



HTML 1

🕒 Date de création	@3 juin 2024 11:30
📁 Cours	HTML
🕒 Type	O'clock
☑ Relue ?	<input type="checkbox"/>
☰ Mise en page ok ?	oui
☑ Prise de note ajoutée	<input checked="" type="checkbox"/>

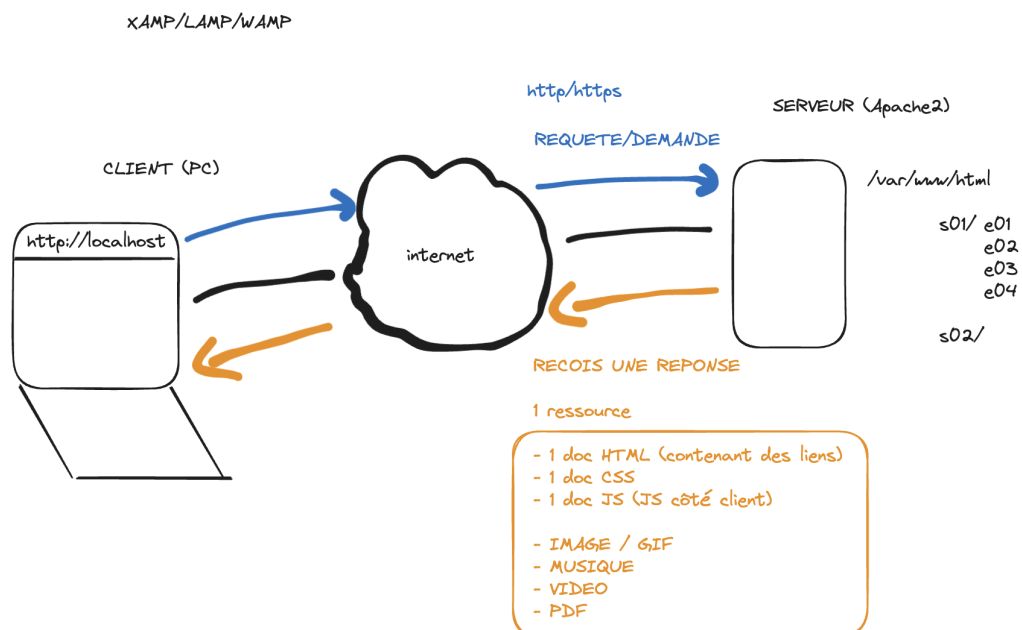
HTML

HTML signifie **HyperText Markup Language** ou Langage de balises pour l'Hypertexte

Il sert à structurer le contenu, il représente le **fond**

Le HTML s'écrit dans un fichier possédant l'extension `.html`

Quand on fait du HTML et CSS il est important de travailler avec un serveur web



- HTML est un **langage de balisage** un peu similaire au Markdown. C'est aussi un langage avec des liens hypertextes ce qui l'a rendu populaire sur la toile cela rend l'utilisation plus fluide

En **MD** ce sont des # qui sont utilisés pour la mise en page des titres alors qu'en **HTML** ce sont les balises `<h1>`, `<h2>` jusqu'à `<h6>`

Comme nous l'avons vu HTML est un langage de balisage pour l'utiliser il faut donc **ouvrir une balise puis la fermer pour pouvoir mettre du texte entre les 2 balises.**

- **HTML** est utilisé pour créer **le fond** du document et **CSS** est utilisé pour la forme

Les commentaires

Pour écrire des commentaires au sein d'un code HTML il faudra utiliser une syntaxe particulière commençant par `<!--` et finissant par `-->`

```
<!-- Je suis un commentaire -->

<!--
Un commentaire peut être écrit
sur plusieurs lignes
-->
```


Les balises

- Il existe une multitude de balises HTML (toutes dispo dans la bible MDN) juste ici :

