

# SUPPORT DE COURS B1-DEV

# HTML



# HTML 5

date	révision
Janvier 2018	Création
19/12/2018	Ajouts de légendes flottantes. Amélioration cosmétiques. Correction colspan=6 (=2)
04/04/2019	Ajout lecture fichier audio
19/09/2019	Correction titre + ajout balise width dans <table>



## TABLE DES MATIÈRES

1	Introduction.....	3
2	Fonctionnement.....	4
3	section HEAD.....	5
3.1	Balise title.....	5
3.2	Balises META.....	6
3.2.1	balise meta charset.....	6
3.2.2	balises meta name.....	7
3.2.3	balise meta name viewport.....	7
3.3	Exemple de page web.....	8
4	Section BODY.....	9
4.1	Les balises titres (H1, H2, H3, H4, H5, H6).....	9
4.2	Les balises paragraphes (P et BR).....	9
4.3	Les balises de formes (B, STRONG, I, U).....	10
4.3.1	Balise <B> et <STRONG>.....	10
4.3.2	Balise <I> et <EM>.....	10
4.3.3	Balise <U> et <MARK>.....	10
4.3.4	Exemple d'utilisation <b>, <strong>, <i> et <u>.....	10
4.4	Autres balises.....	10
4.4.1	Différences balises physiques et logiques.....	11
4.5	Les balises tableaux, liens, images.....	11
4.5.1	Les tableaux.....	11
4.5.2	Les liens.....	13
4.5.3	Les images.....	14
4.5.4	Les vidéos et les musiques.....	14
4.5.5	Les listes.....	14
4.5.6	Les balises conteneurs (div, span).....	15
4.5.7	La balise canevas (canvas).....	15
4.5.8	Les balises HTML5 diverses (progress).....	15
4.5.8.1	Progress Bar.....	15
4.5.9	Liste de toutes les balises HTML.....	15
5	En résumé.....	16
5.1	Page de base en HTML5.....	16
5.2	Les structures.....	16
5.2.1	Titres, paragraphes et listes.....	16
5.2.2	Formatage et inline.....	16
5.2.3	Tables.....	16



## 1 INTRODUCTION

Historiquement, les sites web n'étaient pas prévus pour l'échange d'information. Le modèle était un catalogue ou une encyclopédie : les informations, les produits étaient présentés mais l'achat ou les demandes d'informations passaient par d'autres canaux comme le courrier électronique ou les canaux **IRC**<sup>1</sup> (Internet Relay Chat).

La forte demande d'avoir des présentations toujours plus précises et justes a fait progresser le langage HTML (Hyper Text Markup Language). Ainsi, HTML est devenu un langage de balisage ET de mise en forme.

Malheureusement, ce mélange est à l'opposé d'une bonne structure et rapidement les développeurs de sites web ont cherché à retrouver un standard de développement, pour gagner du temps, de la lisibilité et ainsi pour améliorer la rentabilité.

Avec le web 2.0, l'interactivité est arrivée : il est devenu possible de consulter un site web et d'y ajouter des informations (livres d'or) ou de commander des produits en ligne. Cependant, la réécriture des pages pour adapter un site était toujours aussi compliqué.

Le CERN développera CSS 1.0 seulement en 1995...

1 IRC : [https://fr.wikipedia.org/wiki/Internet\\_Relay\\_Chat](https://fr.wikipedia.org/wiki/Internet_Relay_Chat)

## 2 FONCTIONNEMENT

Un fichier HTML contient des informations que le navigateur ne doit pas afficher mais interpréter. Ces informations sont lisibles dans un éditeur de texte. On utilise les symboles < et > pour encadrer les instructions destinées au navigateur.

< >

L'ensemble (symboles de délimitation < > et l'instruction) est appelé **balise**. Chaque balise ouverte doit être fermée pour créer un bloc. La balise fermante contient le symbole / devant l'instruction.

