



Le HTML Sémantique

De la Structure à la Signification

Au-delà des Balises

Au début, on apprend à voir le HTML
comme ça :

Un Titre

Un paragraphe de texte...

On utilise des balises pour **créer une apparence**.

Ce cours va vous apprendre à le voir
comme ça :

`<h1>`

`<p>`

On utilise des balises pour **décrire un contenu**.

Notre objectif : communiquer le **sens** de notre contenu au navigateur, aux moteurs de recherche et aux technologies d'assistance.

Objectifs du Cours

À la fin de ce module, vous serez capable de :

- Comprendre **pourquoi** la sémantique est cruciale (SEO, accessibilité...).
- Utiliser les **balises structurelles** pour bâtir des pages web robustes (`<header>` , `<main>` , `<nav>` ...).
- Choisir la **bonne balise** pour le bon contenu (texte, citations, images...).
- Appliquer les **bonnes pratiques** pour un code propre, lisible et maintenable.
- **Analyser** une page web existante pour en identifier la structure sémantique.

Le W3C : Gardien du Web

- En 1994, Tim Berners-Lee fonde le **World Wide Web Consortium (W3C)**.
- **Mission** : Standardiser les technologies du Web pour garantir sa croissance et son interopérabilité.
- Le W3C publie les spécifications officielles de HTML, CSS, etc.
- Les navigateurs s'engagent (plus ou moins vite) à suivre ces règles.

L'Arrivée de XHTML

HTML

+

XML

↓

XHTML

Qu'est-ce que XML ?

eXtensible Markup Language

Le but de XML n'est pas d'afficher des données, mais de les **décrire** et de les **structurer**.

Exemple HTML :

```
<h1>Mon Livre Préféré</h1>
<p>
  <strong>Auteur :</strong> Frank Herbert
</p>
```

Les balises décrivent la **présentation** (un titre, un paragraphe, du texte en gras).

Exemple XML :

```
<livre>
  <titre>Dune</titre>
  <auteur>Frank Herbert</auteur>
</livre>
```

Les balises décrivent la **donnée** elle-même (un livre, un titre, un auteur).

Séparation : Contenu vs. Forme

La grande idée de HTML 4

Une règle d'or émerge : **HTML pour le contenu, CSS pour le style.**

À ne plus faire !

```
<!-- HTML qui mélange contenu et style -->
<body background="fond.jpg">
  <center>
    <font color="blue" size="5">
      Titre de la page
    </font>
  </center>
</body>
```

La bonne pratique

```
<!-- HTML pour le contenu -->
<body>
  <h1>Titre de la page</h1>
</body>

/* CSS pour le style */
body {
  background-image: url('fond.jpg');
}
h1 {
  text-align: center;
  color: blue;
  font-size: 2em;
}
```


HTML vs. XHTML : Le Duel

HTML (Flexible)

Insensible à la casse

`<BODY>` est valide

Attributs sans valeur

`<input disabled>`

Fermeture optionnelle

`<p>Un paragraphe`

Balises vides

`<hr>` ou `
`

XHTML (Strict)

