

HTML

HTML , CSS



HTML

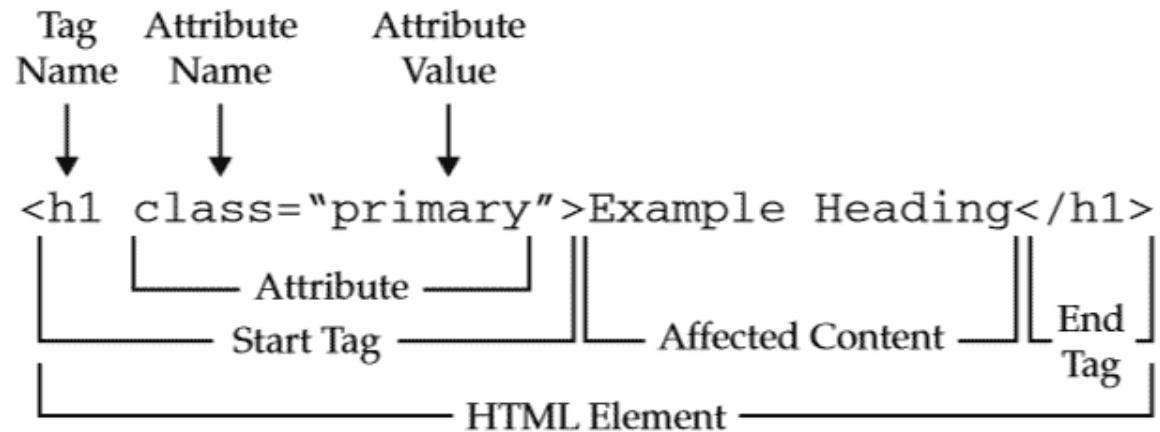
- 📌 Ce n'est pas un langage de programmation.
- 📌 C'est plutôt un langage de description.
 - 📌 Compose de plusieurs balises. (tags)
 - 📌 Interprété par le navigateur
- 📌 Il ne nécessite pas un éditeur de texte particulier.
- 📌 Il n'est pas sensible à la casse.



- 📌 HTML1 : Première version créée par Tim Berners-Lee en 1991.
- 📌 HTML2 : deuxième version, apparue en 1994. On commence à parler de W3C (World Wide Web Consortium, organisme de Standardisation fondé par Tim Berners-Lee).
- 📌 HTML3 : Apparue en 1996 avec plusieurs nouveautés comme les tableaux, les scripts, le positionnement du texte autour des images, etc.
- 📌 HTML4 : apparue en 1998 avec la possibilité : d'utiliser de frames (découpage d'une page en plusieurs parties), des améliorations sur les formulaires, d'utiliser des feuilles de style (CSS).
- 📌 HTML5 : finalisée en octobre 2014 et permet : d'inclure facilement des vidéos, d'ajouter plus de précisions sur les champs d'un formulaire.

Les Éléments HTML se divisent en trois parties distinctes:

HTML Tags



Les balises :

Syntaxe :

```
<balise> objet </balise>
```

Les balises orphelines :

```
[objet]<balise />
```

Les commentaires :




```
<!-- ceci est un commentaire -->
```

Les propriétés des balises :

- 📌 Les balises doivent être ouvertes puis fermées récursivement, comme des parenthèses(`[...]{(...)}`)
- 📌 Tout ce qui est entouré par deux balises est un contenu de balise.
- 📌 Un élément sur une page Web peut être considéré comme une boîte pouvant contenir du texte, des images, des liens, etc.
- 📌 La fermeture et l'ouverture de deux balises ne peuvent pas se croiser. (`[...]...]`)

- 📌 Lorsqu'un navigateur ne reconnaît pas une balise HTML, le texte s'affiche simplement. Cela peut se produire lorsque du texte brut (c'est-à-dire en dehors de tout élément) est rencontré.
- 📌 Les balises non reconnues ne sont toutefois pas affichées.

L'attribut d'une balise:

-  Contrôler le comportement d'un élément html
-  Grouper certains éléments entre eux
-  Modifier l'élément

Chaque attribut attend une valeur, elle se présente comme ceci :

```
<tag attribute="value">...</tag>
```

exemple d'attribut, "style":

```
<tag style="property:value;">...</tag>  
<p style="color:blue;">Ce paragraphe sera affiche en bleu.</p>
```

Ce paragraphe sera affiche en bleu.



Le contenu de la balise sera afficher par notre page internet.

Il sera modifier par les attributs et dépendra de la balise.

exemple de balises :

`<p>` pour paragraphe

`` pour du texte en gras

etc...



L'HTML sert uniquement a la structure d'une page internet.

Il sert a bâtir le “squelette” de notre site internet.

Nous utiliserons ensuite le CSS pour rajouter du style a notre HTML.

