

Langages de description des pages web

le HTML

Hyper Text Markup Language = « langage de balisage pour document hypertexte »

Le HTML est un langage de description de documents : il décrit comment un document est construit, et donc comment un navigateur peut l'afficher



Le navigateur effectue le rendu de la page en interprétant le fichier HTML

HTML ne décrit que la structure et les liens du document.

La forme du document est décrite dans un fichier séparé écrit en CSS (Cascading Style Sheets : feuilles de style en cascade)

⇒ Le développement de pages web se fera en deux temps

- HTML pour créer les éléments à afficher...
- ... Puis CSS pour améliorer le rendu visuel

Historique

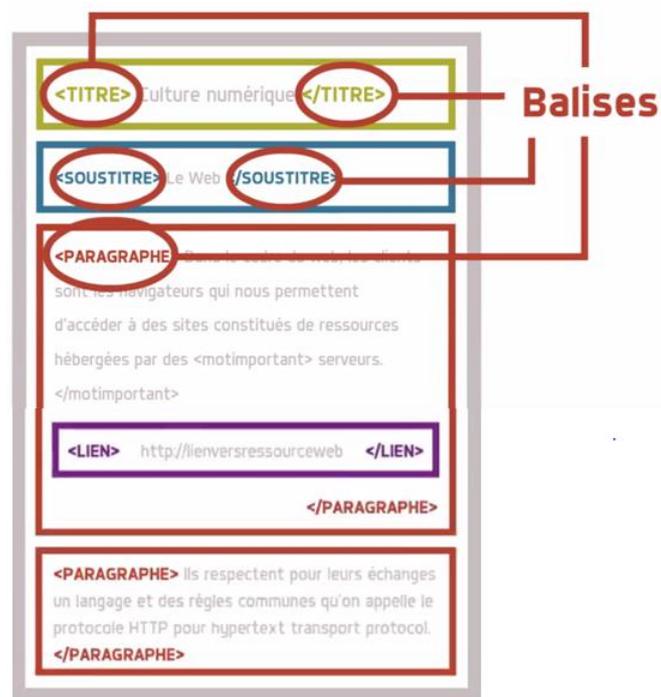
1997 : Le World Wide Web Consortium (W3C) publie la norme HTML 3.2 puis HTML 4.0.

depuis 2007 : HTML 5

HTML est l'une des trois inventions à l'origine du World Wide Web :

- Hypertext Markup Language (HTML)
- Hypertext Transfer Protocol (HTTP)
- Adressage web (URL)

HTML est un Langage à balises ...



qui prennent la forme suivante :

```
Code HTML
<HTML>
  <HEAD>
    <TITLE>Titre de la page</TITLE>
  </HEAD>
  <BODY>
    Contenu de la page
  </BODY>
</HTML>
```

Les balises peuvent avoir plusieurs formes :

- **<balise> ... </balise>** : elles s'ouvrent et se ferment pour délimiter le contenu (début et fin d'un titre, par exemple).
- **<balise />** : balises orphelines (on ne les insère qu'en un seul exemplaire), elles permettent d'insérer un élément à un endroit précis (par exemple une image).

Les balises sont parfois accompagnées **d'attributs** pour donner des indications supplémentaires (exemple : `<image nom="photo.jpg" />`)

- Un attribut est une paire nomAttribut="valeur" ou nomAttribut='valeur'

```
<link rel="stylesheet" type="text/css" href="nsi.css">
```

Un élément se compose de

- une balise ouvrante
- un contenu
- une balise fermante (avec le nom identique à celui de la balise ouvrante)

Exemple : élément "liste à puces" : (liste non ordonnée)

```
<ul>
  <li>Préambule</li>
  <li>Démarche de projet</li>
  <li>Modalités de mise en oeuvre</li>
  <li>Éléments de programme</li>
</ul>
```

→

- Préambule
- Démarche de projet
- Modalités de mise en oeuvre
- Éléments de programme

- Un élément vide est constitué uniquement d'une balise ouvrante et se termine par «/»

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

- Contenu d'un élément :

- Le contenu peut être constitué de texte ou d'autres éléments
- Deux éléments ne peuvent s'entrecouper
 - soit ils sont complètement disjoints
 - soit l'un est inclus dans l'autre

```
<aside>
  <h1>Programme complet de 1<sup>ère</sup></h1>
  <br><center><a href="1e_NSI.pdf"></a></center>
</aside>
```

Règles de bonnes pratiques (même si les navigateurs sont tolérants) :

– Le nom des balises en **minuscule**

– dans les balises :

- Un espace entre chaque attribut
- Pas d'espace entre le nom de l'attribut, le signe '=' et la valeur de l'attribut
- Chaque valeur d'attribut comprise entre " " ou ' '

Les outils pour développer sa page HTML :

- Écrire du code HTML :

– Éditeur de texte simple : Bloc notes (bof), NotePad++, Sublime Text
– IDE : Eclipse, IntelliJ

- Visualiser le résultat :

– dans un navigateur : Internet explorer, Firefox, Chrome, etc.