<HTML> </HTML>

La page web est décrite comme un document web dès la première ligne. Selon la version de HTML utilisé, il existe deux déclarations :

### XHTML (pseudo-HTML4)

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

### HTML5 (version recommandée)

```
<!DOCTYPE HTML>
```

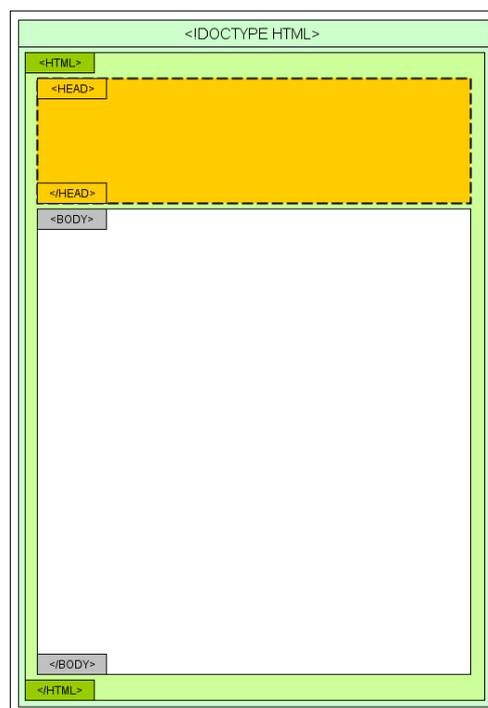
Ensuite, deux sections sont présentes :

- L'entête HTML : cette section contient principalement des informations sur la page.
- Le corps HTML : ici, c'est le contenu de la page à afficher à l'utilisateur.

Il en résulte le code (minimum) suivant :

```
<!DOCTYPE HTML>
<html>
  <head>
  </head>
  <body>
  </body>
</html>
```

Vous allez maintenant découvrir les deux sections (head et body) en détails.



### 3 SECTION HEAD

La section `<HEAD> </HEAD>` contient les informations pour le navigateur mais aussi pour les moteurs de recherches sur Internet. **Elle n'est pas visible par l'utilisateur.**

En effet, Google, Bing, Yahoo (et Lycos en son temps) utilisent les informations qui sont contenues dans cette section pour classer et mémoriser les pages.

**Pour un meilleur référencement sur Internet, il est capital de remplir correctement cette section.**

#### 3.1 BALISE TITLE

La première information est le titre de la page.

```
<title>Bonjour</title>
```

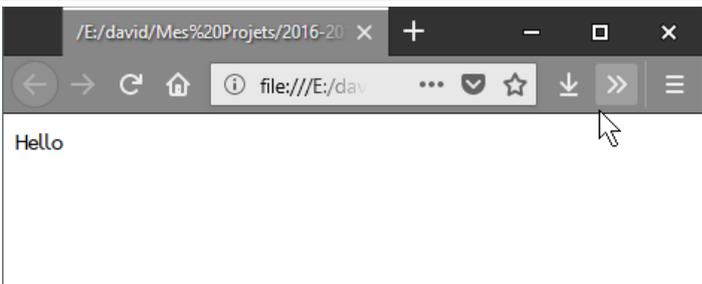
<p><b>Sans titre</b></p>	
<p><b>Avec titre</b> <code>&lt;title&gt;Bonjour&lt;/title&gt;</code></p>	

Tableau 3.1 : Titre de la page dans l'onglet du navigateur

En l'absence d'autres balises, les moteurs de recherches utilisent le contenu de cette balise.



## 3.2 BALISES META

Le préfixe "meta" vient du grec et signifie "au-delà". Ici, ce n'est pas une, mais plusieurs balises, dont le format est `<meta xxxxx="yyyyyy">`

**Notez que les balises META n'ont pas de balises de fermeture ! Même en HTML4, la fermeture est un / placé juste avant le symbole > : `<meta xxxx="yyyy" />`**

**Cette écriture est caduque en HTML5**

L'idée des balises méta est de pouvoir donner des informations "au-delà de la page", c'est-à-dire, impossible à déduire à partir du contenu de la page. Les balises méta n'ont donc pas d'affichage.

### 3.2.1 balise meta charset

Si une balise est à mettre impérativement, c'est cette balise, qui détermine comment le texte contenu dans la page sera interprété par le navigateur.

