document en HTML5 et CSS3

#### Dans cette partie :

- Polices exotiques
- Couleur et image d'arrière-plan
- Gradients linéaires
- Gradient radiaux
- Masquer une image avec un gradient
- RGBA() et HSLA()
- Regrouper des éléments
- etc..

CSS3 permet d'utiliser des polices qui ne sont pas installées sur les "postes clients". C'est la propriété CSS3 <u>@font-face</u> qui autorise cette prouesse.

```
@font-face
{
   font-family: "nom-police";
   [font-style: style-police;]
   [font-variant: petites-capitales;]
   [font-weight: graisse-police;]
   [font-stretch: condensé-étendu;]
   [font-size: taille-police;]
   src: [local('nom-local'),] url('url-police') [format(format-police)];
}
```

- nom-police est le nom de la police, tel qu'il sera utilisé dans la feuille de styles.
- <u>style-police</u> définit le style de la police : <u>normal</u>, <u>italic</u> ou <u>oblique</u>.
- <u>Font-variant:</u> indique la casse des caractères : <u>normal</u> ou <u>small-caps</u>.
- <u>largeur-police</u> indique la largeur de la police : <u>normal, bold, 100, 200, 300, 400, 500, 600, 700, 800</u> ou <u>900</u>.
- <u>condensé-étendu</u> définit l'état de compression/extension horizontale de la police : <u>all</u>, <u>normal</u>, <u>ultra-condensed</u>, <u>extra-condensed</u>, <u>condensed</u>, <u>semi-condensed</u>, <u>semi-expanded</u>, <u>extra-expanded</u> ou <u>ultra-expanded</u>.
- <u>taille-police</u> définit les tailles possibles pour la police : all pour toutes les polices proportionnelles, une ou deux tailles pour les polices bitmap ou devant être rendues dans une taille ou une fourchette de tailles spécifiques.
- <u>nom-local</u> définit le nom local de la police, au cas où cette dernière serait installée sur l'ordinateur.
- <u>url-police</u> définit l'adresse URL de la police accessible sur le serveur.
- <u>format-police</u> définit le format de la police : <u>truedoc-pfr</u> (TrueDoc<sup>TM</sup> Portable Font Resource, .pfr), <u>embedded-opentype</u> (Embedded OpenType, .eot), <u>type-1</u> (PostScript<sup>TM</sup> Type 1, .pfb ou .pfa), <u>truetype</u> (TrueType, .ttf) ou <u>opentype</u> (OpenType, .ttf).

Pour implémenter la police Swiss 721, disponible sur le serveur dans le dossier fonts, sous le nom swiss721lt.pfr, en style Normal et italique et en graisse 200, vous utiliserez les instructions suivantes :

```
@font-face
{
    font-family: "Swiss 721";
    src: url("fonts/swiss721lt.pfr");
    font-style: normal, italic;
    font-weight: 200;
}
```

#### Exercice

Définissez le code CSS pour implémenter la police DeliciousRoman, disponible localement sous le nom "Delicious Roman" et sur le site web http://www.font-face.com/fonts/delicious/, sous le nom Delicious-Roman.otf, en largeur 400.

#### Solution

```
@font-face
{
    font-family: DeliciousRoman;
    src: local('Delicious Roman'), url(http://localhost../Delicious-Roman.otf);
    font-weight:400;
}
```

#### Exercice

Définissez le code CSS pour implémenter la police SketchRockwell, disponible sur le serveur, dans le dossier courant, sous le nom SketchRockwell.ttf.

#### Solution

```
@font-face
{
   font-family: SketchRockwell;
   src: url('SketchRockwell.ttf');
}
```

Pour utiliser une police implémentée *via* <u>@font-face</u>, il suffit d'affecter la propriété <u>font-family</u> définie dans <u>@font-face</u> à un élément ou une classe quelconque. Voici un exemple complet qui implémente la police know\_your\_product.ttf côté serveur.

Le code CSS mis en œuvre est le suivant :

```
{
    font-family: KnowYourProduct;
    src: url('know_your_product.ttf');
}

La propriété @font-face définit la police serveur
    KnowYourProduct (font-family). Le fichier correspondant se
    trouve dans le dossier courant sous le nom
    know_your_product.ttf

h1

{
    font-family: 'KnowYourProduct';
}

La propriété font-family du style h1 est redéfinie pour que
    les éléments de style h1 appliquent la police
    KnowYourProduct.
}
```

Saisissez ce code et sauvegardez-le dans le fichier font-face.css dans le dossier c:\data\html5css3

Pour utiliser ce code CSS3, il suffit de créer un document HTML5 conventionnel et d'utiliser un titre de niveau 1 dans le corps du document.

Saisissez ce code et sauvegardez-le dans le fichier font-face.htm dans le dossier c:\data\html5css3

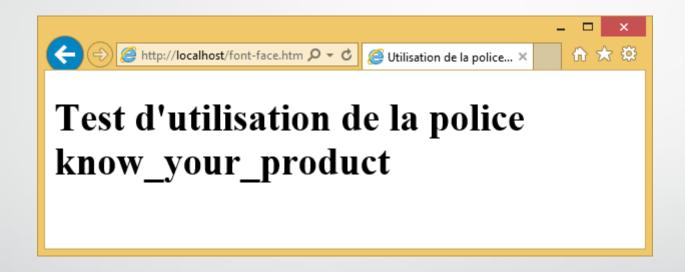
Insérez le fichier know\_your\_product.ttf dans le dossier c:\data\html5css3, puis exécutez le code font-face.htm dans Google Chrome ou Firefox en tapant cette URL :

http://localhost/font-face.htm

Voici ce que vous devriez obtenir :



Affichez ce document dans Internet Explorer. Le résultat est bien décevant :



Cela ne vient pas d'une incompatibilité d'Internet Explorer, mais plutôt d'un mauvais format de police : le navigateur n'est pas en mesure d'exploiter les polices au format TTF. Pour qu'il produise le résultat escompté, la police doit être spécifiée au format EOT.

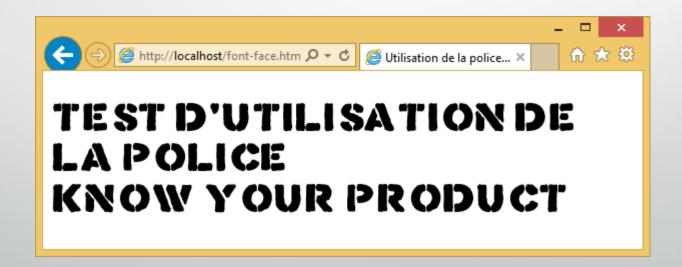
Modifiez le fichier font-face.css en ajoutant une ligne <u>src:</u> dans la définition <u>@font-face</u>:

```
@font-face
{
   font-family: KnowYourProduct;
   src: url('know_your_product.ttf');
   src: url('know_your_product.eot');
}

h1
{
   font-family: 'KnowYourProduct';
}
```

Insérez le fichier know\_your\_product.eot dans le dossier c:\data\html5css3, puis exécutez le code font-face.htm dans Internet Explorer en tapant cette URL : <a href="http://localhost/font-face.htm">http://localhost/font-face.htm</a>

Voici ce que vous devriez obtenir :



Pour assurer la plus grande compatibilité possible, le mieux est de mettre à disposition les versions EOT, TTF, SVG et WOFF de la police. Il vous suffit alors de référencer ces quatre variantes dans le code CSS :

```
@font-face
 font-family: KnowYourProduct;
 src: url('know your product.ttf');
 src: url('know your product.eot');
 src: url('know your product.svg');
 src: url('know your product.woff');
```

# Où trouver des polices exotiques?

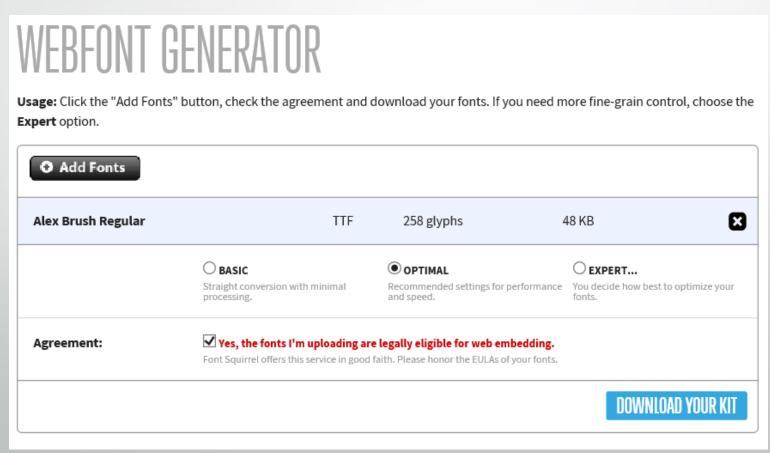
Pour trouver des polices librement téléchargeables et utilisables, je vous conseille d'aller faire un tour sur le site Font Squirrell, à l'adresse <a href="http://www.fontsquirrel.com/fonts/">http://www.fontsquirrel.com/fonts/</a>, où plusieurs milliers de polices libres de droits vous attendent

Choisissez un filtre et un style dans la partie droite du site. Les polices correspondantes sont affichées dans la partie centrale. Cliquez sur le lien Download correspondant à la police de votre choix et procédez comme indiqué à la section précédente. A titre d'exercice, téléchargez la police script **Alex Brush**.



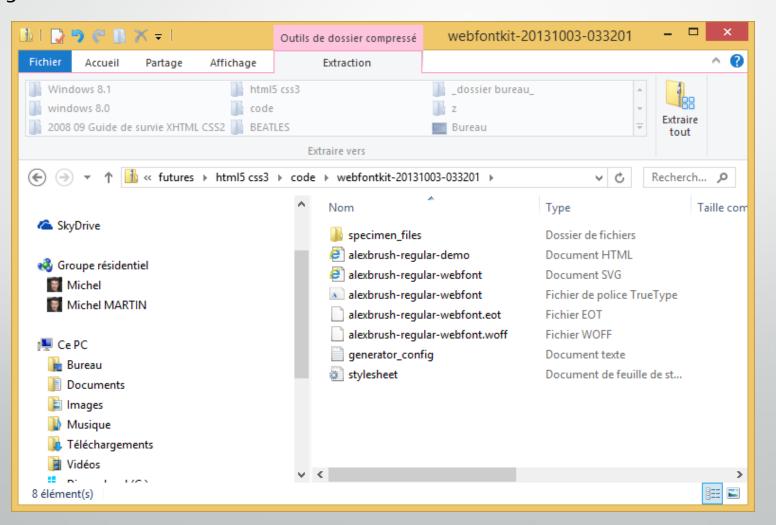
# Où trouver des polices exotiques?

Si vous voulez assurer une compatibilité maximale avec les différents navigateurs disponibles sur le marché, le mieux est de proposer les versions TTF, EOT, SVG et WOFF de vos polices. Là encore, le site Font Squirrell est digne d'intérêt. Cliquez sur **WEBFONT GENERATOR** dans le menu, uploadez une police, générez et téléchargez le kit correspondant en cliquant sur **DOWNLOAD YOUR KIT** :



# Où trouver des polices exotiques?

Le kit téléchargé contient tous les formats nécessaires pour assurer une bonne compatibilité dans les navigateurs.



#### Exercice

Ecrivez le code nécessaire pour interfacer la police alexbrush regular webfont sur tous les navigateurs.

#### Solution

```
<!doctype html>
                                                                                  @font-face
<html>
  <head>
                                                                                    font-family: Alexbrush;
    <meta charset=UTF-8" />
                                                                                    src: url('alexbrush-regular-webfont.eot');
   <title>Utilisation de la police serveur Alexbrush</title>
                                                                                    src: url('alexbrush-regular-webfont.ttf');
    <link rel="stylesheet" href="alexbrush.css">
                                                                                    src: url('alexbrush-regular-webfont.woff');
  </head>
  <body>
    <h1>Test d'utilisation de la police Alexbrush</h1>
                                                                                  h1
 </body>
</html>
                                                                                    font-family: 'Alexbrush';
                                http://localhost/alexbrush.htm
                       Utilisation de la police serve... ×
```

Test d'utilisation de la police Alexbrush

# Couleur et image d'arrière-plan

La couleur d'arrière-plan d'un élément est définie par la propriété <u>background-color</u>. Voici sa syntaxe :

```
background-color: couleur;
```

La valeur <u>couleur</u> peut être spécifiée :

- "En dur", c'est-à-dire sous la forme d'un nom normalisé : <u>yellow</u> ou <u>skyblue</u> par exemple.
- À l'aide d'un code hexadécimal sur 6 digits : <u>#RRVVBB</u> (où RR, VV et BB sont les composantes rouge, vert et bleu de la couleur, codées en hexadécimal entre 00 et FF).
- À l'aide de la fonction  $\underline{RGB(R,V,B)}$ , où  $\underline{R}$ ,  $\underline{V}$  et  $\underline{B}$  sont des valeurs comprises entre  $\underline{0}$  et  $\underline{255}$ .

L'image d'arrière-plan d'un élément est définie par la propriété <u>background-image</u>. Voici sa syntaxe :

```
background-image: url(image);
```

Où image est l'adresse URL de l'image. Par exemple img/fond.jpg, ou encore

http://monsite.com/image/fond.png.

Il n'est pas recommandé d'utiliser des guillemets pour délimiter l'adresse URL de l'image d'arrière-plan car ils peuvent provoquer des dysfonctionnements dans les navigateurs anciens.

# Couleur et image d'arrière-plan

Une image d'arrière-plan peut être répétée horizontalement et/ou verticalement. Pour cela, on donne la valeur <u>repeat</u> (répétition horizontale et verticale), <u>repeat-x</u> (répétition horizontale seulement) ou <u>repeat-y</u> (répétition verticale seulement), à la propriété CSS <u>background-repeat</u>.

La position d'une image d'arrière-plan peut être définie horizontalement et/verticalement : affectez la position horizontale (<u>left</u>, <u>center</u> ou <u>right</u>) et éventuellement la position verticale (<u>top</u>, <u>center</u> ou <u>bottom</u>) à la propriété CSS <u>background-position</u>.

# Couleur et image d'arrière-plan

Dans cet exemple (code005.htm), du texte est affiché au-dessus d'une image en arrière-plan. Nous modifions tour à tour la couleur d'arrière-plan d'un lien hypertexte, d'un paragraphe, puis d'un élément <u>div</u>. Les styles correspondants sont directement définis dans les éléments concernés :

```
couleur jaune
<!DOCTYPE html>
                                                             et celle-ci un arrière-plan bleu
<html>
 <head>
                                                             ciel.
   <meta charset="UTF-8" />
   <title>Couleur d'arrière-plan</title>
 </head>
<body>
 <div style="background-image: url(chat.jpg); font-size: 40px; color: black;">
   La couleur d'arrière-plan de ce texte n'est pas définie. Par contre, <a href="#"</p>
style="background-color: red">ce lien a un arrière-plan de couleur rouge</a>.
   Cette ligne a un arrière-plan de couleur jaune
   <div style="background-color: skyblue;">et celle-ci un arrière-plan bleu ciel.</div>
   </div>
 </div>
</body>
</html>
```

La couleur d'arrière-plan de ce texte n'est pas définie. Par contre,

Cette ligne a un arrière-plan de

couleur rouge

### Gradient vertical

Pour définir un gradient vertical, vous utiliserez ces propriétés CSS:

```
-webkit-linear-gradient(p1, c1, c2);
-ms-linear-gradient(p1, c1, c2) no-repeat;
-moz-linear-gradient(p1, c1, c2) no-repeat;
```

#### Où:

- <u>p1</u> représente les coordonnées du point de départ du gradient : <u>top</u>, <u>left</u>, <u>right</u> et/ou <u>down</u> sur les navigateurs IE et Firefox ; pourcentage horizontal et pourcentage vertical sur les navigateurs Webkit.
- <u>c1</u> est la couleur de départ du gradient.
- <u>c2</u> est la couleur de fin du gradient.

Cette solution n'est pas encore implémentée sur tous les navigateurs et il est nécessaire d'utiliser des préfixes pour s'adresser à chaque navigateur : -webkit pour Chrome, -ms pour IE, -moz pour Firefox.

## Exercice

Définissez un gradient vertical de la couleur rouge à la couleur bleue.

# Solution (codeoo6.htm)

<!doctype html>

<html>

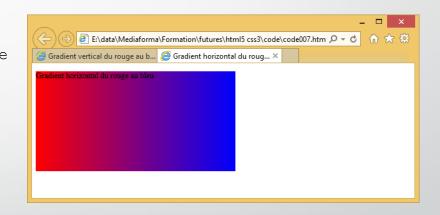
```
<head>
      <meta charset=UTF-8" />
      <title>Gradient vertical du rouge au bleu</title>
                                                                                       Code HTML5
      <link rel="stylesheet" href="gradient-vertical.css">
    </head>
    <body>
      Gradient vertical du rouge au bleu
    </body>
  </html>
                                                                            E:\data\Mediaforma\Formation\futures\html5 css3\code\code006.htm
                           .gradient-vertical
Code CSS<sub>3</sub>
                             width: 400px;
                             height: 200px;
                             background: -ms-linear-gradient(top, red, blue);
                             background: -moz-linear-gradient(top, red, blue);
                             background: -webkit-linear-gradient(top, red, blue);
```

## Exercice

Définissez un gradient horizontal de la couleur rouge à la couleur bleue.

# Solution (codeoo7.htm)

#### Code HTML5



Code CSS<sub>3</sub>

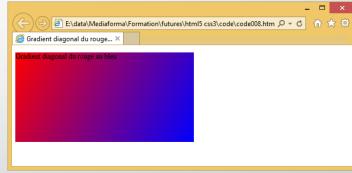
```
.gradient-horizontal
{
  width:400px;
  height: 200px;
  background: -ms-linear-gradient(left, red, blue);
  background: -moz-linear-gradient(left, red, blue);
  background: -webkit-linear-gradient(left, red, blue);
}
```

## Exercice

Définissez un gradient diagonal de la couleur rouge à la couleur bleue.

# Solution (codeoo8.htm)

#### Code HTML5



Code CSS<sub>3</sub>

```
.gradient-diagonal
{
  width:400px;
  height: 200px;
  background: -moz-linear-gradient(top left, red, blue);
  background: -ms-linear-gradient(top left, red, blue);
  background: -webkit-linear-gradient(top left, red, blue);
}
```

# Plusieurs couleurs dans un gradient linéaire

Vous savez définir un gradient linéaire à deux couleurs. Il est possible de lui adjoindre des couleurs intermédiaires, grâce à l'ajout de quelques paramètres dans les propriétés <u>-webkit-linear-gradient</u>, <u>-ms-linear-gradient</u> et <u>-moz-linear-gradient</u>.

# Plusieurs couleurs dans un gradient linéaire

#### **Navigateurs Gecko et IE**

```
-moz-linear-gradient(p1, c1 p1, c2 p2, ..., cN);
-ms-linear-gradient(p1, c1 p1, c2 p2, ..., cN);
-webkit-linear-gradient(p1, c1 p1, c2 p2, ..., cN);
```

#### Où:

- <u>p1</u> à <u>pN-1</u> sont les coordonnées des différentes transitions de couleurs, exprimées en pourcentages.
- <u>c1</u> à <u>cN</u> sont les différentes couleurs du gradient.

### Exercice

Définissez un gradient linéaire horizontal qui passe du rouge au bleu au jaune et au blanc en répartissant de façon égale les différentes transitions. Ce gradient doit être compatible Gecko, IE et Mozilla.

## Solution (codeoog.htm)

Deux points intermédiaires doivent être définis : un à 33 % et un à 66 % de la zone cible.

#### Code HTML5

```
Gradient horizontal multiple,
du rouge au blanc, couleurs
intermédiaires bleu et jaune
```

Code CSS<sub>3</sub>

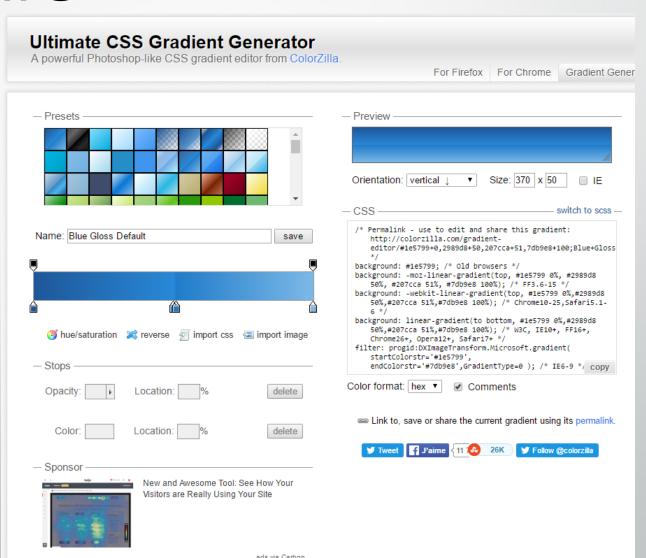
```
.gradient-horizontal-multiple
{
  width:400px;
  height: 200px;
  background: -moz-linear-gradient(left, red 0%, blue 33%, yellow 66%, white);
  background: -ms-linear-gradient(left, red 0%, blue 33%, yellow 66%, white);
  background: -webkit-linear-gradient(left, red 0%, blue 33%, yellow 66%, white);
}
```

# Plusieurs couleurs dans un gradient linéaire

Pour créer vos gradients plus aisément, vous pouvez vous rendre à la page suivante :

http://www.colorzilla.com/gradient-editor/

Quelques réglages élémentaires permettent d'obtenir le code CSS3 correspondant.