Structurer une page HTML:

Une page HTML sera toujours contenu entre deux balises html

Nous construirons nos pages en deux parties :

📌 l'entête 'head'

📌 le corps 'body'

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <title></title>
  </head>
  <body></body>
</html>
```

Le DOCTYPE:

- ✚ n'est pas une balise
- ✚ est facultatif
- ✚ c'est une directive permettant de préciser qu'il s'agit d'un document HTML et indiquant sa version

Nous l'utiliserons de cette manière en html 5

```
<!DOCTYPE html>
```


La Head:

S'utilise de cette manière :

```
<head>  
  ...  
</head>
```

Nous placerons dans la Head des informations nécessaires

- 📌 `<title>` titre du document (affiche par le navigateur en haut de la page)
- 📌 `<link/>` pour référencer un fichier (CSS par exemple)
- 📌 `<style>` pour inclure du code CSS
- 📌 `<meta>` peut contenir :
 - 📌 des informations sur le codage
 - 📌 des informations pour les moteurs de recherche

Le body:

- 📌 Le body contient les informations qui vont s'afficher dans le navigateur
- 📌 nous mettrons toutes les balises à afficher à l'intérieur de celles-ci

il s'utilisera de cette manière:

```
<body>  
  <p>Je suis afficher dans le navigateur</p>  
</body>
```

Cet exemple nous propose un body avec une balise de texte vu précédemment, mais il en existe d'autres :

Les différents types de balises:

Nous allons voir les différents type de balises, chacune d'entre elle a une utilité propre et représente un élément différent.

Tous les exemples qui vous seront présenté seront fait dans ce fichier html comprenant cette structure :



```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <title></title>
  </head>
  <body>
    <!-- Nous ecrivons notre code ici -->
  </body>
</html>
```

Les balises titres:

Ces balises changent la taille des titres allant du plus petit au plus grand :

<h1> <h2> <h3> <h4> <h5>

```
<h1>Premier titre</h1>  
<h2>Second titre</h2>  
<h3>Quatrième titre</h3>  
<h4>Quatrième titre</h4>  
<h5>Cinquième titre</h5>
```

Premier titre

Second titre

Quatrième titre

Quatrième titre

Cinquième titre

Les balises de texte:

Ces balises serviront à remplir de texte les paragraphes qui suivront nos jolis titres !

`<p>` P écrits du texte `</p>`

retourne a la ligne `
`

`` Notre texte mais... en gras ``

`<u>` Notre texte souligné `</u>`

`<i>` Notre texte en italique `</i>`

`<nobr>` Est un p qui ne reviens pas a la ligne `</nobr>`

`<hr>` Une ligne horizontale

`<pre>` Un texte préformaté `</pre>`

```
<hr>
<p> P écrits du texte </p>
<p>
    <hr>
    retourne<br>a la ligne<br>
    <hr>
    <b>
        Notre texte mais... en gras<br>
    </b>
    <hr>
    <u>
        Notre texte souligné<br>
    </u>
    <hr>
    <i>
        Notre texte en italic<br>
    </i>
    <hr>
    <nobr>
        Nobr est la négation du br il ne reviens pas a la ligne:
    </nobr>
    <hr>
</p>
<pre>
    Un texte pré-formaté
</pre>
<hr>
</p>
```

P écrits du texte

retourne
à la ligne

Notre texte mais... en gras

Notre texte souligné

Notre texte en italic

Nobr est la négation du br il ne reviens pas à la ligne:

Un texte pré-formaté

La mise en forme:

```
<p>  
    La mise en forme permet de mettre en valeur différemment,  
    <strong>les choses importantes,</strong>  
    <em> et celles-qui le sont moins.</em>  
    <mark>Demarquer vous!</em>  
</p>
```

La mise en forme permet de mettre en valeur différemment, **les choses importantes**, *et celles-qui le sont moins*. **Demarquer vous!**

Règles générales :

- 📌 Les navigateurs ajoutent automatiquement un espace entre chaque paragraphe.
- 📌 Bien que le texte 'flottant' (non inclus dans un élément de la page) soit affiché par les navigateurs, il vaut mieux pour des raisons de 'style' inclure la totalité du texte utile de la page dans des balises, notamment.

Les listes:

Nous utiliserons `` pour créer une liste non ordonnées, `` pour une liste ordonnée.

Puis `` pour chaque élément de cette liste.

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

Les images:

Nous utiliserons `` pour insérer une image dans notre site web.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>cours</title>
  </head>
  <body>
    
  </body>
</html>
```

Les audios:

Nous utiliserons `<audio>` pour insérer un audio dans notre site web.

```
<audio src="audio.mp3"></audio>
```


Les vidéos:

Nous utiliserons `<video>` pour insérer une vidéo dans notre site web.

```
<video src="fichier.mp4"></video>
```