En effet, entre l'ASCII, l'ANSI (ISO) et maintenant UTF-8, les navigateurs peuvent ne pas afficher correctement les caractères spéciaux, et notamment, les accents.

Cependant, la norme actuelle la plus pratique est le codage en UTF-8. Voici la balise correspondante en HTML 5 :

```
<meta charset="utf-8">
```

<b>Charset UTF-8</b>	Bonjour les étudiants français !
<b>Charset iso-8859-1</b>	Bonjour les Ã©tudiants franÃ§ais !

Tableau 3.2 : différences d'affichage selon la valeur de la balise meta charset

Si vous êtes convaincu qu'il n'y a pas de différence dans les versions d'HTML, voici la même balise en HTML 4 :

```
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
```

**Il est recommandé de placer la balise meta charset dès le début de la section <HEAD> car cette dernière peut déjà contenir des caractères utilisant l'encodage spécifié.**



### 3.2.2 balises meta name

Encore un sous-ensemble contenant plusieurs propriétés :

- **meta name="author"** pour spécifier l'auteur de la page
- **meta name="description"** pour décrire le contenu de la page (un résumé que les moteurs de recherches affichent)
- **meta name="keywords"** contiendra les mots-clés de la page pour les moteurs de recherches. Cette balise **semble désormais obsolète** (les moteurs analysent maintenant le contenu des pages et en ressortent le thème et la fréquence statistique des mots).
- **Meta name="generator"** définit l'outil utilisé pour générer la page. C'est notamment le cas pour les CMS et ressemble un peu à une publicité.
- **Meta name="robots"** donne des instructions pour les robots des moteurs de recherches. On peut par exemple demander qu'une page ne soit pas indexée : `<meta name="robots" content="noindex">`. Cependant, il y a d'autres alternatives (comme le fichier robots.txt à la racine du site)

Exemple de code :

```
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<meta name="author" content="David ROUMANET">
<meta name="description" content="démonstration des balises HTML en module SI6">
<meta name="keywords" content="HTML, balise, XML, JavaScript">
```

Vous trouverez un excellent article sur les balises meta sur

<https://www.alsacreations.com/article/lire/628-balises-meta.html>

### 3.2.3 balise meta name viewport

Seule exception de ce cours, la balise meta name="viewport" : elle est devenue incontournable depuis que les surfaces d'affichage des tablettes et smartphones ont rendues la gestion des pages web plus complexe.

Entre les différents formats d'affichage des constructeurs (320x480 pour iPhone3, 640x1136 pour l'iPhone6 et 1440x2392 pour un LG G3 par exemple) et le fait de pouvoir pivoter l'affichage de 90°, tout en continuant à afficher sur des navigateurs d'ordinateurs (l'écran FullHD ne reflète pas la taille de la fenêtre du navigateur) relève effectivement du casse-tête.

Fin du suspense, voici donc la commande qui résout un grand nombre de problème :

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

Pour aller plus loin : <https://www.alsacreations.com/article/lire/1490-Comprendre-le-Viewport-dans-le-Web-mobile.html>



### 3.3 EXEMPLE DE PAGE WEB

Désormais, on utilisera le modèle HTML5 suivant pour commencer une page web en HTML :

```
<!doctype html>
<html lang="fr">
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta name="author" content="David ROUMANET">
  <meta name="description" content="démonstration des balises HTML en module SI6">
  <meta name="keywords" content="HTML, balise, XML, JavaScript">
  <title>Titre de la page</title>
  <link rel="stylesheet" href="style.css">
  <script src="script.js"></script>
  ...
</head>
<body>
  ...
  <!-- Le reste du contenu -->
  ...
</body>
</html>
```

Je vous recommande de consulter la page <https://www.alsacreations.com/article/lire/1374-html5-structure-globale-document.html> qui donne de très bonnes informations pour les pages web, et notamment des liens vers des sites qui fournissent des modèles (templates).



## 4 SECTION BODY

Commençons par les balises de mise en forme dans HTML. En effet, elles restent fréquemment utilisées et incontournables.

