

Introduction aux langages HTML & CSS

HyperText Markup Language (HTML)

- Langage de balisage standard pour les pages web.
- Les navigateurs web s'en servent pour interpréter et composer du texte.

Historique rapide du HTML (1)

- 1992 : première version.

Quelques balises de mise en forme, balises de lien hypertexte et quelques balises obsolètes aujourd'hui

- 1994 : HTML 2.0

Ajout des images et formulaires

- 1996 : HTML 3.2

- Une version très utilisée
- Comprend la majorité des balises actuelles
- Mise en forme du texte via des images
- Support des applets Java

Historique rapide du HTML (2)

- 1997 : version 4.0
 - Corrigée en 1998 et 1999
 - Support des frames et feuilles de style
- 2000 : spécification eXtended HTML (XHTML)
 - Possibilité de définir ses propres balises comme en XML
- 2008 : HTML5

World Wide Web Consortium (W3C)

- Organisation de standardisation principale du world wide web créée en 1994.
- S'occupe de standardiser le HTML pendant de nombreuses années.
- C'est lui qui décide d'aller vers le XHTML plutôt que de rester sur le HTML.

Web Hypertext Application Technology Working Group (WHATWG)

- Créé en 2004 par les vendeurs de navigateurs leaders, membres du W3C
- En 2007, le W3C et le WHATWG sortent des spécifications du web différentes.
- C'est le WHATWG qui décide d'aller vers HTML5 alors que W3C veut passer sur XHTML2

Quelle spécification suivre ?

- 2019 : accord entre le WHATWG et le W3C
- - Le W3C arrête la publication indépendante de spécifications
 - Collaboration sur une seule version de spécification HTML et DOM
 - Le WHATWG continue le maintien des standards
- La spécification WHATWG l'emporte

La spécification actuelle du WHATWG

Un fichier HTML de base

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8">
    <title>Sample page</title>
  </head>
  <body>
    <h1>Sample page</h1>
    <p>This is a <a href="demo.html">simple</a> sample.</p>
    <!-- this is a comment -->
  </body>
</html>
```

- Information sur la version HTML : balise <!DOCTYPE>
- En-tête déclarative : élément <head>
- Contenu du document : élément <body>

Les bases du HTML

- Il consiste en un arbre d'éléments et de texte
- Le Document Object Model (DOM) représente cet arbre en y affichant les propriétés, les méthodes et les événements de chaque éléments HTML (cf page suivante)
- L'arbre DOM peut être manipulé via des scripts, mais nous ne verrons pas cela dans ce cours

Une introduction rapide au HTML d'après le WHATWG

```

| DOCTYPE: html
| html lang="en"
|   head
|   | #text: ¶
|   | title
|   |   #text: Sample page
|   | #text: ¶
|   | #text: ¶
|   body
|   | #text: ¶
|   | h1
|   |   #text: Sample page
|   | #text: ¶
|   | p
|   |   #text: This is a
|   |   a href="demo.html"
|   |     #text: simple
|   |   #text: sample.
|   | #text: ¶
|   | #comment: this is a comment
|   | #text: ¶ ¶

```

Les éléments HTML

- Chaque élément a une étiquette de début :
`<body>`, `<div>`, `<title>`, ...
- Chaque élément a une étiquette de fin :
`<body>`, `<div>`, `<title>`, ...

Attention, il existe des exceptions : certaines étiquettes peuvent être omises et sont alors implicites comme `
`

- Les éléments sont imbriqués les uns dans les autres
- Les éléments peuvent avoir des attributs dans leur étiquette de début

Les attributs HTML

- Ce sont des noms et leur valeurs séparés par un '='
- La valeur de l'attribut n'a pas besoin d'être entre simple ou double quote si aucun espace blanc ASCII ou si aucun des symboles suivant n'est dedans " ' ` = < >

Élément bloc et inline

- Les éléments blocs sont des conteneurs qui s'étendent selon leur contenu
- Les éléments inline occupent uniquement une ligne
- Les éléments blocs peuvent contenir des éléments blocs et/ou inline
- Les éléments inline ne peuvent contenir que des éléments inline

Le <!Doctype>

- Toute première balise de tout fichier HTML
- Il permet au navigateur d'afficher la page correctement
- Pour l'HTML5 utiliser <!Doctype html>

Les éléments contenu dans le head

- `<meta>` : décrit les métadonnées du document HTML comme le codage des caractères, la description, les mots clés, l'auteur, ... du document
- `<title>` : titre du document
- `<link>` : permet de créer un lien entre le document et une ressource externe

Les éléments contenu dans le body

- Le body est le contenu de votre fichier
- Comporte généralement un <header> avec un menu, un <main> qui est le coeur de votre document et un <footer> contenant des informations supplémentaires

Les éléments fréquents

- `<h1>`, `<h2>`, `<h3>`, ..., `<h6>` sont des titres.
`<h1>` étant le niveau d'importance le plus haut
et `<h6>` le niveau le plus bas

`<h1>Ceci est un titre principal</h1>`

`<h2>Ceci est un sous-titre</h2>`

- `<p>` permet d'écrire un paragraphe :

`<p>Premier paragraphe</p>`

- `<a>` est un lien

`Un lien`

- `` est une image

``

Les tableaux

- `<table>` permet de créer un tableau
- `<tr>` contient une ligne du tableau
- `<th>` contient un élément d'entête du tableau
- `<td>` est une colonne