Nous définirons notre vidéo comme ceci, pour afficher les contrôles et multiplier les sources:

```
<video controls>  
  <source src="fichier.mp4" />  
  <source src="fichier.webm" />  
  <source src="fichier.ogv" />  
</video>
```

Les liens:

Nous utiliserons `<a>`

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>cours</title>
  </head>
  <body>
    <a href="https://fr.wikipedia.org/wiki/Wikip%C3%A9dia:Accueil_principal"
      >lien</a>
  >
</body>
</html>
```

Les liens pour télécharger un fichier:

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>cours</title>
  </head>
  <body>
    <a href="data.csv">lien</a>
  </body>
</html>
```

Les formulaires:

Nous utiliserons le tag `<form>` afin de créer un formulaire.

```
<form method="GET" action="http://www.monsite.com">
```

l'attribut "method" peut valoir:

- 📌 GET : La méthode Get n'est pas fréquemment utilisée car limitée à 255 caractères. Les informations envoyée seront visible dans la zone d'adresse.
- 📌 POST : est généralement utilisée à la place de GET car elle permet d'envoyer de grandes quantités d'informations ou de données dans le formulaire et ne passe pas par la barre d'adresse.

L'attribut "action" spécifie l'adresse de la page ou du programme qui traitera les informations envoyée.

Les zone saisie :

Nous utiliserons le tag<input> pour que l'utilisateur puisse saisir des informations que nous utiliserons dans notre formulaire.

```
<input type="text" name="nom zone texte" />
<input type="email" name="email" />
<input type="number" name="tel" />
<input type="date" name="date" />
<input type="password" name="password" minlength="8" required />
```

Les zones de choix:

Les cases a cocher :

```
<input type="checkbox" name="nom du choix" [checked] />
```

Les boutons radios :

```
<input type="radio" name="nom du choix" [checked] />
```

Les listes déroulantes:







```
<select name="liste">
  <optgroup label="label1">
    <option value="valeur1">valeur1</option>
    ...
    <option value="valeur1">valeurN</option>
  </optgroup>
  <optgroup label="label2">
    <option value="valeur1">valeur1</option>
    ...
    <option value="valeur1">valeurM</option>
  </optgroup>
  ...
</select>
```


Les listes de données:

```
<input list="sports" name="sport" />
<datalist id="sports">
  <option value="football"></option>
  <option value="handball"></option>
  <option value="cross fit"></option>
  <option value="tennis"></option>
  <option value="hockey"></option>
</datalist>
```

```
<form method="post" action="traitement.php">
  <fieldset>
    <legend>Nom complet</legend>
    <!-- partie nom prenom -->
    <label for="nom">nom :</label> <input type="text" id="nom" /><br />
    <label for="prenom">prenom :</label><input type="text" id="prenom" />
  </fieldset>
  <fieldset>
    <legend>Sexe</legend>
    <input type="radio" name="homme" value="homme" />
    <label for="homme"> Homme </label><br />
    <input type="radio" name="femme" value="femme" />
    <label for="femme"> Femme </label>
  </fieldset>
</form>
```

Quelques attributs :




-  **required** : pour indiquer qu'un champ est obligatoire
-  **autofocus** : pour placer le curseur dessus des chargements de la page
-  **autocomplete** : pour indiquer si on autorise l'auto-complétions
-  **multiple** : pour indiquer qu'il est possible de choisir ou d'insérer plusieurs éléments
-  **placeholder** : pour afficher un message indicatif dans un champ
-  **readonly** : pour rendre le champ en lecture seule

Les boutons:

Nous utiliserons le type “button” pour créer un bouton en html a partir d’une balise input.



```
<input type="button" value="Bouton cliquer" />
```

Différents types de boutons:

-  **submit** : envoie le contenu d'un formulaire à la page indique dans l'action.
-  **reset** : remets à zéro (efface) le contenu d'un formulaire.
-  **button** : bouton générique qui permet de réaliser plusieurs tâche pour HTML (quitter une page, ...) ou de déclencher un code JavaScript

Les différentes balises:


Il existe deux types de balises:

-  **block** : un type de balise qui ajoute automatiquement un saut de ligne avant et après.
-  **inline** : un type de balise qui ne reviens pas a la ligne, il sera situe obligatoirement dans une balise block.






Les balises universelles:

Ce sont des balises sans signification, souvent utilisées dans les CSS ou sur les formulaires Web.

 `...` : inline

 `<div>...</div>` : block

Les balises de structuration de page html :

-  `<header>` : l'entête de ma page
-  `<footer>` : le pied de ma page
-  `<nav>` : l'emplacement du menu
-  `<section>` : le corps de la page peut être composé de plusieurs sections
-  `<article>` : dans une section on peut définir un article