Sensible à la casse

`<body>` uniquement

Valeur obligatoire

`<input disabled="disabled">`

Fermeture obligatoire

`<p>Un paragraphe</p>`

Balises vides auto-fermées

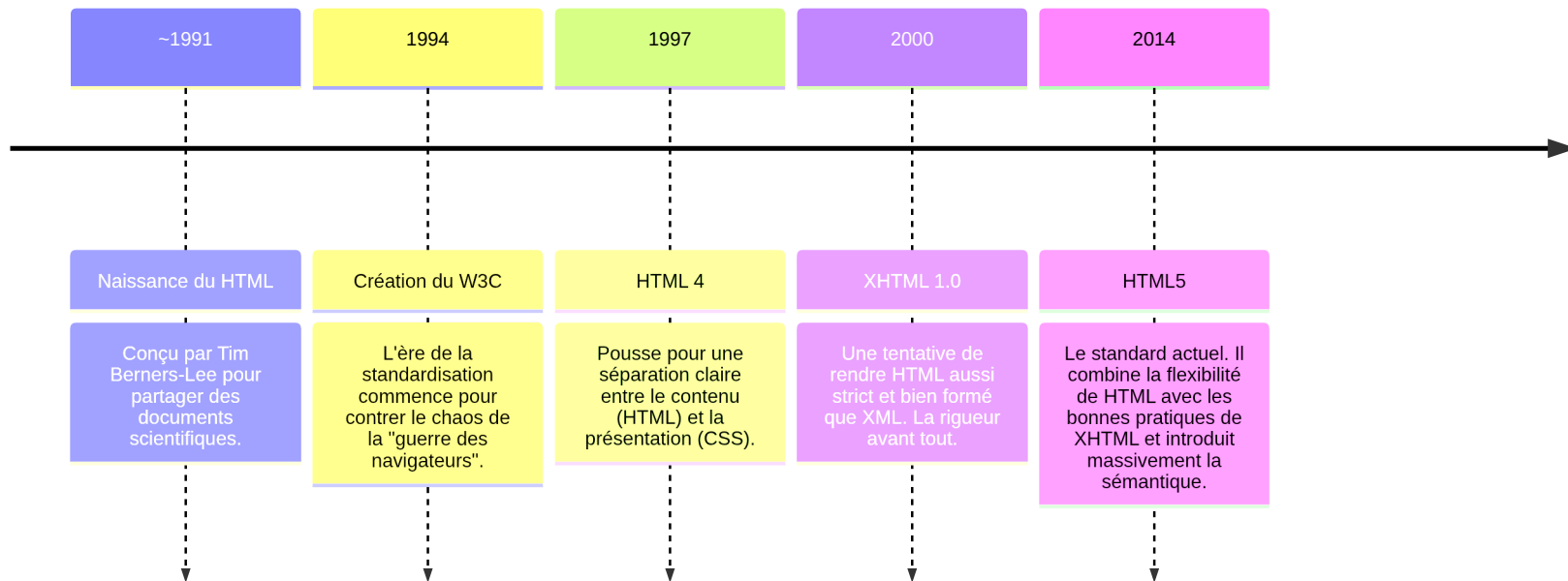
`<hr />` ou `
`

Et Aujourd'hui ? La Leçon Apprise

- XHTML était très (trop ?) strict pour le Web.
- Mais l'idée de **rigueur** et de **structure claire** a fait son chemin.
- **HTML5** est né : il combine la flexibilité de HTML avec les bonnes idées de XHTML.
- Il a surtout introduit un concept fondamental qui est le cœur de notre cours...
- **Le Web Sémantique**

Une Chronologie Visuelle

Évolution des Standards du Web



Qu'est-ce que le HTML Sémantique ?

Donner du sens à la structure

Qu'est-ce que la "Sémantique" ?

- En linguistique, c'est l'étude du **sens des mots**.
- Appliqué au HTML, cela signifie utiliser des balises qui **décrivent leur contenu**, et non leur apparence.
- Une balise sémantique répond à la question : "**Quel est ce contenu ?**"
- Une balise non-sémantique répond à la question : "**À quoi doit ressembler ce contenu ?**" (ce qui est le rôle du CSS !)

L'analogie du journal papier :

Dans un article de journal, vous identifiez instinctivement "le titre principal", "l'auteur", "l'introduction", "une image avec sa légende". Chaque élément a un **rôle**, un **sens**. Le HTML sémantique vise à faire la même chose pour le navigateur

Le Duel : Non-Sémantique vs. Sémantique

L'approche Non-Sémantique

Des boîtes génériques sans signification.

```
<div id="header">
  <div class="logo">Mon Site</div>
  <div class="nav">
    <span>Accueil</span>
    <span>Contact</span>
  </div>
</div>

<div class="main">
  <div class="article">
    Le contenu...
  </div>
</div>
```

Pour une machine, c'est juste un ensemble de `div` .
Le sens ne vient que des `id` ou `class` , qui ne sont pas standardisés.

L'approche Sémantique

Des balises qui décrivent leur rôle.

```
<header>
  <h1>Mon Site</h1>
  <nav>
    <a href="#">Accueil</a>
    <a href="#">Contact</a>
  </nav>
</header>

<main>
  <article>
    Le contenu...
  </article>
</main>
```

Chaque balise a un sens universel. `header` , `nav` ,
`main` , `article` sont compris par les navigateurs,
les moteurs de recherche, etc.