## Gradient radial

Les gradients linéaires ne sont pas les seuls utilisables : on peut également définir des gradients radiaux, qui fixent un dégradé de couleurs d'une zone circulaire vers le reste de l'élément ciblé. Les propriétés à utiliser sont les suivantes :

- <u>-webkit-radial-gradient</u> sur les navigateurs Webkit (Safari, Chrome);
- <u>-ms-radial-gradient</u> sur les navigateurs Internet Explorer ;
- <u>-moz-radial-gradient</u> sur les navigateurs Gecko (Mozilla Firefox).

### Gradient radial

```
background: -webkit-radial-gradient(p1, forme taille, c1, c2);
background: -moz-radial-gradient(p1, forme taille, c1, c2);
background: -ms-radial-gradient(p1, forme taille, c1, c2);
```

#### Où:

- <u>p1</u> représente les coordonnées du point intérieur du gradient.
- <u>forme</u> représente la forme du gradient : <u>circle</u> (cercle) ou <u>ellipse</u> (ellipse).
- <u>taille</u> est la taille du gradient. Ce paramètre peut prendre d'une des valeurs suivantes :
  - <u>closest-side</u> (ou <u>contain</u>) pour que le gradient occupe toute la taille du conteneur.
- <u>closest-corner</u> pour que le gradient occupe tout l'espace disponible jusqu'au coin le plus proche du conteneur.
- <u>farthest-side</u> (ou <u>cover</u>) pour que le gradient occupe tout l'espace disponible jusqu'au coin le plus éloigné du conteneur.
- <u>c1</u> est la couleur intérieure du gradient.
- <u>c2</u> est la couleur extérieure du gradient.

## Exercice

Définissez un gradient vertical du rouge au bleu, compatible Mozilla, Gecko et IE.

## Solution (codeo10.htm)

```
<!doctype html>
<html>
  <head>
    <meta charset=UTF-8" />
    <title>Gradient radial</title>
    <link rel="stylesheet" href="gradient-radial.css">
                                                                             Code HTML5
  </head>
  <body>
    Gradient radial
                                                                E:\data\Mediaforma\Formation\futures\html5 css3\code\code010.htm P → C
  </body>
                                                            Gradient radial
</ht.ml>
                        .gradient-radial
                          width:400px;
                          height: 200px;
                          background: -webkit-radial-gradient(50% 50%, circle closest-side, red, blue);
 Code CSS<sub>3</sub>
                          background: -ms-radial-gradient(50% 50%, circle closest-side, red, blue);
                          background: -moz-radial-gradient (50% 50%, circle closest-side, red, blue);
```

# Plusieurs couleurs dans un gradient radial

Tout comme pour les gradients linéaires, il est possible de définir plusieurs couleurs dans un gradient radial.

Il suffit de préciser les différentes couleurs comme paramètres de la propriété <u>-moz-radial-gradient</u>, <u>-webkit-radial-gradient</u> ou <u>-ms-radial-gradient</u>. Ici, le gradient va du rouge au noir en passant par le jaune et le bleu :

background: -préfixe-radial-gradient(circle, red, yellow, blue, black);

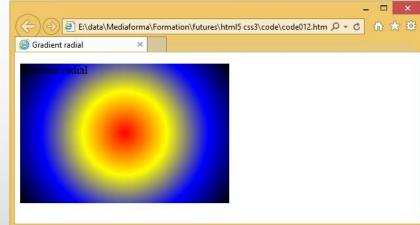
## Exercice

Utilisez les instructions précédentes pour définir un gradient radial multicolore compatible avec IE, Gecko et Mozilla.

## Solution (codeo12.htm)

.gradient-radial

#### Code HTML5



#### Code CSS<sub>3</sub>

```
width:300px;
height: 200px;
background: -webkit-radial-gradient(circle, red, yellow, blue, black);
background: -moz-radial-gradient(circle, red, yellow, blue, black);
background: -ms-radial-gradient(circle, red, yellow, blue, black);
```

En utilisant les fonctions <u>rgba()</u> ou <u>hsla()</u>, on peut appliquer un effet de transparence aux couleurs manipulées dans le code CSS3. Ces fonctions s'appliquent à n'importe quelle propriété dont la valeur est une couleur : <u>color</u>, <u>background-color</u>, <u>border-color</u>, etc.

#### La fonction <a href="mailto:rgba()">rgba()</a> admet quatre paramètres :

```
rgba(red, green, blue, alpha)
```

#### Où:

- <u>red</u>, <u>green</u> et <u>blue</u> sont les composantes rouge, vert et bleu de la couleur. Ces trois informations peuvent prendre 256 valeurs, codées entre 0 et 255.
- <u>alpha</u> est le degré d'opacité de la couleur. Cette information est un nombre décimal codé entre <u>0</u> (transparent) et <u>1</u> (opaque).

La fonction <u>hsla()</u> admet également quatre paramètres :

```
hsla(hue, sat, light, alpha)
```

#### Où:

- <u>hue</u> représente la teinte. Les valeurs autorisées pour ce paramètre sont comprises entre <u>0</u> et <u>360</u>. La valeur <u>0</u> (ou <u>360</u>) correspond au rouge, la valeur <u>120</u> correspond au vert et la valeur <u>250</u> au bleu.
- <u>sat</u> représente la teinte de la couleur, entre <u>0%</u> et <u>100%</u>.
- <u>light</u> représente la luminosité de la couleur, entre <u>0%</u> (noir) et <u>100%</u> (blanc).
- <u>alpha</u> est le degré d'opacité de la couleur. Cette information est un nombre décimal codé entre <u>0</u> (transparent) et <u>1</u> (opaque).

À titre d'exemple, nous allons définir plusieurs éléments <u>div</u> contenant du texte et dont la couleur d'arrièreplan est plus ou moins transparente. Voici le code HTML5 utilisé :

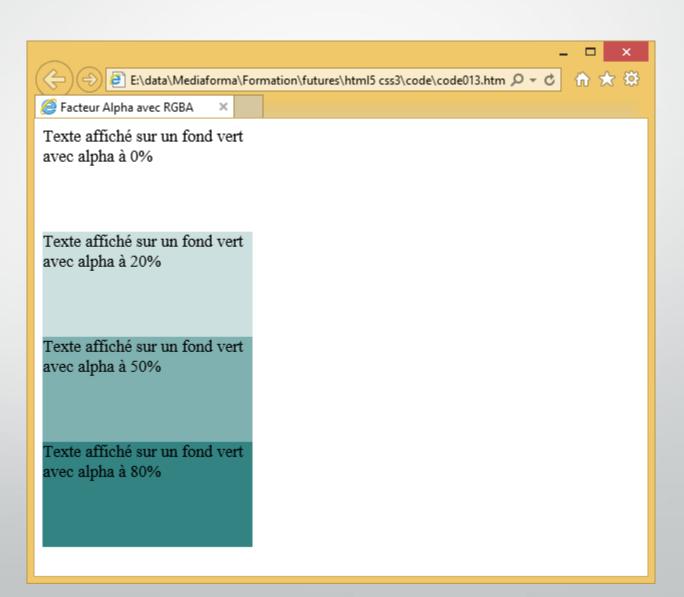
Ce code se contente d'afficher quatre blocs <a href="editor"><u>div></u> de classe <u>alpha00</u>, <u>alpha20</u>, <u>alpha50</u> et <u>alpha80</u>.

Voici le code CSS associé :

```
div
 width: 200px;
 height:100px;
.alpha00
 background-color: rgba(0, 100, 100, 0);
.alpha20
 background-color: rgba(0, 100, 100, .2);
.alpha50
 background-color: rgba(0, 100, 100, .5);
.alpha80
 background-color: rgba(0, 100, 100, .8);
```

Saisissez ce code dans le fichier alpha.css

Voici ce que vous devriez obtenir :



La fonction <u>hsla()</u> est une alternative à la fonction <u>rgba()</u>. Le fichier *alpha.css* précédent pourrait être réécrit comme suit avec la fonction <u>hsla()</u>:

```
div
 width: 200px;
  height:100px;
.alpha00
 background-color: hsla(120, 94%, 18%, 0);
.alpha20
 background-color: hsla(120, 94%, 18%, .2);
.alpha50
 background-color: hsla(120, 94%, 18%, 0.4);
.alpha80
 background-color: hsla(120, 94%, 18%, .8);
```

Pour choisir une couleur plus facilement, vous pouvez vous rendre sur les pages <a href="https://www.colorschemer.com/online.html">www.colorschemer.com/online.html</a> (RBG) ou <a href="https://www.workwithcolor.com/hsl-color-schemer-01.htm">www.workwithcolor.com/hsl-color-schemer-01.htm</a> (HSL).

### Positionner en flux

Lorsque la position d'un élément n'est pas spécifiée, il se place d'après l'ordre dans lequel il apparaît dans le code, en ligne ou en bloc selon son rendu.

Par exemple, l'élément <u>a</u> va tout naturellement se placer à la suite du texte qui le précède.

Cliquez <a href="http://www.google.com"> pour accéder au site de Google</a>

Par contre, l'élément de rendu block <h2> va s'afficher sur la ligne suivante :

Ce texte est suivi<h2> par un titre de style H2</h2>

Ce texte est suivi

par un titre de style H2

## Positionner en flux

Il en va de même des objets placés à l'intérieur d'un conteneur. Ici, par exemple, deux images, un texte, puis deux autres images sont affichés à l'intérieur d'un élément <u>div</u>. Ces éléments se succèdent en ligne. Le passage à la ligne après le mot "templum" a été provoqué par un manque de place dans le navigateur

```
<img src="fgauche.png" style="width:40px">
<img src="fhaut.png" style="width:40px">

Iam summus Pater architectus Deus hanc quam videmus mundanam domum, divinitatis templum augustissimum, archanae legibus sapientiae fabrefecerat.

<img src="fbas.png" style="width:40px">
<img src="fdroite.png" style="width:40px">
```

Iam summus Pater architectus Deus hanc quam videmus mundanam domum, divinitatis templum

augustissimum, archanae legibus sapientiae fabrefecerat.

# Mise en page avec la propriété display

La propriété <u>display</u> permet de spécifier la façon dont un élément est rendu. Elle peut prendre de nombreuses valeurs. Voici les plus courantes :

- <u>none</u> : l'élément n'est pas rendu.
- <u>inline</u> : l'élément est rendu sur la même ligne que son parent.
- block : l'élément est rendu en mode bloc.
- <u>inline-block</u>: l'élément se comporte comme une "boîte en ligne": il peut être dimensionné (<u>width, height</u>) et avoir des marges (<u>margin, padding</u>), mais il se place également sur la ligne de texte de son parent. Vous utiliserez <u>-moz-inline-box</u> pour assurer la compatibilité avec Firefox 2 et supérieur.
- <u>list-item</u>: l'élément est affiché sous la forme d'une liste.
- <u>table</u>, <u>inline-table</u>, <u>table-header-group</u>, <u>table-footer-group</u>, <u>table-row</u>, <u>table-row-group</u>, <u>table-column</u>, <u>table-column-group</u>, <u>table-cell</u>, <u>table-caption</u>: l'élément se comporte comme les composants d'un tableau (attention, ces valeurs sont incompatibles avec Internet Explorer).

```
< ht.ml>
  <head>
                                                                   Premier exemple (codeo14.htm)
     <meta charset=UTF-8" />
    <title>La propriété display</title>
    <style>
                                                                       Saisissez ce code.
       .un
                                                                       Qu'obtenez-vous?
         border: 1px solid #ff0000;
         margin: 10px;
         padding: 10px;
                                                                                                         D + c A ★ A
                                                    E:\data\Mediaforma\Formation\futures\html5 css3\code\code014.htm
        .deux
                                               La propriété display
                                                 ce texte est affiché dans une balise div de type "bloc". Ce texte est affiché dans une balise div dont la propriété display vaut inline
         border: 1px solid #00ff00;
         margin: 2px;
                                                  ce texte est affiché dans une balise div de type "bloc"
         padding: 2px;
       .trois
         border: 1px solid #0000ff;
         margin: 10px;
         padding: 10px;
    </style>
  </head>
  <body>
    <div class="un" style="display: block;">
       ce texte est affiché dans une balise div de type "bloc".
       <div class="deux" style="display: inline">Ce texte est affiché dans une balise div dont la
propriété display vaut inline</div>
       <div class="trois">ce texte est affiché dans une balise div de type "bloc"</div>
    </div>
  </body>
</html>
```

<!DOCTYPE html>

```
<!DOCTYPE html>
<html>
  <head>
                                                                                     Deuxième exemple (codeo15.htm)
    <meta charset=UTF-8" />
    <title>Des tableaux en HTML5</title>
    <style>
       .un
                                                                                                Saisissez ce code
         border: 1px solid #ff0000;
                                                                                                Qu'obtenez-vous?
         margin: 10px;
         padding: 10px;
       .deux
                                                                                                                   ひ → ひ 金 ☆ ☆
                                                      E:\data\Mediaforma\Formation\futures\html5 css3\code\code015.htm
                                                 @ Des tableaux en HTML5
         border: 1px solid #00ff00;
         margin: 2px;
                                                  Ce texte est affiché dans une balise div dont la
         padding: 2px;
                                                  propriété display vaut table.
                                                                                                     ce texte est affiché dans une autre
                                                                               ce texte est affiché dans une balise
                                                  Ce texte est affiché dans une balise div dont la
                                                  propriété display vaut table-row
                                                                               div de type "table-cell"
                                                                                                     balise div de type "table-cell"
       .trois
                                                                               Une première cellule
                                                                                                     Et une deuxième
                                                  On recommence avec une nouvelle ligne
         border: 1px solid #0000ff;
         margin: 10px;
         padding: 10px;
    </style>
  </head>
  <body>
    <div class="un" style="display: table;">
      Ce texte est affiché dans une balise div dont la propriété display vaut table.
      <div class="deux" style="display: table-row">Ce texte est affiché dans une balise div dont la propriété display
vaut table-row
       <div class="trois" style="display: table-cell">ce texte est affiché dans une balise div de type "table-
cell"</div>
      <div class="trois" style="display: table-cell">ce texte est affiché dans une autre balise div de type "table-
cell"</div>
    </div>
    <div class="deux" style="display: table-row">On recommence avec une nouvelle ligne
       <div class="trois" style="display: table-cell">Une première cellule</div>
       <div class="trois" style="display: table-cell">Et une deuxième</div>
      </div>
    </div>
  </body>
</html>
```

### Rendu inline-block

Le rendu <u>inline-block</u> profite des propriétés propres aux rendus <u>inline</u> et <u>block</u>. En effet, il permet aux éléments concernés de s'afficher côte à côte (rendu <u>inline</u>) et d'être dimensionnés et de recevoir des marges externes (rendu <u>block</u>).

Pour définir des blocs de texte de largeur fixe à l'aide de simples éléments <u>p</u>, rien de tel qu'un affichage <u>inline-block</u>.

### Rendu inline-block

</html>

#### codeo16.htm

```
<!DOCTYPE html>
<html>
                                                                                                                                                               _ 🗆 ×
   <head>
                                                                                                                                                          E:\data\Mediaforma\Formation\futures\html5 css3\code\code016.htm
        <meta charset=UTF-8" />
                                                                                     @ Des tableaux en HTML5 @ Un rendu inline-block
       <title>Un rendu inline-block</title>
        <style>
                                                                                         Iam summus Pater
                                                                                                                     Iam summus Pater
                                                                                                                                                 Iam summus Pater
                                                                                         architectus Deus hand
                                                                                                                     architectus Deus han
                                                                                                                                                 architectus Deus hand
                                                                                         quam videmus
                                                                                                                     quam videmus
                                                                                                                                                 quam videmus
                                                                                         mundanam domum,
                                                                                                                     mundanam domum
                                                                                                                                                 mundanam domum
                                                                                         divinitatis templum
                                                                                                                     divinitatis templum
                                                                                                                                                 divinitatis templum
               display:inline-block;
                                                                                         augustissimum
                                                                                                                     anoustissimum
                                                                                                                                                 anoustissimum
                                                                                         archanae legibus
                                                                                                                      archanae legibus
                                                                                                                                                 archanae legibus
                                                                                         sapientiae fabrefecerat.
                                                                                                                      sapientiae fabrefecerat.
                text-align:left;
                                                                                                                                                 sanientiae fabrefecerat
                                                                                         Supercaelestem
                                                                                                                      Supercaelestem
                                                                                                                                                 Supercaelestem
               background: #dce;
                                                                                                                     regionem mentibus
                                                                                                                                                 regionem mentibus
                                                                                         regionem mentibus
                                                                                         decorarat; aethereos
                                                                                                                     decorarat; aethereos
                                                                                                                                                 decorarat; aethereos
               width:150px;
                                                                                         globos aeternis animis
                                                                                                                      globos aeternis animis
                                                                                                                                                 globos aeternis animis
                                                                                         vegetarat:
                                                                                                                     vegetarat:
                                                                                                                                                 vegetarat:
               padding:20px;
                                                                                                                     excrementarias ac
                                                                                         excrementarias ac
                                                                                                                                                 excrementarias ac
                                                                                         feculentas inferioris
                                                                                                                     feculentas inferioris
                                                                                                                                                 feculentas inferioris
               margin: 20px;
                                                                                         mundi partes
                                                                                                                                                 mundi partes
                                                                                         omnigena animalium
                                                                                                                     omnicena animalium
                                                                                                                                                 omnigena animalium
                                                                                         turba complerat.
                                                                                                                     turba complerat.
                                                                                                                                                 turba complerat.
       </style>
   </head>
   <body>
```

Saisissez ce code. Qu'obtenez-vous?

Iam summus Pater architectus Deus hanc quam videmus mundanam domum, divinitatis templum augustissimum, archanae legibus sapientiae fabrefecerat. Supercaelestem regionem mentibus decorarat; aethereos globos aeternis animis vegetarat; excrementarias ac feculentas inferioris mundi partes omnigena animalium turba complerat.

Iam summus Pater architectus Deus hanc quam videmus mundanam domum, divinitatis templum augustissimum, archanae legibus sapientiae fabrefecerat. Supercaelestem regionem mentibus decorarat; aethereos globos aeternis animis vegetarat; excrementarias ac feculentas inferioris mundi partes omnigena animalium turba complerat.

Iam summus Pater architectus Deus hanc quam videmus mundanam domum, divinitatis templum augustissimum,
archanae legibus sapientiae fabrefecerat. Supercaelestem regionem mentibus decorarat; aethereos globos aeternis
animis vegetarat; excrementarias ac feculentas inferioris mundi partes omnigena animalium turba complerat. 
</body>

# Design à deux colonnes

Le principe consiste à définir une boîte <u><div></u> flottante (à gauche ou à droite selon les besoins) et une zone non flottante, dont la marge est légèrement supérieure à la largeur de l'élément flottant.

Dans cet exemple, nous définissons une "colonne" flottante à gauche de largeur 150 pixels, et une colonne non flottante qui occupe le reste de la page. Cette colonne a une marge gauche de 160 pixels, de façon à se détacher légèrement vers la droite de la colonne gauche.

## Design à deux colonnes

```
<!DOCTYPE html>
<ht.ml>
  <head>
    <meta charset=UTF-8" />
   <title>Un design à deux colonnes</title>
    <style>
      #colonne1
        float: left;
        width: 150px;
        background: #66CCFF;
      #colonne2
        margin-left: 160px;
        background: #FFFFCC;
    </style>
  </head>
  <body>
    <div id="colonne1">
```

```
Iam summus Pater
                      Iam summus Pater architectus Deus hanc quam
architectus Deus hanc
                      videmus mundanam domum, divinitatis templum
quam videmus
                      augustissimum, archanae legibus sapientiae
mundanam domum,
                      fabrefecerat. Supercaelestem regionem mentibus
divinitatis templum
                      decorarat; aethereos globos aeternis animis
augustissimum.
                      vegetarat: excrementarias ac feculentas inferioris
archanae legibus
                      mundi partes omnigena animalium turba
sapientiae fabrefecerat
                      complerat. Sed. opere consummato, desiderabat
Supercaelestem
                      artifex esse aliquem qui tanti operis rationem
                      perpenderet, pulchritudinem amaret,
regionem mentibus
decorarat; aethereos
                      magnitudinem admiraretur. Idcirco iam rebus
globos aeternis animis
                      omnibus (ut Moses5 Timaeusque6 testantur)
                      absolutis, de producendo homine postremo
excrementarias ac
                      cogitavit. Verum nec erat in archetypis unde
feculentas inferioris
                      novam sobolem effingeret, nec in thesauris quod
mundi partes
                      novo filio hereditarium largiretur, nec in subselliis
omnigena animalium
                      totius orbis, ubi universi contemplator iste sederet.
turba complerat. Sed,
opere consummato,
desiderabat artifex ess
aliquem qui tanti
operis rationem
perpenderet,
pulchritudinem amaret
```

codeo17.htm

Saisissez ce code. Qu'obtenez-vous?

