

Introduction au HTML

structurer ses données

Introduction au HTML

Rôle

HTML signifie **H**yper **T**ext **M**arkup **L**anguage soit en français : langage de marquage de document hyper-text. Il s'agit de **structurer** les données en fonction de leur **sémantique** (sens) : est-ce que j'ai un titre ? un paragraphe ? un lien (qui fait qu'un texte est un hyper-texte) ? une image ? etc.

Introduction au HTML

Règle absolue

Le HTML ne doit pas être utilisé pour faire de la mise en forme ! Les balises sont utilisées pour leur sémantique, **jamais** pour leur rendu.

Introduction au HTML

Structure d'une page

```
<!DOCTYPE html>
<html>
  <head>
    <!-- ici des méta-données = des informations utiles pour le document mais
         qui ne s'affiche pas et ne doivent pas s'afficher dans le navigateur -->
  </head>

  <body>
    <!-- ici notre document visible dans le navigateur -->
  </body>
</html>
```

Les éléments sémantiques

On structure la donnée grâce à différents **éléments sémantiques** :

- corps du document
- titre principal ou de niveau 1
- sous-titre ou titre de niveau 2
- paragraphe
- liste
- lien
- image
- citation
- ...

Introduction au HTML > **Structure d'une page**

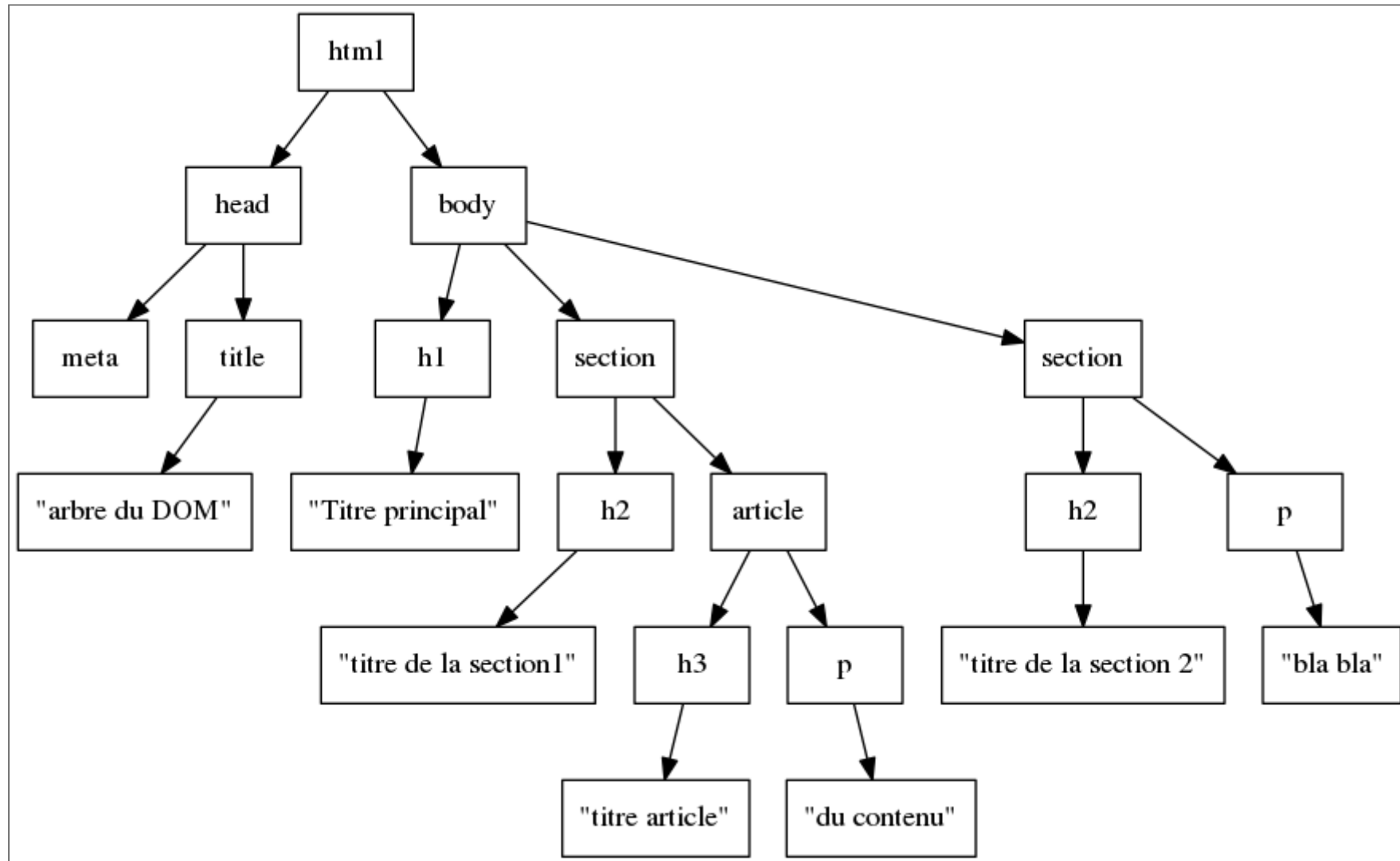
Des éléments aux balises

Les éléments sémantiques sont matérialisés dans le document par des balises. Il s'agit d'un petit texte encadré par des chevrons : `< >`. On en a déjà rencontré 3 :

- la balise `<html>` ... avec sa balise **fermante** `</html>`
- la balise `<head>` ... avec sa balise **fermante** `</head>`
- la balise `<body>` ... avec sa balise **fermante** `</body>`

Introduction au HTML > **Structure d'une page**

Les balises imbriquées créent une arborescence



Introduction au HTML > **Structure d'une page**

Un peu de vocabulaire

- Entre une balise **ouvrante** et la balise **fermante** correspondante, il y a divers éléments de structuration : du texte et/ou d'autres balises.

Exemple

La balise `<html>... </html>` ne contient que 2 balises : `<head>` et `<body>`, **rien d'autre**.

- Une balise sans balise fermante est dite **orpheline**
 - `<hr>` permet d'insérer une ligne horizontale
