

# Chapitre : Introduction à HTML5

## I. Qu'est-ce que le HTML5 ?

HTML5 n'est pas un système distinct, mais plutôt la dernière version du langage de balisage HyperText **HTML**.

Historiquement, on est passé du HTML version 1 en 1992 au HTML version 2 en 1994 au HTML version 3 en 1997 et puis HTML version 4 en 1999 jusqu'au HTML5. En fait, le HTML5 n'est pas très récent lui-même, puisqu'il est sorti il y a plusieurs années, en 2014. HTML est un langage facile à apprendre et est généralement appris en parallèle avec CSS.

Concernant le CSS, nous sommes passés de la version 1 en 1996 au CSS2 en 1998 puis au CSS2.1 en 2011 et finalement au CSS3 en Novembre 2014. Chaque nouvelle version inclut les niveaux précédents, et ainsi de suite.

Le HTML et le CSS sont deux langages qui se complètent, le premier permet de créer et de structurer du contenu, le second permet de faire de la mise en forme visuelle et dynamique.

HTML fonctionne en utilisant diverses balises. Exceptionnellement, plusieurs balises sémantiques de structure de page et de structure de contenu ont été ajoutées à HTML5, permettant ainsi un découpage plus précis et plus fin de notre page web, ce qui permet de fournir une structure plus claire et plus significative à son contenu, et améliorer par conséquent l'accessibilité, l'indexation par les moteurs de recherche et la compréhension globale du document. En effet, le fait de baliser notre document HTML expressivement va permettre aux robots d'indexation de mieux interpréter notre page web, et donc de cibler le contenu réel que nous souhaitons mettre en évidence.

Ainsi, la sémantique a permis de donner un rôle précis à certaines nouvelles balises.

## II. Qu'est-ce qu'une balise sémantique ?

Une balise sémantique est une balise qui donne une indication et un sens au contenu qu'elle entoure, elle permet d'organiser et de structurer au mieux notre code et permet donc par la suite de concevoir de pages web optimisées, le tout en respectant les dernières normes HTML 5 en vigueur.

### III. Structure d'un document HTML :

#### 1. Structure simple : Un document HTML5 minimal :

Les bases d'un code HTML5 ressemblent beaucoup à celles rédigées à l'aide des précédentes versions. On y retrouve donc un **doctype** suivi des éléments les plus courants :

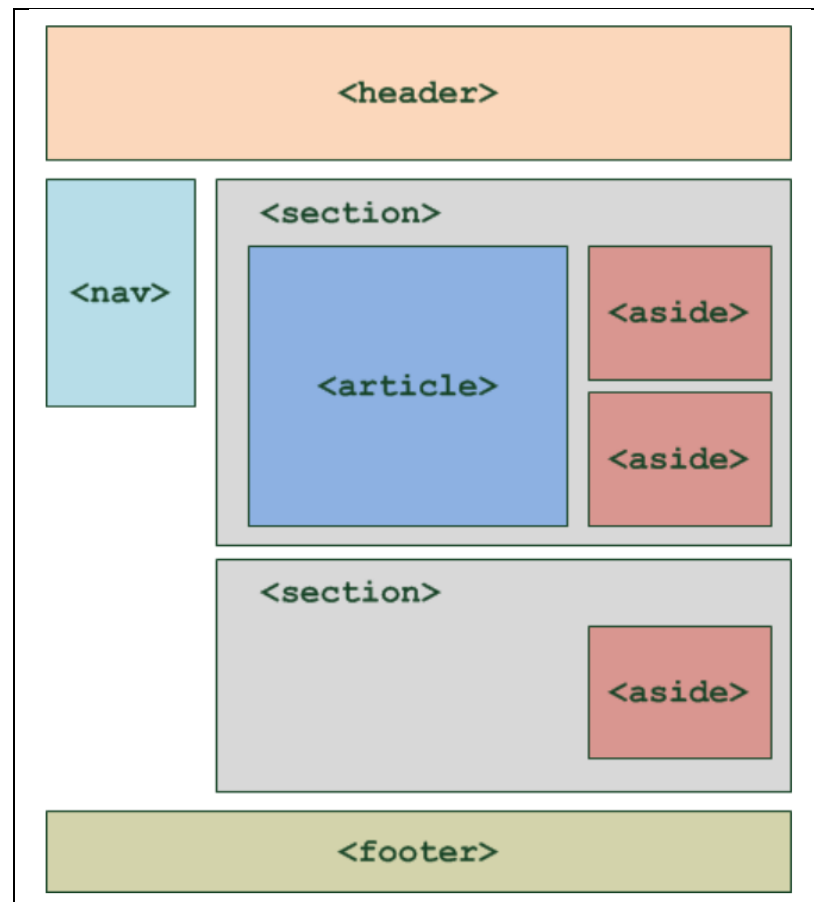
```
1 <!doctype html>
2 <html>
3 <head>
4   <meta charset="utf-8">
5 </head>
6 <body>
7   <p>structure minimale d'une page web en HTML5 </p>
8   <!--corps de la page web-->
9 </body>
10 </html>
```