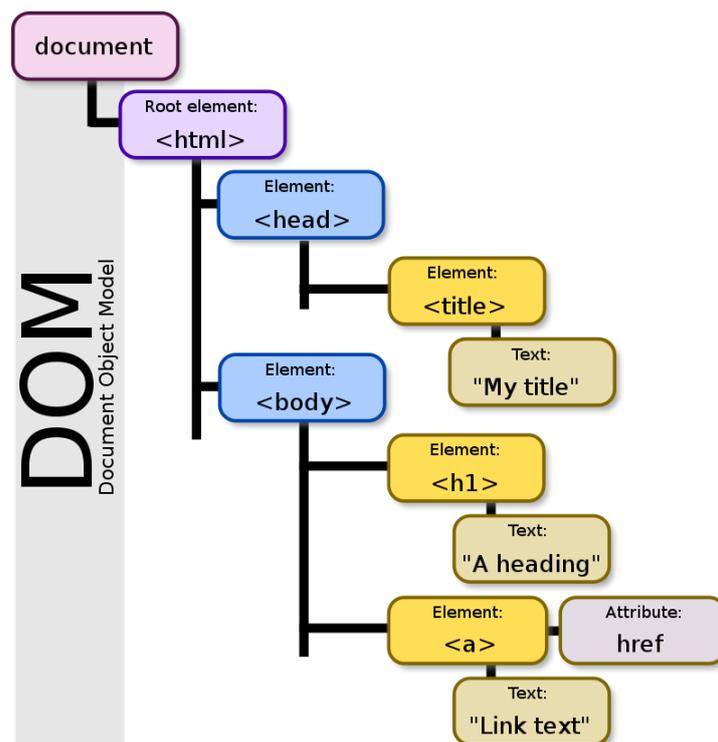
- Corriger les éventuelles erreurs : dans le menu du navigateur : «Plus d'outils», «Outils de développement»
- Vérifier la validité / conformité de son document :
<https://validator.w3.org/>

Structure d'un fichier HTML :

- Première ligne du fichier : le doctype
- Indique qu'il s'agit d'une page au format HTML et donne la version

Un fichier HTML suit une structure arborescente :

- L'élément racine : **html**
- Constitué d'un en-tête, défini par l'élément **head**
- Et d'un contenu, décrit par l'élément **body**



By Birger Eriksson - Own work, CC BY-SA 3.0

La structure du document HTML est décrite par le **DOM** : *Document Object Model* (DOM) ou « **modèle d'objets de document** ».

Cette structure est normalisée permettant à des **scripts** d'examiner ou de modifier les objets de la page.

ex : mise à jour de l'affichage de l'heure...

L'en-tête

Contient des informations sur le document HTML, mais ne sont pas affichées sur la page HTML

– Le titre de la page : élément **title**

```
<title>Programme NSI - Classe de première</title>
```

– L'encodage utilisé : élément **meta**, attribut *charset*

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

– Les liens avec des fichiers extérieurs (CSS, Javascript...)

Le contenu

Ce sont les informations qui seront affichées sur la page

Le contenu est structuré à partir de deux types d'éléments

– Éléments de type «**inline**»

- ne commence pas sur une nouvelle ligne
- contient du texte et/ou d'autres éléments «inline»

– Éléments de type «**block**»

- structure le contenu en bloc
- un bloc commence sur une nouvelle ligne
- peut contenir d'autres blocs ou d'autres éléments de type «inline»

Structuration du texte

• 6 niveaux de titre possibles : éléments de type block **h1** à **h6**

```
<h1> Titre de niveau 1 </h1>
<h2> Titre de niveau 2 </h2>
<h3> Titre de niveau 3 </h3>
<h4> Titre de niveau 4 </h4>
<h5> Titre de niveau 5 </h5>
<h6> Titre de niveau 6 </h6>
```



