

---

# HTML

---



# Une page Web, c'est quoi ?

Un fichier HTML est un format de fichier « texte » qui s'enregistre en .html

Il peut contenir des images, du texte, des liens, des médias, des polices d'écritures...

Il peut renvoyer à d'autres pages html via des liens.

# Un site Web, c'est quoi ?

Plusieurs pages web liées entre elles.

Il peut être accessible depuis n'importe quel endroit via une connection internet.

# HTML : définition

Hypertext Markup Language (HTML)

Langage de balises qui permet de structurer des pages

Différentes versions depuis 1989 - Aujourd'hui :

-XHTML (2000 – 2006)

-HTML5

# Editeurs de texte:

Il existe plusieurs éditeurs de texte spécialisés:

- Atom (gratuit et open source)

- Visual Studio Code (gratuit et open source)

- Sublime Text (version démo)

- Dreamweaver (payant)

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8" />
5     <title>Titre</title>
6   </head>
7
8   <body>
9
10  </body>
11 </html>
```

# Structure d'un document HTML

- L'information dans un document HTML est délimitée par des **balises** (écrites en minuscules).
- Toute balise `<quelquechose>` doit être fermée par une balise `</quelquechose>`.
- La paire de balises `<html> </html>` encadre le document.
- Le document est composé de deux parties :
- l'entête délimité par la paire de balises `<head> </head>`
- le corps délimité par la paire de balises `<body> </body>`
- Les balises peuvent être imbriquées : une paire de balise qui s'ouvre dans une autre paire de balises se referme dans cette paire de balises. Les balises intérieures se referment avant.
- On insère un commentaire (ignoré par le navigateur) à l'intérieur de : `<!-- le commentaire ici -->`

# Balises génériques

Les balises génériques sont des balises qui n'ont pas de sens sémantique.

En effet, toutes les autres balises XHTML ont un sens : `<p>` signifie "Paragraphe", `<h2>` signifie "Sous-titre" etc.

Parfois, on a besoin d'utiliser des balises génériques (aussi appelées balises universelles) car aucune des autres balises ne convient. On utilise le plus souvent des balises génériques pour construire son design.

Il y a 2 balises génériques : l'une est inline, l'autre est block.

- La balise `<span>` est la balise générique de type inline, on l'utilise notamment pour mettre en évidence une portion de texte dans un paragraphe ou dans un lien
- La balise `<div>` est la balise générique de type block, on l'utilise notamment pour diviser la page en plusieurs zones (bannière, menu, pied de page, colonne latérale,...)

# Balises génériques

Ces balises ont un intérêt uniquement si on leur donne un attribut `class`, `id` ou `style` :

- **class** : indique le nom de la classe CSS à utiliser.
- **id** : donne un nom à la balise. Ce nom doit être unique sur toute la page car il permet d'identifier la balise. On peut se servir de l'ID pour de nombreuses actions, comme par exemple : pour un lien vers une ancre, pour un style CSS de type ID, pour des manipulations en Javascript, etc.
- **style** : cette balise permet d'indiquer directement le code CSS à appliquer. On n'est donc pas obligé d'avoir une feuille de style à part puisqu'on peut juste mettre directement les attributs CSS. MAIS, il est préférable de ne pas utiliser cet attribut et de passer à la place par une feuille de style externe car cela rend le site plus facile à mettre à jour par la suite.



# Les balises HTML

# Balises de premier niveau

`<html>`

Balise principale

`<head>`

En-tête de la page

`<body>`

Corps de la page

# Balises d'en-tête

`<link />`

Liaison avec une feuille de style

`<meta />`

Métadonnées de la page web (charset, mots-clés, etc.)

`<script>`

Code JavaScript

`<style>`

Code CSS

`<title>`

Titre de la page

# Balises de structuration du texte

<abbr>

Abréviation

<blockquote>

Citation (longue)

<cite>

Citation du titre d'une œuvre ou d'un événement

<q>

Citation (courte)

<sup>

Exposant

<sub>

Indice

# Balises de structuration du texte

`<strong>`

Mise en valeur forte

`<em>`

Mise en valeur normale

`<mark>`

Mise en valeur visuelle

`<h1>`

Titre de niveau 1

`<h2>`

Titre de niveau 2

`<h3>`

Titre de niveau 3

(...)

# Balises de structuration du texte

`<img />`

Image

`<audio>`

Son

`<video>`

Vidéo

`<source>`

Format source pour les balises `<audio>` et `<video>`

# Balises de structuration du texte

<code>&lt;a&gt;</code>	Lien hypertexte
<code>&lt;br /&gt;</code>	Retour à la ligne
<code>&lt;p&gt;</code>	Paragraphe
<code>&lt;hr /&gt;</code>	Ligne de séparation horizontale
<code>&lt;address&gt;</code>	Adresse de contact
<code>&lt;del&gt;</code>	Texte supprimé
<code>&lt;ins&gt;</code>	Texte inséré

# Balises de listes

- |                         |                              |
|-------------------------|------------------------------|
| <code>&lt;ul&gt;</code> | Liste à puces, non numérotée |
| <code>&lt;ol&gt;</code> | Liste numérotée              |
| <code>&lt;li&gt;</code> | Élément de la liste à puces  |



# Balises génériques

`<span>`

Balise générique de type inline

`<div>`

Balise générique de type block

Ces balises ont un intérêt uniquement si vous leur associez un attribut "class", "id" ou "style":

- **class:** indique le nom de la classe CSS à utiliser.
- **id:** donne un nom à la balise. Ce nom doit être unique sur toute la page car il permet d'identifier la balise. Vous pouvez vous servir de l'ID pour de nombreuses choses, par exemple pour créer un lien vers une ancre, pour un style CSS de type ID, pour des manipulations en JavaScript, etc.
- **style:** cet attribut vous permet d'indiquer directement le code CSS à appliquer. Vous n'êtes donc pas obligés d'avoir une feuille de style à part, vous pouvez mettre directement les attributs CSS. Notez qu'il est préférable de ne pas utiliser cet attribut et de passer à la place par une feuille de style externe, car cela rend votre site plus facile à mettre à jour par la suite.

Ces trois attributs ne sont pas réservés aux balises génériques : vous pouvez aussi les utiliser sans aucun problème dans la plupart des autres balises.

- 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" />`).
- Une page web est constituée de deux sections principales : un en-tête (`<head>`) et un corps (`<body>`).
- On peut afficher le code source de n'importe quelle page web en faisant un clic droit puis en sélectionnant "Afficher le code source de la page".

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

  <body>
    <p>Hello world!</p>
  </body>
</html>
```