```
<table>
  <tr>
    <th>First column</th>
    <th>2nd column</th>
  </tr>
  <tr>
    <td>Something</td>
    <td>Anything</td>
  </tr>
</table>
```

| First column | 2nd column |
|--------------|------------|
| Something | Anything |

Les listes

- `` : une liste non ordonnée
- `` : une liste ordonnée
- `` : un élément de liste

```
<ul>  
  <p>ToDo list</p>  
  <li>eat</li>  
  <li>sleep</li>  
  <li>code</li>  
  <li>repeat</li>  
</ul>
```

ToDo list

- eat
- sleep
- code
- repeat

Conteneurs : élément bloc courant

- `<div>` : conteneur le plus courant. Bas niveau sémantique
- `<section>` : contient des choses ayant un liens ensemble. Bon niveau sémantique.

Utile pour des chapitres, une introduction, des nouveautés, des informations de contact, ...

- `<article>` : un article fait du sens par lui même. Haut niveau sémantique.

Un post de forum/blog, un commentaire, un carte de produit, un article de journal, ...

Autres éléments sémantiques

- `<nav>` : un groupe de liens de navigation

Quasiment toujours utilisé pour faire un menu

- `<aside>` : permet d'apporter des informations supplémentaires sur le contenu auquel il est lié
- `<figure>` : contient généralement une image et sa légende

`<figcaption>` : légende

`` : image

Valider votre HTML

- Utilisez un validator permet de vérifier que son code est cohérent

Validator conseillé par le WHATWG

A vous de jouer !

- Essayez de faire un premier site
- Faites un cv, un blog, un forum, ...
- Faites un menu
- Faites un tableau
- Utilisez des images
- Faites un header, un main et un footer

Le style

- L'attribut style permet de modifier le style de vos éléments : couleurs, couleurs de fond, taille, positionnement, ...
- L'utilisation de cet attribut est plus que déconseillé par le WHATWG. Utilisez plutôt un fichier de style type CSS

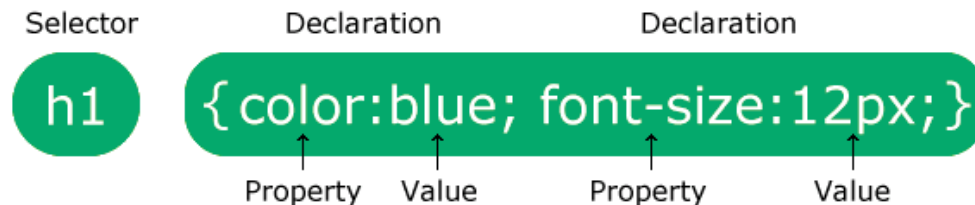
Cascading Style Sheets (css)

- Ce fichier contient les styles à appliquer sur vos éléments HTML
- Dans votre `<head>` ajoutez le lien vers votre CSS

`<link rel="stylesheet" href="index_style.css">`

- Pour agir sur le style d'un élément, il faut utiliser des sélecteurs CSS

- Syntax :



Sélecteur CSS

```
/* un commentaire CSS */

/* selecteur des éléments table */
table {
    border: 1px solid black;
}

/*selecteur multiple */
th, td {
    border: 1px solid black;
}

/*selecteur des enfants li dans les listes non ordonnees */
ul li {
    transform: 1.1;
}
```

L'attribut class

- Dans les éléments HTML, l'attribut class permet de sélectionner cette classe spécifique dans votre CSS

```
<ul>
  <p class=todo>ToDo list
  <li>eat</li>
  <li>sleep</li>
  <li>code</li>
  <li>repeat</li>
</ul>
<p>no class for this para
```

```
17
18  .todo {
19      margin: 50px;
20      color: blueviolet;
21  }
```

ToDo list

- eat
- sleep
- code
- repeat

no class for this paragraph

Quelques propriété CSS à essayer

- font-family
- font-weight
- font-style
- color
- background-color
- position
- width
- height
- margin
- padding

Le sélecteur :hover

- a:hover permet de modifier les propriétés de l'élément a quand votre curseur est sur cet élément
- Permet de faire des transitions

```
a:hover {  
    transform: scale(1.5);  
}
```

A vous de jouer !!

- Ajoutez un fichier CSS à votre fichier HTML
- Modifiez le style de votre HTML selon vos envies
- Mettez des couleurs de fond, essayez de faire un gradient
- Changez la police de caractère
- Mettez en avant les éléments importants