## 2. Structure détaillée :

La nouvelle version du HTML entraine comme à son habitude quelques modifications. De nouvelles balises et attributs viennent s'ajouter et d'autres qu'on déconseille. Ci-dessous, le schéma d'un site classique ainsi que son squelette en HTML5 :

```
<!DOCTYPE html>
<html lang="fr">
<head>
  <meta charset="utf-8"> <title>contenu statique du haut de page</title>
</head>
<body>
  <header>
    <h1>Nom de la page</h1>
  </header>
  <nav>
    <ul>
      <li>Menu 1</li> <li>Menu 2</li><li>Menu 3</li>
    </ul>
  </nav>
  <main>
    <!--contenu principal-->
    <article>
      <header>
        <h1>Titre de l'article</h1>
        <p>Description de l'article</p>
      </header>
      <p>Contenu de l'article</p>
      <footer>Pied de l'article </footer>
    </article>
  </main>
  <section>
    <h2>Section Thématique</h2>
    <p>Cette section traite des sujets similaires.</p>
  </section>
  <aside>
    <h1>Nom de la colonne latéral</h1>
    <p> contenu complémentaire au contenu principal</p>
  </aside>
  <footer>
    <p>Bas de page.</p>
    <p>©2023 Mon Site. Tous droits réservés.</p>
  </footer>
</body>
</html>
```

Le schéma ci-dessous propose un exemple d'organisation de la page :



Sections de la page identifiées par les balises

## IV. Les balises sémantiques de structure de page en html5 :

La plupart des nouvelles balises sémantiques ajoutées à HTML5 sont des conteneurs principaux d'une page qu'on appelle des balises de sectionnement ou des balises de délimitation de zones dans une page. En effet, elles permettent de séparer les principales parties d'une page ce qui permet par la suite de garantir un meilleur balisage de sa structure.

Nous allons lister dans ce qui suit **sept** balises sémantiques de structure de page ajoutées à **HTML5**.

### 1. L'entête du site avec <header> :

Généralement, un site internet possède deux types de contenu :

- ✚ Une partie statique, en haut de page, que l'on retrouve sur l'ensemble des pages du site, c'est-à-dire qui s'affichera quelle que soit la page sur laquelle on se trouve.
- ✚ Une partie dynamique, qui va changer selon la page.

Les informations statiques en haut de page sont à mettre dans la partie haute, appelée ‘entête’ ou ‘**header**’ en anglais. Cette zone peut alors être définie à l’aide de la balise **<header>** où on y retrouvera souvent le logo du site, et au minimum sa navigation principale, mais ça peut aussi être des éléments de titre, des boutons d’outils pour la connexion, la gestion du profil, un formulaire de recherche, le nom d’auteur, la date, et beaucoup d’autres choses.

**Attention** , à ne pas confondre ou comparer avec l’élément **<head>** ! Contrairement au **<head>**, le **<header>** se place dans le **<body>** et affiche du contenu sur la page.