Tam summus Pater architectus Deus hanc quam videmus mundanam domum, divinitatis templum augustissimum, archanae legibus sapientiae fabrefecerat. Supercaelestem regionem mentibus decorarat; aethereos globos aeternis animis vegetarat; excrementarias ac feculentas inferioris mundi partes omnigena animalium turba complerat. Sed, opere consummato, desiderabat artifex esse aliquem qui tanti operis rationem perpenderet, pulchritudinem amaret, magnitudinem admiraretur. Idcirco iam rebus omnibus (ut Moses5 Timaeusque6 testantur) absolutis, de producendo homine postremo cogitavit. Verum nec erat in archetypis unde novam sobolem effingeret, nec in thesauris quod novo filio hereditarium largiretur, nec in subselliis totius orbis, ubi universi contemplator iste sederet.

```
</div>
<div id="colonne2">
```

Iam summus Pater architectus Deus hanc quam videmus mundanam domum, divinitatis templum augustissimum, archanae legibus sapientiae fabrefecerat. Supercaelestem regionem mentibus decorarat; aethereos globos aeternis animis vegetarat; excrementarias ac feculentas inferioris mundi partes omnigena animalium turba complerat. Sed, opere consummato, desiderabat artifex esse aliquem qui tanti operis rationem perpenderet, pulchritudinem amaret, magnitudinem admiraretur. Idcirco iam rebus omnibus (ut Moses5 Timaeusque6 testantur) absolutis, de producendo homine postremo cogitavit. Verum nec erat in archetypis unde novam sobolem effingeret, nec in thesauris quod novo filio hereditarium largiretur, nec in subselliis totius orbis, ubi universi contemplator iste sederet.

```
</div>
</body>
</html>
```

# Design à 3 colonnes dont 2 flottantes

Pour réaliser un design à trois colonnes dont deux sont flottantes, ancrées sur les bords gauche et droit de la page, vous utiliserez trois éléments <u>div</u>:

- un premier de largeur fixe (width), ancré sur le bord gauche de la page (float: left);
- un deuxième de largeur fixe (width), ancré sur le bord droit de la page (float: right);
- un troisième qui occupera librement l'espace situé entre les deux autres.

#### Exercice

Ecrivez le code pour réaliser un design à 3 colonnes :

- Colonne 1 : 150 px flottante à gauche
- Colonne 2 : 300 px flottante à droite
- Colonne 3 : La place restante

#### Solution

codeo18.htm

```
E:\data\Mediaforma\Formation\futures\html5 css3\code\code018.htm
🥰 Un design fluide à 3 colonnes ×
 am summus Pater Iam summus Pater architectus Deus hanc quam videmus Iam summus Pater architectus Deus hanc
architectus Deus hanc mundanam domum, divinitatis templum augustissimum, quam videmus mundanam domum, divinitatis
                      archanae legibus sapientiae fabrefecerat.
                                                                           templum augustissimum, archanae legibus
 nundanam domum, Supercaelestem regionem mentibus decorarat; aethereos sapientiae fabrefecerat. Supercaelestem
divinitatis templum globos aeternis animis vegetarat, excrementarias ac regionem mentibus decorarat, aethereos
                      feculentas inferioris mundi partes omnigena animalium globos aeternis animis vegetarat;
 monstissimum
archanae legibus
                     turba complerat. Sed. opere consummato, desiderabat excrementarias ac feculentas inferioris mundi
 apientiae fabrefecerat. artifex esse aliquem qui tanti operis rationem
                                                                            partes omnigena animalium turba complerat
                     perpenderet, pulchritudinem amaret, magnitudinem
                                                                           Sed, opere consummato, desiderabat artifex
  gionem mentibus
                      admiraretur. Idcirco iam rebus omnibus (ut Moses5
                                                                           esse aliquem qui tanti operis rationem
 ecorarat: aethereos Timaeusque6 testantur) absolutis, de producendo
                                                                           perpenderet, pulchritudinem amaret,
  obos aeternis animis homine postremo cogitavit. Verum nec erat in
                                                                            magnitudinem admiraretur. Idcirco iam rebus
                      archetypis unde novam sobolem effingeret, nec in
                                                                           omnibus (ut Moses5 Timaeusque6 testantur)
                      thesauris quod novo filio hereditarium largiretur, nec in absolutis, de producendo homine postremo
 eculentas inferioris subselliis totius orbis, ubi universi contemplator iste
                                                                           cogitavit. Verum nec erat in archetypis unde
                                                                           novam sobolem effingeret, nec in thesauris
mundi partes
omnigena animalium
                                                                            anod novo filio hereditarium largiretur, nec in
turba complerat. Sed
                                                                            subselliis totius orbis, ubi universi
                                                                            contemplator iste sederet.
 lesiderabat artifex es
aliquem qui tanti
operis rationem
 nagnitudinem
```

Iam summus Pater architectus Deus hanc quam videmus mundanam domum, divinitatis templum augustissimum, archanae legibus sapientiae fabrefecerat. Supercaelestem regionem mentibus decorarat; aethereos globos aeternis animis vegetarat; excrementarias ac feculentas inferioris mundi partes omnigena animalium turba complerat. Sed, opere consummato, desiderabat artifex esse aliquem qui tanti operis rationem perpenderet, pulchritudinem amaret, magnitudinem admiraretur. Idcirco iam rebus omnibus (ut Moses5 Timaeusque6 testantur) absolutis, de producendo homine postremo cogitavit. Verum nec erat in archetypis unde novam sobolem effingeret, nec in thesauris quod novo filio hereditarium largiretur, nec in subselliis totius orbis, ubi universi contemplator iste sederet.

```
</div>
<div id="colonne2">
```

Iam summus Pater architectus Deus hanc quam videmus mundanam domum, divinitatis templum augustissimum, archanae legibus sapientiae fabrefecerat. Supercaelestem regionem mentibus decorarat; aethereos globos aeternis animis vegetarat; excrementarias ac feculentas inferioris mundi partes omnigena animalium turba complerat. Sed, opere consummato, desiderabat artifex esse aliquem qui tanti operis rationem perpenderet, pulchritudinem amaret, magnitudinem admiraretur. Idcirco iam rebus omnibus (ut Moses5 Timaeusque6 testantur) absolutis, de producendo homine postremo cogitavit. Verum nec erat in archetypis unde novam sobolem effingeret, nec in thesauris quod novo filio hereditarium largiretur, nec in subselliis totius orbis, ubi universi contemplator iste sederet.

```
</div>
<div id="colonne3">
```

Iam summus Pater architectus Deus hanc quam videmus mundanam domum, divinitatis templum augustissimum, archanae legibus sapientiae fabrefecerat. Supercaelestem regionem mentibus decorarat; aethereos globos aeternis animis vegetarat; excrementarias ac feculentas inferioris mundi partes omnigena animalium turba complerat. Sed, opere consummato, desiderabat artifex esse aliquem qui tanti operis rationem perpenderet, pulchritudinem amaret, magnitudinem admiraretur. Idcirco iam rebus omnibus (ut Moses5 Timaeusque6 testantur) absolutis, de producendo homine postremo cogitavit. Verum nec erat in archetypis unde novam sobolem effingeret, nec in thesauris quod novo filio hereditarium largiretur, nec in subselliis totius orbis, ubi universi contemplator iste sederet.

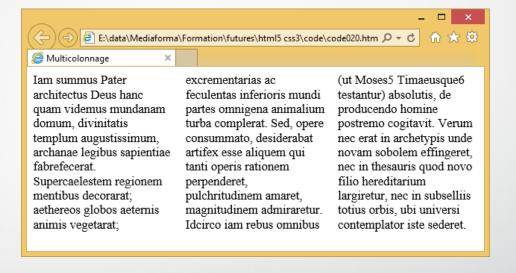
```
</div>
</body>
</html>
```

codeo20.htm

# Multicolonnage CSS3

Pour définir un multicolonnage CSS3, il suffit de fixer la largeur des colonnes et l'espace entre deux colonnes. Voici le code HTML5/CSS3 utilisé :

```
<!DOCTYPE html>
< html>
  <head>
    <meta charset=UTF-8" />
    <title>Multicolonnage</title>
    <style>
      .multi-col
        column-width: 150px;
        -moz-column-width: 150px;
        -webkit-column-width: 150px;
        -moz-column-gap: 20px;
        -webkit-column-gap: 20px;
    </style>
  </head>
 <body>
    <div class="multi-col">
```



Tam summus Pater architectus Deus hanc quam videmus mundanam domum, divinitatis templum augustissimum, archanae legibus sapientiae fabrefecerat. Supercaelestem regionem mentibus decorarat; aethereos globos aeternis animis vegetarat; excrementarias ac feculentas inferioris mundi partes omnigena animalium turba complerat. Sed, opere consummato, desiderabat artifex esse aliquem qui tanti operis rationem perpenderet, pulchritudinem amaret, magnitudinem admiraretur. Idcirco iam rebus omnibus (ut Moses5 Timaeusque6 testantur) absolutis, de producendo homine postremo cogitavit. Verum nec erat in archetypis unde novam sobolem effingeret, nec in thesauris quod novo filio hereditarium largiretur, nec in subselliis totius orbis, ubi universi contemplator iste sederet.

```
</div>
</body>
</html>
```

# Multicolonnage CSS3

Si le nombre de colonnes doit rester fixe, utilisez les propriétés CSS3 <u>column-count</u>, <u>-moz-column-count</u> et <u>-webkit-column-count</u>. Par exemple, pour définir trois colonnes, quelle que soit la largeur de la fenêtre, ajoutez les deux lignes suivantes dans la classe <u>multi-col</u>:

```
column-count: 3;
-moz-column-count: 3;
-webkit-column-count: 3;
```

#### Exercice

Modifiez le code précédent pour que le nombre de colonne soit fixe et égal à 4.

## Solution

#### Modifiez le style .multi-col comme ceci :

```
<style>
.multi-col
{
    column-width: 150px;
    -moz-column-width: 150px;
    -webkit-column-width: 150px;
    -moz-column-gap: 20px;
    -webkit-column-gap: 20px;
    column-count: 4;
    -moz-column-count: 4;
    -webkit-column-count: 4;
}
</style>
```

# Multicolonnage CSS3

Si nécessaire, vous pouvez également insérer un trait séparateur entre les colonnes en utilisant les propriétés CSS3 <u>column-rule</u>, <u>-moz-column-rule</u> et <u>-webkit-column-rule</u> dont voici la syntaxe :

```
column-rule: largeur style couleur;
-moz-column-rule: largeur style couleur;
-webkit-column-rule: largeur style couleur;
```

#### Où:

- <u>largeur</u> est la largeur du trait en pixels.
- <u>style</u> est le style du trait. Il peut prendre l'une des valeurs suivantes : <u>dotted</u>, <u>dashed</u>, <u>solid</u>, <u>insert</u>, <u>double</u>, <u>groove</u>, <u>ridge</u> ou <u>outset</u>.
- <u>couleur</u> est la couleur du trait.

Par exemple, pour définir un trait séparateur continu noir et d'épaisseur 1 pixel, vous utiliserez les deux propriétés suivantes :

```
column-rule: 1px solid black;
-moz-column-rule: 1px solid black;
-webkit-column-rule: 1px solid black;
```

Entraînez-vous à utiliser ces instructions dans le code précédent.

# Multicolonnage CSS3

Pour terminer, sachez qu'il est possible de définir la couleur d'arrière-plan du texte avec la propriété <u>background</u> et l'alignement du texte dans les colonnes avec la propriété <u>text-align</u>. Par exemple, pour affecter un arrière-plan gris et pour justifier le texte dans les colonnes, servez-vous des propriétés suivantes :

```
background:#ccc;
text-align:justify;
```

## Exercice

En utilisant le code HTML précédent, définissez un multicolonnage fixe sur trois colonnes séparées entre elles par un trait de 1 pixel, justifiées et d'arrière-plan #ccc.

### Solution

La classe multi-col doit être modifiée comme ceci :

```
.multi-col
   column-width: 150px;
   -moz-column-width: 150px;
   -webkit-column-width: 150px;
   -moz-column-gap: 20px;
   -webkit-column-gap: 20px;
   column-rule: 1px solid black;
   -moz-column-rule: 1px solid black;
   -webkit-column-rule: 1px solid black;
   column-count: 3;
   -moz-column-count: 3;
   -webkit-column-count: 3;
  background: #ccc;
   text-align: justify;
          E:\data\Mediaforma\Formation\futures\html5 css3\code\code023.htm
    Iam summus Pater architectus Deus mundi partes omnigena animalium turba postremo cogitavit. Verum nec erat in
            videmus mundanam complerat. Sed, opere consummato, archetypis unde novam sobolem
            divinitatis templum desiderabat artifex esse aliquem qui effingeret, nec in thesauris quod novo
              archanae legibus tanti operis rationem perpenderet, filio hereditarium largiretur, nec ii
    sapientiae fabrefecerat. Supercaelestem | pulchritudinem amaret, magnitudinem
    globos aeternis animis vegetarat; (ut Moses5 Timaeusque6 testantur)
    excrementarias ac feculentas inferioris absolutis, de producendo homine
```

# Affichage fluide, Grid Layout et FlexBox

# Principe d'un affichage fluide

On parle d'**affichage fluide** lorsque le contenu et la disposition sont élastiques. Pour arriver à ce résultat, on utilise des blocs dont la largeur est exprimée en pourcentages de la largeur totale de la fenêtre (ordinateur) ou de l'écran (mobile).

#### **Exercice**

Définissez quelques lignes de HTML5 et de CSS3 pour obtenir la mise en page suivante :



Voici la largeur des différents blocs :

• Couleur jaune : 20%;

Couleur bleue: 80%;

Couleurs rouget et verte : 100%.

### **Solution**

Le code HTML5 est très simple : il met en place quatre zones distinctes : header, article, aside et footer :

```
<!DOCTYPE html>
<html>
   <head>
    <meta charset="UTF-8" />
    <title>Un document HTML5 avec en-tête, bas de page, zone de contenu et zone annexe</title>
    <link rel="stylesheet" href="styles004.css">
   </head>
   <body>
    <header>
      <h1>En-tête du document</h1>
      Texte de l'en-tête
    </header>
    <article>
      <h2>Premier article</h2>
        Texte du premier article
    </article>
    <aside>
      Texte affiché dans la partie droite de la page avec la balise <aside&gt; <br>
    </aside>
    <footer>
      Copyright et e-mail du Webmaster
    </footer>
   </body>
 </ht.ml>
```

Le code CSS3 est également très simple. Il indique que les quatre zones seront affichées en mode block, puis il définit les caractéristiques de chaque zone :

```
header, article, footer, aside {
 display: block;
header {
 background-color: red;
article {
 height: 300px;
 float: left;
 background-color: #66FFFF;
 width:80%;
aside {
 height: 300px;
 float:right;
 width:20%;
 background-color:yellow;
footer {
clear: both;
 background-color: #99FF66;
```

Vérifiez que l'affichage s'adapte bien à l'espace horizontal disponible en redimensionnant la fenêtre du navigateur.

### Remarque

Il est possible de définir les marges extérieures (margin) et intérieures (padding) d'un élément HTML5 quelconque (img, div, nav, etc.) sous la forme d'un pourcentage de leur conteneur pour leur conférer un caractère responsive.

### Principe d'un affichage Grid Layout

Le principe de fonctionnement du Grid layout consiste à diviser l'écran en plusieurs zones, comme dans un tableau, à ceci près que le découpage se fait en CSS et non en HTML. Voici un exemple de découpage Grid layout (propre à Internet Explorer et Microsoft Edge) :



Par exemple, pour découper la page en quatre zones (logo, menu, sidebar et contenu) comme dans la figure précédente, vous utiliserez ce code :

```
<!DOCTYPE html>
<html>
   <head>
     <meta charset="UTF-8" />
    <title>Gestion des débordements</title>
    <style>
      body {
         display: -ms-grid;
         -ms-grid-columns: 200px 400px;
         -ms-grid-rows: 100px 300px;
      header, menu, aside, article { display: block; margin: 0; padding: 0;}
      header { -ms-grid-column: 1; -ms-grid-row: 1; background-color: yellow;}
      menu { -ms-grid-column: 2; -ms-grid-row: 1; background-color: red;}
      aside { -ms-grid-column: 1; -ms-grid-row: 2; background-color: grey; }
      article { -ms-grid-column: 2; -ms-grid-row: 2; background-color: green; }
    </style>
   </head>
   <body>
     <header>
       Logo
     </header>
     <menu>
       Menu
     </menu>
     <aside>
       Sidebar
     </aside>
    <article>
       Contenu
    </article>
  </body>
 </html>
```

# Principe de la grille flexible (flexbox)

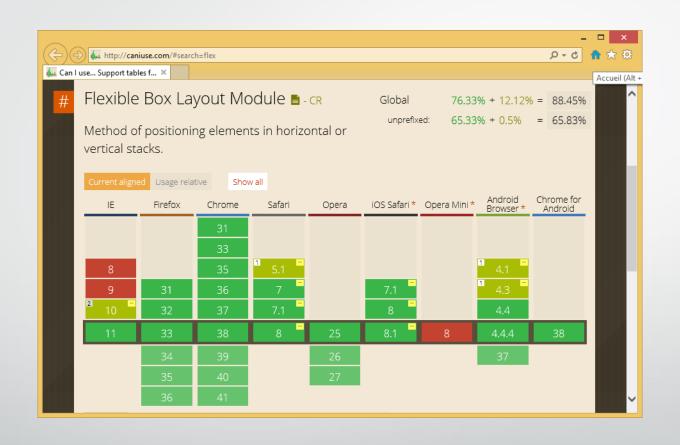
L'affichage Flexbox est un nouveau type de rendu défini dans la propriété CSS display :

```
display: flex;
```

Il permet de gérer la mise en page de plusieurs éléments enfants dans un élément parent. Avec un affichage flexbox, vous pouvez entre autres :

- mettre en place plusieurs éléments sur une ligne sans vous soucier de leur taille et/ou de leurs marges;
- modifier le sens de lecture vertical/horizontal;
- aligner les éléments à gauche, à droite ou au centre du parent ;
- modifier l'ordre d'affichage des éléments enfants.

Le mode d'affichage flex est utilisable sur la plupart des navigateurs récents. Pour vous assurer de sa compatibilité avec votre navigateur, allez sur la page <a href="http://caniuse.com/#search=flex">http://caniuse.com/#search=flex</a> :



### Affichage flexbox – display: flex

Pour mettre en place un affichage flexbox, commencez par définir le mode d'affichage **flex** pour le conteneur :

.conteneur { display: flex; }

### En ligne ou en colonne – flex-flow

Si vous voulez que les enfants s'affichent en ligne, affectez la valeur **row** à la propriété fl**ex- flow** du conteneur :

```
.conteneur {
  display: flex;
  flex-flow: row;
}
```

Si vous voulez que les enfants s'affichent en colonne, affectez la valeur **column** à la propriété **flex-flow** du conteneur :

```
.conteneur {
  display: flex;
  flex-flow: column;
}
```

### Direction de l'affichage – flex-direction

La propriété **flex-direction** définit l'ordre d'affichage des enfants dans le parent. Elle peut prendre les valeurs suivantes :

Valeur de flex-direction	Signification
Row	de gauche à droite
row-reverse	de droite à gauche
Column	de haut en bas
column-reverse	de bas en haut

### Wrapping des enfants – flex-wrap

Par défaut, les enfants essayerons de s'afficher sur une même ligne (ou une même colonne). Vous pouvez modifier ce comportement en agissant sur la propriété flexwrap :

Valeur de flex-wrap	Signification
nowrap	Sur une seule ligne ou une seule colonne
wrap	Sur plusieurs lignes (ou plusieurs colonnes) si
	nécessaire
wrap-reverse	Sur plusieurs lignes (ou plusieurs colonnes) si
	nécessaire, dans l'ordre inverse d'affichage

### Alignement des enfants sur l'axe principal – justify-content

La propriété justify-content définit l'alignement des enfants sur l'axe principal (horizontal ou vertical selon la valeur affectée à la propriété flex-flow). Elle peut prendre les valeurs suivantes :

Valeur de justify-content	Signification
flex-start	Alignement au début du conteneur
flex-end	Alignement à la fin du conteneur
center	Centré dans le conteneur
space-between	Espacement entre les enfants
space-around	Espacement avant, entre et après les
	enfants

### Alignement des enfants sur l'axe secondaire— align-items

La propriété align-items définit l'alignement des enfants sur l'axe secondaire (l'axe perpendiculaire à l'axe principal). Elle peut prendre les valeurs suivantes :

Valeur de align-items	Signification
flex-start	Alignement au début du conteneur
flex-end	Alignement à la fin du conteneur
center	Centré dans le conteneur
stretch	Etiré pour occuper toute la dimension de
	l'axe secondaire
baseline	Aligné sur la baseline

### Alignement des enfants sur l'axe secondaire—align-content

La propriété align-content définit l'alignement des enfants sur l'axe secondaire (l'axe perpendiculaire à l'axe principal). Elle n'a aucun effet lorsque les enfants occupent une seule ligne. Cette propriété peut prendre les valeurs suivantes :

Valeur de align-items	Signification
flex-start	Alignement au début du conteneur
flex-end	Alignement à la fin du conteneur
center	Centré dans le conteneur
stretch	Etiré pour occuper toute la dimension de
	l'axe secondaire
space-between	Espacement entre les enfants
space-around	Espacement avant, entre et après les
	enfants

### Ordre d'affichage d'un enfant - order

Par défaut, les enfants s'affichent dans l'ordre où ils apparaissent dans le code.

Cependant, vous pouvez modifier cet ordre en définissant la propriété **order** dans un des enfants :

#enfant3 { order: 5; }

### Taille des enfants – flex-grow

La propriété **flex-grow** définit la taille des éléments affichés sur une même ligne (ou une même colonne). Si tous les éléments d'une même ligne ont une propriété **flex-grow** initialisée à **1**, ils auront la même largeur. Si l'un d'entre eux a une propriété **flex-grow** initialisée à **2** et les autres à **1**, il aura une largeur deux fois plus grande que les autres éléments.

