

Les divisions de la page : l'élément <div>

L'élément DIV

L'élément **DIV** permet de spécifier l'aspect fonctionnel d'un paragraphe (une barre de navigation par exemple), et de regrouper plusieurs paragraphes ensemble. Nous l'utiliserons surtout en relation avec des feuilles de style.

Comme l'élément `<p>`, l'élément `<div>` peut être directement inclut dans le corps du document `<body>`. Il crée une division de la page. Ce type de division permet de grouper dans un seul bloc un ensemble composé soit de texte soit d'éléments inclus, auxquels on pourra appliquer globalement des styles particuliers. À la différence des paragraphes, une division créée avec `<div>` permet d'inclure une très grande variété d'éléments XHTML, comme du texte brut, les éléments en ligne et la totalité des éléments de niveaux bloc comme l'autorise `<body>`, mais également les titres, les listes et les formulaires, ce qui est interdit dans un paragraphe. Les tableaux 3-5 et 3-6 donnent respectivement la liste de tous les éléments enfants et parents de l'élément `<div>`. C'est donc un élément très riche qui se prête bien à la création de la structure d'une page en grands blocs distincts auxquels il est possible d'appliquer par la suite des styles propres et des positions précises (voir le chapitre 13, *Créer une mise en page : le dimensionnement et le positionnement*).

Tableau 3-5. Liste des éléments enfants de l'élément <div>

Texte, a, abbr, acronym, address, b, bdo, big, blockquote, br, button, cite, code, del, dfn, div, dl, em, fieldset, form, h1, h2, h3, h4, h5, h6, hr, i, img, input, ins, kbd, label, map, object, ol, p, pre, q, table, samp, script, select, small, span, strong, sub, sup, textarea, tt, ul, var

Tableau 3-6. Liste des éléments parents de l'élément <div>

blockquote, body, dd, del, div, fieldset, form, ins, li, map, noscript, object, td, th

L'élément `<div>` admet l'ensemble des attributs communs dont `id`, `class`, `title`, `dir`, `xml:lang`, qui sont les plus utilisés. Une des applications les plus courantes des divisions `<div>` consiste à créer des calques auxquels on applique des styles de positionnement permettant de remplacer avantageusement les cadres. Il nous sera, par exemple, possible de diviser la page web en zones bien définies ayant chacune un rôle précis. Comme nous le verrons dans la deuxième partie de cet ouvrage, l'application de styles à des divisions permettra de créer facilement un en-tête, une zone de menu, une zone de contenu et un pied de page distincts qu'il sera possible de répéter dans chaque page (voir le chapitre 13).

Il nous sera également possible à l'aide de scripts simples de créer des effets graphiques sur des divisions en les faisant apparaître ou disparaître en fonction des actions du visiteur.

Notons aussi que, contrairement aux paragraphes `<p>`, la fin d'une division n'entraîne pas un saut de ligne par défaut, mais seulement un retour à la ligne. Les contenus des différentes divisions peuvent donc se succéder sans rupture. L'exemple 3-3 reprend le même contenu que l'exemple 3-2 en remplaçant les éléments `<p>` par des divisions

`<div>` (repères `_`, `_` et `_`). On notera la différence d'affichage obtenue entre la figure 3-3 (paragraphes `<p>`) et la figure 3-4 (blocs `<div>`). Nous ne retrouvons plus entre deux blocs `<div>` le saut de ligne qui existe entre deux paragraphes. Dans cet exemple, la réalisation d'une séparation du texte entre les deux premiers blocs `<div>` ne se justifie que si l'on veut leur appliquer des styles différents par la suite.

Exemple 3-3. Les divisions en blocs `<div>`

```
<?xml version="1.0" encoding="iso-8859-1"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
<html xml:lang="fr" xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Les divisions div</title>
<meta http-equiv="Content-type" content="text/html; charset=iso-8859-1" />
<link rel="shortcut icon" type="images/x-icon" href="../images/favicon.ico" />
</head>
<body>
<h1>Chapitre 1</h1>
<h2>La création du monde</h2>
<div>In principio creavit Deus caelum et terram terra autem erat inanis
```

```

➤ et vacua et tenebrae super faciem abyssi et spiritus Dei ferebatur super
➤ aquas dixitque Deus fiat lux et facta est lux et vidit Deus lucem ...
</div>_
<div>Et vocavit Deus aridam terram congregationesque aquarum appellavit maria
➤ et vidit Deus quod esset bonum et ait germinet terra herbam virentem et
➤ facientem semen et lignum pomiferum faciens fructum iuxta ...
</div>_
<h2>Le huitième jour</h2>
<div>In principio creavit Deus caelum et terram terra autem erat inanis et vacua
➤ et tenebrae super faciem abyssi et spiritus Dei ferebatur super aquas ...
</div>_
</body>
</html>

```

*Les divisions en ligne *

L'équivalent en ligne de l'élément de bloc `<div>` peut être réalisé à l'aide de l'élément ``. C'est le plus employé des éléments créant des divisions sémantiques en ligne. Il n'a pas de rôle prédéfini et peut remplacer nombre des éléments précédents de cette section à condition de lui attribuer un style particulier adapté à chaque besoin de présentation.

