

# HTML 4/HTML 5

## 1. Site Statique et dynamique

Un site est dit statique ou à contenu statique, si les pages le contenant ne sont modifiées que par le webmaster. Il est dynamique ou à contenu dynamique si au moins une de ses pages est générée en temps réel sur requête d'un utilisateur par un programme exécutable ou un script interprété lancé par le serveur web. Qu'il soit statique ou dynamique, la structure des pages constituant un site est similaire. La structure était réalisée traditionnellement grâce à la balise `<frameset>` incluant plusieurs balises `<frame>` pour chaque partie d'une page. Cette technique populaire au début du web a laissé place plus tard à l'utilisation des tableaux html avec la balise `<table>` plus esthétique. Mais l'apparition puis la généralisation des terminaux de nouvelle génération (mobiles, tablettes, etc.) a encore une fois poussé à l'émergence d'une nouvelle version du HTML. Avec le HTML5, la structure est basée sur de nouvelles balises plus compréhensible par les robots des moteurs de recherches et plus facilement utilisée par les outils d'accessibilité et les appareils autres que le PC.

## 2. Le HTML 4 rapidement et autrement

Hypertext Markup Language, ou langage qui permet d'augmenter les documents hypertexte (contenant des liens) avec des mots clés appelés soit 'balise' soit 'attribut' est le langage décrivant les pages du world wide web. Ecepté `<img>`, `<br>`, `<hr>` et `<link>`, toutes les balises ont forme de début et une forme de fin (la forme de début est celle de début avec une barre oblique avant le nom de la balise comme `<html>...</html>`).

Toute page web ou document HTML est structuré comme suit :

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

Le document HTML constitue le corps d'un message de réponse HTTP d'un serveur web vers le client web (navigateur).

La partie corps de la page située entre les balises `<body>` contient toute l'information visible à l'utilisateur dans la fenêtre du navigateur (à l'exception du titre de la page qui est inséré dans `<head>`.)

Le titre est exprimé avec la balise `<title>`Titre significatif du contenu de la page `</title>`. Il faut toujours veiller à la renseigner faute de quoi, l'utilisateur aura du mal à se situer dans quel page d'un site il est, son navigateur affichera 'Sans titre' ou 'Untitled' comme titre de la fenêtre et les moteurs de recherche comme google affichent le contenu de cette balise lorsqu'ils affichent le lien vers une page comme un résultat de recherche.

## Chapitre 2 : HTML 4/HTML 5

La balise `<meta>` est aussi très importante. Par exemple `<meta charset= 'UTF-8'>` permet d'afficher correctement les caractères d'une langue comme le français ou autre.  
`<meta name='viewport', content="initial-scale=1.0">` (HTML5) permet d'adapter tout contenu HTML à la taille actuelle du terminal utilisateur.

De plus, le head peut contenir des balises de définition de style avec `<style> </style>`, d'inclusion d'autre fichiers avec `<link rel= 'stylesheet' href='URL' />` pour une feuille de style séparée, `<script type= 'text/css' src='URL d'un fichier css' />`, ou déclaration d'entités (variables ou fonctions) javascript en utilisant `<script type='text/javascript' src='URL d'un fichier js' />`

La balise `<body>` est destinée à recevoir le contenu édité statiquement ou généré dynamiquement par un script côté serveur.

Le contenu d'un document est constitué de textes explicatifs déclarés avec la balise paragraphe `<p>...</p>`. Les paragraphes (de 3 à 5) constituent des sections déclarées avec la balise `<h>` Titre de section ou de sous-section `</h>` où  $x=1..6$  selon le niveau de la section. Les tableaux HTML présentant des données ou des résultats avec la balise `<table>...</table>` où chaque ligne est déclarée avec `<tr>... </tr>`. Une ligne (row) contient un ensemble de cellules chacune déclarée avec `<td>..</td>`. Les figures présentent des résultats ou des des illustrations (desseins) en faisant appel à la balise `<img src='URL image' alt='Texte alternatif informatif de l'image' />`.