**Remarque** : La balise **<header>** peut être utilisée dans la balise **<section>** et dans la balise **<article>**.

**Exemple :**

```
1 <html>
2   <head>
3   </head>
4   <body>
5     <header>
6       <!-- contenu statique du haut de page -->
7     </header>
8   </body>
9 </html>
```

## 2. Les menus de navigation avec **<nav>** :

Au sein de la balise **<header>** qu’on vient juste de définir, nous pouvons incorporer une zone permettant de délimiter la navigation principale du site web. Cette zone peut être définie à l’aide de la balise sémantique **<nav>**.

Un document peut avoir plusieurs éléments **<nav>**, par exemple un pour la navigation principale sur le site où on présente des liens vers d’autres pages, et un autre destiné à la navigation au sein de la même page.

Il faut donc savoir que la balise **<nav>** n’est pas forcément placée dans le **header** d’un site, on peut la retrouver également à plusieurs endroits de la page comme par exemple dans le pied de page ou dans une barre latérale de contenu (sidebar).

**Exemple :**

```
<header>
  <nav>
    <a href="p1.html">Lien 1</a>
    <a href="p2.html">Lien 2</a>
    <a href="p3.html">Lien 3</a>
    <a href="p4.html">Lien 4</a>
  </nav>
</header>
```

### 3. Le contenu principal avec <main> :

Après avoir structuré la partie haute du site, il faut s'intéresser au contenu principal de la page (soit la partie dynamique d'un site).

L'élément HTML **<main>** sert à déterminer la zone, généralement étendue de votre page, où vous allez coder la majorité de contenu utile et important et directement en relation avec le sujet principal de la page (soit la fonctionnalité principale d'une application ou d'un site web).

Un document ne peut pas avoir plus d'un seul élément **<main>** qui représentera le contenu majoritaire du **<body>**. Ce contenu doit être unique dans le document. Les balises **<section>**, **<article>**, **<aside>**, **<footer>**, **<header>** ou **<nav>** ne peuvent pas contenir la balise **<main>**, par contre, on peut mettre ces balises à l'intérieur de **<main>**, à l'exception du **header** principal ainsi que le **footer** principal qui sont des parties différentes.

À la différence d'éléments tels que **<body>**, les niveaux de titre comme **<h2>**, ...etc. l'élément **<main>** n'affecte pas la structure même de la page, il s'agit d'un élément purement informatif qui ne contribue pas au plan du document.

#### Exemple :

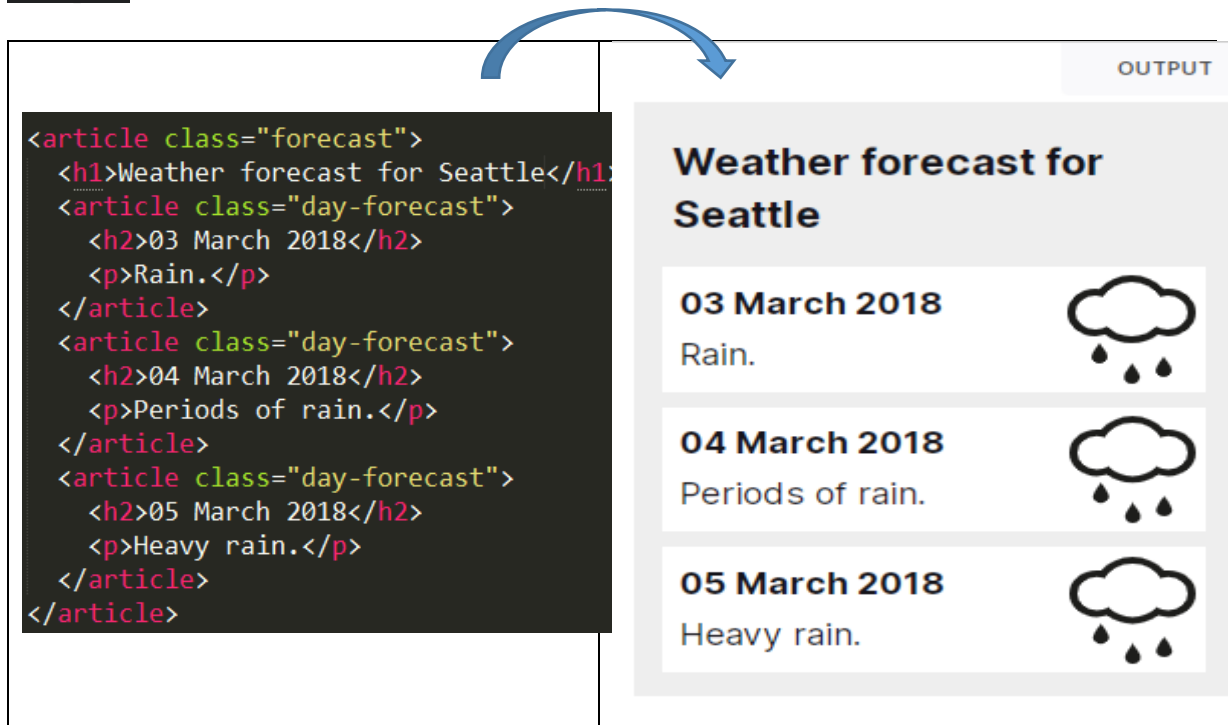
```
<html>
  <head>
  </head>
  <body>
    <header>
      <!-- contenu statique du haut de page -->
    </header>
    <main>
      <!-- contenu principal -->
    </main>
  </body>
</html>
```

