



# HTML 4 (Partie II)



---

# Balises

## BALISES DU DOCUMENT

---

Balises	Description
<html>	Balise d'ouverture de toute page web
<head>	En-tête de la page
<body>	Corps de la page

## BALISES D'EN-TETE

Balises	Description
<link />	Permet d'insérer une feuille de style ou d'afficher une icône de favicon
<meta />	Fixe les propriétés du document (auteur, description de la page)
<script>	Permet d'insérer un script (le plus souvent en JavaScript)
<noscript>	Cette balise permet d'afficher un texte alternatif aux scripts de la page web (si le javascript a été désactivé par exemple)
<style>	Permet de définir une zone de mise en forme via du code CSS
<title>	Titre du document (ne s'affichant pas dans la page web mais sur la zone de titre du navigateur). Le texte contenu dans cette balise a une influence sur le référencement

## BALISES DE LISTING

Balises	Description
<ul>	Liste à puces non numérotée
<ol>	Liste à puces numérotée
<li>	Permet de créer un élément de la liste
<dl>	Liste de définitions
<dt>	Terme à définir
<dd>	Définition du terme

## BALISES DE TABLEAU

Balises	Description
<table>	Délimite un tableau
<caption>	Permet de donner un titre au tableau
<tr>	Ligne de tableau
<th>	Cellule d'en-tête du tableau (généralement mise en gras)
<td>	Cellule du tableau
<thead>	permet d'insérer l'en-tête du tableau
<tbody>	permet d'insérer le corps du tableau
<tfoot>	permet d'insérer le pied du tableau

## BALISES DE FORMULAIRE

Balises	Description
<form>	La balise form délimite un formulaire. Elle s'accompagne de deux attributs : method : comment vont circuler les données action : Où va-t-on arriver après soumission du formulaire
<fieldset>	Permet de regrouper plusieurs éléments d'un formulaire
<legend>	Titre d'un groupe dans un formulaire A utiliser à l'intérieur d'un <fieldset>
<label>	Titre d'un élément de formulaire. Cette balise s'accompagne de l'attribut for pour indiquer l'ID de l'élément auquel correspond le label.
<input />	Champ de formulaire. Il existe de nombreux types de champs différents.
<textarea>	Zone de saisie multi ligne. La taille peut être définie grâce aux attributs rows et cols.
<select>	Liste déroulante.
<option>	Élément d'une liste déroulante.
<optgroup>	Permet de regrouper plusieurs options de même type.

## BALISES OBJET

Balises	Description
<code>&lt;object&gt;</code>	Un objet peut être une animation flash, un applet java ou autre
<code>&lt;param /&gt;</code>	La balise <code>&lt;param /&gt;</code> définit les paramètres d'objet dans une page web html.



## BALISES DE TEXTE

Balises	Description
<h1>	Titre de niveau 1
<h2>	Titre de niveau 2
<h3>	Titre de niveau 3
<h4>	Titre de niveau 4
<h5>	Titre de niveau 5
<h6>	Titre de niveau 6
<acronym>	Cette balise sert à définir des acronymes (exemple : TGV = Train à grande vitesse). Elle est accompagnée de l'attribut title.
<abbr>	Cette balise sert à déclarer une abréviation dans une page web. L'attribut title permet de donner la signification de l'abréviation et l'attribut lang la langue de l'abréviation.
<blockquote>	Citation taille longue
<cite>	Citation taille moyenne
<q>	Citation taille courte
<sup>	Mise en exposant (exemple : m <sup>2</sup> )
<sub>	Mise en indice

## BALISES DE TEXTE

Balises	Description
<strong>	Strong permet de faire ressortir une partie de texte (texte en gras)
<em>	Em permet de faire ressortir une partie du texte (texte en italique)
 	Cette balise permet d'effectuer un retour à la ligne
<p>	La balise p pour paragraphe nous permettra d'insérer du texte
<hr />	La balise hr marque d'un trait la séparation entre deux « parties »
<address>	Cette balise permet de montrer une adresse (le texte est mis en italique)
<del>	Cette balise barre une partie d'un texte
<ins>	Cela permet d'indiquer qu'une partie du texte a été insérée (texte souligné)
<dfn>	Cela permet d'indiquer que le texte est une définition
<kbd>	La balise "keyboard" ( clavier ) permet d'indiquer un code que doit taper le visiteur
<pre>	Insère un saut de ligne à chaque fois que l'on passe à la ligne dans le document HTML