 - `` est la balise pour insérer une image
 - `
` pour passer à la ligne à l'intérieur d'un paragrapheparfois on ajoute un `/` avant le chevron fermant, comme ceci : `
` on voit de suite qu'il s'agit d'une balise orpheline.

Attribut de balise

- C'est une information qui n'apparaît pas dans la page mais qui est donnée à une balise sous la forme d'une propriété associée à une valeur. Certains attributs sont indispensables (comme par exemple celui pour spécifier à la balise `` quel fichier on souhaite afficher). La valeur est séparée de sa propriété par un caractère `=` et est **toujours** entre guillemets `" "` même quand il s'agit de nombres.
- Certains attributs sont universels (commun à toutes les balises ou presque) d'autres sont spécifiques.

Exemple

```

```

Les balises incontournables

- `<html>...</html>` la première balise, contient `<head>...</head>` et `<body>...</body>`
- `<head>...</head>` pour les méta-données
- `<body>...</body>` pour notre contenu ; **attention** ne contient que des balises de type *block*
- Les balises de titres, sous-titres, etc. : `<h1>...</h1>`, `<h2>...</h2>` ... jusqu'à `<h6>...</h6>`
- La balise pour un paragraphe : `<p>...</p>`
- La balise pour une image : ``
- La balise pour un lien : `...`
- Les balises de listes :
 - pour une liste non ordonnée : `... `

- pour une liste ordonnée : `...`
- Les items de liste : `...`
- Les balises de tableau (détaillé plus bas)

Les balises `<h1>...</h1>...`

- **type** : *block*
- **contenu** : du texte ou des balises de type *inline*
- **bonne pratique** : on veillera à utiliser les différents niveaux dans l'ordre, sans sauter de niveau
- **exemple** :

In [3]:

```
%%HTML
<h1>Les langages de programmation</h1>
  <h2>Python</h2>
  <h2>Javascript</h2>
<h1>Les langages de balisage</h1>
  <h2>XML</h2>
  <h2>HTML</h2>
```

Les langages de programmation

Python

Javascript

Les langages de balisage

XML

HTML

La balise `<p>...</p>`

- **type** : *block*
- **contenu** : du texte ou des balises de type *inline*
- **exemple** :

```
<p>Bonjour. Nous sommes ici pour apprendre le <strong>HTML</strong>.</p>
```