### Rétrécissement des enfants – flex-shrink

La propriété **flex-shrink** indique si un élément peut être rétréci.

### Taille d'un élément – flex-basis

La propriété **flex-basis** définit la taille par défaut d'un élément avant que l'espace restant ne soit distribué :

```
#enfant2 { flex-basis: 200px; }
```

ou:

#enfant2 { flex-basis: auto; // Valeur par défaut}

### Alignement d'un enfant – align-self

La propriété **align-self** permet de définir l'alignement d'un enfant. Elle peut prendre les valeurs suivantes : **auto**, **flex-start**, **flex-end**, **center**, **baseline** ou **stretch**.

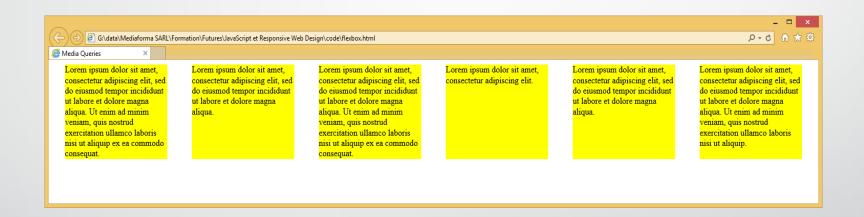
Et maintenant, vous allez vous entraîner pour comprendre le fonctionnement des **flexbox**.

### Commencez par saisir ce code HTML :

```
<body>
 <div class="conteneur">
   <div class="enfant">
   Lorem ipsum dolor sit amet, consectetur adipiscing 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 consequat.
    </div>
    <div class="enfant">
   Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et
dolore magna aliqua.
    </div>
    <div class="enfant">
   Lorem ipsum dolor sit amet, consectetur adipiscing 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 consequat.
    </div>
    <div class="enfant">
   Lorem ipsum dolor sit amet, consectetur adipiscing elit.
   </div>
    <div class="enfant">
   Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et
dolore magna aliqua.
    </div>
    <div class="enfant">
   Lorem ipsum dolor sit amet, consectetur adipiscing 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.
   </div>
 </div>
</body>
</html>
```

#### **Exercice**

Définissez les classes conteneur et enfant pour mettre en place une mise en page flexbox dans laquelle tous les enfants ont une largeur de 200 pixels, une couleur d'arrière-plan yellow et se partagent équitablement l'espace horizontal restant.



#### Indices

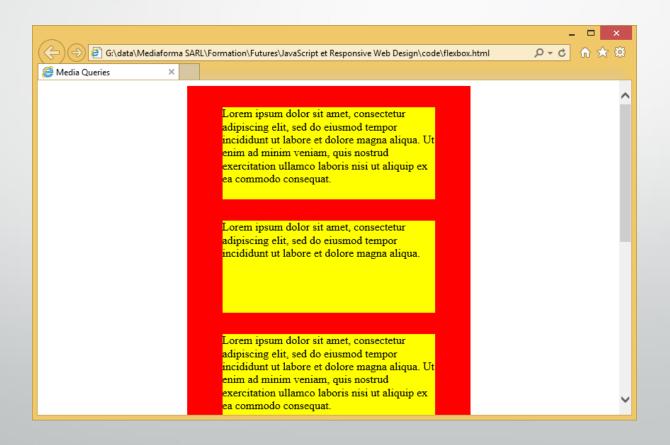
```
.conteneur {
 display: flex;
 flex-flow: row|colum; // enfants en ligne ou en colonne
 flex-direction: row|row-reverse|column|colum-reverse; // Ordre d'affichage des enfants dans le parent
 flex-wrap: nowrap|wraop|wrap-reverse; // Sur 1 ligne/col|plusieurs lignes/col|plusieurs lignes/col inversé
 justify-content: flex-start|flex-end|center|space-between|space-around; // align enfants sur axe princ
  align-items: flex-start|flex-end|center|stretch|baseline; // alignement des enfants sur l'axe secondaire
  align-content: flex-start|flex-end|center|stretch|space-between|space-around; // esp. enfants sur axe sec.
 flex-grow: 1[2|3|... //Largeur des éléments affichés sur 1 même ligne/col (n X + large)
 flex-shrink: 1[2|3|... //Largeur des éléments affichés sur 1 même ligne/col (n X + étroit)
#enfant3 {order: 5;} //ordre d'affichage d'un enfant
#enfant2 { flex-basis: 200px; } // Largeur par défaut d'un élément
#enfant2 {align-self: auto|flex-start|flex-end|center|baseline|stretch;} //Alignement d'un enfant
```

# **Solution**

```
<style>
.conteneur {
    display: flex;
    justify-content: space-between;
}
.enfant {
    flex-basis: 200px;
    background-color: yellow; }
</style>
```

#### **Exercice**

Définissez les classes conteneur et enfant pour mettre en place une mise en page flexbox sur une colonne centrée de largeur 400 px et d'arrière-plan red. Les enfants auront une hauteur de 130px, une largeur de 300px et un arrière-plan yellow. Chaque enfant sera séparé du suivant par une marge verticale de 30 px et sera centré dans son parent.



#### Indices

```
.conteneur {
 display: flex;
 flex-flow: row|colum; // enfants en ligne ou en colonne
 flex-direction: row|row-reverse|column|colum-reverse; // Ordre d'affichage des enfants dans le parent
 flex-wrap: nowrap|wraop|wrap-reverse; // Sur 1 ligne/col|plusieurs lignes/col|plusieurs lignes/col inversé
 justify-content: flex-start|flex-end|center|space-between|space-around; // align enfants sur axe princ
  align-items: flex-start|flex-end|center|stretch|baseline; // alignement des enfants sur l'axe secondaire
  align-content: flex-start|flex-end|center|stretch|space-between|space-around; // esp. enfants sur axe sec.
 flex-grow: 1[2|3|... //Largeur des éléments affichés sur 1 même ligne/col (n X + large)
 flex-shrink: 1[2|3|... //Largeur des éléments affichés sur 1 même ligne/col (n X + étroit)
#enfant3 {order: 5;} //ordre d'affichage d'un enfant
#enfant2 { flex-basis: 200px; } // Largeur par défaut d'un élément
#enfant2 {align-self: auto|flex-start|flex-end|center|baseline|stretch;} //Alignement d'un enfant
```

# **Solution**

```
<style>
  .conteneur {
   display: flex;
   flex-flow: column;
   width: 400px;
   background-color: red;
   margin: auto;
  .enfant {
   height: 130px;
   width: 300px;
   background-color: yellow;
   margin-top: 30px;
   margin-left: auto;
   margin-right: auto;
</style>
```

# Gestion des débordements de contenus

Qu'il s'agisse de texte, d'images ou de tout autre type de contenu, les éléments placés dans un conteneur peuvent déborder de ce conteneur. Vous devez alors gérer les débordements, en particulier si le contenu peut provenir de sources que vous ne contrôlez pas.

#### Examinez le code suivant :

```
_ 🗆 X
<!DOCTYPE html>
                                                                 <html>
                                                           Gestion des débordements
   <head>
                                                           Lorem ipsum dolor sit amet.
                                                           ce mot est extrèmement long intentionnellement pour illustrer la coupure de mots
      <meta charset="UTF-8" />
                                                            consectetur adipiscing elit, sed
      <title>Gestion des débordements</title>
                                                           do eiusmod tempor incididunt
                                                           ut labore et dolore magna
      <style>
                                                           aliqua. Ut enim ad minim
        #mabox {
                                                           veniam, quis nostrud
                                                            exercitation ullamco laboris
           width: 200px;
                                                           nisi ut aliquip ex ea commodo
                                                           consequat. Duis aute irure
           height: 200px;
                                                           dolor in reprehenderit in
           border: 1px dotted black;
                                                            voluptate velit esse cillum
                                                           dolore eu fugiat nulla pariatur.
           background-color: yellow;
                                                            Excepteur sint occaecat
                                                            cupidatat non proident, sunt in
                                                           culpa qui officia deserunt
      </style>
                                                           mollit anim id est laborum
   </head>
   <body>
       <div id="mabox">
         Lorem ipsum dolor sit amet,
ce mot est extrèmement long intentionnellement pour illustrer la coupure de mots
consectetur adipiscing 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 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>
   </body>
 </html>
```

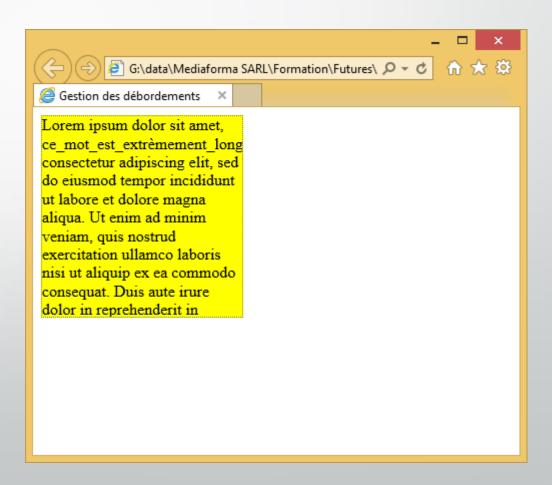
Manifestement, la balise <div> conteneur n'est pas assez grande pour héberger tout le texte. Etant donné que ses dimensions ont été fixées "en dur" (width et heught), le texte déborde. Plusieurs technique vont permettre de gérer les débordements.

# Première technique : overflow: hidden

Tout ce qui dépasse du conteneur est affacé avec cette instruction CSS :

```
#mabox {
   overflow: hidden;
```

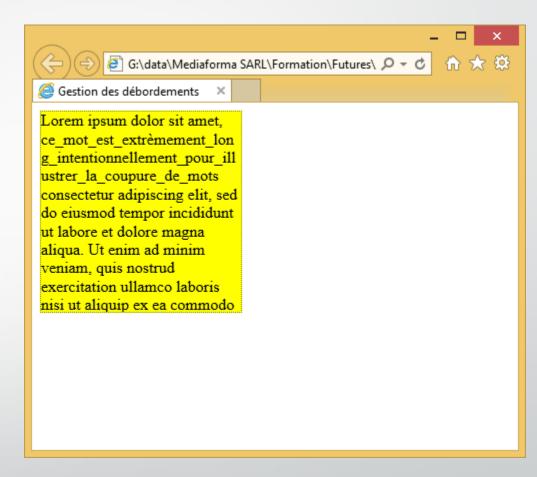
Voici le résultat :



### Deuxième technique : passage à la ligne des mots longs

Lorsqu'elle est initialisée à break-word, la propriété CSS overflow-wrap permet de découper proprement les mots trop longs pour le conteneur :

```
#mabox {
  overflow: hidden;
  word-wrap: break-word;
```



Comme vous pouvez le constater, le mot long n'est pas partiellement dissimulé par le propriété overflow: hidden;. Au contraire, le mot est renvoyé à la ligne autant de fois que nécessaire pour s'afficher dans son intégralité.

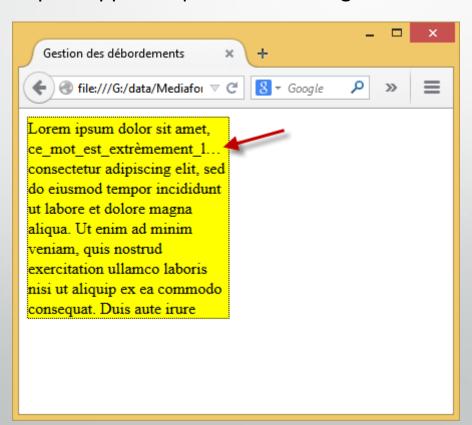
### Troisième technique : points de suspension

En affectant la valeur **ellipsis** à la propriété **text-overflow**, des points de suspension seront affichés chaque fois qu'un texte est rogné :

```
#mabox {
  overflow: hidden;
  text-overflow: ellipsis;
```

#### Attention

Cette propriété n'est pas supportée par tous les navigateurs.



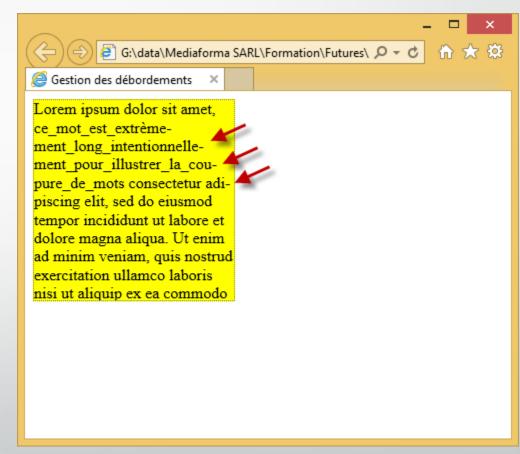
# Quatrième technique : césure de mots

Affectez la valeur **auto** à la propriété **hyphens** pour activer la césure de mots :

```
#mabox {
  overflow: hidden;
  hyphens: auto;
  -ms-hyphens: auto;
  -moz-hyphens: auto;
  -webkit-hyphens: auto;
```

#### Attention

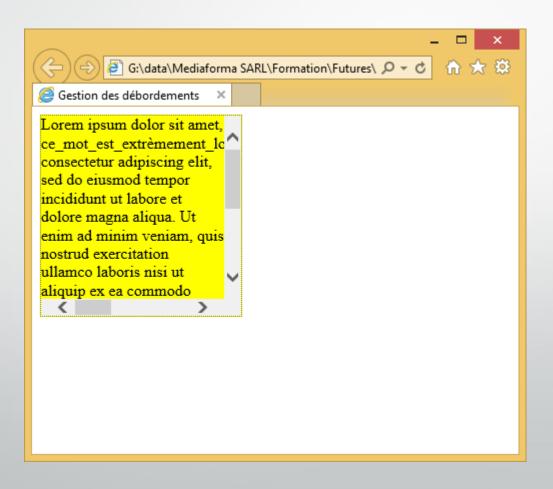
Cette propriété n'est pas encore supportée par tous les navigateurs. Pensez à utiliser les préfixes habituels pour améliorer sa compatibilité.



# Cinquième technique : barres de défilement

En affectant la propriété **auto** à la propriété **overflow**, des barres de défilement permettent d'accéder au contenu qui déborder du conteneur :

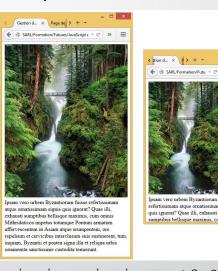
```
#mabox {
   overflow: auto;
```



# Composants graphiques

# Images fluides

Pour que la taille d'une image soit automatiquement redimensionnée en fonction de l'espace dont elle dispose, il suffit de définir sa largeur maximale à 100% de son conteneur. Voici un exemple de code :



Ipsam vero urbem Byzantiorum fuisse refertissimam atque ornatissimam signis quis ignorat? Quae illi, exhausti sumptibus bellisque maximis, cum omnis Mithridaticos impetus totumque Pontum armatum affervescentem in Asiam atque erumpentem, ore repulsum et cervicibus interclusum suis sustinerent, tum, inquam, Byzantii et postea signa illa et reliqua urbis ornanemta sanctissime custodita tenuerunt.

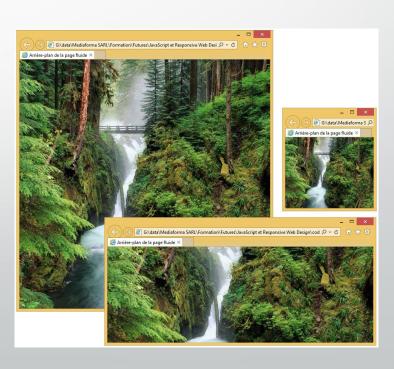
```
</body>
```

Redimensionnez la fenêtre. Comme vous pouvez le voir, l'image se redimensionne automatiquement en fonction des dimensions de la fenêtre :

# Arrière-plan de page fluide

Pour définir un arrière-plan de page fluide, la technique est un peu différente :

```
<!DOCTYPE html>
<html>
   <head>
     <meta charset="UTF-8" />
     <title>Arrière-plan de la page fluide</title>
     <style>
       body{
         background-image:url(foret.jpg);
         background-attachment: fixed;
         background-repeat:no-repeat;
         background-position:center center;
         background-size:cover;
     </style>
   </head>
   <body>
   </body>
</html>
```



# Textes et polices

#### Dans cette partie :

- Eléments redéfinis en HTML5
- Nouveaux éléments en HTML5
- Bien afficher les caractères accentués

### Eléments redéfinis en HTML5

<a>

Dans les versions précédentes du langage, l'élément <u>a</u> avait un modèle de contenu <u>inline</u>. En d'autres termes, il ne pouvait envelopper que des éléments de type <u>inline</u>.

En HTML5, il peut envelopper des éléments de tous niveaux : <u>p</u>, <u>hr</u>, <u>pre</u>, <u>ul</u>, <u>ol</u>, <u>dl</u>, <u>div</u>, <u>h1</u>, <u>h2</u>, <u>h3</u>, <u>h4</u>, <u>h5</u>, <u>h6</u>, <u>hgroup</u>, <u>address</u>, <u>blockquote</u>, <u>ins</u>, <u>del</u>, <u>object</u>, <u>map</u>, <u>noscript</u>, <u>section</u>, <u>nav</u>, <u>article</u>, <u>aside</u>, <u>header</u>, <u>footer</u>, <u>video</u>, <u>audio</u>, <u>figure</u>, <u>table</u>, <u>form</u>, <u>fieldset</u>, <u>menu</u>, <u>canvas</u> ou <u>details</u>.

## <a> Exemples

```
<!-- Lien traditionnel -->
<a href="http://www.siteweb.com">texte</a>
<!-- Lien sur un tableau -->
<a href="http://www.siteweb.com">
 </a>
<!-- Lien avec identifiant sur une image. L'image s'affiche dans une fenêtre complémentaire -->
<a id="monlien" "http://www.siteweb.com" target=" blank"><img src="monimage.gif"></a>
<!-- Lien pour tous les types de média sauf print -->
<a href= "http://www.siteweb.com" media="not print">next</a>
```

## Eléments redéfinis en HTML5

<b>, <strong>, <i> et <small>

L'élément <u>b</u> met en avant de façon modérée une portion de texte. L'élément <u>strong</u> met en avant un texte plus important.

L'élément <u>i</u> met en italique une portion de texte sans toutefois le mettre en avant. Il sera utilisé pour afficher des termes techniques, des pensées, des désignations taxonomiques, des expressions idiomatiques, etc.

L'élément <u>small</u> est essentiellement utilisé pour afficher des commentaires secondaires et des mentions légales. Le texte encadré par cet élément apparaît en caractères de petite taille.

## Nouveaux éléments HTML5

### time

L'élément <u>time</u> définit une information horaire ou une date. Voici sa syntaxe :

```
<time>date ou heure</time>
```

#### ou

<time datetime="date ou heure">texte</date>

#### Quelques exemples:

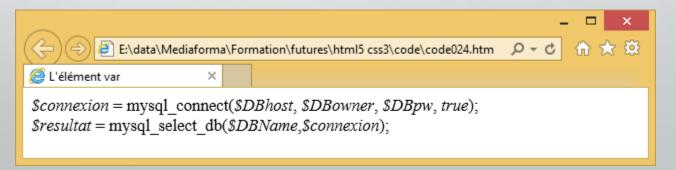
```
Le magasin ouvre à <time>09:00</time> du lundi au vendredi.
```

Que faites-vous à <time datetime="25-12-2010">Noël</time> ?

## Nouveaux éléments HTML5

#### var

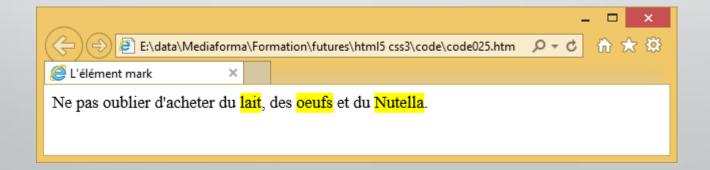
L'élément <u>var</u> permet de différencier du reste du texte les variables définies dans du code informatique ou des équations mathématiques. Par défaut, le texte ciblé apparaît en italique, mais il est possible de modifier cet attribut avec des déclarations de styles CSS.