### 4. Les contenus indépendants avec <article> :

Le contenu éditorial de votre site peut être balisé à l'aide de la balise **<article>** qui représente un contenu itérable. En français, un élément itérable est un élément qui peut être répété.

L'élément **<article>** représente une composition autonome dans un document, une page, une application ou un site, ayant son propre sens, différenciée facilement d'un autre contenu et destinée à être distribuée ou réutilisée de manière indépendante. Cette balise a d'autant plus d'importance sur un site e-commerce, un site d'actualité ou un blog. Elle va permettre de signaler aux robots d'indexation quels éléments sont en lien avec le sujet de la page.

Exemple :



Un document donné peut contenir plusieurs articles :

- Par exemple, sur un blog qui affiche le texte de chaque article l'un après l'autre au fur et à mesure que le lecteur fait défiler, chaque message sera contenu dans un élément **<article>**.
- Un autre exemple, sur un site e-commerce, tu auras une page avec la liste des produits proposés. Bien que les produits soient différents en contenu, en apparence ils sont normalement tous construits de la même façon, avec une image, un nom, une description, un prix et d'autres informations. On peut donc dire que les produits sont itérables, leurs balises d'affichages se répètent avec la même mise en forme, et simplement un contenu textuel adapté.

Remarque : La balise **<article>** peut avoir son propre **header** et **footer**.