Enfin, un auteur peut être amené à énumérer un ensemble de faits soit de même importance avec `<ul> </ul>` ou d'importance hiérarchique avec les listes ordonnées déclarées avec la balise `<ol>...</ol>`. Une liste ordonnée ou non ordonnée contient des éléments (item) chacun déclaré avec la balise `<li>` nom de l'élément `</li>`.

HTML5 : Une nouvelle façon d'organiser le contenu, de contrôler les erreurs et de s'adapter aux nouveaux appareils

Avec HTML5, les tableaux ne sont plus utilisés pour structurer le contenu (statique ou dynamique) d'une page, ni les balises `<frameset>` et `<frame>` qui sont éliminées définitivement du HTML. A la place : des balises ayant du sens (balises sémantiques) sont introduites.

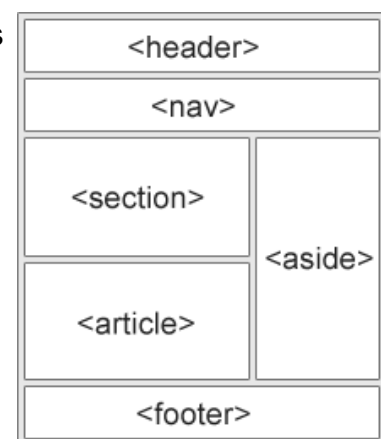
header, nav, article, aside et footer permettent de désigner les différentes parties d'une page où header va contenir l'en-tête (un logo, le nom de l'entreprise, un slogan et une zone de recherche éventuelle), le menu, le corps de la page, l'information supplémentaire (pub ou événements ou autre), et le bas de page (informations légales, droits des utilisateurs, licence d'utilisation etc...), respectivement.

### 3. Nouvelles balises HTML 5

Le HTML5 offre de nouvelles balises sémantiques pour désigner les différentes parties d'une page web.

Celles-ci sont :

- `<header>`
- `<article>`
- `<details>`



## Chapitre 2 : HTML 4/HTML 5

- `<figcaption>`
- `<figure>`
- `<footer>`
- `<aside>`
- `<main>`
- `<mark>`
- `<nav>`
- `<section>`
- `<summary>`
- `<time>`

### 3.1 L'élément `<header>`

Selon le W3C, l'élément header spécifie un en-tête pour le document ou pour une partie de la page. Par conséquent, il peut apparaître soit au début de `<body>` soit à l'intérieur d'une balise `<section>` ou `<article>`.

Exemple :

```
<article>
  <header>
    <h1>What Does WWF Do?</h1>
    <p>WWF's mission:</p>
  </header>
  <p>WWF's mission is to stop the degradation of our planet's natural
environment, and build a future in which humans live in harmony with
nature.</p>
</article>
```

On peut avoir plusieurs `<header>` dans une page web.

### 3.2 L'élément `<footer>`

L'élément footer désigne un bas-de-page pour un document HTML soit pour une partie d'une page HTML. Il doit contenir des informations sur l'élément qui le contient. Généralement, il contient l'auteur du document, des informations sur le copyright, des liens vers les conditions d'utilisation du document (informations légales), des informations de contact, etc.

Exemple :

```
<footer>
  <p>Posted by: Hege Refsnes</p>
  <p>Contact information: <a href="mailto:someone@example.com">
someone@example.com</a>.</p>
</footer>
```

### 3.3 L'élément `<section>`

L'élément `<section>` définit une section (au sens de partie) d'un document. Selon la documentation

## Chapitre 2 : HTML 4/HTML 5

du W3C sur HTML5, « une section est un groupe thématique de contenu, typiquement avec un titre »

Exemple <section>

```
<section>
  <h1>WWF</h1>
  <p>The World Wide Fund for Nature (WWF) is....</p>
</section>
```

### 3.4 L'élément <article>

L'élément <article> spécifie un contenu indépendant et auto-contenu qui peut être distribué séparément (sur un autre site par exemple) des autres éléments de la page. Un article doit avoir du sens et pouvoir être lu seul, indépendamment des autres contenus d'un site.