## Nouveaux éléments HTML5

### mark

L'élément <u>mark</u> surligne une portion de texte. Au moment de l'écriture de cet ouvrage, il n'est supporté que par les navigateurs Webkit (Chrome et Safari). Voici sa syntaxe : <mark>texte</mark>



### Bien afficher les caractères accentués

Pour que les caractères accentués apparaissent correctement dans tous les navigateurs du monde, vous pouvez utiliser :

```
Á Á
á á
À À
é é
&à à
 Â
â â
etc.
```

Vous pouvez aussi utiliser le code ASCII des caractères :

Caractère

Code

Ò

#### codeo26.htm

```
<meta charset=UTF-8" />
  <title>Les codes des caract&egrave; res</title>
 </head>
 <body>
  <
           32 :  
                   33 : !
                           34 : "
                                  35 : #
                                          36 : $
                                                  37 : %
                                                          38 : &
        41 : )
                42 : *
                        43 : +
                               44: ,
                                       45 : -
                                               46: .
                                                       47 : /
                                                               48 : 0
                                                                       49: 1
40 : (
50: 2
        51 : 3
                52 : 4
                        53 : 5
                                54 : 6
                                       55 : 7
                                               56: 8
                                                       57: 9
                                                               58: :
                                                                       59: &#59;
60 : <
        61 : =
                62: >
                        63: ?
                                64 : @
                                       65 : A
                                               66 : B
                                                       67: C
                                                               68: D
                                                                       69: E
70: F
        71 : G
                72 : H
                        73 : I
                                74: J
                                       75 : K
                                               76: L
                                                       77 : M
                                                               78: N
                                                                       79: O
        81 : Q
                82 : R
                        83: S
                                84 : T
                                       85 : U
                                               86: V
                                                       87: W
80 : P
                                                               88 : X
                                                                       89 : Y
                92 : \
90: Z
        91 : [
                        93 : ]
                                94 : ^
                                       95 : _
                                               96: `
                                                       97 : a
                                                               98: b
100 : d 101 : e 102 : f 103 : g 104 : h 105 : i 106 : j 107 : k 108 : l 109 : m
110 : n 111 : o 112 : p 113 : q 114 : r 115 : s 116 : t 117 : u 118 : v 119 : w
120 : x 121 : y 122 : z 115 : s 124 : | 125 : } 126 : ~ 127 :  128 : € 129 : 
130 : ' 131 : ƒ 132 : " 133 : … 134 : † 135 : ‡ 136 : ˆ 137 : ‰ 138 : Š 139 : ‹
140 : Œ 141 :  142 : Ž 143 :  144 :  145 : ' 146 : ' 147 : " 148 : " 149 : •
150 : – 151 : — 152 : ˜ 153 : ™ 154 : š 155 : › 156 : œ 157 :  158 : ž 159 : Ÿ
160 :   161 : ¡ 162 : ¢ 163 : £ 164 : ¤ 165 : ¥ 166 : ¦ 167 : § 168 : ¨ 169 : ©
170 : ª 171 : « 172 : ¬ 173 : ­ 174 : ® 175 : ¯ 176 : ° 177 : ± 178 : ² 179 : ³
180 : ´ 181 : µ 182 : ¶ 183 : · 184 : ¸ 185 : ¹ 186 : º 187 : » 188 : ¼ 189 : ½
190 : ¾ 191 : ¿ 192 : À 193 : Á 194 : Â 195 : Ã 196 : Ä 197 : Å 198 : Æ 199 : Ç
200 : È 201 : É 202 : Ê 203 : Ë 204 : Ì 205 : Í 206 : Î 207 : Ï 208 : Ð
210 : Ò 211 : Ó 212 : Ô 213 : Õ 214 : Ö 215 : × 216 : Ø 217 : Ù 218 : Ú 219 : Û
220 : Ü 221 : Ý 222 : Þ 215 : × 224 : à 225 : á 226 : â 227 : ã 228 : ä 229 : å
150 : – 151 : — 152 : ˜ 153 : ™ 154 : š 155 : › 156 : œ 157 :  158 : ž 159 : Ÿ
240 : ð 241 : ñ 242 : ò 243 : ó 244 : ô 245 : õ 246 : ö 247 : ÷ 248 : ø 249 : ù
250: ú 251: û 252: ü 253: ý 254: þ 255: ÿ
```

<!DOCTYPE html>

<html> <head>

</body>

# Regroupement du contenu

#### Dans cette partie:

- Concept de boîte
- Styles de liste HTML
- Choisir le style d'une liste
- Des images à la place des puces
- Légendage
- Texte non proportionnel
- Citations
- Ruptures thématiques

### Concept de boîte

Les "boîtes de bloc" sont des zones rectangulaires qui se comportent comme autant de zones d'affichage indépendantes. Chaque boîte peut être dotée de barres de défilement, d'une image d'arrière-plan, de styles indépendants et être positionnée précisément sur la page pour obtenir un affichage digne d'un professionnel.

Les boîtes sont mises en place avec l'élément div. Leurs principales propriétés sont les suivantes :

- width: largeur;
- <u>height</u>: hauteur;
- <u>margin</u>: marges extérieures;
- <u>padding</u>: marges intérieures;
- <u>background-color</u>: couleur d'arrière-plan.

### Exercice

Définissez une boîte dont les caractéristiques sont les suivantes :

- largeur : 250 pixels;
- hauteur: 60 pixels;
- marges extérieures : 20 pixels ;
- marges intérieures : 10 pixels ;
- couleur d'arrière-plan : jaune.

Affichez du texte dans cette boîte.

### Solution

codeo27.htm

```
<!DOCTYPE html">
<html>
  <head>
    <meta charset=UTF-8" />
    <title>Le concept de boîtes</title>
  </head>
  <body >
    <div style="width:250px; height=60px; margin: 20px; padding: 10px; background-color: yellow;">
    Ce texte est affiché à l'intérieur de la balise div
    </div>
  </body>
</html>
                               E:\data\Mediaforma\Formation\futures\html5 css3\code\code027.htm
                        Le concept de boîtes
                           Ce texte est affiché à l'intérieur de la
                           balise div
```

### Listes HTML5

Le langage HTML5 compte trois types de listes :

- non ordonnées ou listes à puces : éléments <u>ul</u> et <u>li</u> ;
- ordonnées : éléments <u>ol</u> et <u>li</u> ;
- de description : éléments <u>dl</u>, <u>dt</u>, <u>dd</u> et <u>dfn</u>.

Les listes de description étaient déjà présentes dans HTML 4.x. Souvent boudées et utilisées de manière non sémantique, elles ont subi un lifting pour être remises au goût du jour dans HTML5. L'élément <u>dfn</u> est essentiellement utilisé pour encapsuler les mots d'un glossaire, à l'intérieur de l'élément <u>dt</u>. Notez qu'on peut également définir plusieurs descriptions (<u>dd</u>) pour un seul nom (<u>dt</u>), ou encore regrouper plusieurs noms pour une seule description.

#### Cet exemple illustre ces trois types de listes :

```
Styles de listes HTML
                                                Liste non ordonnée
<!DOCTYPE html">
                                                   · Premier élément
<html>
                                                   · Deuxième élément
  <head>
                                                   · Troisième élément
    <meta charset=UTF-8" />
    <title>Styles de listes HTML</title>
                                                Liste ordonnée
  </head>

    Premier élément

  <body>
                                                  Deuxième élément
    <b>Liste non ordonnée</b>
                                                  3. Troisième élément
    <111>
      Premier élément
                                                Liste de termes et de descriptions
      Deuxième élément
      Troisième élément
                                                Bit
                                                    Information représentée par un symbole à deux valeurs généralement notées 0 et 1,
    associées aux deux états d'un dispositif.
    <b>Liste ordonnée</b>
                                                Octet
    <01>
                                                    Ensemble ordonné de huit éléments binaires traités comme un tout.
      Premier élément
      Deuxième élément
      Troisième élément
    <br/>b>Liste de termes et de descriptions</b>
    <d1>
      <dt><dfn>Bit</dfn></dt>
      <dd>Information représentée par un symbole à deux valeurs généralement notées 0 et 1,
associées aux deux états d'un dispositif.</dd>
      <dt><dfn>Octet</dfn></dt>
      <dd>Ensemble ordonné de huit éléments binaires traités comme un tout.</dd>
    </dl>
  </body>
</html>
```

E:\data\Mediaforma\Formation\futures\html5 css3\code\code028.htm

### Choisir le style d'une liste

CSS3 est bien plus riche que HTML5 en matière de listes et la propriété <u>list-style-type</u> permet de choisir le style d'une liste. Elle admet de nombreuses valeurs.

#### Pour une liste à puces (élément <u>ul</u>) :

- <u>disc</u>: puce ronde remplie;
- <u>circle</u>: puce ronde vide;
- <u>square</u> : carré rempli ;
- <u>none</u>: aucun signe.

#### Pour une liste ordonnée alphabétique (élément ol) :

- <u>lower-roman</u>: i, ii, iii, iv;
- <u>upper-roman</u>: I, II, III, IV;
- lower-greek :  $\alpha$ ,  $\beta$ ,  $\chi$ ,  $\delta$ ,  $\epsilon$ ;
- <u>lower-alpha</u>: a, aa, aaa;
- <u>upper-alpha</u>: A, AA, AAA;
- <u>lower-latin</u>: a, b, c;
- <u>upper-latin</u>: A, B, C;
- <u>none</u>: aucune numérotation.

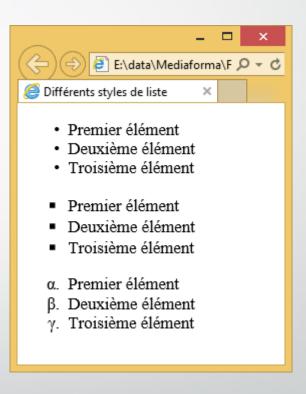
#### Pour une liste ordonnée décimale (élément ol) :

- <u>decimal</u>: 1, 2, 3;
- <u>decimal-leading-zero</u>: 01, 02, 03;
- <u>hebrew</u>: numérotation hébraïque traditionnelle;
- <u>armenian</u>: numérotation arménienne traditionnelle;
- <u>georgian</u>: numérotation géorgienne traditionnelle;
- <u>hiragana</u>: a, i, u, e, o, ka, ki, ku;
- <u>katakana</u>: A, I, U, E, O, KA, KI, KU;
- <u>hiragana-iroha</u>: i, ro, ha, ni, ho, he;
- <u>katakana-iroha</u>: I, RO, HA, NI, HO, TE;
- <u>none</u>: aucune numérotation.

Exercice : Entraînez-vous à utiliser la propriété list-styletype dans des listes ul et ol.

#### codeo29.htm

```
<!DOCTYPE html>
<ht.ml>
 <head>
   <meta charset="UTF-8" />
  <title>Différents styles de liste</title>
  <style>
    .rond {list-style-type: disc;}
    .carre {list-style-type: square;}
    .lgreek {list-style-type: lower-greek;}
  </style>
 </head>
 <body>
  Premier élément
     Deuxième élément
     Troisième élément
  Premier élément
     Deuxième élément
     Troisième élément
  Premier élément
     Deuxième élément
     Troisième élément
  </body>
</html>
```



## Des images à la place des puces

Si vous vous sentez limité par les styles de puces accessibles par défaut en CSS3, sachez que la propriété <u>list-style-image</u> permet d'utiliser n'importe quelle image comme symbole de puce. Voici sa syntaxe :

```
list-style-image: url(adresse);
```

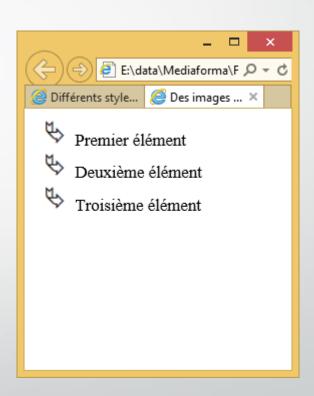
Où <u>adresse</u> est l'adresse URI de la puce, au format GIF ou JPEG.

### Exercice

Trouvez sur le Web un clipart de puce et utilisez-le dans une liste à puces.

### Solution

```
<!DOCTYPE html">
<html>
 <head>
   <meta charset=UTF-8" />
   <title>Des images à la place des puces</title>
   <style>
     .puce {list-style-image: url(puce.gif);}
   </style>
</head>
 <body>
   Premier élément
     Deuxième élément
     Troisième élément
   </body>
</html>
```



## Des images à la place des puces

Si l'image spécifiée dans la propriété <u>list-style-image</u> n'est pas trouvée, la puce par défaut la remplacera. Si vous souhaitez une puce circle, square ou aucune puce, vous devez initialiser la propriété <u>list-style-type</u> en conséquence. Par exemple :

```
.puce
{
    list-style-image: url(puces.gif);
    list-style-type: square;
}
```

## Légendage

Dans les livres et les magazines, les illustrations, graphiques, tableaux, exemples de code, etc., sont souvent repérés par une légende. Jusqu'ici, il n'existait aucun moyen de marquer sémantiquement ce genre de contenu dans un document HTML. Pour combler cette lacune, deux nouveaux éléments font leur apparition dans HTML5 : <u>figure</u> et <u>figcaption</u>.

#### Vous les utiliserez comme ceci :

```
<figure>
  <img>, , <video>, <code>, etc.
  <figcaption>La légende de l'élément.</figcaption>
</figure>
```

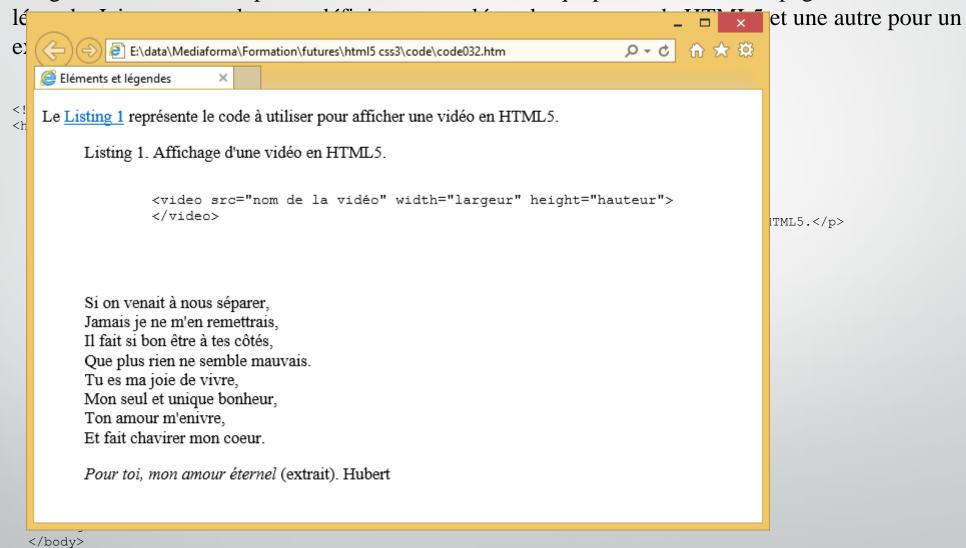
#### **Attention**

<figcaption> doit être défini comme premier ou dernier enfant de l'élément figure.

```
<!DOCTYPE html>
<html>
 <head>
   <meta charset=UTF-8" />
   <title>Eléments et légendes</title>
 </head>
 <body>
   <figure>
     <img src="grenouille.jpg" width="200px" border="1" />
     <figcaption>Légende de la Figure 1</figcaption>
   </figure>
    >
   <figure>
     <img src="cabane.jpg" height="200px" border="1" />
     <figcaption>Légende de la Figure 2</figcaption>
    </figure>
    >
   <figure>
     <video src="video1.mp4" width="200" border="1"></video>
     <figcaption>Légende de la vidéo 1</figcaption>
    </figure>
    >
   <figure>
     <video src="video2.mp4" width="200" border="1" ></video>
     <figcaption>Légende de la vidéo 2</figcaption>
   </figure>
 </body>
</html>
```



Images et vidéos ne sont pas les seuls éléments HTML5 qui peuvent être accompagnés d'une



</html>

## Texte non proportionnel

Lorsqu'un texte est encadré avec les éléments <u>pre</u> et <u>/pre</u>, ses caractères sont affichés avec une police non proportionnelle, et les passages à la ligne respectent ceux rencontrés dans le code. Les éléments <u>br</u> ne sont donc plus nécessaires. Cet élément est généralement utilisé pour différencier le code informatique sur une page web.

Le code ci-après affiche du texte et du code HTML. Il est utilisé deux fois pour montrer l'importance des caractères <u>&lt</u>; et <u>&gt</u>; et pour mettre en évidence l'interprétation des caractères d'espacement dans l'élément <u>pre</u>.

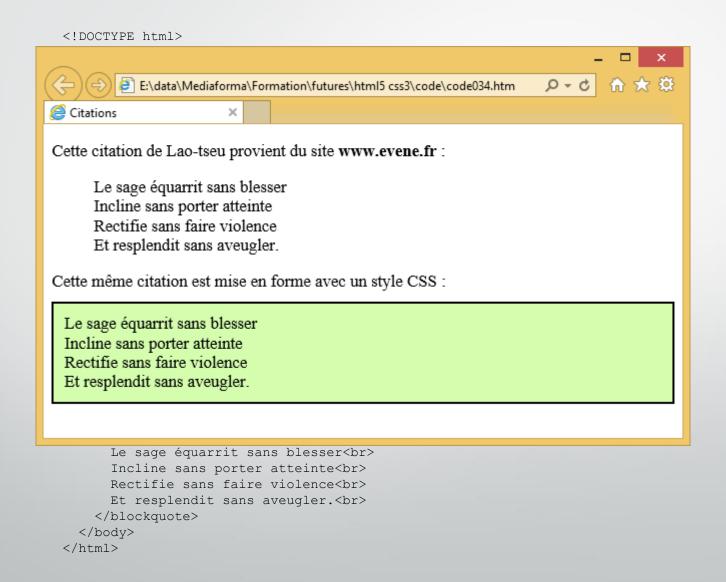
```
<!DOCTYPE html>
<html>
  <head>
   <meta charset=UTF-8" />
   <title>Balise pre</title>
  </head>
  <body>
   Ce code affiche une imagette sur laquelle le visiteur peut cliquer pour afficher la vidéo
"Enregistrer le son du PC" :<br>
   <
   <a href="http://www.google.com">
     <img src="https://cdn.pixabay.com/photo/2016/12/06/15/26/christmas-cookies-1886760 960 720.jpg"</pre>
border=0>
   </a>
  <br/>br>
  Ce code affiche une imagette sur laquelle le visiteur peut cliquer pour afficher la vidéo "Enregistrer
le son du PC" :<br>
  <
   <a href="http://www.google.com"&gt;
      <img src="https://cdn.pixabay.com/photo/2016/12/06/15/26/christmas-cookies-1886760 960 720.jpg"
border=0&gt
    <a&qt;
  </body>
```

### Citations

L'élément <u>blockquote</u> permet d'insérer une citation dans une page web, ce qui est intéressant pour faire référence à une source d'informations extérieure ou pour mettre en valeur des blocs de texte dans vos pages.

Le code ci-après affiche une citation provenant d'un site externe, sans mise en forme CSS puis avec mise en forme CSS:

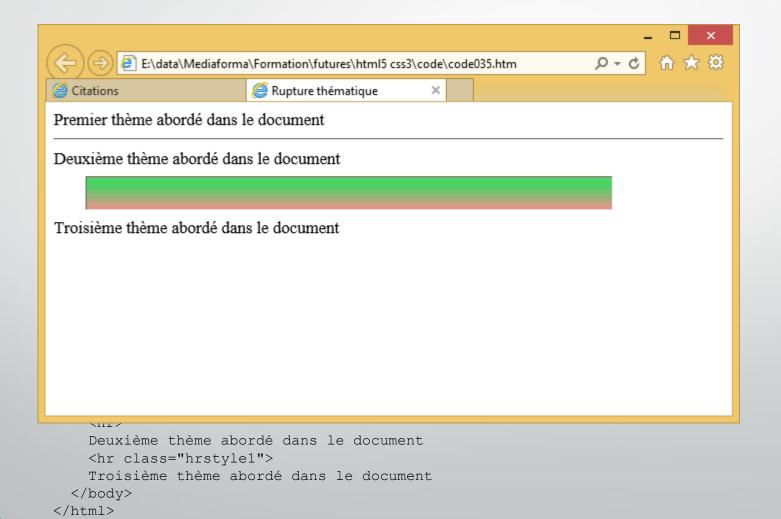
Codeo34.htm



## Rupture thématique

L'élément <u>hr</u> insère une ligne horizontale qui représente une rupture thématique dans le contenu de la page. Les attributs HTML 4.01 <u>align</u>, <u>noshade</u>, <u>size</u> et <u>width</u> ne sont plus supportés dans la version HTML5 de cet élément. Pour modifier les caractéristiques d'un élément <u>hr</u>, vous devrez désormais passer par le code CSS.

### Ce code affiche un élément <u>hr</u> standard et un autre dont les caractéristiques ont été modifiées par quelques lignes de CSS



Voici une version agrandie de l'image \_fond.gif :

## Les éléments interactifs

Certains éléments HTML5 sont interactifs de façon native, c'est-à-dire qu'ils ne nécessitent pas de code CSS, JavaScript ou jQuery supplémentaire. Ils ont été regroupés dans cette partie :

- Version résumée et étendue d'un texte
- Eléments éditables
- Menus HTML5

Code 036.htm

### <details> et <summary>

Les éléments <u><details></u> et <u><summary></u> permettent d'afficher des versions résumée et étendue d'un texte. Voici la syntaxe à utiliser :

Essayez ce code dans Google Chrome, Firefox et Internet Explorer/Microsoft Edge.

### Eléments éditables

Si vous affectez la valeur <u>true</u> à l'attribut <u>contenteditable</u> d'éléments comme <u>span</u>, <u>div</u> et <u>p</u>, vous pourrez les modifier.

Si le cœur vous en dit, vous pouvez même ajouter quelques lignes de JavaScript basées sur la fonction <u>getElementById()</u> pour récupérer le texte modifié afin de lui appliquer un traitement.

### Exercice

Définissez un document HTML contenant une balise <div> que vous rendrez éditable.

Insérez un bouton et une deuxième balise <div> dans le document.

Lorsque l'utilisateur clique sur le bouton, copiez le contenu de la première <div> dans la deuxième. Pour cela, vous utiliserez la fonction JavaScript getElementById().