Un Exemple Concret : Un Titre

Visuellement, ces deux lignes peuvent être **identiques** grâce au CSS :

Ceci est un titre
Ceci est un titre

Mais pour une machine, leur **sens** est radicalement différent :

```
<!-- Ceci est juste un texte.  
Aucune importance particulière. -->  
<div style="font-size: 2em; font-weight: bold;">  
  Ceci est un titre  
</div>  
<!-- CECI EST LE TITRE PRINCIPAL DE LA PAGE.  
TRÈS IMPORTANT. -->  
<h1>Ceci est un titre</h1>
```

👉 Utiliser `h1` informe tout le monde (développeurs, Google, lecteurs d'écran) que ce contenu est la chose la plus importante de la page.

Pourquoi est-ce si important ?

Les 4 piliers du HTML Sémantique

1. SEO (Référencement)

Les moteurs de recherche (Google, Bing...) comprennent mieux la structure de votre page. Un contenu dans une balise `<h1>` ou `` a plus de poids.

2. Accessibilité (a11y)

Les lecteurs d'écran pour les personnes malvoyantes utilisent la sémantique pour naviguer. Ils peuvent lister les titres (`<h1>` , `<h2>` ...) ou sauter directement à la navigation (`<nav>`).

3. Maintenabilité

Un code sémantique est plus facile à lire et à comprendre pour les autres développeurs (et pour vous-même dans 6 mois !). Il est auto-documenté.

4. Clarté et Simplicité

Le code est plus propre et plus léger. On évite la "div-itis", cette maladie qui consiste à imbriquer des `<div>` à l'infini.

Les Balises Sémantiques Structurelles

Bâtir le squelette de la page

La Structure d'une Page Sémantique

`<header>` : En-tête du site (logo, titre...)

`<nav>` : Navigation principale

`<main>` : Contenu principal et unique de la page

`<aside>`
Contenu
secondaire

`<footer>` : Pied de page (copyright, liens...)

<header>

L'en-tête de la page

La balise `<header>` représente un groupe de contenus introductifs ou d'aides à la navigation.

Il peut contenir un titre, un logo, un formulaire de recherche, le nom de l'auteur, etc.

Important : Une page peut avoir plusieurs

`<header>` (par exemple, un pour la page et un pour chaque `<article>`).

```
<header>
  
  <h1>Le Titre de mon Site Web</h1>
  <p>Mon slogan accrocheur</p>
</header>
```

<nav>

La navigation principale

La balise `<nav>` est destinée aux blocs de navigation majeurs.

Elle est idéale pour le menu principal du site, une table des matières, ou un fil d'Ariane.

Les lecteurs d'écran peuvent l'utiliser pour permettre à l'utilisateur de "sauter la navigation" et d'aller directement au contenu.

```
<nav>
  <ul>
    <li><a href="/">Accueil</a></li>
    <li><a href="/produits">Nos produits</a></li>
    <li><a href="/contact">Contact</a></li>
  </ul>
</nav>
```

`<main>`

Le contenu unique de la page

La balise `<main>` doit contenir le contenu **dominant** et **unique** du document.

Tout ce qui se répète sur plusieurs pages (menus, logos, footers) doit être **en dehors** de `<main>`.

Règle d'or : Il ne doit y avoir qu'une seule balise `<main>` visible par page.

```
<main>
  <h2>Titre de mon article</h2>
  <p>
    Ici se trouve le paragraphe principal de ma page,
    le contenu pour lequel l'utilisateur est venu.
  </p>
</main>
```

`<footer>`

Le pied de page

La balise `<footer>` représente le pied de page de son ancêtre sectionnel le plus proche (ou de la page entière).

On y trouve typiquement les informations de copyright, les liens vers les mentions légales, un plan du site, ou des informations de contact.

```
<footer>
  <p>© 2025 Mon Entreprise Inc.</p>
  <nav>
    <a href="/mentions-legales">Mentions Légales</a>
    <a href="/confidentialite">Politique de confidentiali
  </nav>
</footer>
```

Les Blocs Principaux

■ `<header>`

L'en-tête de la page ou d'une section. Contient souvent le logo, le titre principal, un champ de recherche.