Titre de niveau 1

Titre de niveau 2

Titre de niveau 3

Titre de niveau 4

Titre de niveau 5

Titre de niveau 6

• Les paragraphes sont délimités par les balises **<p> ... </p>** de type block

• Retour à la ligne : balise **
**

• Liste non-ordonnée (voir exemple plus haut page 3) ou ordonnée :

```
<ol>
  <li> Element 1 </li>
  <li> Element 2 </li>
  <li> Element 3 </li>
</ol>
```



1. Element 1
2. Element 2
3. Element 3

Éléments spécifiques du texte

Éléments «inline» qui permettent d'ajouter de l'information structurelle à un fragment de texte

- **em**: emphase
- **strong**: emphase plus forte
- **abbr**: indique une abréviation
- **acronym**: indique un acronyme
- **adress**: indique une adresse
- sub**: mettre le contenu en indice
- sup**: mettre le contenu en exposant

Éléments pour afficher du code

Éléments permettant de mettre en évidence les chaînes de caractères liées à de la programmation

- code**: Contient un fragment de code de programmation
- samp**: Contient un exemple de sortie de programme
- kbd**: Indique un texte à saisir par l'utilisateur
- var**: Indique une variable ou un argument dans un programme

Caractères spéciaux

Pour afficher les caractères difficilement accessibles via les touches du clavier, il faut les saisir au moyen d'un code précédé de & et suivi d'un ;

Quelques exemples :

- **: espace insécable
- «** : «
- »** : »
- ©** : ©

Lien vers un autre document

Les liens sont caractérisés par l'élément de type inline **a**

Le contenu de l'élément sera la zone cliquable

Plus d'informations sur le site web du [Rectorat](http://www.ac-toulouse.fr/)

Les attributs :

– **href** permet d'indiquer vers quelle page on souhaite être dirigé :

- une page externe en indiquant son URL
- une autre page du site en indiquant le chemin relatif pour y accéder

- `target="_blank"` permet d'ouvrir le document lié au lien dans une nouvelle fenêtre ou onglet
- `title` permet de définir un texte qui sera ouvert dans une infobulle lorsque le curseur survolera le lien

Lien vers un élément du document HTML

Pour pouvoir pointer vers un élément du document, il faut que celui-ci soit identifié

- Attribut `id` avec une valeur unique

```
<h3 id="preamble">Préambule</h3>
```

Lien vers un élément du document : `#id`

```
<a href="#preamble">Préambule</a>
```

Lien vers un élément d'un autre document :

lien_du_doc#id

Image

Privilégier les images au format JPEG pour les photos, PNG pour les graphiques GIF pour les images animées

Ajouter une image : élément vide de type inline **img**

```

```

Attributs de la balise `img`

- `src`: définit l'emplacement de l'image à afficher (*obligatoire*)
- `alt`: Donne une information textuelle sur l'image (*optionnel*)

Figure

Permet de mettre une légende à une image

```
<figure>  
    
  <figcaption>Plan de l'Université</figcaption>  
</figure>
```

Élément «sémantique» de type block **figure**

- contient l'image (*élément img*)
- suivi de sa légende (*élément figcaption*)

Déclaration d'un tableau :

Élément de type block **table**

... qui contient les différentes lignes du tableau

→ Créer une ligne : élément **tr**

... qui Contient les cellules de la ligne

→ Définir une cellule : élément **td**

```
<table border="1">
  <tr>
    <td>Produits</td>
    <td>kcal/100g</td>
  </tr>
  <tr>
    <td>Abricot</td>
    <td>43</td>
  </tr>
  <tr>
    <td>Ananas</td>
    <td>55</td>
  </tr>
  <tr>
    <td>Artichaut</td>
    <td>47</td>
  </tr>
  <tr>
    <td>Asperge</td>
    <td>18</td>
  </tr>
</table>
```



Produits	kcal/100g
Abricot	43
Ananas	55
Artichaut	47
Asperge	18

Ressources utilisées pour ce cours :



Diaporama "Langages de description des pages web" - Franck Silvestre & Mathieu Raynal
Département Informatique - Faculté des Sciences et Ingénierie
UNIVERSITE TOULOUSE III – Paul Sabatier

Diaporamas "Technologies du Web" - Laure Soulier
Laboratoire LIP6 - MLIA - Sciences Sorbonne Université

Vidéo Culture Numérique - Alain Preux - Romaric Gaudel - Marc Tommasi
Université de Lille