### Solution

```
<!doctype html>
<html>
 <head>
   <meta charset=UTF-8" />
   <title>Eléments éditables</title>
   <style>
      div
       border: dotted 1px #666;
       padding: 5px 10px;
       margin-top: 40px;
       margin-bottom: 40px;
   </style>
   <script>
     function traitement()
         var contenu = document.getElementById('texte-editable').innerHTML;
         document.getElementById('cible').innerHTML = 'Texte lu dans la div : ' + contenu;
   </script>
 </head>
 <body>
   <div id="texte-editable" contenteditable="true">Le texte de cette div peut être modifié</div>
   <button onclick="traitement()">Cliquez ici pour accéder au texte de la div</button>
   <div id="cible"></div>
 </body>
</html>
```

### Des menus en HTML5

Les éléments <u>menu</u> et <u>command</u> font leur apparition dans HTML5. Grâce à eux, il est possible de mettre en place un menu en quelques lignes de code.

L'élément <u>menu</u> définit le menu et les éléments <u>command</u> associent commandes de menus et actions.

Malheureusement, ces balises sont très mal supportées dans les navigateurs actuels. C'est pourquoi vous devrez utiliser des techniques plus conventionnelles pour mettre en place un menu.

# Des menus en HTML5

Pour créer un menu horizontal, le plus simple consiste à utiliser une liste <u></u> dont la propriété <u>display</u> vaut <u>inline</u>. Ainsi, les éléments de la liste ne seront pas affichés verticalement, en tant que blocs, mais horizontalement, sur une même ligne.

```
<!DOCTYPE html>
<html>
<head>
<style type="text/css">
 ul#horizontal
   list-style-type: none;
 ul#horizontal li
   display: inline;
   padding-right: 2em;
</style>
</head>
<body>
 <a href="page1.htm">Page 1</a>
   <a href="page2.htm">Page 2</a>
   <a href="page3.htm">Page 3</a>
   <a href="page4.htm">Page 4</a>
   <a href="page5.htm">Page 5</a>
 </body>
</html>
```

# Un menu déroulant

Le principe de base consiste à imbriquer deux listes <u></u>, la première contenant les entrées principales du menu et la deuxième les entrées secondaires.

```
<style type="text/css">
  #principal
   background: #E0FCC8;
   width: 800px;
   height: 30px;
   position: absolute;
   top: 0;
   left: 0;
  #principal li
   list-style-type: none;
    float: left;
   width: 150px;
   border-style: dashed;
   border-width: 1px;
  #principal li a:link, #principal li a:visited
  display: block;
  background: #E0FCC8;
   margin: 0;
  padding: 5px 6px;
   text-decoration: none;
   color: black;
  #principal li a:hover
   background-color: #A5F95B;
  #principal .secondaire
   display: none;
   list-style-type: none;
   margin: 0;
   padding: 0;
    border: 0;
```

#### Codeo38.htm

```
#principal .secondaire li
   float: none;
   border: 0;
  #principal .secondaire li a:link, #principal .secondaire li a:visited
   display: block;
   text-decoration: none;
  #principal .secondaire li a:hover
   background-color: #A5F95B;
  #principal li:hover > .secondaire
   display: block;
</style>
<body>
<a href="#">Page 1</a>
  <a href="page1-1.htm">Page 1 - Sous-page 1</a>
  <a href="page1-2.htm">Page 1 - Sous-page 2</a>
  <a href="page1-3.htm">Page 1 - Sous-page 3</a>
 <a href="#">Page 2</a>
  <a href="page2-1.htm">Page 2 - Sous-page 1</a>
  <a href="page2-2.htm">Page 2 - Sous-page 2</a>
  <a href="page2-3.htm">Page 2 - Sous-page 3</a>
 <a href="#">Page 3</a>
 <a href="page3-1.htm">Page 3 - Sous-page 1</a>
  <a href="page3-2.htm">Page 3 - Sous-page 2</a>
  <a href="page3-3.htm">Page 3 - Sous-page 3</a>
 </body>
```

# Les liens hypertextes

Avec les liens hypertextes, vos pages web peuvent faire référence à des pages situées sur le même site web ou sur des sites annexes. Ce chapitre montre comment personnaliser vos éléments XHTML <u>a</u> en définissant un ou plusieurs styles CSS associés.

# Liens hypertextes

Plusieurs types de liens peuvent être utilisés :

Lien vers	Exemple
un document local	<a href="nom-du-document.htm">texte du lien</a>
un signet dans un document	<a href="nom-du-document.htm#nom-du-signet">texte du lien</a>
local	(le signet est défini à l'aide de l'attribut id ou name d'un élément
	quelconque)
un document sur un autre site	<a href="http://nom-du-site/nom-du-dossier/nom-du-&lt;/th&gt;&lt;/tr&gt;&lt;tr&gt;&lt;th&gt;&lt;/th&gt;&lt;th&gt;document.htm">texte du lien</a>
un fichier déposé sur un	<a href="ftp://adresse-du-fichier">Texte du lien</a>
serveur FTP	
un correspondant e-mail	<a href="mailto:adresse-du-correspondant">Texte du lien</a>

# Liens hypertextes

En HTML 4.01, l'élément <u>a</u> pouvait être utilisé pour définir un lien ou une ancre. La nuance se faisait au niveau de l'attribut <u>href</u>.

Cette ligne définit un lien vers le site Google :

```
<a href="http://www.google.fr">Cliquez ici pour aller sur Google</a>
```

Par contre, cette ligne définit l'ancre "plusloin":

```
<a name="plusloin">Texte quelconque</a>
```

Pour faire afficher la portion de page qui commence à l'ancre plusloin, le code à utiliser est le suivant :

```
<a href="#plusloin">Cliquez ici pour aller plus loin</a>
```

Supposons que l'ancre ait été définie sur une autre page HTML nommée page2.htm. Pour afficher la portion de page qui commence à l'ancre plusloin dans page2.htm, le code à utiliser est le suivant :

```
<a href="page2.htm#plusloin">Cliquez ici pour aller plus loin</a>
```

# Liens hypertexte en HTML5

En HTML5, l'attribut <u>name</u> utilisé sur des éléments autres que <u>input</u> devient obsolète. L'élément <u>a</u> ne peut donc plus être utilisé pour définir des ancres. Rassurez-vous : définir des ancres est toujours possible si on affecte un identifiant à un élément et si on y fait référence dans un élément <u>a</u> :

# Liens hypertextes en HTML5

En HTML5, l'élément <u>a</u> peut contenir un ou plusieurs éléments quelconques, mis à part les éléments interactifs. Le code ci-après est ainsi totalement licite :

```
<!DOCTYPE html>
<html>
  <head>
  </head>
  <body>
    <aside>
     Atteindre
      <a href="http://www.google.com/windows8">
        Les formations
        <h2>Une minute par jour</h2>
        Une minute tous les jours pour apprendre à utiliser Windows 8
      </a><br>
      Cette page donne des trucs et astuces en vidéo pour bien utiliser Windows 8.
    </aside>
  </body>
</html>
```

Codeo39.htm

### Mettre en forme des liens

Par défaut, les liens hypertextes apparaissent en caractères bleus soulignés lorsqu'ils n'ont pas été visités. Ils sont affichés en caractères bordeaux soulignés lorsqu'ils ont été visités.

Vous utiliserez la propriété CSS <u>color</u> pour choisir d'autres couleurs :

- <u>a {color}</u> pour les liens non visités ;
- <u>a:visited {color}</u> pour les liens visités.

Le type de soulignement peut également être choisi avec la propriété CSS <u>text-decoration</u> :

- <u>a {text-decoration}</u> pour les liens non visités ;
- <u>a:visited {text-decoration}</u> pour les liens visités.

La propriété <u>text-decoration</u> peut prendre les valeurs suivantes :

- <u>none</u>: aucun soulignement;
- <u>underline</u> : soulignement par défaut, en dessous du lien ;
- <u>overline</u>: soulignement au-dessus du lien;
- <u>overline underline</u>: soulignement double, en dessous et au-dessus du lien;
- <u>line-through</u>: barre transversale sur toute la longueur du lien.

Définissez quatre liens hypertexte de classes type1 à type4.

Définissez les classes type1 à type4 et type1.visited à type4.visited pour obtenir les caractéristiques suivantes :

- type1 non visité : couleur rouge, décoration none
- type 1 visité : couleur bleue, décoration underline
- type2 non visité : couleur bleue, décoration overline
- type2 visité : couleur rouge, décoration none
- type 3 non visité : couleur rouge, décoration overline et underline
- type 3 visité : couleur bleue, décoration overline
- type 4 non visité : couleur bleue décoration line-through
- type4 visité : couleur rouge décoration underline

# Solution

```
<!DOCTYPE html>
<html>
 <head>
    <meta charset=UTF-8" />
    <title>Mise en forme des liens hypertextes</title>
    <style>
      a.type1 {color: red; text-decoration: none; }
      a.type1:visited {color: blue; text-decoration: underline; }
      a.type2 {color: blue; text-decoration: overline; }
      a.type2:visited {color: red; text-decoration: none; }
      a.type3 {color: red; text-decoration: overline underline; }
      a.type3:visited {color: blue; text-decoration: overline; }
      a.type4 {color: blue; text-decoration: line-through; }
      a.type4:visited {color: red; text-decoration: underline; }
    </style>
 </head>
 <body>
    <a href="http://www.page1.fr" class="type1">Lien vers la page 1</a>
    <a href="http://www.page2.fr" class="type2">Lien vers la page 2</a>
    <a href="http://www.page3.fr" class="type3">Lien vers la page 3</a>
    <a href="http://www.page4.fr" class="type4">Lien vers la page 4</a>
 </body>
</html>
```

# Mettre en forme des liens

Les propriétés relatives au texte sont également utilisables dans un style relatif aux liens hypertextes. Vous utiliserez en particulier les propriétés suivantes :

- <u>font-family</u>: police(s) à utiliser;
- <u>font-size</u> : taille des caractères ;
- <u>font-style</u>: orientation des caractères (<u>normal</u>, <u>italic</u> ou <u>oblique</u>);
- <u>font-weight</u>: graisse de la police;
- <u>font-variant</u>: caractères par défaut (<u>normal</u>) ou petites majuscules (<u>small-caps</u>);
- <u>text-transform</u>: casse des caractères (<u>none</u>, <u>capitalize</u>, <u>uppercase</u> ou <u>lowercase</u>).

Enfin, notez qu'il est possible d'utiliser la pseudo-classe <u>a:hover</u>, associée à la propriété <u>color</u>, pour changer la couleur des liens hypertextes lorsque le pointeur les survole.

# Coins arrondis et ombres

Avec CSS<sub>3</sub>, réaliser des boîtes à coins arrondis et ajouter des ombrages sur des éléments quelconques devient un jeu d'enfant. Ce chapitre montre comment procéder en l'état actuel de l'avancement des spécifications et de leur adoption par les différents navigateurs du marché.

#### Dans cette section:

- Boîte à coins arrondis
- Mise en page avec arrondis
- Ombrage
- Ombrage de texte

# Boîte à coins arrondis

La méthode consiste à utiliser la propriété <u>border-radius</u>. En l'état actuel du développement et de l'adoption de la norme, vous devrez spécifier des préfixes sur cette propriété, en fonction du navigateur :

- <u>-moz</u> pour les navigateurs Gecko (Mozilla Firefox);
- <u>-webkit</u> pour les navigateurs Webkit (Safari, Chrome);
- <u>-khtml</u> pour les navigateur Konqueror;
- <u>-ms</u> pour Internet Explorer 9.

La syntaxe de la propriété <u>border-radius</u> est la suivante :

```
border-radius: r1 r2 r3 r4;
```

Où <u>r1</u>, <u>r2</u>, <u>r3</u> et <u>r4</u> représentent respectivement le rayon des coins supérieur gauche, supérieur droit, inférieur droit et inférieur gauche de l'élément.

Si un seul rayon est spécifié, il s'applique aux quatre coins de l'élément.

Définissez une bordure noire d'épaisseur 1 px autour d'un élément <u>div</u> en utilisant la propriété <u>border-radius</u>.

Codeo41.htm

# Solution

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset=UTF-8" />
    <title>Une boîte à coins arrondis en CSS3</title>
    <style>
      .arrondi
        -moz-border-radius: 15px;
        -webkit-border-radius: 15px;
        -khtml-border-radius: 15px;
        -ms-border-radius: 15px;
        border-radius: 15px;
        border: 1px solid black;
    </style>
  </head>
  <body>
    <div class="arrondi">Boîte à bords arrondis en CSS3</div>
  </body>
</html>
```

L'arrondi s'étend sur toute la largeur de la page. Il est possible de limiter sa largeur avec la propriété <u>width</u> et même de définir sa hauteur avec la propriété <u>height</u>. D'autre part, le texte à l'intérieur de l'élément <u>div</u> est un peu trop collé sur le bord gauche. Pour résoudre ce problème, il suffit de définir un <u>padding</u>. Pour finaliser la mise en forme, vous pouvez également modifier l'alignement du texte avec la propriété <u>text-align</u> et colorer l'arrière-plan avec la propriété <u>background</u>.

Modifiez le code précédent en tirant profit de toutes ces propriétés.

# Solution

Codeo42.htm

```
<!DOCTYPE html>
<html>
 <head>
    <meta charset=UTF-8" />
    <title>Une boîte à coins arrondis en CSS3</title>
    <style>
      .arrondi
        -moz-border-radius: 15px;
        -webkit-border-radius: 15px;
        -khtml-border-radius: 15px;
        -ms-border-radius: 15px;
        border-radius: 15px;
        border: 1px solid black;
        padding: 10px;
        width: 250px;
        background:cyan;
        text-align: center;
    </style>
  </head>
  <body>
    <div class="arrondi">Boîte à bords arrondis en CSS3</div>
 </body>
</html>
```

L'arrière-plan peut également être une image, répétée si nécessaire pour couvrir toute la surface de l'élément. Pour cela, vous utiliserez la propriété background :

```
Background: url("url de l'image") repeat;
```

Remplacez l'arrière-plan de la boîte par une image répétée.

# Solution

Codeo43.htm

```
<!DOCTYPE html>
<html>
 <head>
    <meta charset=UTF-8" />
    <title>Une boîte à coins arrondis en CSS3</title>
    <style>
      .arrondi
        -moz-border-radius: 15px;
        -webkit-border-radius: 15px;
        -khtml-border-radius: 15px;
        -ms-border-radius: 15px;
        border-radius: 15px;
        border: 1px solid black;
        padding: 10px;
        width: 250px;
        background: url(" fond.gif") repeat;
        text-align: center;
    </style>
  </head>
  <body>
    <div class="arrondi">Boîte à bords arrondis en CSS3</div>
 </body>
</html>
```

Définissez une boîte et appliquez-lui quatre coins arrondis de rayons différents.

#### Codeo44.htm

# Solution

```
<!DOCTYPE html>
<html>
 <head>
   <meta charset=UTF-8" />
   <title>Une boîte à coins arrondis en CSS3</title>
   <style>
      .arrondi
       -moz-border-radius: 50px 50px 200px 50px;
       -webkit-border-radius: 50px 50px 200px 50px;
       -khtml-border-radius: 50px 50px 200px 50px;
       -ms-border-radius: 50px 50px 200px 50px;
       border-radius: 50px 50px 200px 50px;
       border: 1px solid black;
   </style>
  </head>
  <body>
   <hr>
     Boîte à bords arrondis en CSS3
     <br><br><br>>
   </body>
</html>
```

# Mise en page avec arrondis

Tous les éléments HTML5 peuvent être arrondis, y compris les nouveaux : <u>header, footer, nav, aside</u> et <u>article</u>. Dans cet exemple, des arrondis sont appliqués à tous les éléments <u>header, nav, article, section, footer, aside</u> et <u>img</u> du document.

#### Codeo45avant.htm

Saisissez le code HTML et le code CSS suivants. Complétez la feuille de styles mep.css pour définir des angles arrondis sur les éléments header, nav, article, section, footer, aside et img.

```
<!doctype html>
<html>
  <head>
   <meta charset=UTF-8" />
   <title>Mise en page en arrondis</title>
   <link rel="stylesheet" href="mep.css">
 </head>
 <body>
   <nav>
     <111>
       <a href="#">Accueil</a>
       <a href="#">Page1</a>
       <a href="#">Page 2</a>
       <a href="#">Contact</a>
     </111>
   </nav>
 <aside>
   Contenu texte ajouté sur le côté avec la balise <aside&qt;
 </aside>
  <article>
   <header>
     <h2>Titre de l'article</h2>
   </header>
   Contenu principal de l'article, suivi d'une image.
   <img src="grenouille.jpg" width="300px" /><br>
   <footer>
     <a href="#">Afficher les commentaires</a>
   </footer>
  </article>
</body>
</html>
```

```
header
 background-color: red;
nav
  float:left;
  width:20%;
 background-color:yellow;
article
 background-color: #66FFFF;
  width:80%;
aside
 float:right;
  width:20%;
  background-color:yellow;
footer
  clear:both;
  background-color: #99FF66;
```

# Solution

Codeo45apres.htm

Voici le code CSS à insérer dans la feuille de styles mep.css :

```
header, nav, article, section, footer, aside, img
{
    display: block;
    -moz-border-radius: 15px;
    -webkit-border-radius: 15px;
    -khtml-border-radius: 15px;
    border-radius: 15px;
    border: 1px solid black;
    padding: 10px;
}
```

# Ombrage sur un élément

L'ombrage des éléments HTML5 non textuels se fait via la propriété CSS3 box-shadow :

```
box-shadow: offset-horiz, offset-vert, rayon-ombrage, couleur
```

#### Où:

- <u>offset-horiz</u> est le décalage horizontal de l'ombrage en pixels.
- <u>offset-vert</u> est le décalage vertical de l'ombrage en pixels.
- <u>rayon-ombrage</u> est le rayon de l'ombrage en pixels.
- <u>couleur</u> est la couleur de l'ombrage.

Cette propriété présente deux avantages : elle est compatible avec la plupart des navigateurs actuels (y compris Internet Explorer) et elle s'applique à tous les éléments HTML5. Notez toutefois que vous aurez besoin des habituels préfixes <u>-moz-</u>, <u>-webkit-</u> et <u>-khtml-</u> pour (respectivement) les navigateurs Gecko, Webkit et Konqueror.

Appliquez un ombrage box-shadow à un élément img.

# Solution

#### Codeo46.htm

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset=UTF-8" />
    <title>Ombrage CSS3</title>
    <style>
      .ombrageCSS3
        width: 230px;
        box-shadow: 10px 10px 5px #bbb;
        -moz-box-shadow: 10px 10px 5px #bbb;
        -webkit-box-shadow: 10px 10px 5px #bbb;
        -khtml-box-shadow: 10px 10px 5px #bbb;
    </style>
 </head>
  <body>
    <img class="ombrageCSS3" src="grenouille.jpg" width="400px" />
 </body>
</html>
```

Essayez de modifier les quatre paramètres de la propriété box-shadow.

# Ombrage de texte

L'ombrage CSS3 d'éléments textuels se fait via la propriété <u>text-shadow</u>:

```
text-shadow : offset-horiz, offset-vert, épaisseur, couleur
```

#### Où:

- <u>offset-horiz</u> est le décalage horizontal de l'ombrage en pixels.
- <u>offset-vert</u> est le décalage vertical de l'ombrage en pixels.
- <u>épaisseur</u> est l'épaisseur de l'ombrage en pixels.
- <u>couleur</u> est la couleur de l'ombrage.

Appliquez la propriété text-shadow à un élément p qui contient du texte.

#### Codeo47.htm

# Solution

```
<!DOCTYPE html>
<html>
 <head>
   <meta charset=UTF-8" />
   <title>Ombrage de texte en CSS3</title>
   <style>
     .otexte
       font-size: 60px;
       text-shadow: 4px 4px 4px grey;
   </style>
 </head>
 <body>
   Ce texte est ombré
 </body>
</html>
```

# Mise en page avec arrondis et ombrages

Il est tout à fait possible d'appliquer conjointement les techniques CSS3 permettant de définir des arrondis sur des éléments, celles permettant d'ombrer des éléments non textuels et celles permettant d'ombrer des éléments textuels. Il suffit pour cela de définir les styles CSS nécessaires dans la feuille de styles.

# Transformations 2D

Avec CSS<sub>3</sub>, il est désormais très simple d'appliquer des transformations (translations, rotations, changements d'échelle). Et, bonne nouvelle, les transformations sont supportées par la plupart des navigateurs de dernière génération : Webkit (Safari, Chrome), Gecko (Firefox), Opera et Internet Explorer 9 et ultérieur.

Les transformations n'affectent pas le flux. En d'autres termes, les divers éléments qui environnent un élément ne seront pas déplacés suite à sa transformation. Cependant, vous pourrez utiliser la propriété <u>overflow</u> de l'élément transformé pour déterminer le rendu.

#### Dans cette section:

- Translations
- Rotations
- Changements d'échelle
- Inclinaisons
- Matrices

### Translations

Tous les éléments HTML5 peuvent subir une translation par l'intermédiaire de la fonction <u>translate()</u> appliquée à la propriété <u>transform</u>. Voici la syntaxe à utiliser :

```
transform: translate(h, v);
```

Où  $\underline{h}$  définit le déplacement horizontal (positif pour un déplacement vers la droite, négatif pour un déplacement vers la gauche) et  $\underline{v}$  le déplacement vertical (positif pour un déplacement vers le bas, négatif pour un déplacement vers le haut).