Exemples de contenus qui peuvent être mis dans un <article> : un post dans un forum ou un blog, un article d'un journal.

Exemple :

```
<article>
  <h1>What Does WWF Do?</h1>
  <p>WWF's mission is to stop the degradation of our planet's natural
environment,
  and build a future in which humans live in harmony with nature.</p>
</article>
```

### 3.5 L'élément <nav>

L'élément <nav> spécifie un ensemble de liens de navigation.

Exemple :

```
<nav>
  <a href="/html/">HTML</a> |
  <a href="/css/">CSS</a> |
  <a href="/js/">JavaScript</a> |
  <a href="/jquery/">jQuery</a>
</nav>
```

### 3.6 L'élément <aside>

L'élément <aside> définit un contenu quelque peu à côté du contenu dans lequel il est placé (comme une barre latérale). Le contenu 'à côté' doit être en relation avec le contenu qui l'entoure.

Exemple

```
<p>My family and I visited The Epcot center this summer.</p>
```

```
<aside>
  <h4>Epcot Center</h4>
  <p>The Epcot Center is a theme park in Disney World, Florida.</p>
</aside>
```

### 3.7 L 'élément <figure> et <figcaption>

Dans les livres et les journaux, il est courant de trouver des explications avec les images.

L'objectif de <figcaption> est de rajouter une explication visuelle à l'image. En HTML5, l'image et son titre peuvent être regroupés dans une balise <figure>.

Exemple :

```
<figure>
  
  <figcaption>Fig1. - The Pulpit Rock, Norway.</figcaption>
</figure>
```

### 3.8 L'élément <main>

L'élément <main> contient le contenu principal d'une page.

L'élément <mark> désigne un texte marqué / sur-élevé.

### 3.9 Les éléments <details> et <summary>

Destiné à présenter un contenu qui peut avoir des détails.

Exemple :

```
<details>
  <summary>Copyright 1999-2014.</summary>
  <p>- by Refsnes Data. All Rights Reserved.</p>
  <p>All content and graphics on this web site are the property of the company
  Refsnes Data.</p>
</details>
```

### 3.10 L'élément <time>

L'élément HTML5 <time> contient une date/heure lisible/compréhensible par un être humain. Un lecteur d'écran détectera que c'est une heure et prononcera correctement l'heure au lieu de prononcer seulement des nombres.

Exemple :

```
<p>We open at <time>10:00</time> every morning.</p>
```

Un des avantages majeurs de ces balises sémantiques est de permettre aux programmes informatiques de détecter des parties clés d'une page comme le contenu principal ou la barre de

navigation. Ces informations sont essentielles pour les lecteurs d'écrans, par exemple.


### 3.11 Les éléments <meter> et <progress>

L'élément <progress> permet d'indiquer le degré de progrès d'une tâche (ou un processus dynamique)

Exemple

Downloading progress :

```
<progress value="22" max="100"></progress>
```

Downloading progress: 

L'élément <meter> permet d'indiquer un niveau (d'une jauge) et est utilisé pour indiquer des valeurs qui ne changent pas très vite (il n'y a pas de processus en cours modifiant la valeur)

Exemple

```
<meter value="2" min="0" max="10">2 out of 10</meter><br>
```

```
<meter value="0.6">60%</meter>
```



### 3.12 Nouvelles balises relatives au contenu audio/video

Les balises <audio> <source> et <video> sont introduites avec le HTML5 et avec elles, les attributs autoplay, loop, controls, type, muted et src (auxquels il faut rajouter width + height pour <video>)

Exemple audio :

```
<audio controls>
  <source src="horse.ogg" type="audio/ogg">
  <source src="horse.mp3" type="audio/mpeg">
  Your browser does not support the audio tag.
</audio>
```

Exemple video

```
<video width="320" height="240" controls>
  <source src="movie.mp4" type="video/mp4">
  <source src="movie.ogg" type="video/ogg">
  Your browser does not support the video tag.
</video>
```

### 3.13 Nouvelles balises et attributs relatifs aux formulaires

Les éléments HTML5 <fieldset>, <legend> et <label> :

Ces éléments HTML5 sont utilisés dans les formulaires pour regrouper visuellement des champs, ensemble, qui seront entourés d'un cadre fin qui contiendra un titre exprimé avec <legend>. Les éléments de type <input> ont l'attribut 'id' qui est utilisé pour leur associer une étiquette avec <label> qui apparaîtra au-dessus de l'input dans un mobile et à leur gauche sur un ordinateur (notez l'attribut for de <label> qui est égal à la valeur de l'id de l'input qu'il doit toujours accompagner).