### 4.1 LES BALISES TITRES (H1, H2, H3, H4, H5, H6)

Comme sous Microsoft Office ou LibreOffice, ces balises constituent les titres de la page, de manière hiérarchique.

Testez le code suivant (nous allons le ré-utiliser) :

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>Exemple de titres</title>
  </head>
  <body>
    <h1>Titre principal</h1>
    <h2>Titre secondaire</h2>
    <h3>Titre sous-titre secondaire</h3>
    <h4>Titre sous-sous titre</h4>
    <h5>Titre de niveau 5</h5>
    <h6>Titre de niveau 6</h6>
  </body>
</html>
```

### 4.2 LES BALISES PARAGRAPHES (P ET BR)

La balise <p> et son terminateur </p> délimitent un paragraphe. Cela signifie notamment un saut de ligne automatique entre chaque paragraphe. La balise <br/> ordonne un simple retour à la ligne.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>Exemple de paragraphe</title>
  </head>
  <body>
    <h1>Les paragraphes et sauts de ligne</h1>
    <p>Ceci est un paragraphe</p>
    <p>Ceci est un autre paragraphe<br/>avec un retour à la ligne.</p>
    <p>Et enfin, voici un dernier paragraphe.</p>
  </body>
</html>
```

Attention, un style sur la balise <br/> ne sert à rien (c'est une balise de fin).

## 4.3 LES BALISES DE FORMES (B, STRONG, I, U)

### 4.3.1 Balise <B> et <STRONG>

D'apparence similaire, les deux balises donnent de l'importance à un texte. Cependant, sur une lecture vocale, STRONG augmentera la force de prononciation, là où B n'agit que pour mettre du texte en gras.

La balise <b> signifie gras (bold en anglais) et nécessite un terminateur </b>.

La balise <strong> signifie fort et nécessite un terminateur </strong>.

### 4.3.2 Balise <I> et <EM>

La balise <i> et sa terminaison </i> affiche un texte en italique.

### 4.3.3 Balise <U> et <MARK>

La balise <u> et sa terminaison </u> affiche un texte souligné (underline en anglais).

### 4.3.4 Exemple d'utilisation <b>, <strong>, <i> et <u>

Voici un exemple :

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>Exemple de paragraphe</title>
  </head>
  <body>
    <h1>Les paragraphes et sauts de ligne</h1>
    <p>Ceci est un paragraphe</p>
    <p>Ceci est un autre paragraphe<br/>avec un retour à la ligne.</p>
    <p>Et enfin, voici un dernier paragraphe.</p>
  </body>
</html>
```

## 4.4 AUTRES BALISES

Il existe d'autres balises pour la mise en forme en HTML :

- <sup> et </sup> pour mettre un texte en exposant. Exemple,  $E = MC^2$ .
- <sub> et </sub> pour mettre un texte en indice. Exemple,  $H_2O$ .
- <pre> et </pre> pour un paragraphe formaté. Exemple, tapez **dir** pour afficher le répertoire. Un effet similaire avec <tt> et </tt> s'applique à un texte dans un paragraphe.
- <strike> et </strike> pour barrer un texte. Exemple, ceci est ~~barré~~.

#### 4.4.1 Différences balises physiques et logiques

Il faut noter la différence entre les balises physiques et les balises logiques :

Balises physiques	Balises logiques
<ul style="list-style-type: none"> <li>• <code>&lt;b&gt;</code> et <code>&lt;bold&gt;</code></li> <li>• <code>&lt;i&gt;</code></li> <li>• <code>&lt;u&gt;</code></li> </ul>	<ul style="list-style-type: none"> <li>• <code>&lt;strong&gt;</code>, <code>&lt;em&gt;</code> et <code>&lt;mark&gt;</code></li> <li>• <code>&lt;cite&gt;</code></li> <li>• <code>&lt;address&gt;</code></li> </ul>

Tableau 4.1 : les différentes balises physiques et logiques

Pour les balises logiques, c'est le navigateur qui décide de mettre le style italique, gras ou souligné.

### 4.5 LES BALISES TABLEAUX, LIENS, IMAGES...

#### 4.5.1 Les tableaux

Les tableaux sont très utilisés pour afficher des informations avec un certain formalisme. Il faut cependant savoir qu'avant l'arrivée de CSS, ils étaient employés pour formater les pages web.

Menu vertical	Titre	contact
Pied de page	publicité	

Tableau 4.2 : exemple de formatage web 1.0

Les balises d'un tableau sont `<table>` et `</table>` : à l'intérieur il faut définir les colonnes et les lignes.

`<caption>` et `</caption>` permettent de donner une ligne de titre au tableau

`<tr>` et `</tr>` définissent une ligne (TR signifiant Table Row).

`<td>` et `</td>` définissent les divisions en colonnes (TD signifiant Table Division)

**A savoir : les lignes contiennent plusieurs colonnes (au minimum, une) et non l'inverse.**

**Cela implique que les balises `<tr>` contiennent les balises `<td>` : `<tr><td>col1</td><td>col1</td></tr>`**

Il est possible d'utiliser `<th>` et `</th>` pour les entêtes de colonnes, permettant un formatage d'entête.

Il y a aussi des *paramètres* à placer dans la balise `<table>` pour obtenir des espaces et des bordures différentes (les propriétés sont cumulables dans la même ligne) :

- `<table border=1>` affiche une bordure de un pixel
- `<table cellspacing=0>` supprime les espaces entre les cellules
- `<table width=400>` affiche un tableau ayant 400 pixels de large

Pour bien maîtriser l'agencement des balises : <http://cyberzoide.developpez.com/html/table.php3>

Voici un exemple de code et son résultat :