Il possède l'ensemble des attributs communs et c'est son attribut `class` qui sera systématiquement utilisé pour lui attribuer un style.

Le code suivant :

```

<p>Le langage <span class="gras">XHTML</span> a pour complément indispensable
➤ Les styles <span class="gras">CSS 2</span></p>

```

Crée deux divisions en ligne qui utilise un style CSS particulier, défini par ailleurs dans l'élément `<style>` de l'en-tête (repère `_`). Leurs contenus sont donc affichés dans une police plus grande que celle du paragraphe parent et en gras.

C'est donc un élément passe-partout à usage très divers que nous retrouverons à chaque fois qu'il s'agira d'appliquer un style de façon ponctuelle à un contenu réduit en ligne.

Exemple

```
<?xml version="1.0" encoding="iso-8859-1"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
<html xml:lang="fr" xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Les éléments sémantiques en ligne</title>
<meta http-equiv="Content-type" content="text/html; charset=iso-8859-1" />
<link rel="shortcut icon" type="images/x-icon" href="../images/favicon.ico" />
<style type="text/css" >
.gras {font-size:larger; font-weight:bold;}_
</style>
</head>
<body>
<!-- L'élément abbr -->
<h1>Abréviations</h1>
<p>La masse se mesure en <abbr title="kilogramme" onmouseover="alert('Unité
➡ de masse: kilogramme')">kg</abbr>_
La force se mesure en <abbr title="Newton" onmouseover="alert('Unité de force:
➡ Newton')">N</abbr><br />_
</p>
<!-- L'élément acronym -->
<h1>Acronymes</h1>
<p>Utilisez vite <acronym title="eXtensible Hypertext Markup Language">
➡ XHTML 1.1</acronym> promis à un grand avenir.! _
</p>
<!-- L'élément bdo -->
<h1>Sens de lecture</h1>
<p>Lire de gauche à droite
<bdo dir="ltr" xml:lang="fr">XHTML</bdo> ou de droite à gauche
```

```

➤ <bdo dir="rtl" xml:lang="ar">XHTML</bdo>_
</p>
<!-- L'élément cite -->
<h1>Citations</h1>
<p>Comme le disait Boris Vian
<cite title="Vian 1920-1959"> La vérité n'est pas du coté du plus grand nombre
➤ car on ne veut pas qu'elle y soit. Le jour où il sera à même, par sa culture
➤ et ses connaissances, de choisir lui même sa vérité, il y a peu de chance
➤ pour qu'il se trompe.</cite>_
</p>
<!-- L'élément code -->
<h1>Code source en ligne</h1>
<p>Pour créer une boîte d'alerte en JavaScript, nous écrivons par exemple:
<code>alert('Bonjour')</code>. Dans ce cas l'exécution du script s'arrête.
</p>
<!-- L'élément dfn -->
<h1>Définition en ligne</h1>
<p>Ce bâtiment a la forme d'un pentagone
<dfn>(polygone à cinq cotés)</dfn> et abrite des gens peu recommandables._
</p>
<!-- L'élément kbd -->
<h1>Saisies clavier</h1>
<p>Si vous en avez assez, tapez : <kbd>Ctrl+Alt+Suppr</kbd>, et au revoir!</p>_
<!-- L'élément q -->
<h1>Citations courtes</h1>
<p>Comme Hamlet posons nous la question : <q cite="http://www.funhtml.com/
➤ hamlet.html" title="Hamlet : William Shakespeare" onclick="alert
➤ ('Voir '+this.cite)'">Être ou ne pas &ecirc;tre </q></p>
<!-- L'élément samp -->
<h1>Exemples en ligne</h1>
<p>Le type de l'équation du premier degré à deux inconnues est :
➤ <samp> ax+by = c</samp>. Elle n'a pas de solution unique</p>_
<!-- L'élément span -->

```

```
<h1>Le conteneur span</h1>
<p>Le langage <span class="gras">XHTML</span> a pour complément
indispensable
↳ les styles <span class="gras">CSS 2</span></p>_
<!-- L'élément var -->
<h1>Variables</h1>
<p> Dans l'équation <code> ax+by = c </code> les variables sont <var> x </var>
↳ et <var> y </var> </p>_
</body>
</html>
```

L'élément `<p>` est enfant de `<div>` et les éléments `` et `` sont eux-mêmes enfants de `<p>`. Par héritage et bien sûr faute d'avoir définis des styles propres pour les éléments `<p>`, `` et ``, ceux-ci ont un contenu qui possède les styles définis pour leur parent direct ou indirect `<div>`. Ils sont donc tous en blanc sur fond bleu. Si nous créons un style différent pour l'élément `<p>`, ses éléments enfants héritent alors de ces styles et non plus de ceux de l'élément `<div>`.