■ `<nav>`

Définit un bloc de liens de navigation principaux (menu, fil d'Ariane...).

■ `<main>`

Le contenu **unique et central** de la page. Il ne doit y en avoir qu'un seul par page !

■ `<footer>`

Le pied de page. Contient des informations comme le copyright, les mentions légales, un plan du site...

```
<body>
  <header>
    <h1>Mon Super Blog</h1>
  </header>
  <nav>
    <ul>
      <li><a href="/">Accueil</a></li>
      <li><a href="/a-propos">À Propos</a></li>
    </ul>
  </nav>

  <main>
    <h1>Titre de la page</h1>
    <p>Mon Super Contenu</p>
  </main>

  <footer>
    <p>© 2025 - Mon Super Blog</p>
  </footer>
</body>
```



Mon Super Blog

- Accueil
- À Propos

Titre de mon article

Mon Super Contenu

© 2025 - Mon Super Blog

<article>

Un contenu autonome

La balise `<article>` encapsule une composition autonome dans un document, une page, ou un site.

L'idée est que son contenu pourrait être redistribué indépendamment et garderait tout son sens.

Exemples : un article de blog, un post de forum, un produit e-commerce, un commentaire.

```
<article>
  <h3>Mon Premier Article de Blog</h3>
  <p>
    Ceci est le début de mon article.
    Il a du sens tout seul.
  </p>
  <footer>
    <p>Publié le
      <time datetime="2025-10-04">4 Octobre 2025</time>
    </p>
  </footer>
</article>
```

<section>

Un regroupement thématique

La balise `<section>` est utilisée pour regrouper des contenus qui ont un lien thématique entre eux.

Contrairement à `<article>`, son contenu n'est pas forcément autonome. Son sens dépend du contexte de la page.

Une section devrait presque toujours avoir un titre (`<h1>` - `<h6>`) pour identifier son thème.

```
<section>
  <h2>Notre Équipe</h2>
  <p>
    Découvrez les membres qui composent notre entreprise.
  </p>
  <!-- Fiches des membres de l'équipe ici -->
</section>
```

<aside>

Contenu complémentaire

La balise `<aside>` représente une portion du document dont le contenu n'est que tangentiellement lié au contenu principal.

C'est souvent représenté comme une barre latérale ("sidebar").

Exemples : liens connexes, biographie de l'auteur, publicités.

```
<main>
  <p>Le contenu principal de la page...</p>
</main>
<aside>
  <h3>Articles similaires</h3>
  <ul>
    <li><a href="#">Autre article</a></li>
    <li><a href="#">Encore un autre</a></li>
  </ul>
</aside>
```

Sémantique du Contenu et du Texte

Donner du sens à chaque mot

Importance vs. Apparence

Sémantique : Donne du sens

`` : Indique une **forte importance**.
Crucial pour le SEO et l'accessibilité.

`` : Marque une *emphase*, change le sens
d'une phrase.

Visuel : Ne change que le style

`` : Met le texte en **gras** sans lui donner plus
d'importance. À éviter sauf si purement
stylistique.

`<i>` : Met en *italique*. Utile pour un terme
technique, un mot étranger, un nom de navire...

Exemple : "N'appuyez **surtout pas** sur le bouton rouge !" vs "Le *Tyrannosaurus Rex* est un dinosaure."

Légender Contenus et Images

La paire `<figure>` et `<figcaption>`

La balise `<figure>` est utilisée pour encapsuler un contenu (image, diagramme, bloc de code...) qui illustre le propos.

`<figcaption>` lui fournit une légende directement associée.