Référence des éléments HTML - HTML (HyperText Markup Language) | MDN

Cette page répertorie tous les éléments HTML, qui sont créés à l'aide de balises. Ils sont regroupés par fonction pour vous aider à trouver facilement ce que vous avez en tête. Une liste alphabétique de tous les éléments est fournie dans la barre latérale de la

 <https://developer.mozilla.org/fr/docs/Web/HTML/Element>

 mdn web docs

Un exemple d'élément HTML

```
<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
```

Composé le plus souvent de 2 balises, ici pour notre élément de paragraphe

- balise ouvrante `<p>`
- balise fermante `</p>`

Le contenu de l'élément se trouve entre ces 2 balises

Certains éléments HTML, n'ayant **pas vocation à contenir du texte** ou d'autres éléments, s'écrivent avec **une balise dite autofermante**

```
<br> <!-- Saut de ligne -->
<hr> <!-- Ligne de séparation horizontale -->
...
```

- Il est possible d’imbriquer des valeurs les unes dans les autres
- On peut également **rajouter des attributs dans les balises** et il est possible d’en mettre autant que l’on veut.
- Si l’on doit faire 3 titres H4 et 3 paragraphes on peut faire :
 - `(h4+p)*3`

Les attributs

Les attributs viennent compléter un élément HTML avec des informations supplémentaires
Par exemple un lien, aura besoin d’un contenu mais également d’une adresse de destination

```
<a href="https://github.com">Un lien vers GitHub</a>
```

Pour un lien, l’adresse de destination sera exprimée avec l’attribut `href`

Il existe un certain nombre d’attributs universels comme

- `class` permet d’ajouter une liste de classes utilisable en CSS ou en JavaScript
- `id` permet d’indiquer un identifiant unique qui peut être ciblé par un lien, du CSS ou du JavaScript
- `title` permet d’ajouter un texte descriptif du contenu souvent représenté par une bulle flottante
- etc ...

La liste exhaustive des **attributs universels** sur Mozilla Developer Network

Un certain nombre d’éléments HTML possède des attributs spécifiques, le plus simple est de se référer aux pages spécifiques des différents **éléments HTML** existant.

Un peu d’espace que diable !

Le navigateur interprète le code HTML et livre le résultat de cette interprétation. `<p>Mon contenu</p>` dans la source HTML sera interprété puis affiché comme ceci `Mon contenu`.

Lors de son interprétation, le navigateur ignore également un certain nombre de chose comme les sauts de lignes, les tabulations ou les espaces répétés

```
<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
```

```

<p>
  Lorem ipsum
  dolor

  sit amet,

  consectetur    adipiscing

                elit.
</p>

```

Les 2 codes ci-dessus donneront le même résultat

Indentation

Sachant que les tabulations, les espaces répétés ou les sauts de lignes ne sont pas interprétés, il convient de structurer la source HTML afin de la rendre plus lisible et plus simple à maintenir.

```
<article><p>Lorem ipsum dolor <strong>sit amet</strong>, consectetur adipiscing elit.</p></article>
```

```

<article>
  <p>
    Lorem ipsum dolor <strong>sit amet</strong>,
    consectetur adipiscing elit.
  </p>
</article>

```

En comparaison, les extraits ci-dessus sont équivalents en terme de HTML par contre au niveau de la lisibilité...

La famille c'est important !

Le HTML est par nature voué à être structuré par encapsulation ou par imbrication.

```
<body> contient <article> qui contient <h1> (un titre) et <p> (un paragraphe de texte).
```

```

<body>
  <article>
    <h1>
      Lorem ipsum
    </h1>
    <p>
      Vivamus vel vestibulum mauris.
      Nam sed est a orci iaculis mattis eu non ante.
      Vivamus aliquet purus non tincidunt posuere.
    </p>

```

```
</article>
</body>
```

- `<body>` est
 - le parent de `<article>`
 - l'ancêtre de `<h1>`
 - l'ancêtre de `<p>`
- `<article>` est
 - l'enfant de `<body>`
 - le parent de `<h1>`
 - le parent de `<p>`
- `<p>` est
 - le frère de `<h1>`
 - l'enfant de `<article>`
 - le descendant de `<body>`