## 5. Créer un découpage plus précis avec **<section>** :

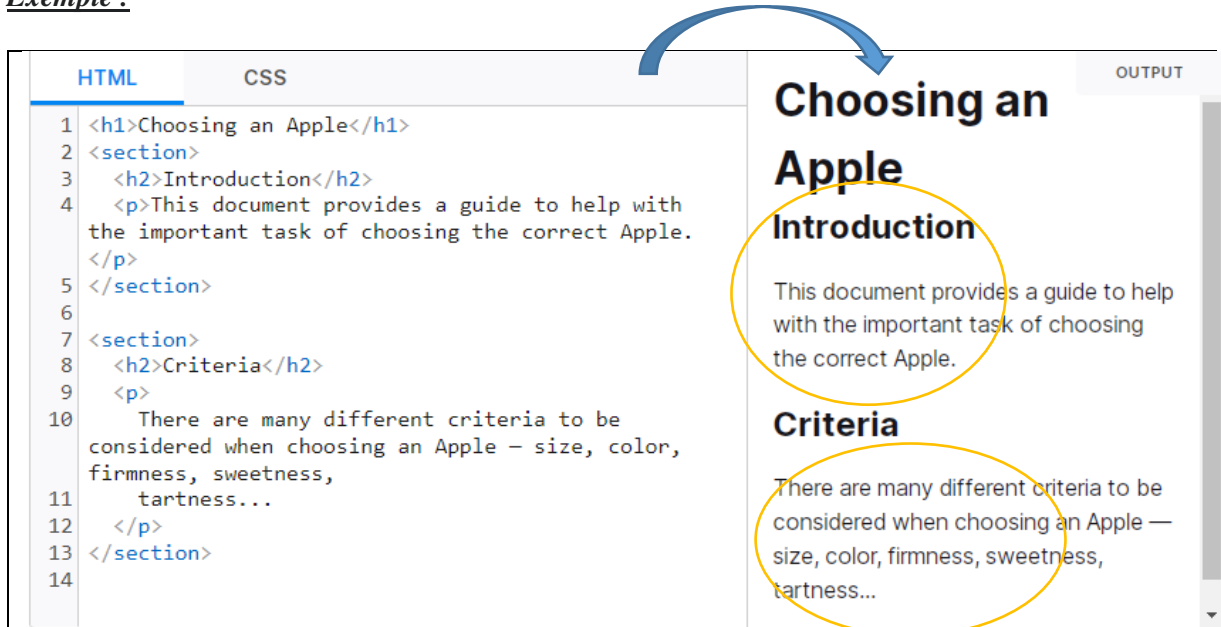
Quand le contenu de votre site présent dans une des balises **<main>** ou **<article>** commence à être plus ou moins complexe, il est important de correctement structurer le contenu principal de la page en plusieurs zones. Pour cela, il existe l'élément HTML **<section>** qui représente un bloc de contenu générique dans un document et qui permet de faire un découpage thématique (division sémantique) pour catégoriser et identifier convenablement les principales parties d'une page. Ainsi, chaque nouvelle balise **<section>** indique aux robots d'indexation une variété de contenus différents pour une même page.

Chaque élément **<section>** devrait être identifié, généralement grâce à un élément de titre (**<h1>**-**<h6>**).

Quand on crée une section, il faut que tout son contenu soit sur le même thème, c'est-à-dire que tous les éléments à l'intérieur doivent parler de la même chose.

La balise `<section>` peut être utilisée autant de fois que nécessaire, mais sans en abuser.

**Exemple :**



## **6. Le contenu annexe avec `<aside>` :**

Pour certaines pages, il est possible de proposer un encart de contenu qui est 'à côté' du contenu principal de la page, c'est-à-dire qui lui n'est pas directement lié, il est possible ainsi d'encadrer ce contenu tout simplement à l'aide de la balise `<aside>` (en français, « aparté »).

La balise `<aside>` est souvent utilisée pour un contenu complémentaire au contenu principal, par exemple pour une barre d'outils, ou un menu de navigation secondaire positionné à côté du `<main>`, ou pour délimiter des sidebars (les barres latérales de contenu annexes que l'on retrouve souvent sur des blogs),.