## BALISES DE CONTENU

Balises	Description
<a>	Lien hypertexte. L'indication de l'url de destination s'effectue grâce à l'attribut href
<img>	Insère une image. Les attributs src (pour indiquer l'adresse de l'image) et alt (pour indiquer un texte de remplacement) viennent compléter cette balise.
<map>	La balise map regroupe les différentes zones réactives d'une même image.
<area>	La balise area possède obligatoirement un attribut alt. L'attribut shape permet de définir quel type de zone nous souhaitons dessiner : "rect" (rectangle), "circle" (cercle) ou "poly" (polygone). L'attribut coords permet de définir les coordonnées associées à la figure géométrique choisie
<audio>	Audio
<video>	Vidéo
<source>	Format source pour les balises <audio> et <video>

## BALISES GENERIQUE DE CONTENEUR

---

Balises	Description
<span>	Balise générique permettant de former un conteneur de type inline
<div>	Cette balise est en fait souvent utilisée pour regrouper d'autres balises de manière à former un conteneur. Il est ensuite intéressant d'utiliser le style CSS pour le positionner n'importe où.
<header>	En-tête
<nav>	Balise menu permettant de regrouper les différents liens de navigation
<footer>	Pied de page
<section>	Section de page
<article>	Article (contenu)
<aside>	Partie destinée aux informations complémentaires



## BALISE AUTO-FERMANTE

---

Les balises de type AUTOFERMANTE HTML sont des balises qui sont ouvrantes et fermantes en même temps. Cela signifie qu'elles ne contiennent pas de contenu. Nous fermons donc la balise en ajoutant un espace et un slash (/) à la fin de la balise.

Quelques exemples de balises auto-fermantes :

`<area />` , `<br />` , `<hr />` , `<img />` , `<input />` , `<link />` , `<meta />` , `<param />`



---

# Conception

# TYPE BLOC ET TYPE INLINE

---

## Type Bloc

C'est un élément (ou une zone) qui prend par défaut toute la largeur de la page.

Les balises de type BLOC HTML sont des boîtes d'éléments qui peuvent contenir d'autres éléments tels que du texte, des images, des liens ou autres... Elles peuvent contenir d'autres balises de type bloc et aussi contenir d'autres balises de type inline.

Par défaut, les blocs se placent toujours l'un en dessous de l'autre.

Quelques balises de type BLOC « block » : <p>, <ul>, <li>, <div>, <form>, <input>, <blockquote>, <table>, <h1>...<h6>,...

## Type Inline

C'est un élément (ou une zone) qui prend uniquement la place dont il a besoin

Une balise inline ne peut contenir uniquement une (ou plusieurs) balise inline, et n'a pas de dimensions à proprement parler (aucune largeur et aucune hauteur. Elle n'occupe que la place minimum nécessaire à son contenu).

Par défaut, les inline se placent toujours l'un à côté de l'autre.

Quelques exemples de balises de type Inline « en ligne » : <a>, <strong>, <em>, <span>, <img>, Description block, inline, etc.

# STRUCTURE

---

## Déclaration DOCTYPE

Dans un document HTML vous devrez spécifier le DOCTYPE.

```
<!DOCTYPE html>
<html>
  <head>
    <title>Titre du site</title>
    <link rel="stylesheet" href="structure.css" />
  </head>
  <body>
  </body>
</html>
```



### Dans la partie HEAD (en-tête) :

Nous pouvons voir des informations sur le document, telles que le titre général de la page, le lien vers la feuille de style CSS.

Nous pourrions également y inscrire certaines balises meta donnant des informations pour le référencement du site ainsi qu'appeler des fichiers JavaScript.

### Dans la partie BODY (corps) :

Nous pouvons observer que le contenu de notre document se trouve plus particulièrement à l'intérieur des balises `<body></body>`.

Le corps (balise body) contient ce qui sera affiché sur la page web.

Il est possible d'ajouter des commentaires, ce sont des annotations dans le code-source qui ne seront pas interprétées par le navigateur

# META

---

Les balises <meta /> contiennent les métas informations de la page web HTML.

Ainsi, la balise <meta /> vous donne des infos sur la description de la page web, ses mots-clés, son auteur, les règles spécifiques destinées aux robots des moteurs de recherche, la langue de la page web... L'attribut content est obligatoire.

```
<meta name="keywords" lang="fr" content="motcle1,mocle2" />
<meta name="description" content="Description de ma page web." />
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<meta http-equiv="Content-Language" content="fr" />
<meta http-equiv="Content-Script-Type" content="text/javascript" />
```

Les balises Meta description et Meta keywords sont importantes pour donner une description de son site web et indiquer des mots-clés. Il en existe d'autres.

## TEXTE, TITRE ET COULEURS

---

Nous nous situons à l'intérieur des balises `<body>` `</body>` et nous souhaitons afficher un titre et un texte.

### Exemple

# Titre de niveau 1

Paragraphe...

## Titre de niveau 2

Second paragraphe...

Certaines balises héritent de propriétés, c'est le cas de `h1` ou de `p` qui héritent de marges, cela est bien sûr modifiable via les feuilles de style CSS.