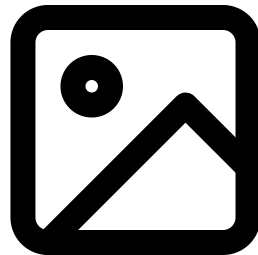


Fig. 1 - Logo illustratif de notre concept.

```
<figure>
  
  <figcaption>
    <strong>Fig. 1</strong> - Logo illustratif de notre concept.
  </figcaption>
</figure>
```

Balises de texte spécifiques

`<mark>` sert à **surligner** un texte pour le faire ressortir, comme avec un marqueur.

```
<p>Le moteur de recherche a surligné le terme <mark>Slidev</mark>.</p>
```

`<time>` représente une date/heure. L'attribut `datetime` la rend lisible par les machines.

```
<p>La conférence commence à <time datetime="2025-11-20T09:00">9h le 20 Novembre</time>.</p>
```

<address>

Informations de contact

La balise `<address>` fournit des informations de contact pour le document courant ou pour l' `<article>` parent le plus proche.

Le contenu doit inclure un nom, une adresse postale, une URL, une adresse e-mail, un numéro de téléphone, etc.

```
<address>
  Pour plus d'informations, contactez Jane Doe.<br>
  <a href="mailto:jane.doe@example.com">jane.doe@example.
  <a href="tel:+32123456789">+32 123 456 789</a>.
</address>
```


<q> et <blockquote>

Citer une source

La balise `<q>` (quote) est utilisée pour de courtes citations **en-ligne**, directement dans un paragraphe. Les navigateurs ajoutent automatiquement des guillemets.

```
<p>
  Quand il a vu le code, il a simplement dit :
  <q>C'est une excellente idée !</q>.
</p>
```

La balise `<blockquote>` est utilisée pour des citations longues qui forment un **bloc** à part entière. Les navigateurs l'affichent souvent avec un retrait.

```
<blockquote>
  <p>
    La seule façon de faire du bon travail est d'aimer
    ce que vous faites. Si vous n'avez pas encore trouvé,
    continuez à chercher. N'abandonnez pas.
  </p>
</blockquote>
```

`<cite>` : La Source de la Citation

La balise `<cite>` sert à indiquer la source d'une citation ou le titre d'une œuvre créative (un livre, un film, une chanson...).

Elle ne contient **pas** la citation elle-même, mais la référence à son origine.

On la combine très souvent avec `<blockquote>` pour fournir un contexte complet.

```
<blockquote>
  <p>
    Un ordinateur mérite que vous l'appeliez "vous" ;
    les machines ne fonctionnent jamais si vous les tutoy
  </p>
  <footer>
    – <cite>La Foire aux immortels</cite>
  </footer>
</blockquote>
```

Note : Dans l'exemple ci-dessus, le nom de l'auteur n'est pas dans la balise `<cite>` , car il n'est pas le titre de l'œuvre. C'est une bonne pratique à respecter.

Exemple :

Un ordinateur mérite que vous l'appeliez "vous" ; les machines ne fonctionnent jamais si vous les tutoyez.
— *La Foire aux immortels*

<time> : Rendre le Temps Lisible

Pour les humains et les machines

La balise `<time>` permet de baliser une date, une heure ou une durée de manière sémantique.

Son contenu est ce que l'humain lit. L'attribut `datetime` fournit une version standardisée pour la machine (navigateur, calendrier, moteur de recherche).

```
<!-- Une date simple -->
<p>
  Le cours a lieu le
  <time datetime="2025-10-28">28 Octobre</time>.
</p>

<!-- Une heure précise -->
<p>
  Rendez-vous à
  <time datetime="2025-10-28T14:30">14h30</time>.
</p>

<!-- Une durée -->
<p>
  Le film dure <time datetime="PT2H30M">2h 30m</time>.
</p>
```

Les Conteneurs Non-Sémantiques

Quand il n'y a pas d'autre choix : `<div>` et ``

`<div>` et `` : Les Couteaux Suisses

- Jusqu'ici, nous avons vu des balises qui donnent un **sens** à leur contenu : `<header>` , `<h1>` , `<article>` ...
- Mais parfois, on a juste besoin de **grouper des éléments** pour des raisons purement visuelles (style CSS) ou interactives (JavaScript).
- Aucune balise sémantique ne correspond à "une icône à côté d'un mot".
- C'est là que les conteneurs **génériques** et **non-sémantiques** interviennent.

`<div>`

Un conteneur générique de type **bloc**.