```
<table border=1 cellspacing=0 cellpadding=5>
  <caption>Répertoire des contacts</caption>
  <tr bgcolor="#FF7700">
    <th>Nom</th>
    <th>téléphone</th>
    <th>Ville</th>
  </tr>
  <tr>
    <td>Batman</td>
    <td bgcolor="#FF7777">555 3333</td>
    <td>Gottam city</td>
  </tr>
  <tr>
    <td>Superman</td>
    <td>555 0000</td>
    <td>Metropolis</td>
  </tr>
</table>
```

Répertoire des contacts

Nom	téléphone	Ville
Batman	555 3333	Gottam city
Superman	555 0000	Metropolis

Il existe d'autres balises comme **<thead>**, **<tbody>** et **<tfoot>** mais il est préférable d'aller consulter la page d'OpenClassroom sur le sujet :

<https://openclassrooms.com/courses/apprenez-a-creeer-votre-site-web-avec-html5-et-css3/les-tableaux-1>

De même, il est possible de créer des cellules de tailles différentes avec les *attributs* **colspan** et **rowspan**.

- **<td colspan=2>** occupe 2 colonnes sur la même ligne.  
Dans l'exemple ci-contre, la cellule grise.
- **<th rowspan=2>** occupe 2 lignes sur la même colonne.  
Dans l'exemple ci-contre, les cellules vertes et roses.

<td rowspan=2>Adore...</td>

<td colspan=2></td>

Choses préférées et détestées

		Bob	Alice
Adore...	Couleur	Bleu	Rose
	Parfum	Boisé	Vanille
Déteste...	>Couleur	Jaune	Marron
	>Parfum	Menthe	Noix de coco

Vous trouverez plus d'exemples sur <https://html.com/tables/rowspan-colspan/>



## 4.5.2 Les liens

Lorsqu'il est nécessaire de cliquer pour aller sur une autre page, on parle de lien.

```
<A HREF="http://www.ac-grenoble.fr">lien vers rectorat</A>
```

Il est possible de créer des liens, qui pointent vers section à l'intérieur d'une même (et longue) page.

```
<h2 id="C4">Chapitre 4 - Les liens</h2>
<p>blabla bla</p>
...
<h2 id="C5">Chapitre 5 - les retours</h2>
voici le lien vers <a href="#C4">Chapitre 4</a>
```

Si vous devez pointer une section depuis une autre page, il suffit de préciser la page destination :

```
voici le lien vers <a href="pageCours.html#C4">Chapitre 4</a>
```

Le lien peut également pointer une adresse de courrier électronique.