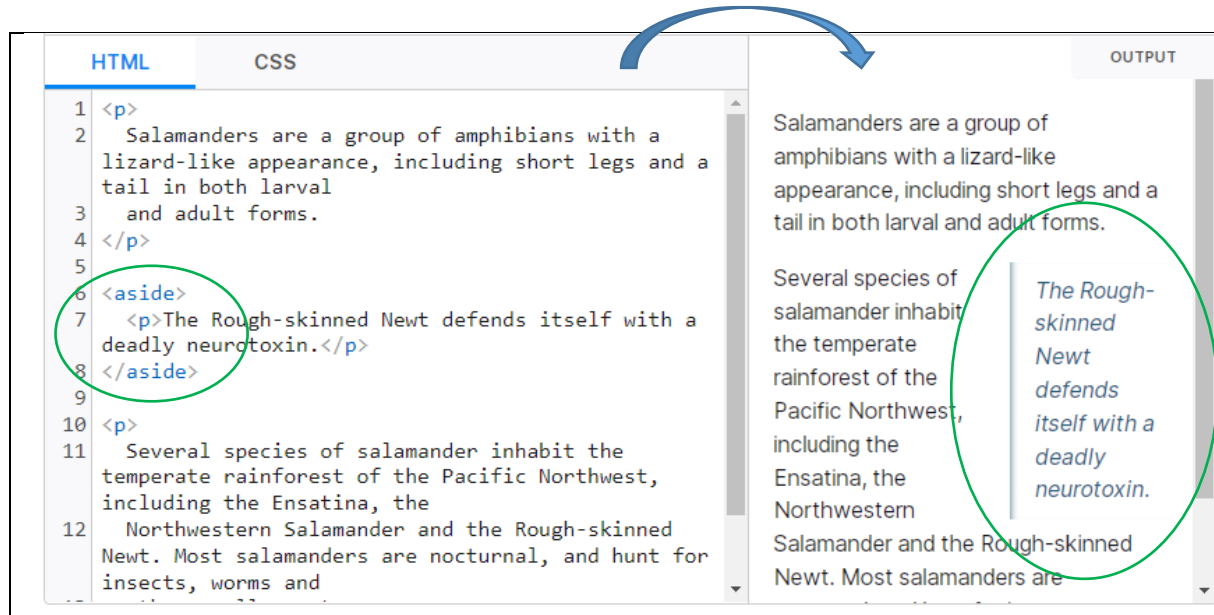
Les apartés sont fréquemment présents sous la forme d'encadrés. Il n'y a pas de règle quant à l'endroit où utiliser cette balise, à vous de choisir librement son emplacement en fonction de l'ergonomie de votre site web.

Il faut utiliser la balise `<aside>` avec précaution, il ne faut pas l'utiliser par exemple pour marquer du texte entre parenthèses, puisque ce type de texte est considéré comme faisant partie du flux principal. Encore, il ne faut pas confondre un contenu de la page qui serait positionné sur la droite ou sur la gauche avec un `<aside>`.

Lorsque la balise `<aside>` se trouve dans un article, son contenu se réfère à l'article lui-même et non au site Web (par exemple des notes de pages, un glossaire ou tout élément relatifs à l'article).