La balise `<img...>`

- **type** : *inline*
- **contenu** : aucun
- **attributs indispensables** : `src` pour indiquer le chemin du fichier, `alt` pour un texte alternatif en cas d'impossibilité soit d'afficher l'image, soit de visualiser l'image (navigateur textuel ou en audio description)
- **exemple** :

```

```

La balise `<a...>`

- **type** : *inline*
- **contenu** : texte ou balises de type *inline*
- **attributs indispensables** : `href` pour indiquer la cible du lien, on parle d'URL (Uniform Resssource Locator). Cette cible peut-être une page web externe, un page web interne, un endroit précis d'une page web (ancree)

- **exemple URL externe** :

```
<a href="http://www.univ-reunion.fr">Site web de l'Université de la  
Réunion</a>
```

- **exemple URL interne** :

```
<a href=" ../autres_pages/exemples.html">Notre page d'exemples</a>
```

- **exemple ancre interne à la page** :

```
<a href="#haut">Notre page d'exemples</a>
```


La balise `...` (*unordered list*)

- **type** : *block*
- **contenu** : uniquement des balises `...`
- **exemple** :

In [1]:

```
%%HTML
<ul>
  <li>3kg de mangues</li>
  <li>1 pack de lait</li>
  <li>2 paquets de café</li>
</ul>
```

- 3kg de mangues
- 1 pack de lait
- 2 paquets de café

La balise `...` (*ordered list*)

- **type** : *block*
- **contenu** : uniquement des balises `...`
- **exemple** :

In [2]:

```
%%HTML
<ol>
  <li>Dans un récipient adapté au four à micro-ondes, mélangez 350ml
    d'eau et 400ml de lait en salant à votre convenance</li>
  <li>Versez le contenu du sachet en mélangeant bien</li>
  <li>Faites chauffer 5 minutes</li>
  <li>Ajoutez une noix de beurre, mélangez, c'est prêt</li>
</ol>
```

1. Dans un récipient adapté au four à micro-ondes, mélangez 350ml d'eau et 400ml de lait en salant à votre convenance
2. Versez le contenu du sachet en mélangeant bien
3. Faites chauffer 5 minutes

4. Ajoutez une noix de beurre, mélangez, c'est prêt

La balise `...` (*list item*)

- **type** : *block*
- **contenu** : texte ou tout type de balise, mais ne se trouve qu'à l'intérieur des balises `` et ``
- **exemple** :

In [11]:

```
%%HTML
<ul>
  <li>Entrées
    <ul>
      <li>Salade du chef -- 8€</li>
      <li>Foie gras maison -- 14€</li>
    </ul>
  </li>
  <li>Pizzas
    <ul>
      <li>Reine -- 11€</li>
      <li>Végétarienne -- 12€</li>
    </ul>
  </li>
</ul>
```

- Entrées
 - Salade du chef -- 8€
 - Foie gras maison -- 14€
- Pizzas
 - Reine -- 11€
 - Végétarienne -- 12€

Tableau

Plusieurs balises sont nécessaires. **Attention** un tableau HTML ne doit être utilisé **que** pour des données sous forme de tableau, et non pour faire de la mise en page.

- `<table>...</table>` la balise pour créer le tableau
- `<tr>...</tr>` pour une *table row* ie une ligne
- `<td>...</td>` pour une cellule de donnée (*table data*)

Introduction au HTML > Structure d'une page > **Tableau**

In [1]:

```
%%HTML
<table>
  <tr><td>Pays</td><td>Capitale</td></tr>
  <tr><td>France</td><td>Paris</td></tr>
  <tr><td>Belgique</td><td>Bruxelles</td></tr>
  <tr><td>Suisse</td><td>Berne</td></tr>
</table>
```

Pays	Capitale
France	Paris
Belgique	Bruxelles
Suisse	Berne

Attention le rendu du tableau est propre au notebook ici, il diffère du rendu **par défaut** d'un navigateur.