```
voici le lien vers <a href="mailto:toto@wanaskype.de">Toto HALDERMAN</a>
```

Deux dernières astuces à propos des liens : la première est qu'il est possible d'utiliser une image à la place d'un texte à cliquer, la seconde est qu'il est possible d'ouvrir le lien dans une nouvelle page cible.

```
<a href="index.html">
  </img>
</a>
```

L'exemple ci-dessus ouvre la page index.html lorsque l'utilisateur clique sur l'image smiley.gif.

```
<a href="https://www.w3schools.com/" target="_blank">Visit W3Schools!</a>
```

Celui-ci ouvrira le site W3School dans un nouvel onglet ou une nouvelle fenêtre, grâce à l'attribut target.

Enfin, une méthode originale pour créer plusieurs liens en fonction de zones cliquables sur une image : c'est la fonction image-map !

Ex : code avec un rectangle qui pointe vers image-map.net et un polygone qui pointe... devinez :

```

<map name="image-map">
  <area href="https://www.image-map.net/" coords="835,888,769,681" shape="rect">
  <area href="https://www.qwant.com/"
coords="720,673,737,576,710,536,739,483,767,541,795,525,800,541" shape="poly">
</map>
```

Le plus simple reste d'utiliser un éditeur en ligne : <https://www.image-map.net/>



### 4.5.3 Les images

Les images utilisent une balise très logique `<img>` et `/img>`. Il est recommandé d'ajouter l'attribut **alt** pour afficher un texte alternatif (si l'image ne s'affiche pas ou bien si l'utilisateur est malvoyant et utilise un système de description audio).

```
</img>

```

Un procédé d'encodage<sup>2</sup> (base64) permet également de stocker l'image dans le fichier HTML (plutôt que dans le même répertoire). Ce procédé augmente le poids (octets) de l'image, à réserver à de petites images, uniques.

```

```

### 4.5.4 Les vidéos et les musiques

Pour les vidéos ou les musiques, il s'agit de balises HTML5 : dans les versions HTML précédentes, l'affichage de vidéo se faisait avec du code JavaScript. Désormais, il suffit d'écrire :

```
<video src="victoire_BTS-SIO.mp4" width="375" height="280" controls></video>
<audio src="sf_ocean.mp3" type="audio/mpeg" controls autoplay loop></audio>
```

Les attributs **width** et **height** sont optionnels, comme l'attribut **controls** qui affiche une barre de contrôle de lecture de la vidéo. La balise **autoplay** sert à démarrer automatiquement un média.

### 4.5.5 Les listes

Il est possible d'utiliser des puces d'énumération ou bien une numérotation automatique). La balise `<ul>` signifie "unordered" et `<ol>` signifie "ordered". Voici les deux formes :

Liste non-numérotées (à puces)	Liste numérotées
<pre>&lt;ul&gt;   &lt;li&gt;Café&lt;/li&gt;   &lt;li&gt;Thé&lt;/li&gt;   &lt;li&gt;Chocolat&lt;/li&gt; &lt;/ul&gt;</pre>	<pre>&lt;ol&gt;   &lt;li&gt;Café&lt;/li&gt;   &lt;li&gt;Thé&lt;/li&gt;   &lt;li&gt;Chocolat&lt;/li&gt; &lt;/ol&gt;</pre>

Tableau 4.3 : les 2 formes de liste

Il est utile de préparer le travail qui sera fait en CSS : les listes sont souvent utilisées comme des menus cliquables (ce qui n'est pas le cas en HTML pur).

Il est possible (en HTML) d'avoir différents types pour les deux méthodes : des puces carrées, des numérotations romaines, etc.

Voici un lien utile : <http://cyberzoide.developpez.com/html/list.php3>

2 Encoder une image en base64 : <https://www.motobit.com/util/base64-decoder-encoder.asp>



### 4.5.6 Les balises conteneurs (div, span)

Il existe des balises de type conteneur : par défaut, elles n'affectent pas la présentation des données, mais elles permettent certaines interactions avec CSS. Ces balises sont `<div>...</div>` et `<span>...</span>`.

Tandis que `<div>` est un conteneur qui occupe la largeur de la page (c'est un bloc dont la largeur est 100 % de la page par défaut), `<span>` s'insère dans un flux texte : on peut donc placer plusieurs conteneurs `<span>` les uns à la suite des autres.

Pour `<div>` c'est une balise typée "block" tandis que `<span>` est typée "inline".

### 4.5.7 La balise canevas (canvas)

Cette balise n'existe qu'avec HTML5 et permet de créer une surface de dessin (surface de rendu).

Un canevas n'a que deux *paramètres*, **width** et **height** (outre les attributs id, class, name).

Ensuite, il est possible de dessiner sur ce canevas avec JavaScript : c'est pour cela que le fonctionnement des canevas sera étudié dans le support de cours JavaScript.

Voici la déclaration de la zone canevas avec les balises `<canvas></canvas>` :