### Exemple :

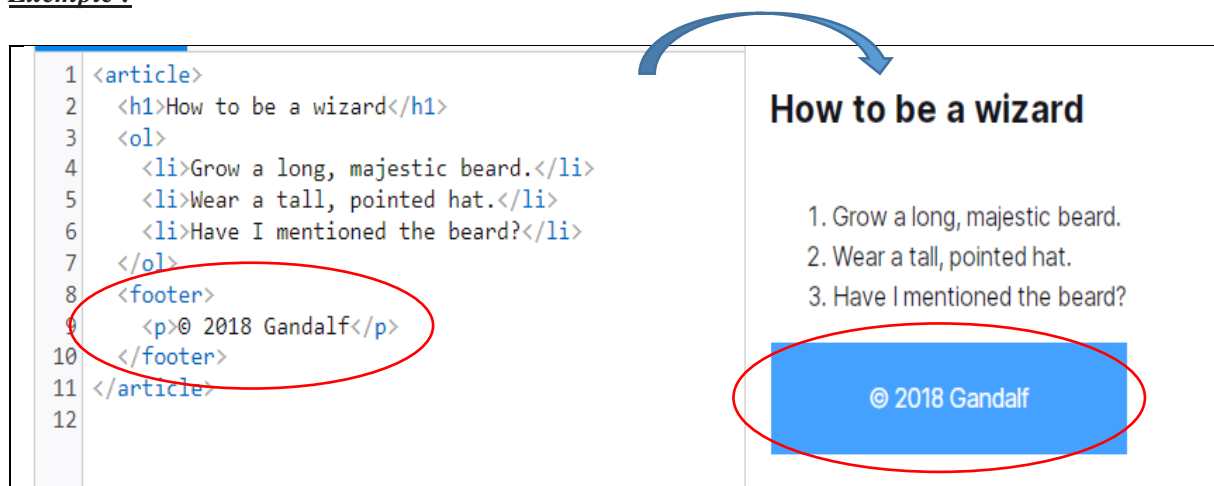


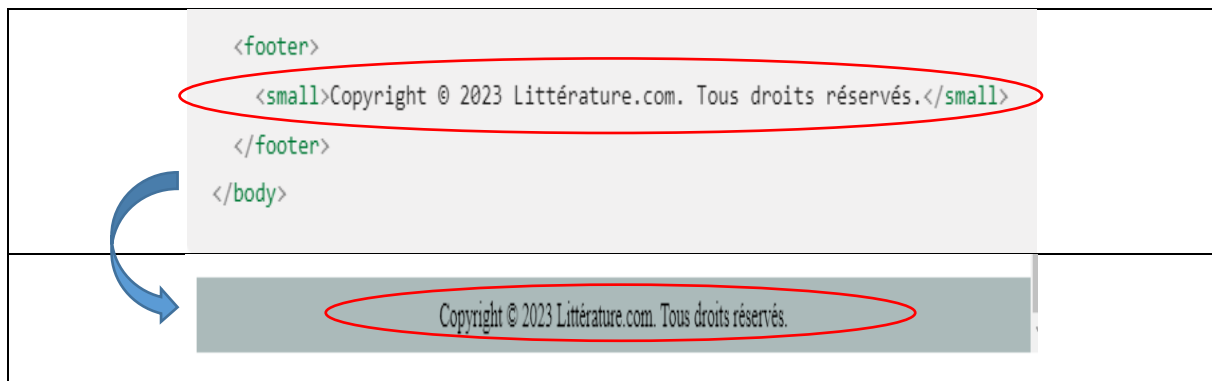
## 7. Le pied de page ou de section avec <footer> :

La balise `<footer>` sert principalement à délimiter le pied de page comme on peut l'utiliser pour baliser une fin d'article ou de section.

Un élément `<footer>` contient habituellement des informations sur l'auteur de la section, des données relatives au droit d'auteur (*copyright*), des liens de contact, le logo du site, etc. On peut également placer des balises `<nav>` dans le footer afin de délimiter des zones de navigation secondaire à travers des liens vers d'autres documents en relation.

### Exemple :





## V. Les principaux avantages offerts par HTML5 :

HTML5 a introduit des changements extraordinairement bienvenus non seulement pour les développeurs mais aussi pour les utilisateurs finaux. Dans cette partie, nous allons passer en revue les améliorations importantes apportées par cette nouvelle version :

- ✚ **Améliorations dans la gestion des erreurs :** La gestion des erreurs a été codifiée et redécouverte complètement sur HTML5. Le traitement cohérent des erreurs a fait une énorme différence à cet égard, les algorithmes d'analyse améliorés utilisés dans HTML5 aident les développeurs de navigateurs à normaliser et à réduire le temps ainsi que l'argent pour afficher les choses de manière cohérente sur une page Web.
- ✚ **Syntaxe simplifiée et règles d'analyse syntaxique améliorée :** HTML5 a amélioré les règles d'analyse syntaxique pour une meilleure compatibilité.
- ✚ **Amélioration de la sémantique des éléments :** Des améliorations ont été apportées aux rôles sémantiques de divers éléments existants en HTML pour améliorer l'insinuation du code.
- ✚ **De meilleurs contrôles des formulaires Web améliorés** avec des entrées de texte optimisées, des boîtes de recherche et différents champs pour divers usages, et ceci grâce à de nouveaux ajouts comprenant plusieurs entrées comme les suivantes : e-mail, numéro de téléphone, URL, diaporama, sélecteur de date et d'heure et sélecteur de couleur et autre...
- ✚ **Prise en charge améliorée des fonctionnalités des applications Web :** Permettre aux navigateurs Web de fonctionner comme des plateformes d'application en offrant aux développeurs un meilleur contrôle des performances de leurs sites Web par le moyen de nouveaux éléments en HTML5 fournissant toutes les fonctionnalités nécessaires pour se faire.
- ✚ **Support mobile renforcé :** Auparavant, à l'époque du HTML original, l'accès à l'internet par des appareils autres que les ordinateurs n'était pas encore même proposé ; aujourd'hui, tout est accessible à partir de nos téléphones, montres et télévisions, grâce en partie à la vitesse du HTML5. La nouvelle version a apporté de meilleures normes pour assurer un fonctionnement plus fluide et plus souple des sites web sur les appareils plus petits et moins puissants, et par conséquent, le Web sur mobile est simplifié.
- ✚ **La prise en charge d'éléments multimédias tels que l'audio, la vidéo, les graphiques vectoriels, les animations et les jeux :** Pour la vidéo et l'audio, l'intégration d'un lecteur est aussi simple que l'ajout d'une simple balise, vous êtes ainsi capables de créer des jeux vidéo à part entière avec HTML5, comme vous pouvez également intégrer des graphiques vectoriels.
- ✚ **Le stockage Web remplace les cookies :** Avant HTML5, les données des applications devaient être stockées dans des cookies. Avec le stockage Web, les applications Web peuvent stocker des données localement dans le navigateur de l'utilisateur. HTML5 prend également en charge le stockage dans une base de données web SQL. Notons que le stockage Web est plus sécurisé et de grandes quantités de données peuvent être stockées localement, sans affecter les performances du site Web.
- ✚ **Le taux de plantage des navigateurs mobiles est réduit.**