Types d'éléments

Les 2 types principaux sont `block` et `inline`

Par exemple

block	inline
<code><p><h1> ... <h6><main><section><article><div> ...</code>	<code><a><abbr> ...</code>

Les éléments "en bloc"

- commencent sur une nouvelle ligne
- occupent le maximum d'espace disponible (la largeur de leurs conteneurs)
- peuvent contenir des éléments `block` ou `inline`

Les éléments "en ligne"

- suivent le fil du texte
- occupent l'espace de leur contenu (la largeur de leurs contenus)
- ne peuvent contenir que des éléments `inline` ; seule exception, le lien `<a>` qui lui peut contenir des éléments de type `block`

Les autres types

D'autres types existent comme

- **list item** ``

- **table** `<table>`
- etc...

Pour aller plus loin avec les types d'éléments, la page sur les catégories de contenu du MDN est très complète

Les liens

Les liens hypertextes sont la base du web, ils permettent de naviguer au sein d'un même site ou vers des sites distants.

Plusieurs types de cibles sont donc possible

- Liens dans la même page | **les ancres**
- Liens dans d'autres pages du même site | **les chemins relatifs**
- Liens vers des sites distants | **les chemins absolus**

Quelque soit la cible, l'élément HTML reste le même, il s'agit de `<a>`; l'attribut `href` permettra de différencier les différentes destinations

Les ancres

Les ancres permettent de naviguer dans la même page grâce à l'attribut `id`

```
<header id="header-page">
  <h1>Le blog</h1>
  ...
</header>

...

<a href="#header-page">Retour en haut de la page</a>
```

URL relative

Le chemin relatif vers un fichier depuis la page courante

```
+ articles
  - hero-corp.html
  - jessica-jones.html
- index.html
```

En simulant une navigation entre les différents fichiers, la structure relative est plus claire

Depuis **index.html**

```
<nav>
  <a href="index.html">Accueil</a> <!--page courante-->
  <a href="articles/hero-corp.html">Hero Corp</a>
  <a href="articles/jessica-jones.html">Jessica Jones</a>
</nav>
```

Pour atteindre `Hero Corp` il faut d'abord aller dans `articles/`, le répertoire qui le contient

Depuis **articles/hero-corp.html**

```
<nav>
  <a href=" ../index.html">Accueil</a>
  <a href="hero-corp.html">Hero Corp</a> <!--page courante-->
  <a href="jessica-jones.html">Jessica Jones</a>
</nav>
```

Pour retourner à `Accueil` il faut d'abord remonter d'un le répertoire avec `../`

Pour aller à `Jessica Jones` il suffit de le cibler directement, il se trouve dans le même répertoire

Depuis **articles/jessica-jones.html**

```
<nav>
  <a href=" ../index.html">Accueil</a>
  <a href="hero-corp.html">Hero Corp</a>
  <a href="jessica-jones.html">Jessica Jones</a> <!--page courante-->
</nav>
```

URL absolue

Les URLs absolues sont composées de 3 parties

Exemple : `http://blog-heroique.com/articles/hero-corp.html`

- **le protocole** `http://`
- **le domaine** `blog-heroique.com`
- **le chemin du fichier** `articles/hero-corp.html`

Le document HTML

- Un raccourci très utile pour l'encodage des documents html est `"!"`
- Doctype → type du document dans certains cas cela peut être aussi xml
- L'attribut `"lang"` dans la balise html peut comporter des valeurs comme `"fr"` ou `"en"` pour définir la langue du document
- Les caractères spéciaux sont quelque chose de très "français" comme tous les accents (ex : éàù..) → il existe une convention qui est d'encoder en UTF-8 pour qu'ils soient pris en

compte.