### Codeo49.htm

# Voici un exemple

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset=UTF-8" />
   <title>Translations</title>
    <style>
      div
        width: 200px;
        height: 60px;
        background-color: SkyBlue;
        margin: 20px;
        border-color: black;
        border-width: 5px;
        border-style: dashed;
      .droite
        -webkit-transform : translate(20px, 0px);
        -moz-transform : translate(20px, 0px);
        -o-transform : translate(20px, 0px);
        -ms-transform : translate(20px, 0px);
      .gauche
        -webkit-transform : translate(-20px, 0px);
        -moz-transform : translate(-20px, 0px);
        -o-transform : translate(-20px, 0px);
        -ms-transform : translate(-20px, 0px);
   </style>
 </head>
<body>
<div>
     Boîte sans déplacement
    </div>
   <div class="droite">
     Boîte déplacée à droite de 20px
   </div>
   <div class="gauche">
     Boîte déplacée à gauche de 20px
   </div>
  </body>
</html>
```

Entrainez-vous à effectuer d'autres translations en modifiant les paramètres de la

fonction translate.

### Rotations

Tous les éléments HTML5 peuvent subir une rotation *via* la fonction <u>rotate()</u> appliquée à la propriété <u>transform</u>. Voici la syntaxe à utiliser :

```
transform: rotate(rdeg);
```

Où  $\underline{\mathbf{r}}$  est l'angle de la rotation en degrés. Une valeur positive provoque une rotation dans le sens horaire, et une valeur négative provoque une rotation dans le sens trigonométrique.

# div { width: 200px; height: 60px; background-color: SkyBlue; margin-top: 120px; border-color: black; border-width: 5px; border-style: dashed; display: inline-block;

-webkit-transform : rotate(45deg);

-webkit-transform : rotate(-9odeg);

-moz-transform : rotate(-9odeg);
-o-transform : rotate(-9odeg);
-ms-transform : rotate(-9odeg);

<div style="border-width: 5px;">
La &lt;div&gt; originale

La <div&gt; après une rotation à droite de 45°

La <div&gt; après une rotation à gauche de 90°

<div class="rotdroite45">

<div class="rotgauche90">

-moz-transform : rotate(45deg);
-o-transform : rotate(45deg);

-ms-transform : rotate(45deg);

<!DOCTYPE html>

.rotdroite45

.rotgauche90

</style>
</head>
<body>

</div>

</div>

</div>
</body>

<meta charset=UTF-8" /> <title>Rotations</title>

<html>

<style>

Voici un

exemple

#### Codeo50.htm

Entrainez-vous à effectuer d'autres rotations en modifiant les paramètres de la fonction rotate.

# Changements d'échelle

Les éléments HTML5 peuvent subir un changement d'échelle quand la fonction <u>scale()</u> est appliquée à la propriété CSS3 <u>transform</u>. Voici la syntaxe à utiliser :

```
transform: scale(ex, ey);
```

Où <u>ex</u> et <u>ey</u> représentent les facteurs d'échelle selon les axes des abscisses et des ordonnées. Par défaut, si <u>ey</u> n'est pas spécifié, il est supposé égal à <u>ex</u>.

Codeo51.htm

### Voici un exemple

```
<html>
 <head>
 <meta charset=UTF-8"/>
 <title>Changements d'échelle</title>
 <style>
  .e200
   -webkit-transform : scale(2);
   -moz-transform : scale(2);
   -o-transform : scale(2);
   -ms-transform : scale(2);
  .e50
   -webkit-transform : scale(.5);
   -moz-transform : scale(.5);
   -o-transform : scale(.5);
   -ms-transform: scale(.5);
 </style>
 </head>
 <body>
 <img src="grenouille.jpg">
 <img src="grenouille.jpg" class="e200">
 <imq src="grenouille.jpg" class="e50">
 </body>
</html>
```

<!DOCTYPE html>

#### Inclinaison

Les éléments HTML5 peuvent être inclinés avec la fonction <u>skew()</u> appliquée à la propriété CSS3 <u>transform</u>. Voici la syntaxe à utiliser :

```
transform: skew(ax, ay);
```

Où <u>ax</u> et <u>ay</u> représentent les angles d'inclinaison selon les axes des abscisses et des ordonnées. Les valeurs positives de <u>ax</u> inclinent l'élément dans le sens trigonométrique. Quant aux valeurs positives de <u>ay</u>, elles l'inclinent vers l'arrière. Par défaut, si <u>ay</u> n'est pas spécifié, il est supposé égal à <u>0</u>. Dans ce cas, il n'y a donc pas d'inclinaison selon l'axe des ordonnées.

#### Exercice

Définissez le code HTML5 et CSS3 nécessaire pour incliner une image comme ceci :

- 15 degrés sur les axes X et Y
- -25 degrés sur les axes X et Y

Voici le résultat à obtenir



#### Solution

Codeo52.htm

```
<!DOCTYPE html>
<html>
 <head>
    <meta charset=UTF-8" />
   <title>Inclinaison</title>
    <style>
     imq
       margin-top: 60px;
       margin-left: 40px;
      .xy15
        -webkit-transform : skew(15deg, 15deg);
        -moz-transform : skew(15deg, 15deg);
        -o-transform : skew(15deg, 15deg);
        -ms-transform : skew(15deg, 15deg);
      .xy-25
        -webkit-transform : skew(-25deg, -25deg);
        -moz-transform : skew(-25deg, -25deg);
        -o-transform : skew(-25deg, -25deg);
        -ms-transform : skew(-25deg, -25deg);
   </style>
 </head>
  <body>
   <imq src="grenouille2.jpg">
   <img src="grenouille2.jpg" class="xy15">
   <imq src="grenouille2.jpg" class="xy-25">
 </body>
</html>
```

### Solution

#### Codeo53.htm

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset=UTF-8" />
   <title>Matrices de transformation</title>
    <style>
      imq
        margin-top: 60px;
        margin-left: 40px;
      .mat90
        -webkit-transform: matrix(0, 1, -1, 0, 0, 0);
        -moz-transform : matrix(0, 1, -1, 0, 0, 0);
        -o-transform : matrix(0, 1, -1, 0, 0, 0);
        -ms-transform : matrix(0, 1, -1, 0, 0, 0);
   </style>
  </head>
  <body>
   <img src="chat2.jpg">
   <img src="chat2.jpg" class="mat90">
  </body>
</html>
```

# Transitions

#### Dans cette section:

- Modification progressive d'une propriété
- Animer une transformation
- Déclencher une animation en JavaScript

#### **Transition**

Vous pouvez animer les transformations CSS3 avec des pseudo-classes ou du code JavaScript comme facteurs déclenchants. Ainsi par exemple, une rotation ou une translation peut être affichée progressivement lorsque l'utilisateur déplace le pointeur ou clique sur un élément quelconque. Cette fabuleuse nouveauté réside dans la propriété CSS3 <u>transition</u> dont voici la syntaxe :

```
transition: propriété durée type délai;

Où:

propriété est une propriété CSS: background, opacity ou width par exemple.

durée est la durée de l'animation en secondes.

type est le type de l'animation. Ce paramètre peut prendre l'une des valeurs suivantes:

- linear: aucun effet;

- cubic-bezier(x1, y1, x2, y2): courbe de Bézier dont les points Po et P3 ont pour valeur (o,o) et (1,1);

- ease: équivalent à cubic-bezier(0.25, 0.1, 0.25, 1.0);

- ease-in: (0.42, 0, 1.0, 1.0);

- ease-out: (0, 0, 0.58, 1.0);

- ease-in-out: (0.42, 0, 0.58, 1.0).

délai est le délai avant le déclenchement de l'animation.
```

# Modification progressive d'une couleur

Pour modifier progressivement une couleur, vous devez :

- définir une classe contenant la propriété <u>transition: color</u> et l'affecter à l'élément concerné ;
- définir la pseudo-classe associée et fixer la couleur à atteindre.

```
<!DOCTYPE html>
< ht.ml>
 <head>
   <meta charset=UTF-8" />
   <title>Animation de la couleur</title>
   <style>
      .anim-couleur
       font-size: 40px;
       transition: color 2s ease-in;
       -moz-transition: color 2s ease-in;
       -webkit-transition: color 2s ease-in;
       -o-transition: color 2s ease-in;
      .anim-couleur:hover
       color: red;
   </style>
  </head>
 <body>
   Placez le pointeur sur ce texte
 </body>
</ht.ml>
```

Codeo54.htm

### Modification progressive de la transparence

La propriété <u>opacity</u> permet de régler la transparence d'un élément. Pour faire disparaître un élément au survol, il suffit de définir :

- Une classe dans laquelle la propriété <u>transition: opacité</u> est initialisée et l'affecter à l'élément concerné.
- La pseudo-classe correspondante dans laquelle l'opacité cible est spécifiée.

### Exercice

Insérez une image dans un document HTML5 et faites-la disparaître progressivement lorsqu'elle est survolée par le pointeur de la souris.

Pour cela, vous utiliserez la propriété CSS3 transition.

#### Solution

#### Codeo55.htm

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset=UTF-8" />
    <title>Animation de la transparence</title>
    <style>
      .anim-opacite
        transition: opacity 2s linear;
        -moz-transition: opacity 2s linear;
        -webkit-transition: opacity 2s linear;
        -o-transition: opacity 2s linear;
      .anim-opacite:hover
        opacity: 0;
    </style>
  </head>
  <body>
    <img class="anim-opacite" src="grenouille.jpg" />
  </body>
</html>
```

### Animer une transformation

En utilisant conjointement les propriétés CSS3 <u>transform</u> et <u>transition</u>, vous n'aurez besoin que de quelques lignes de code pour animer une transformation.

#### Exercice

Insérez une image dans un document HTML5 et déplacez-la progressivement de 50 pixels vers la droite lorsqu'elle est survolée par la souris.

#### Solution

<!DOCTYPE html>

```
<html>
  <head>
    <meta charset=UTF-8" />
    <title>Animation d'une translation</title>
    <style>
      .animation
        -moz-transition: all 2s linear;
        -webkit-transition: all 2s linear;
        -o-transition: all 2s linear;
        transition: all 2s linear;
      .animation:hover
        -moz-transform: translate(50px, 0);
        -webkit-transform: translate(50px, 0);
        -o-transform: translate(50px, 0);
        transform: translate(50px, 0);
    </style>
  </head>
  <body>
    <img class="animation" src="grenouille2.jpg" />
  </body>
</html>
```

#### Codeo56.htm

La valeur <u>all</u> étant appliquée au premier paramètre, toutes les propriétés modifiées pour cet élément seront automatiquement animées. Ici, seule une translation est appliquée à l'image, mais on aurait pu également ajouter une disparition progressive, une rotation ou un quelconque autre effet.

Vous pourriez remplacer <u>all</u> par <u>transform</u> pour obtenir le même effet, car seule la propriété transform est affectée dans cette transition

### Exercice

Modifiez le code de l'exercice précédent pour effectuer simultanément trois transformations lorsque l'image est survolée :

- Une translation de 50 pixels;
- Une rotation de 360 degrés;
- Une modification de l'opacité pour que l'image devienne entièrement transparente.

### Solution

Codeo57.htm

Seul le code de la pseudo-classe .animation:hover doit être modifié. Remarquez l'utilisation conjointe des fonctions translate() et rotate()

```
.animation:hover
{
    -moz-transform: translate(400px, 0) rotate(360deg);
    -webkit-transform: translate(400px, 0) rotate(360deg);
    -o-transform: translate(400px, 0) rotate(360deg);
    transform: translate(400px, 0) rotate(360deg);
    opacity: 0;
}
```

## Déclencher une animation en JavaScript

Il peut être intéressant de déclencher une animation par une fonction JavaScript. Cette fonction étant associée à la propriété <u>onClick</u> d'un élément HTML <u>button</u> par exemple, l'animation sera déclenchée lorsque l'utilisateur cliquera sur un bouton.

Le code de la diapositive suivante montre comment déclencher une animation sur l'attribut <u>top</u> d'une <u><div></u> à la suite d'un clic sur un <u><button></u>, puis comment restituer la position initiale de la <u><div></u> à la suite d'un clic sur un autre bouton.

```
<!DOCTYPE html>
<ht.ml>
  <head>
    <meta charset=UTF-8" />
   <title>Animation déclenchée en JavaScript</title>
                                                                              Codeo58.htm
    <style>
      div
       position: absolute;
       left: 100px;
       top: 100px;
       height: 100px;
       width: 100px;
       background: Yellow;
        -webkit-transition: top 1s ease-in;
        -moz-transition: top 1s ease-in;
        -o-transition: top 1s ease-in;
        transition: top 1s ease-in;
   </style>
    <script>
      function animer()
       var el = document.getElementById('madiv');
        el.setAttribute("style", "top: 150px;");
      function reinit()
       var el = document.getElementById('madiv');
        el.setAttribute("style", "top:100px;");
   </script>
  </head>
  <body>
    <button onclick="animer()">Cliquez ici pour animer la &lt;div&qt;</putton>
   <button onclick="reinit()">Cliquez ici pour réinitilaiser la &lt;div&gt;</button>
   <div id="madiv">Une simple &lt;div&gt;</div>
  </body>
</html>
```

# Formulaires

#### Dans cette section:

- Eléments dédiés aux formulaires
- Placeholder
- Autofocus
- Validation de données
- Labels implicites
- Aligner les éléments d'un formulaire
- Arrondir les angles des éléments d'un formulaire
- Modifier l'allure d'un bouton

### Formulaires

Si les pages web servent à diffuser des informations, elles sont également en mesure d'en recueillir par l'intermédiaire de formulaires. Les données saisies sont alors communiquées à un programme ou expédiées sans traitement dans une boîte e-mail.

Les diapos suivantes résument les différentes balises utilisables en HTML5.

Balise	Signification
<pre><fieldset>contenu</fieldset></pre>	Groupe d'options permettant de rassembler plusieurs champs.
<pre><legend>texte</legend></pre>	Légende affichée dans un élément <u>fieldset</u> .
<a href="mailto:square;"><label>texte</label></a> /label>	Simple légende textuelle.
<pre><input id="identifiant" name="Nom" type="text" value="valeur"/></pre>	Zone de texte.
<input <="" id="identifiant" name="Nom" p="" type="button"/> value="Légende du bouton">	Bouton.
<pre><input id="identifiant" src="image.jpg" type="image"/></pre>	Un bouton contenant une image.
<pre><input id="identifiant" name="Nom" type="password" value="valeur par défaut"/></pre>	Zone de saisie d'un mot de passe.
<pre><input id="identifiant" name="Nom" type="checkbox" value="Valeur"/></pre>	Case à cocher.
<pre><input id="identifiant" name="Nom" type="radio" value="Valeur"/></pre>	Bouton radio.

Balise	Signification
<input id="identifiant" name="Nom" type="submit" value="Texte affiché sur le&lt;/p&gt;&lt;/th&gt;&lt;th&gt;Bouton d'envoi, pour envoyer les&lt;/th&gt;&lt;/tr&gt;&lt;tr&gt;&lt;th&gt;bouton"/>	données du formulaire au serveur.
<input id="identifiant" name="Nom" type="reset" value="Texte affiché sur le bouton"/>	Bouton de réinitialisation du formulaire.
<input id="identifiant" name="Nom" title="Date(jj-mm-aaaa)" type="date" value="Une&lt;/p&gt;&lt;/th&gt;&lt;th&gt;Champ spécialisé dans la saisie de&lt;/th&gt;&lt;/tr&gt;&lt;tr&gt;&lt;th&gt;&lt;u&gt;date"/>	dates (seulement pour les
	navigateurs Opera et Chrome alors
	que j'écris ces lignes [voir
	Figure 12.1]).
<input <="" id="identifiant" name="Nom" p="" title="Heure(hh-mm)" type="time"/>	Champ spécialisé dans la saisie
value="10:00">	d'heures (seulement pour les
	navigateurs Opera et Chrome alors
	que j'écris ces lignes).
<pre><input id="identifiant" name="Nom" type="datetime"/></pre>	Champ spécialisé dans la saisie de
	date et heures (seulement pour les
	navigateurs Opera et Chrome alors
	que j'écris ces lignes).
<input id="identifiant" name="Nom" type="number" value="55"/>	Champ spécialisé dans la saisie des
	nombres.
<input id="identifiant" name="Nom" type="color" value="red"/>	Palette de couleurs (aucun navigateur
	n'est encore compatible).
<u><input id="identifiant" list="datalist" name="Nom" type="search" value="Valeur par&lt;/u&gt;&lt;/th&gt;&lt;th&gt;Zone de texte permettant d'effectuer&lt;/th&gt;&lt;/tr&gt;&lt;tr&gt;&lt;th&gt;&lt;u&gt;défaut"/></u>	des recherches.

Balise	Signification
<input id="identifiant" name="Nom" type="email"/>	Champ de saisie spécialisé pour
	recevoir des adresses e-mail.
<input id="identifiant" name="Nom" type="url"/>	Champ de saisie spécialisé pour
	recevoir des adresses URL.
<pre><input id="identifiant" name="Nom" type="file"/></pre>	Désignation d'un fichier local.
<pre><datalist id="identifiant"><option value="valeur1"><option< pre=""></option<></option></datalist></pre>	Facilite la saisie en implémentant
<u>value="valeurN"&gt;</u>	l'autocomplete sur plusieurs valeurs
	prédéfinies dans des éléments <u>option</u>
	(aucun navigateur n'est encore
	compatible).
<textarea cols="Nombre colonnes" id="identifiant" rows="Nombre lignes">Texte&lt;/th&gt;&lt;th&gt;Zone de saisie multiligne.&lt;/th&gt;&lt;/tr&gt;&lt;tr&gt;&lt;th&gt;par défaut</textarea>	
<pre><select id="identifiant" name="Nom"><option value="valeur1"><option< pre=""></option<></option></select></pre>	Liste déroulante.
value="valeurN">	
<pre><select id="identifiant" name="Nom" size="4"><option value="valeur1"><option< pre=""></option<></option></select></pre>	· ·
value="valeurN">	affichés).
<pre><option id="identifiant" name="Nom" size="4"><option< pre=""></option<></option></pre>	Zone de liste (ici, quatre éléments sont
value="valeur1"> <option value="valeurN"></option>	affichés).
<output name="Nom"></output>	Affichage d'un résultat.

#### Formulaires

Les éléments qui composent un formulaire doivent obligatoirement être inclus entre des balises <form> et </form> dont la syntaxe est la suivante :

```
<form name="nom" action="programme ou action" method="get|post">
</form>
```

#### Où:

- <u>name</u> est le nom du formulaire.
- <u>action</u> est le nom du programme auquel les données doivent être transmises ou l'action à accomplir (<u>mailto:adresse e-mail</u> par exemple).
- <u>method</u> est la méthode de transmission des données du formulaire : <u>get</u> poste les données dans l'adresse URL et <u>post</u> dans le corps de la requête. La seconde méthode est préférable à la première car elle n'affiche pas en clair les données saisies dans le formulaire (ce qui pourrait être gênant si des données confidentielles sont transmises). De plus, la taille des données n'est pas limitée, alors qu'elle ne peut dépasser 256 octets dans la méthode <u>get</u>.

```
Voici un exemple de formulaire
```

```
<!DOCTYPE html>
<ht.ml>
  <head>
    <meta charset=UTF-8" />
   <title>Un formulaire</title>
 </head>
  <body>
   <form name="MonFormulaire" method="post" action="traitement.php">
      <fieldset>
        <legend>Légende du &lt;fieldset&gt;</legend>
       Contenu du du <fieldset&gt;
      </fieldset>
      <br><label>Un champ texte</label>
      <input type="text" name="Nom" value="valeur">
      <br><label>Un bouton</label>
      <input type="button" name="Nom" value="Légende du bouton">
      <br><label>Un bouton image</label>
      <input type="image" src="commande.jpg">
      <br><label>Un champ mot de passe</label>
      <input type="password" name="Nom" value="valeur par défaut">
      <br><label>Une case à cocher</label>
      <input type="checkbox" name="Nom" value="Valeur">
      <label>Un bouton radio</label>
      <input type="radio" name="Nom" value="Valeur">
      <br><label>Des boutons &lt;submit&qt; et &lt;reset&qt;</label>
      <input type="submit" name="Nom" value="Envoyer">
      <input type="reset" name="Nom" value="Annuler">
      <br><label>Champs spécialisés date et heure</label>
      <input type="date" name="Nom" title="Date(aaaa-mm-jj)" value="2010-11-22">
      <input type="time" name="Nom" title="Heure(hh-mm)" value="10:00">
      <input type="datetime" name="Nom">
      <br><label>Saisie de nombres entiers autour de la valeur 55</label>
      <input type="number" name="Nom" value="55">
```

...suite <u>Codeo59.htm</u>

```
<br><label>Une zone de texte spécialisée recherche</label>
      <input type="search" name="Nom" list="datalist" value="Valeur par</pre>
défaut">
      <br/>clabel>Un curseur pour sélectionner des valeurs numériques/label>
      <input type="range" min="-500" max="500" value="0" step="10"</pre>
name="Nom">
      <br><label>Un bouton pour désigner un fichier local</label>
      <input type="file" name="Nom">
      <br><label>Une zone de saisie multiligne</label>
      <textarea cols="40" rows="4">Texte par défaut</textarea>
      <br><label>Liste déroulante et zone de liste</label>
      <select name="Nom">
        <option>Valeur 1</option>
        <option>Valeur 2</option>
        <option>Valeur 3</option>
        <option>Valeur 4</option>
        <option>Valeur 5</option>
      </select>
      <select size="2" name="Nom">
        <option>Valeur 1</option>
        <option>Valeur 2</option>
        <option>Valeur 3</option>
        <option>Valeur 4</option>
        <option>Valeur 5</option>
      </select>
      <br><label>Résultat d'un calcul : </label>
      <output name="resultat">123</output>
    </form>
  </body>
</html>
```

### Placeholder

#### Codeo6o.htm

L'attribut <u>placeholder</u> des balises <u><input type="text"></u> permet d'ajouter un texte grisé dans la zone de saisie. Ce texte est visible à l'affichage de la page, puis il disparaît lorsque la zone de saisie a le focus, laissant ainsi l'utilisateur taper son texte.

Essayez ce code

### Autofocus

#### Codeo61.htm

Pour donner le focus à un champ d'un formulaire, il suffit d'utiliser l'attribut autofocus sur le champ concerné.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset=UTF-8" />
    <title>Utilisation de l'attribut Autofocus</title>
  </head>
  <body>
    <form action="action.php" method="post">
      <fieldset>
        <legend>Positionnement du point d'insertion dans le premier champ input</legend>
        <label>Nom</label>
        <input type="text" id="champ1" autofocus>
        <label>Prénom</label>
        <input type="text" id="champ2">
        <label>Adresse
        <input type="text" id="champ3">
        <label>Ville</label>
        <input type="text" id="champ4">
      </fieldset>
    </form>
  </body>
</html>
</html>
```

Essayez ce code

#### Validation de données

Lorsqu'un champ de saisie doit obligatoirement être rempli, il suffit de lui affecter l'attribut <u>required</u>. Dans ce formulaire, le champ de saisie <u>Nom</u> est obligatoire. Si l'utilisateur clique sur le bouton Submit sans l'avoir renseigné, un message d'erreur est généré et le formulaire n'est pas envoyé.