### Code

```
<h1>Titre de niveau 1</h1>  
<p>Paragraphe...</p>
```

```
<h2>Titre de niveau 2</h2>  
<p>Second paragraphe...</p>
```

## FOND DE PAGE

---

Le fond d'une page se fixe sur la balise ouvrante : <body>.

Nous pouvons à la fois définir une image de fond ou une couleur de fond.

### Exemple



## COMMENTAIRES

---

En programmation (ou en langage de conception comme ici), les commentaires sont des portions du code source ignorées par le compilateur ou l'interpréteur, car ils ne sont pas nécessaires à l'exécution du programme.

Les commentaires sont le plus souvent utilisés pour expliquer le code ; certains outils permettent d'extraire l'information disponible dans les commentaires pour générer automatiquement la documentation du programme.

Les commentaires en HTML se déclarent avec `<!-- texte -->`

### Code

```
<!-- EXEMPLE -->
```

# LISTES

---

Les listes sont des outils bien utiles pour présenter des informations. HTML en offre 3 types différents. Le premier type de liste est la liste non ordonnée, autrement appelée liste à puces.

## Exemple

- element 1
- element 2
- element 3

## Code

```
<ul>  
<li>1er élément</li>  
<li>2ème élément</li>  
<li>3ème élément</li>  
</ul>
```

## LISTES

---

Le second type de liste est la liste ordonnée. Ce type de liste sera présentée avec des petites puces avant le texte de chaque élément.

### Exemple

1. element 1
2. element 2
3. element 3

### Code

```
<html>
<ol>
<li>element 1</li>
<li>element 2</li>
<li>element 3</li>
</ol>
</html>
```

## LISTES

---

Le dernier type de liste est la liste de définitions, permettant de créer des listes possédant un terme et sa définition.

### Exemple

- element 1
- element 2
- element 3

### Code

```
<html>
<dl>
<li>element 1</li>
<li>element 2</li>
<li>element 3</li>
</dl>
</html>
```



# TABLES HTML

---

Souvent, les intégrateurs web vous mettront en garde contre l'utilisation des tables HTML (appelées aussi tableau HTML).

Pourquoi ? A l'origine les « tables » ou « tableaux » ont été créés pour présenter des données tabulaires. Ils ont été détournés par les designers afin de structurer des pages web complexes.

L'inconvénient était qu'il n'était pas simple de mettre en place un design complexe à l'aide de ce système.

Aujourd'hui, et depuis déjà quelques années, la conception d'un design en structure de page web est beaucoup plus simple et se fait par des balises prévues à cet effet telles que les balises *div*.

Les avantages d'utiliser des blocs *div* à la place des « tables » ou « tableaux » sont multiples :

- Avoir des sources sémantiquement correctes
- Le code est beaucoup plus maintenable et moins complexe
- Les pages sont plus accessibles

Néanmoins, si nous revenons à la fonction d'origine des « tables » ou « tableaux » (de représenter des données tabulaires), il est tout à fait normal et autorisé par les normes W3C/HTML de les utiliser.

Tant que nous ne cherchons pas à construire son design de site via des tables et que nous ne les emboîtons pas les unes dans les autres, le code sera valide w3c.

Un tableau HTML est découpé en lignes contenant des cellules.

### Exemple

Ma premiere table

ligne 1, colonne 1	ligne 1, colonne 2	ligne 1, colonne 3
ligne 2, colonne 1	ligne 2, colonne 2	ligne 2, colonne 3
ligne 3, colonne 1	ligne 3, colonne 2	ligne 3, colonne 3

### Code

```
<html>

  <table border="1" cellpadding="5" cellspacing="10">

    <caption>Ma premiere table</caption>

    <tr>
      <th>ligne 1, colonne 1</th>
      <th>ligne 1, colonne 2</th>
      <th>ligne 1, colonne 3</th>
    </tr>

    <tr>
      <td>ligne 2, colonne 1</td>
      <td>ligne 2, colonne 2</td>
      <td>ligne 2, colonne 3</td>
    </tr>

    <tr>
      <td>ligne 3, colonne 1</td>
      <td>ligne 3, colonne 2</td>
      <td>ligne 3, colonne 3</td>
    </tr>

  </table>

</html>
```

Les cellules peuvent en recouvrir d'autres, que ce soit en largeur ou en hauteur. Pour qu'une cellule occupe x cellules vers la fin de la ligne, nous utilisons l'attribut `colspan="x"`.

Pour qu'une cellule occupe x cellules vers la fin de la colonne, nous utilisons l'attribut `rowspan="x"`.