Autres balises

- `...` l'emphase permet de mettre du *texte en avant*
- `...` pour une information **importante**
- `
` pour insérer un passage à la ligne
- `<hr>` pour insérer une ligne horizontale
- `<blockquote>...</blockquote>` pour une citation longue (plusieurs lignes)
- `<q>` citation courte (une ligne)

Balises structurantes de type *block*

- `<main>...</main>` contenu principal de la page. Devrait être unique et pas comporter de contenu répété dans les autres pages (menu de navigation, informations style copyright, logos etc.)
- `<header>...</header>` définit l'en-tête d'une page. On pourra y trouver le titre principal, un logo ou une bannière, le menu de navigation...
- `<footer>...</footer>` définit le pied-de-page ; on y trouve les crédits, lien vers une éventuelle FAQ, le contact etc.

Balises structurantes de type *block*

- `<section>...</section>` une section du document
- `<article>...</article>` définit une portion indépendante de contenu, que l'on pourrait réutiliser telle quelle ailleurs, dans un autre site. Une section peut contenir plusieurs articles.
- `<aside>...</aside>` définit un complément d'information ; en lien avec le contenu principal mais facultatif et qui pourrait être retiré sans que cela nuise à l'ensemble.
- `<nav>...</nav>` définit un menu de navigation. En général contient une liste non ordonnée.

Balises *boîtes*, sans signification

- `<div>...</div>` balise de type *block* pour regrouper plusieurs éléments de la page
- `...` balise de type *inline* pour regrouper plusieurs caractères de la ligne

Introduction au HTML > **Structure d'une page**

Les méta-données

Parmi les plus usuelles, on a :

- le `<title>...</title>` qui le titre de votre page, le texte qui apparaît dans la fenêtre ou l'onglet de votre navigateur. C'est aussi ce titre qui sera sauvegardé dans vos *bookmarks*
- l'encodage via la balise `<meta ...>` : c'est une balise qui change de rôle en fonction des attributs qu'on lui donne. Pour l'encodage on fera :

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

- le(s) lien(s) avec la/les feuille(s) de style CCS. Pour cela on utilise la balise `<link ...>` :

```
<link rel="stylesheet" href="nom_feuille_de_style">
```

Autres méta-données

- auteur :

```
<meta name="author" content="Sébastien Hoarau">
```

- description :

```
<meta name="description" content="Introduction au langage HTML.">
```

- *favicon* : raccourci pour *favorite icon*. Cette petite icône, sorte de logo de votre site apparaîtra dans la barre de titre de l'onglet ou la fenêtre et à côté de l'URL dans les bookmarks. Elle s'obtient en ayant un fichier image de type `.ico` et avec la balise `<link ...>` :

```
<link rel="shortcut icon" href="nom_du_fichier_icone" type="image/x_icon">
```

Bien que la méta pour les mots-clés existe, elle n'est plus utilisée : les moteurs de recherche n'en tiennent pas compte à cause des *spams* qui remplissent ce champs avec des centaines de mots clés biaisant complètement le résultat.

Attributs universels

Ce sont des attributs qui peuvent s'utiliser avec toutes les balises.

- `class` permet de mettre des sortes de *catégories* aux éléments.
- `id` permet de donner à une balise un identifiant **unique** dans la page.
- `title` permet d'ajouter une information le plus souvent visible au survol de l'élément

In [3]:

```
%%HTML  
<p class="bleu encadrer">Ce texte doit être écrit en bleu et encadré.  
Ce sera le rôle de la CSS</p>
```

Ce texte doit être écrit en bleu et encadré. Ce sera le rôle de la CSS

Dans la feuille de style on fera référence à la catégorie *bleu* : `.bleu` ou *encadrer* : `.encadrer` (notez bien le point avant le nom de la classe).

In [7]:

```
%%HTML
<h1 id="top">Titre principale de ma page</h1>
...

<!-- plus loin dans la page -->

<p><a href="#top">Retour en haut</a></p>
```

Titre principale de ma page

...

Retour en haut

In [6]:

```
%%HTML  
<p>Le <abbr title="World Wide Web Consortium">W3C</abbr> préconise le HTML5.</p>
```

Le W3C préconise le HTML5.

Introduction au HTML

Bibliographie et outils

- **Le site W3School** : tutoriel (en anglais) et manuel de référence (liste de balises)
- **Le site du W3C** : l'organe officiel des technologies du web
- **L'outil de validation du W3C** : permet de vérifier que votre source est conforme HTML5
- **HTML5 Rocks** : bibliothèque très fournie de ressources pour HTML5 (et plus)
- **html5test** : Jusqu'à quel point votre navigateur supporte le HTML5 ?