- ✚ **La géolocalisation de l'utilisateur est désormais offerte et possible** : L'API permet aux sites web et aux applications basées sur un navigateur compatible HTML5 d'accéder à la position géographique de l'utilisateur via le navigateur.

**Cependant**, HTML5 est compatible uniquement avec les navigateurs les plus récents, car il y a beaucoup de nouvelles balises et de nouveaux éléments que seuls certains navigateurs prennent en charge.

En plus des caractéristiques énumérées ci-dessus, la version HTML5 a vu d'autres changements comme les suivants :

- Certains éléments ont été supprimés, comme `noframes`, `acronym`, `font`, `frame`, `frameset`,..
- De nombreux nouveaux éléments ont été ajoutés notamment `video`, `canvas`, `audio`, `nav`, `aside`, `progress`, `section`, `time`,....
- De nouveaux attributs ont été introduits, tels que `translate`, `async`, `manifest`, `sandbox`, `srcdoc`, `hidden`, `reversed`,..

Veuillez visiter le lien suivant pour plus de détails sur la liste des balises HTML5 :

(<https://jaetheme.com/tutoriels-ui/liste-des-balises-html5/>)

## **VI. Les langages complémentaires à HTML5 :**

D'autres langages, complémentaires au HTML5, sont nécessaires pour réaliser des projets. Ils peuvent être soit directement intégré dans la page soit écrits sur une autre page et reliés à la page codée en HTML5. Parmi ces principaux langages, on retrouve :

- ✚ **CSS** : CSS est l'acronyme de « Cascading Style Sheets » ce qui signifie « feuilles de style en cascade ». C'est un langage informatique qui permet la mise en forme des pages web.
- ✚ **Javascript** : JavaScript est un langage de programmation de scripts principalement employé dans les pages web interactives en complément de HTML et CSS, et donc il présente une partie essentielle des applications web. Il permet de créer du contenu mis à jour de façon dynamique, de contrôler le contenu multimédia, d'animer des images, d'afficher une fenêtre d'alerte en cliquant sur un lien,...
- ✚ **PHP** : pour Hypertext Preprocessor, plus connu sous son sigle PHP. Il s'agit d'un langage de programmation principalement utilisé pour produire des pages Web dynamiques via un serveur web. C'est un langage interprété, cela veut dire que celui-ci doit être interprété par un serveur afin d'afficher correctement le contenu demandé. Il s'intègre naturellement au HTML pour générer et gérer des applications web interactives.
- ✚ **SQL** : sigle pour Structured Query Language, « langage de requêtes structurées » C'est un langage informatique utilisé pour manipuler des bases de données relationnelles.

## **Bibliographie :**

<https://developer.mozilla.org/>

<https://jaetheme.com/balises-html5/>