### Exemple

ligne 1 et 2, colonne 1	ligne 1, colonne 2	ligne 1, colonne 3
	ligne 2, colonne 2	ligne 2, colonne 3
ligne 3, colonne 1	ligne 3, colonne 2 et 3	

### Code

```
<html>

  <table border="1" cellpadding="5" cellspacing="10">

    <tr>
      <td rowspan="2">ligne 1 et 2, colonne 1</td>
      <td>ligne 1, colonne 2</td>
      <td>ligne 1, colonne 3</td>
    </tr>

    <tr>
      <td>ligne 2, colonne 2</td>
      <td>ligne 2, colonne 3</td>
    </tr>

    <tr>
      <td>ligne 3, colonne 1</td>
      <td colspan="2">ligne 3, colonne 2 et 3</td>
    </tr>

  </table>

</html>
```

## LIENS

---

L'aspect le plus intéressant du Web est cette formidable capacité à créer des liens de page en page, de site en site. Pour créer un lien, nous utilisons la balise a.

### Lien externe

Ce type de lien pointe vers un autre site.

### Code

```
<a href="http://www.tf1.fr/news">TF1 actualités</a>
```

# LIENS

---

## Lien local

Ce type de lien permet de lier deux pages de votre site entre elles.

Supposons que nous nous trouvons dans une page nommée « page.html »

## Code

```
<a href="page2.html "> Accédez à ma page 2</a>
```

## LIENS

---

### Lien ouvrant une nouvelle fenêtre

L'attribut target permettra d'ouvrir la page dans une nouvelle fenêtre et ainsi ne pas quitter la page affichée à l'écran.

### Code

```
<a href="http://www.google.com/" target="_blank">Google</a>
```

## LIENS

---

### Lien vers une adresse email

Ce type de lien permet d'ouvrir le client de messagerie de l'utilisateur.

A noter qu'ils ne sont plus très utilisés car ils ne sont plus dans l'air du temps et sont susceptibles de récolter des spams.

### Code

```
<a href="mailto:nom@site.com"> contactez-moi</a>
```

## LIENS

---

### Lien vers un fichier

Ce type de lien permet de pointer vers un fichier à ouvrir ou à télécharger sur son ordinateur.

### Code

```
<a href="un_fichier.xlsx"> consulter le fichier excel </a>
```



# LIENS

---

## Lien « ancre »

Ce type de lien permet d'accéder rapidement à l'information qui nous intéresse en ciblant l'ancre (id) auquel un texte dépend.

## Code

```
<a href="#ici">lire le texte</a> <p id="ici"> texte </p>
```

## LIENS

---

### Lien sur image

Vous pouvez aussi faire d'une image un lien hypertexte.

Par exemple le logo de votre organisation servant de lien pour revenir à la page d'accueil.

### Code

```

```

# FORMULAIRES

---

Les formulaires interactifs permettent aux auteurs de pages Web de doter leur page web d'éléments interactifs permettant par exemple un dialogue avec les internautes.

Le lecteur saisit des informations en remplissant des champs ou en cliquant sur des boutons, puis appuie sur un bouton de soumission (submit) pour l'envoyer soit à un URL, c'est-à-dire de façon générale à une adresse e-mail ou à un script de page web dynamique tel que PHP.

Un formulaire peut servir à recueillir les avis des visiteurs, leur e-mail en vue d'une newsletter. Cela peut également servir pour une inscription, un recrutement, un bon de commande, un sondage, un jeu... La balise <form> qui permet de construire un formulaire possède au moins deux attributs « method » et « action ».

## METHOD

METHOD indique sous quelle forme seront envoyées les réponses

« POST » est la valeur qui correspond à l'envoi de données côté « serveur », tandis que « GET » correspond à un envoi des données codées dans l'URL.

## ACTION

Indique l'adresse d'envoi. Où arrivons-nous une fois le formulaire soumis ?

### Informations

Nom \*

PrÃ©nom \*

Adresse

### Date de Naissance

Jour

Mois

Annee

### Sexe

homme:

☐

femme:

☐

## MULTIMEDIA

---

Certains éléments peuvent être intégrés aux pages web tels qu'une carte interactive (code googlemap fournit par google).

Les vidéos et les sons audio possèdent leur propre balise.

```
<video preload controls poster="cic.gif">
  <source src="cic.ogv" type="video/ogg" />
</video>
```

```
<audio src="jazz.mp3" controls></audio>
```

# SCRIPT

---

Un script peut être inséré dans le HTML (généralement il s'agit de code JavaScript)

Là aussi, plusieurs méthodes s'offrent à nous :

## Exemple 1

```
<script>  
...  
</script>
```

1. Script inséré directement dans le contenu de la page (à répéter sur toutes les pages HTML).

## Exemple 2

```
<script src = "url/fichier.js"> </SCRIPT>
```

2. Script inséré via un fichier externe (solution recommandée).