- Head → ce sont les métadonnées (l'enveloppe de la lettre) c'est très important pour le navigateur mais ce n'est pas affiché
- Body → tout ce qui sera affiché à l'écran

La balise `<title>` → pour mettre le titre de la page (celui qui s'affiche dans l'onglet)

Pour avoir les meme settings que Yannick →

Ma configuration Visual Studio Code

Article ► <https://grafikart.fr/tutoriels/vscode-settings-2096>

Abonnez-vous ► <https://bit.ly/GrafikartSubscribe>

► https://www.youtube.com/watch?v=GFmE5_8xypY&ab_channel=Grafikart.fr



- Pour plus de mise en page comme en MD on peut utiliser les balises de titre qui vont de `<h1>` à `<h6>` . Il est très important tout comme en MD de n'avoir qu'un seul titre de niveau 1 par document HTML et ceci pour des raisons de référencement notamment.
- Pour une meilleure lisibilité du document il faut respecter des règles de sémantique : toujours les titres de niv 1 en haut puis niv 2 et ainsi de suite.
- Dans VS code pas de visualisation directement comme pour le MD, il faut toujours coder d'un côté et de l'autre utiliser chrome pour voir le résultat

Les textes

Les textes dans une page HTML représentent la majeure partie du contenu.

Dans les différents éléments servant à gérer le contenu il est possible de qualifier finement le contenu

- Une notion importante ``
- Une emphase ``
- Une citation dans une phrase `<q>`
- Une abbréviation `<abbr>`
- Une heure ou une date `<time>`
- Un exposant `<sup>`
- Un indice `<sub>`
- etc...

Paragraphe MDN

L'élément HTML de texte par excellence, `<p>` sert à délimiter un paragraphe de texte


```
<p>
  Lorem ipsum dolor sit amet, consectetur adipiscing elit. Pellentesque s
  celerisque suscipit nibh quis porttitor.
  Integer iaculis mi urna, a pulvinar quam adipiscing ut.
</p>
```

Titres MDN

Les éléments HTML de titres allant du niveau 1 au niveau 6

- `<h1>` Titre principal
- `<h2>` Sous-titre
- `<h3>` Sous sous-titre
- `<h4>` ...
- `<h5>`
- `<h6>`

Citations

Bloc de citation [MDN](#)

```
<p>William L. McKnight, PDG de 3M entre 1949 et 1966, a déclaré en 1948 :
</p>
<blockquote>
  <p>
    "As our business grows, it becomes increasingly necessary to delegate
    responsibility and to encourage men and women to exercise their initiativ
    e.

    This requires considerable tolerance. Those men and women, to whom we
    delegate authority and responsibility, if they are good people, are going
    to want to do their jobs in their own way.
  </p>
  <p>
    "Mistakes will be made. But if a person is essentially right, the mis
    takes he or she makes are not as serious in the long run as the mistakes
    management will make if it undertakes to tell those in authority exactly
    how they must do their jobs."
  </p>
  <p>
    "Management that is destructively critical when mistakes are made kil
    ls initiative. And it's essential that we have many people with initiativ
    e if we are to continue to grow."
```

```
</p>
</blockquote>
```

Citation dans un paragraphe MDN

```
<p>
  William L. McKnight encourage 3M à <q>"delegate responsibility and encourage
</p>
```

Listes

3 types de Listes

- Liste non-ordonnée ``
- Liste ordonnée ``
- Liste de définition `<dl>`

Les listes non-ordonnées `` et ordonnées `` ne peuvent avoir comme seuls enfants que des `` items de liste . Les `` contiendront les autres éléments.

Listes sans ordre

- item 1
- item 2
- item 3

```
<ul>
  <li>item 1</li>
  <li>item 2</li>
  <li>item 3</li>
</ul>
```

Liste ordonnée

1. item 1
2. item 2
3. item 3

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

Liste de définition

Liste d'items comportant un terme et une définition de ce terme, idéal pour les glossaires

```
<dl>
  <dt>Terme à définir</dt>
  <dd>Définition</dd>
  <dt>HTML</dt>
  <dd>Langage pour concevoir des pages web</dd>
</dl>
```