## Chapitre 2 : HTML 4/HTML 5

Notez aussi les nouveaux types de `<input>` : tel, email, number, date. A noter ici, que sur un smartphone, un champs de type tel/number fera apparaître un clavier numérique, alors que le champs de type text fera apparaître le clavier alphabétique. On voit ici un exemple de l'adaptation de HTML5 aux mobiles.

Exemple

```
<form>
<fieldset>
<legend>Informations personnelles</legend>
<label for='nom1'> Nom:</label>
<input type='text' id='nom1' name='nom' />
<label for='prenom1'>Prénom:</label>
<input type='text' id='prenom1' name='prenom' />
</fieldset>
<fieldset>
<legend>Contact</legend>
<label for='adr1'> Adresse: </label>
<input type='text' id='adr1' name='adr' />
<label for='adr2'> Email: </label>
<input type='email' id='adr2' name='email' />
<label for='numtel1'> Tél: </label>
<input type='tel' id='numtel1' name='numtel' />
</fieldset>
</form>
```

Ce code produira l'apparence suivante :

<b>Informations personnelles</b>		
Nom:	<input type="text"/>	Prénom: <input type="text"/>
<b>Contact</b>		
Adresse:	<input type="text"/>	Email: <input type="text"/> Tél: <input type="text"/>

En HTML5, on a de nouveaux types d'attributs qui sont : color, date, datetime, datetime-local, month, week, time, email, number, range, search, tel, et url.

Ses types se rajoutent aux types HTML4 : button, radio, checkbox, text, textarea, password, file, submit et reset.

### Autres attributs HTML5

draggable : un élément peut être saisi par un clique long sur la souris et déposé ailleurs

dropzone : définit une zone où l'on peut déposer des éléments pris avec la souris.

spellcheck : vérifie l'orthographe d'un élément pour lequel l'utilisateur saisit le contenu

required : pour un élément de type input, rend le champs obligatoire, le navigateur ne postera pas le formulaire tant que ce champ n'est pas correctement rempli.

placeholder : pour un élément de type input, texte qui apparaîtra dans la champs tant que l'utilisateur n'y a rien saisi.

translate='yes|no' : indique si le contenu de l'élément doit être traduit ou non

contenteditable='true|false' : le contenu d'un élément peut être édité (modifié) ou non

## 4. Attributs globaux (HTML 4 et HTML5)

Les attributs globaux sont des attributs qui appartiennent à toutes les balises HTML.

lang : la langue du contenu texte d'une balise

hidden : pour un élément de type input, rend le champs invisible à l'utilisateur.

dir : la direction du contenu d'une balise (rtl ou ltr)

id : identificateur d'une balise ; très important par exemple pour la manipulation d'une balise avec le javascript à travers le DOM.

style : permet d'appliquer un style particulier à une balise qui va surpasser tout ce qui a été défini comme style auparavant.

tabindex : spécifie l'ordre d'un élément lorsque l'utilisateur visite les éléments de la page avec la touche tabulation (tab)

accesskey : définit un raccourci clavier vers un élément HTML, utilisable lorsque l'utilisateur ne dispose pas de souris pour cliquer directement sur l'élément.

## 5. Conclusion

Dans cette partie du cours, nous avons parlé du langage du web, le HTML qui est maintenant à sa version 5. Le HTML5 sert à structurer les pages et énoncer du contenu informatif ou des données mais l'apparence (la présentation) dépend des styles CSS qui ont été séparés du HTML pour des raisons que nous allons aborder dans le chapitre « Styles et CSS. »