<h1>Titre super important</h1>

<h2>Titre important</h2>

<h3>Titre un peu moins important (sous-titre)</h3>

<h4>Titre pas trop important</h4>

<h5>Titre pas important</h5>

<h6>Titre vraiment pas important du tout</h6>

## Les liens:

- Lorsque deux fichiers sont dans le même répertoire, il suffit d'appeler le nom du fichier comme cible.
- Ce sont des liens relatifs, car relatifs au site et dossier dans lequel ils se trouvent.

```
<a href="page1.html" title="ma 1ere page"> ma page</a>
```

```
<a href="index.html" title="accueil du site">l'accueil</a>
```

# Exercice time!

- Créez un dossier « mon\_site »
- Créez y index.html
- Placez-y un second fichier appelé page1.html
- Ouvrez index.html, ajoutez lui un titre h1 et ajoutez un lien vers la page1.html
- Ensuite ouvrez page1.html, ajoutez lui un titre h1 et ajoutez un lien de retour à l'index

```
<h1> Mon index ! </h1>  
<p> Lien vers <a href="page1.html" title="ma 1er page">ma page</a></p>
```

# Mon index !

Lien vers [ma page](#)

```
<h1> Ma page 1 </h1>  
<p> Retour à <a href="index.html" title="accueil du site">l'accueil</a></p>
```

# Ma page 1

Retour à [l'accueil](#)