```
<canvas id="maZone" width="500" height="400"></canvas>
```

### 4.5.8 Les balises HTML5 diverses (progress)

Il existe d'autres balises amusantes et intéressantes avec un langage de programmation comme JavaScript.

#### 4.5.8.1 Progress Bar

Permet d'insérer une barre de progression.

```
<progress width="200" height="15" value="40%" max="200">40%</progress>
```

### 4.5.9 Liste de toutes les balises HTML

L'essentiel est indiqué dans ce cours, cependant, il existe de nombreuses autres balises HTML.

Voici quelques sites qui les énumèrent.

<http://www.codeshttp.com/baliseh.htm>

<https://www.w3schools.com/html/>

## 5 EN RÉSUMÉ

Cette section résume ce qu'il faut absolument se rappeler par cœur :

### 5.1 PAGE DE BASE EN HTML5

La structure de la page est définie comme dans le schéma ci-contre :

```
<!doctype html>
```

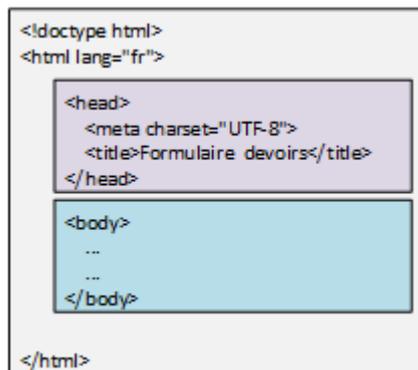
```
<html lang="fr"> ... </html>
```

A l'intérieur des balises HTML, on distingue :

l'entête `<head> </head>` qui contient les informations sur le contenu de la page.

Le corps `<body> </body>` qui est le contenu de la page à afficher.

La notion d'indentation est importante pour retrouver facilement chaque section.



### 5.2 LES STRUCTURES

#### 5.2.1 Titres, paragraphes et listes

<pre>&lt;h1&gt; à &lt;h6&gt; ↵ &lt;p&gt; &lt;/p&gt; ↵ &lt;div&gt;&lt;/div&gt; ↵ &lt;br&gt; ↵</pre>	<pre>&lt;ol&gt; pour les listes ordonnées   &lt;li&gt;blabla &lt;/li&gt;   &lt;li&gt;blabla &lt;/li&gt; &lt;/ol&gt;</pre>	<pre>&lt;ul&gt; pour les listes à puce   &lt;li&gt;blabla &lt;/li&gt;   &lt;li&gt;blabla &lt;/li&gt; &lt;/ul&gt;</pre>
--	---	--

#### 5.2.2 Formatage et inline

Gras : `<b></b>`

Italique : `<i></i>`

image: `<img></img>`

lien : `<a href=""></a>`

#### 5.2.3 Tables

La création d'une table commence et termine par `<table> </table>`

Chaque ligne est créée par `<tr>` et `</tr>`.

Chaque colonne est créée par `<td>` et `</td>`.

La première ligne du tableau peut contenir des colonnes d'entête, qui utilisent les balises `<th>` et `</th>` pour permettre un formatage différent.