``

Un conteneur générique de type **en-ligne** (inline).

<div> : Le Conteneur de Bloc

<div> est l'élément de groupage le plus courant. Étant de type **bloc**, il force un retour à la ligne avant et après lui.

On l'utilise principalement pour :

- Créer la **mise en page** (wrapper, colonnes, grille).
- Regrouper des éléments qui forment un "composant" visuel, comme une carte de produit.
- Appliquer un style de fond ou des marges à un ensemble d'éléments.

Exemple : Créer une "carte" de profil

```
<!-- Aucune balise sémantique ne décrit une  
"carte de profil" -->  
<div class="profile-card">  
    
  <h2>Jean Dupont</h2>  
  <p>Développeur Web</p>  
</div>
```

`` : Le Conteneur en Ligne

`` est un conteneur **en-ligne**. Il ne provoque aucune rupture dans le flux du texte.

On l'utilise pour cibler une petite portion de contenu au sein d'un bloc plus grand :

- Styliser un ou plusieurs mots (changer la couleur, la police...).
- Associer une icône à un texte.

Exemple : Mettre en évidence une notification

```
<p>
  Vous avez
  <span class="notification-badge">3</span>
  nouveaux messages.
</p>
```

La Règle d'Or : Sémantique d'Abord !

Avant d'écrire `<div>` ou ``, posez-vous toujours cette question :

Existe-t-il une balise HTML sémantique qui décrit la **nature** de mon contenu ?

Ceci est une navigation ?

OUI → Utilisez `<nav>`

NON → Évitez `<div class="nav">`

Ceci est un titre important ?

OUI → Utilisez `<h2>`

NON → Évitez `<div class="titre">`

Ceci est le contenu principal ?

OUI → Utilisez `<main>`

NON → Évitez `<div id="main-content">`

C'est juste un conteneur pour du style ?

ALORS → OK, utilisez `<div>` ou ``

Bonnes Pratiques en HTML

Sémantique

Peaufiner la structure et l'accessibilité

La Hiérarchie des Titres

Le plan de votre document

- Les titres (`<h1>` , `<h2>` , ..., `<h6>`) structurent votre contenu comme les chapitres d'un livre.
- Ils ne doivent **jamais** être choisis pour leur apparence (taille, gras), mais pour leur **niveau hiérarchique**.
- **Règle n°1** : Un seul `<h1>` par page. C'est le titre principal de votre document.
- **Règle n°2** : Ne sautez jamais de niveau. On ne passe pas d'un `<h2>` à un `<h4>` sans `<h3>` entre les deux.
- Cette structure est fondamentale pour l'accessibilité (navigation au lecteur d'écran) et le SEO.

Structure des Titres : Correct vs. Incorrect

✅ La Bonne Pratique

La hiérarchie est logique et ininterrompue.

```
<h1>Le Titre Principal de la Page</h1>
<h2>Section 1</h2>
  <h3>Sous-section 1.1</h3>
  <h3>Sous-section 1.2</h3>
<h2>Section 2</h2>
  <h3>Sous-section 2.1</h3>
```

❌ Les Erreurs à Éviter

Niveaux sautés et `h1` multiples.

```
<!-- ERREUR : On saute de h1 à h3 -->
<h1>Titre Principal</h1>
  <h3>Sous-section sans section parente ?</h3>

<!-- ERREUR : Plusieurs h1 -->
<h1>Le Titre de la Page</h1>
<p>...</p>
<h1>Un autre titre majeur ? Non !</h1>
```

L'Attribut `alt` : Une Voix pour les Images

L'attribut `alt` (texte alternatif) est **obligatoire** sur les balises ``. Il a trois rôles vitaux :

Accessibilité

Les lecteurs d'écran lisent le texte de l'attribut ``alt`` aux utilisateurs malvoyants, leur décrivant ainsi le contenu de l'image.

Image Cassée

Si l'image ne peut pas se charger, le navigateur affiche le texte de l'attribut ``alt`` à la place. L'utilisateur sait ce qu'il a manqué.

SEO

Les moteurs de recherche comme Google utilisent ce texte pour comprendre le sujet de l'image et mieux référencer votre page.