Essayez ce code

#### Validation de données

Le champ d'action des attributs <u>required</u> est bien plus étendu, puisqu'il permet également de valider des données complexes (telles que des adresses e-mail ou des URL). À titre d'exemple, le formulaire ci-après teste la validité des champs <u>email</u> et <u>url</u>. Lorsqu'on clique sur le bouton Submit, un message d'erreur s'affiche si un de ces deux champs n'est pas conforme.

```
<!DOCTYPE html>
                                                      Codeo63.htm
<html>
  <head>
    <meta charset=UTF-8" />
    <title>Validation e-mail et url</title>
  </head>
  <body>
  <form id="validation">
   <label>Nom</label> <input name="Nom"><br>
   <label>e-mail</label> <input type=email name="email" required><br>
   <label>Adresse URL</label> <input type=url name="url" required><br>
   <label>Commentaire</label> <textarea name="comment"></textarea><br>
   <input type=submit value="Valider">
  </form>
  </body>
                                                            Essayez ce code
</html>
```

### Validation de données

Pour faciliter la reconnaissance des champs obligatoires, il est possible d'utiliser quelques lignes de CSS. Par exemple, cette ligne affecte un arrière-plan de couleur jaune aux champs <input> dont l'attribut required est spécifié :

```
input:required { background:yellow }
```

# Labels implicites et explicites

Les labels spécifiés dans un formulaire peuvent être reliés de façon implicite ou explicite à un champ de saisie, qu'il s'agisse d'un <u><input type="text"></u>, d'un <u><input type="checkbox"></u>, d'un <u><textarea></u> ou d'un quelconque autre champ de saisie.

On réalise la liaison implicite en définissant le champ de saisie comme enfant de l'élément <u>label</u>, c'est-à-dire en l'insérant entre les balises <u><label></u> et <u></label></u>.

La liaison explicite se fait à travers l'attribut <u>for</u> de l'élément <u>label</u>, qui fait référence à l'<u>id</u> du champ cible.

# Labels implicites et explicites

```
<!DOCTYPE html>
             <ht.ml>
               <head>
                 <meta charset=UTF-8" />
                 <title>Alignement d'éléments dans un formulaire</title>
               </head>
               <body>
              <form id="validation">
Label implicite
               <label>Nom<input id="Nom"></label><br>
                                                                 Label explicite
                <label>Prénom</label> <input id="prenom"><br> 
                <label for="adresse">Adresse</label><textarea id="adresse"></textarea><br>
               </form>
               </body>
             </html>
```

Les labels implicites provoquent des problèmes d'accessibilité. Vous utiliserez des labels explicites légèrement modifiés.

# Labels implicites et explicites

La valeur de l'attribut <u>id</u> de l'élément <u>input</u> et celle de l'attribut <u>for</u> de l'élément <u>label</u> doivent être identiques.

# Aligner les éléments d'un formulaire

D'une façon générale, les divers éléments utilisés dans un formulaire (labels, zones de texte, cases à cocher, listes déroulantes, etc.) sont de type <u>inline</u>. Pour pouvoir modifier leur taille en utilisant des propriétés CSS, vous devez les redéfinir en <u>inline-block</u>.

Dans la diapo suivante, nous allons aligner à gauche deux zones de texte et un <u>textarea</u> en fixant la largeur des labels qui précèdent chacun d'eux.

### Codeo64.htm

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset=UTF-8" />
    <title>Labels implicites et explicites</title>
    <style>
     label
        display: inline-block;
       width: 80px;
    </style>
  </head>
  <body>
    <form id="validation">
    <label>Nom</label><input id="Nom"><br>
    <label>Prénom</label><input id="Prenom"><br>
    <label>Adresse//textarea id="Adresse" rows="8" cols="30"></textarea><br><br>
    <input type="button" name="Nom" value="Valider">
  </form>
  </body>
</html>
```

Essayez ce code

# Aligner les éléments d'un formulaire

Une autre technique consiste à utiliser un positionnement absolu.

Sans la diapositive suivante, les éléments <u>input</u> et <u>textarea</u> sont alignés à 85px du bord gauche.

La propriété <u>top</u> de l'élément <u>button</u> doit également être initialisée pour éviter un chevauchement du <u>textarea</u> et du <u>button</u>.

### Codeo65.htm

```
<!DOCTYPE html>
<html>
 <head>
   <meta charset=UTF-8" />
   <title>Alignement d'éléments dans un formulaire</title>
   <style>
     input, textarea
       position: absolute;
      left: 85px;
     .button
       position: absolute;
      left: 85px;
       top: 200px;
   </style>
 </head>
 <body>
   <form id="validation">
   <label>Nom</label><input id="Nom"><br>
   <label>Prénom</label><input type=email id="email"><br>
   <label>Adresse
   <input type="button" name="Nom" value="Valider" class="button">
  </form>
 </body>
</html>
```

Essayez ce code

### Arrondir les angles des éléments d'un formulaire

La propriété CSS3 <u>border-radius</u> permet d'arrondir les angles de tous les éléments HTML5, y compris des éléments utilisés dans un formulaire. Ici par exemple, les angles de deux zones de texte, d'un <u>textarea</u> et d'un bouton sont arrondis.

```
<!DOCTYPE html>
<html>
                                                                                Codeo66.htm
  <head>
    <meta charset=UTF-8" />
    <title>Arrondir les angles d'un formulaire</title>
    <style>
      .arrondi
        -moz-border-radius: 15px;
        -webkit-border-radius: 15px;
        -khtml-border-radius: 15px;
        -ms-border-radius: 15px;
       border-radius: 15px;
        border: 1px solid black;
    </style>
 </head>
  <body>
    <form id="validation">
    <label>Nom</label><input id="Nom" class="arrondi"><br>
    <label>Prénom</label><input id="Prenom" class="arrondi"><br>
    <label>Adresse/label><textarea id="adresse" class="arrondi" rows="8" cols="30"></textarea><br><br><br><br><br>
    <input type="button" name="Nom" value="Valider" class="arrondi">
   </form>
  </body>
</html>
```

Essayez ce code

## Modifier l'allure d'un bouton avec un style

Plusieurs techniques CSS3 peuvent être appliquées pour définir le style d'un bouton :

- incorporation de polices sur le serveur ;
- gradient de couleur pour définir l'arrière-plan;
- différents arrondis sur les angles ;
- ombre portée.

### Exercice

Ecrivez du code HTML pour donner l'allure suivante à un bouton :



```
Codeo67.htm
```

```
<!DOCTYPE html>
<html>
 <head>
    <meta charset=UTF-8" />
   <title>Un bouton de formulaire entièrement redéfini</title>
   <style>
     @font-face {
       font-familly: algerian;
       src: url('ALGER.TTF');
      .arrondi {
       width: 220px;
       height: 50px;
       -moz-border-radius: 50px 50px 200px 50px; -webkit-border-radius: 50px 50px 200px 50px;
       -khtml-border-radius: 50px 50px 200px 50px; -ms-border-radius: 50px 50px 200px 50px;
       border-radius: 50px 50px 200px 50px; border: 1px solid black;
       background: -moz-linear-gradient(top, #40FFFF, #C0FFFF);
       background: -webkit-gradient(linear, 0% 0%, 0% 100%, from(#40FFFF));
       box-shadow: 5px 5px 5px grey;
       -moz-box-shadow: 5px 5px 5px grey; -webkit-box-shadow: 5px 5px 5px grey;
       -khtml-box-shadow: 5px 5px 5px grey; box-shadow: 5px 5px grey;
       font-family: 'algerian'; font-size: 40px;
      .arrondi:hover {
       background: -moz-linear-gradient(top, #COFFFF, #40FFFF);
       background: -ms-linear-gradient(top, #COFFFF, #40FFFF);
       background: -webkit-gradient(linear, 0% 0%, 0% 100%, from(#COFFFF)), to(#40FFFF));
   </style>
 </head>
  <body>
   <form id="validation">
   <input type="button" name="bouton1" value="Valider" class="arrondi">
  </form>
 </body>
</html>
```

## Tableaux

En tant que développeur, vous utilisez certainement des éléments <u>table</u> pour organiser vos données tabulaires. En y ajoutant quelques lignes de CSS<sub>3</sub>, vous allez voir à quel point les tableaux que vous manipuliez jusqu'ici peuvent changer d'allure.

#### Dans cette section:

- Bordures autour des cellules
- Bordures séparées ou fusionnées
- Espacer les cellules
- Mettre en forme les cellules vides
- Habiller un tableau
- Arrière-plan des cellules
- Différentier lignes paires et impaires
- etc.

### Bordures des tableaux

La propriété <u>border</u>, utilisable dans les éléments <u>table</u>, <u>th</u> et <u>td</u> permet (respectivement) de définir la bordure autour du tableau, des cellules de titre et des cellules de données. Voici sa syntaxe :

```
border: épaisseur style couleur;
```

#### Où:

- <u>épaisseur</u> est l'épaisseur de la bordure : <u>thin</u>, <u>medium</u>, <u>thick</u> ou une valeur numérique exprimée en pixels (<u>px</u>), points (<u>pt</u>), pourcentage (<u>%</u>), ou une autre unité de mesure CSS.
- <u>style</u> est le style de la bordure : <u>dashed</u>, <u>dotted</u>, <u>double</u>, <u>solid</u>, <u>insert</u>, <u>groove</u>, <u>ridge</u> ou <u>outset</u>.
- <u>couleur</u> est la couleur de la bordure : une couleur prédéfinie, comme <u>Silver</u> ou <u>DarkBlue</u>, ou une valeur hexadécimale, comme <u>#341276</u>.

### Exercice

Définissez un tableau de 3 lignes sur 3 colonnes. Tracez une bordure pointillée noire de 4 pixels d'épaisseur autour d'un tableau et une bordure continue rouge de 1 pixel d'épaisseur autour des cellules de titre <a href="mailto:th>"><a href="mailto:th)"><a href

## Solution

```
<!DOCTYPE html>
<ht.ml>
 <head>
  <meta charset=UTF-8" />
  <title>Bordures autour d'un tableau</title>
  <style>
    table
     border: 4px dashed black;
    th, td
     border: 1px solid red;
  </style>
 </head>
 <body>
 Titre colonne 1
    Titre colonne 2
    Titre colonne 3
  Elément 1 ligne 1
    Elément 2 ligne 1
    Elément 3 ligne 1
  Elément 1 ligne 1
    Elément 2 ligne 1
    Elément 3 ligne 1
  </body>
</html>
```

Codeo68.htm

## Bordures séparées ou fusionnées

La propriété <u>border-collapse</u> détermine si les cellules d'un tableau sont séparées (valeur <u>separate</u>) ou fusionnées (valeur <u>collapse</u>).

Les deux tableaux représentés ci-dessous sont identiques, à ceci près que le premier a une propriété border-collapse initialisée à separate et le second à collapse.

Titre colonne 1	Titre colonne 2	Titre colonne 3
Elément 1 ligne 1	Elément 2 ligne 1	Elément 3 ligne 1
Elément 1 ligne 1	Elément 2 ligne 1	Elément 3 ligne 1

Titre colonne 1	Titre colonne 2	Titre colonne 3
Elément 1 ligne 1	Elément 2 ligne 1	Elément 3 ligne 1
Elément 1 ligne 1	Elément 2 ligne 1	Elément 3 ligne 1

## Espacer les cellules

La propriété border-spacing détermine l'espace qui sépare les bordures des cellules adjacentes d'un tableau. Dans cet exemple, nous définissons un tableau encadré par une bordure continue noire épaisse de 1 pixel. Les cellules () sont séparées (border-collapse: separate;) et espacées de 10 pixels (border-spacing: 10px;).

Codeo69.htm

```
<!DOCTYPE html>
<ht.ml>
 <head>
   <meta charset=UTF-8" />
  <title>Espacer les cellules d'un tableau</title>
  <style>
    table
      border: 1px solid black;
      text-align: center;
      border-collapse: separate;
      border-spacing: 10px;
    td
      width: 120px;
      border: 1px solid red;
  </style>
 </head>
 <body>
 <t.r>
   Date
   Articles
   Prix
   <t.r>
   12/08
   21
   5412
   </t.r>
   17/08
   14
   1519
  </body>
</html>
```

## Mettre en forme les cellules vides

Selon la valeur qui lui est affectée, la propriété <u>empty-cells</u> affiche (<u>show</u>) ou cache (<u>hide</u>) les bordures autour des cellules vides d'un tableau.

#### Dans le code de la diapositive suivante :

- Le style <u>table</u> définit les caractéristiques générales des tableaux : bordure (<u>border</u>) et couleur d'arrière-plan (<u>background-color</u>).
- Les styles <u>#table1</u> et <u>#table2</u> définissent respectivement le style des tableaux d'identificateur <u>table1</u> et <u>table2</u>.
- Le style <u>td</u> définit l'encadrement des cellules.

Le code HTML affiche deux tableaux d'identificateur <u>table1</u> et <u>table2</u>. Dans le premier les cellules vides sont encadrées. Dans le second, l'encadrement n'est pas visible

```
<!DOCTYPE html>
<ht.ml>
 <head>
  <meta charset=UTF-8" />
  <title>Bordure des cellules vides</title>
  <style>
    table
     border: 2px solid red;
     background-color: #dfd;
    #table1 { empty-cells: show; }
    #table2 { empty-cells: hide; }
    td { border: 1px dashed black;}
  </style>
 </head>
 <body>
  Elément 1 ligne 1
     Elément 3 ligne 1
    Elément 2 ligne 2
     Elément 3 ligne 2
    <br>
    Elément 1 ligne 1
     Elément 3 ligne 1
    Elément 2 ligne 2
     Elément 3 ligne 2
    </body>
</html>
```

### Codeo70.htm

## Habiller un tableau

Pour définir les caractéristiques d'un tableau, il suffit d'initialiser les propriétés des éléments <u>table</u>, <u>caption</u>, <u>th</u>, <u>tr</u> et <u>td</u> en conséquence. Vous pouvez également utiliser un ou plusieurs sélecteurs d'identificateurs (<u>#</u>, <u>first-child</u>, etc.) pour formater différemment certains éléments du tableau.

Le code de la diapositive suivante modifie le style des éléments usuels d'un tableau pour lui donner une tout autre allure.

```
<!DOCTYPE html>
                                         tr#dernier
<ht.ml>
 <head>
                                            background-color: silver;
   <meta charset=UTF-8" />
  <title>Habillage d'un tableau</title>
                                           </style>
                                         </head>
   <style>
   table
                                         </body>
                                           border: 1px solid silver;
                                            <caption>Recettes budg&eacute;taires par r&eacute;gion/caption>
    background-color: #dfd;
                                             En millions de F
   caption
                                            <th rowspan = 2>1985 en F/hab.</th>
                                             text-align: left;
                                             color: red;
                                             1984
    font-weight: bold;
                                            1985
    font-size: 130%;
                                             <t.r>
    border-bottom: 5px solid blue;
                                            Ile-de-France
    margin-bottom: 10px;
                                             399258
                                             415920
                                            178,8
   th
                                             border: 1px solid red;
                                             <t.r>
    empty-cells: hide;
                                            Midi-Pyré nné es
    background-color: yellow;
                                             73961
                                             63826
                                            219,4
   tr
                                             text-align: center;
                                             Centre
   td
                                             43638
                                            55802
    border: 1px dashed black;
                                            185,9
                                             td:first-child
                                           </body>
    color: red;
                                        </html>
    font-weight: bold;
    border: 0;
    text-align: right;
```

Codeo71.htm

## Arrière-plan des cellules

L'arrière-plan des cellules peut être une couleur uniforme ou une image. Dans le premier cas, vous utiliserez la propriété <u>background-color</u>, et dans le second, la propriété <u>background: url</u>.

Dans le code de la diapositive suivante, nous appliquons trois textures d'arrière-plan, librement téléchargées sur le site www.clipart-fr.com. La texture *fond1.jpg* est appliquée à la première ligne du tableau, la texture *fond2.jpg* à la première colonne du tableau (hormis la première ligne) et la texture *fond3.jpg* à toutes les autres cellules .

### Codeo72.htm

```
<!DOCTYPE html>
<html>
 <head>
 <meta charset=UTF-8"/>
 <title>Arrière-plan des cellules</title>
 <style>
  table
   border: 1px solid black;
   text-align: center;
   color: blue;
   font-weight: bold;
   font-size: 200%;
  td {width: 200px;}
  td.fond1 { background:url(fond1.jpg); }
  td.fond2 { background:url(fond2.jpg); }
  td.fond3 { background:url(fond3.jpg); }
 </style>
 </head>
```

```
<body>
Date
Articles
Prix
12/08
21
5412
14/08
10
2215
16/08
15
3219
19/08
17
2719
</body>
</html>
```

# Différentier lignes paires et impaires

Pour alterner la couleur des lignes d'un tableau, il suffit d'utiliser deux <u>background-color</u>. Le plus simple consiste à employer le sélecteur CSS3 <u>tr:nth-child()</u>.

### Exercice

Définissez un tableau de 5 lignes et 3 colonnes et écrivez le code nécessaire pour obtenir le résultat suivant :

Date	Articles	Prix
12/08	21	5412
14/08	10	2215
16/08 19/08	15	3219
19/08	17	2719

### Codeo73.htm

```
<!DOCTYPE html>
<html>
<head>
  <meta charset=UTF-8"/>
  <title>Différenciation des lignes paires/impaires</title>
  <style>
   table
    border: 1px solid black;
   tr:nth-child(odd)
    background-color: silver;
   tr:nth-child(even)
    background-color: white;
   td
    width: 200px;
  </style>
 </head>
```

```
<body>
Date
Articles
Prix
12/08
21
5412
14/08
10
2215
16/08
15
3219
19/08
17
2719
</body>
</html>
```

## Aligner les cellules dans un tableau

Vous utiliserez les propriétés <u>text-align</u> et <u>vertical-align</u> pour (respectivement) définir l'alignement horizontal et vertical des cellules dans un tableau.

La propriété <u>text-align</u> peut prendre les valeurs <u>left</u> (gauche), <u>right</u> (droite), <u>center</u> (centré) ou <u>justify</u> (justifié).

La propriété <u>vertical-align</u> peut prendre les valeurs <u>top</u> (alignement sur la partie supérieure de la cellule), <u>bottom</u> (alignement sur la partie inférieure de la cellule) ou <u>middle</u> (centré sur la hauteur de la cellule).

L'exemple de la diapositive suivante utilise les trois valeurs possibles pour les propriétés <u>text-align</u> et <u>vertical-align</u> à travers les classes <u>c1</u>, <u>c2</u> et <u>c3</u>, successivement appliquées aux cellules des première, deuxième et troisième lignes de la table.

Afin que vous puissiez bien visualiser l'effet des différents types d'alignements, la largeur des cellules a été fixée à 200 pixels (width: 200px;) et la hauteur des cellules à 40 pixels (height: 40px;).

```
<!DOCTYPE html>
<html>
 <head>
  <meta charset=UTF-8" />
  <title>Alignement des cellules</title>
                                  <body>
                                     <style>
   table { border: 1px solid black; }
                                      <t.r>
   td
                                      Date
                                      Articles
     border: 1px dashed red;
                                      Prix
     width: 200px;
                                      height: 40px;
                                      12/08
    td.c1
                                      21
     text-align: center;
                                      5412
     vertical-align: middle;
                                      17/08
   td.c2
                                      14
     text-align: left;
                                      1519
                                      vertical-align: top;
                                     td.c3
                                   </body>
                                  </html>
     text-align: right;
     vertical-align: bottom;
  </style>
 </head>
```

Codeo74.htm

# JavaScript

Cette section va vous montrer comment utiliser le langage :

- JavaScript et HTML
- Objets, propriétés, fonctions et méthodes
- Définition d'un nouveau type d'objet
- Utilisation d'un objet par défaut
- Tour d'horizon du langage JavaScript