Comment Rédiger un Bon Attribut alt ?

Pour une image informative :

Décrivez brièvement mais précisément ce que l'image montre. Imaginez que vous la décrivez à quelqu'un au téléphone.

```
<!-- ☒ Bien -->  
  
  
<!-- ☐ Mauvais -->  

```

Pour une image décorative :

Si l'image n'apporte aucune information et n'est là que pour le style (un motif, une icône purement visuelle), laissez l'attribut `alt` vide. Le lecteur d'écran l'ignorerait simplement.

```
<!-- ☒ Parfait pour une image de décoration -->  

```

L'Attribut `lang` : Indiquer la Langue

L'attribut `lang` est placé sur la balise `<html>` pour déclarer la langue principale du document.

Pourquoi est-ce important ?

- **Accessibilité** : Aide les lecteurs d'écran à utiliser la bonne prononciation.
- **Navigateurs** : Peut proposer une traduction si la langue du document est différente de celle du navigateur.
- **SEO** : Aide les moteurs de recherche à cibler le bon public linguistique.

```
<!DOCTYPE html>
<html lang="fr">
  <head>
    <meta charset="UTF-8">
    <title>Ma Page en Français</title>
  </head>
  <body>
    <p>Bonjour le monde !</p>
  </body>
</html>
```

Les Commentaires : Notes pour les Humains

Les commentaires sont des notes dans le code source qui ne sont pas affichées par le navigateur.

- Ils servent à expliquer une portion de code complexe, à laisser des rappels, ou à désactiver temporairement du code.
- La syntaxe est `<!-- Votre commentaire ici -->`.
- **Attention** : Les commentaires sont visibles par tous ceux qui inspectent le code source de la page. N'y mettez jamais d'informations sensibles (mots de passe, etc.).

```
<!--  
  Ceci est un commentaire sur plusieurs lignes.  
  Il explique le rôle de la section ci-dessous.  
-->  
<section class="user-profile">  
  <h2>Profil Utilisateur</h2>  
  
  <!-- TODO: Remplacer ces données par des vraies venant de l'API -->  
  <p>Nom : John Doe</p>  
  
  <!-- <p>Ancienne information, désactivée pour le moment.</p> -->  
</section>
```

L'Indentation : La Propreté du Code

L'indentation (le décalage des lignes de code) n'a **aucun impact** sur l'affichage de la page dans le navigateur. Cependant, elle est **fondamentale** pour la lisibilité et la maintenance du code.

Code non indenté (illisible)

```
<body>
<header>
<h1>Mon Site</h1>
</header>
<main>
<article>
<h2>Titre de l'article</h2>
<p>Un paragraphe.</p>
</article>
</main>
</body>
```

Code indenté (clair)

```
<body>
  <header>
    <h1>Mon Site</h1>
  </header>
  <main>
    <article>
      <h2>Titre de l'article</h2>
      <p>Un paragraphe.</p>
    </article>
  </main>
</body>
```

L'indentation montre clairement la **structure d'imbrication** et la relation parent-enfant entre les balises. C'est une convention professionnelle non négociable.

Les Règles d'Or de l'Écriture HTML

- **Toujours en minuscules** : Les balises et attributs s'écrivent en minuscules . (`<body>` , pas `<BODY>`).
- **Guillemets doubles** : Utilisez les guillemets doubles (") pour les valeurs d'attributs. (`class="exemple"`).
- **Fermeture systématique** : Toutes les balises doivent être fermées, même si c'est optionnel pour le navigateur (`...`).
- **Balises auto-fermantes** : N'utilisez pas de (/) pour les balises comme `` , `
` , `<hr>` .
- **Indentation** : Imbriquez et indentez le code pour une lisibilité parfaite.
- **Hiérarchie des titres** : Respectez l'ordre logique des titres (`<h1>` à `<h6>`) sans sauter de niveau.

La Structure Fondamentale d'un Document

Toute page HTML doit commencer par un "squelette" qui déclare les informations essentielles au navigateur.

1. `<!DOCTYPE html>` : Indique au navigateur que le document est du HTML5.
2. `<html lang="fr">` : Définit la langue du contenu pour l'accessibilité (synthèse vocale) et le SEO.
3. `<meta charset="UTF-8">` Spécifie l'encodage des caractères pour afficher correctement les accents et symboles.

```
<!DOCTYPE html>
<html lang="fr">
  <head>
    <meta charset="UTF-8">
    <title>Titre de la Page</title>
  </head>
  <body>
    <!-- Le contenu de votre page ici -->
  </body>
</html>
```

Réduire les Balises au Minimum

Éviter les imbrications inutiles

Un code plus simple est plus rapide à charger pour le navigateur et plus facile à maintenir pour le développeur.

✖ Mauvais Exemple

La balise `<div>` est complètement superflue. Le `<h1>` peut porter la classe et la sémantique tout seul.

```
<div class="titrePrincipal">  
  <h1>About Us</h1>  
</div>
```

✔ Bon Exemple

Le code est plus léger, plus direct et plus facile à lire. Moins de balises, c'est mieux !

```
<h1 class="titrePrincipal">About Us</h1>
```

L'Ordre des Attributs

Une convention pour la lisibilité

Pour le navigateur, l'ordre n'a pas d'importance. Pour les humains qui lisent le code, une convention rend tout plus simple et prévisible.

Attributs dans le désordre

```
<ul>
  <li data-auteur="J.R.R Tolkien" class="livre" id="livre1">
    Le Hobbit
  </li>
  <li id="livre2" data-auteur="F. Herbert" class="livre">
    Dune
  </li>
  <li class="livre" id="livre3" data-auteur="H.P. Lovecraft">
    L'appel de Cthulhu
  </li>
</ul>
```

Attributs ordonnés

```
<ul>
  <li class="livre" id="livre1" data-auteur="J.R.R Tolkien">
    Le Hobbit
  </li>
  <li class="livre" id="livre2" data-auteur="F. Herbert">
    Dune
  </li>
  <li class="livre" id="livre3" data-auteur="H.P. Lovecraft">
    L'appel de Cthulhu
  </li>
</ul>
```

Ordre Recommandé des Attributs

Il n'y a pas de règle absolue, mais voici une convention logique et largement adoptée.

1. `class` : Le plus commun, pour la liaison avec le CSS.
2. `id` , `name` : Les identifiants uniques.
3. `data-*` : Les attributs de données personnalisées.
4. `src` , `href` , `for` , `type` , `value` : Attributs de contenu ou de comportement.
5. `title` , `alt` : Attributs de contenu textuel pour l'utilisateur.
6. `role` , `aria-*` : Attributs d'accessibilité.

Adopter une convention rend le code plus prévisible et plus rapide à analyser.

Exercice Pratique

L'Objectif de l'Exercice

- Nous avons couvert la théorie des balises sémantiques.
- Il est temps de voir comment cela s'applique dans un cas concret.
- Le but est d'entraîner votre œil à **reconnaître les structures et les rôles** des différents blocs d'une page web.
- Votre mission : associer mentalement la bonne balise sémantique à chaque section de la page suivante.

Mon Carnet de Voyage

Récits et photographies de mes aventures autour du monde.

[Accueil](#)[Destinations](#)[À Propos](#)[Contact](#)

Voyage en Islande : Terre de Glace et de Feu

Publié par Alex le 15 mars 2025



La fameuse cascade Seljalandsfoss où l'on peut marcher derrière.

Le Cercle d'Or, un classique incontournable

Notre première étape fut le fameux Cercle d'Or. C'est un **incontournable** pour tout premier voyage en Islande. Nous avons été émerveillés par le geyser Strokkur, qui jaillit avec une régularité impressionnante toutes les quelques minutes.

La magie des aurores boréales

Le clou du spectacle fut sans conteste la chasse aux aurores boréales. Loin de la pollution lumineuse des villes, le ciel nocturne s'est transformé en une toile vivante, peinte de rubans verts et violets.

C'est un spectacle que des mots ne peuvent décrire, une expérience humble face à la puissance de la nature.

— Citation tirée de mon *Journal de bord personnel*

Catégories : [Europe](#), [Nature](#)

À propos de l'auteur

Alex, photographe et voyageur passionné, explore le monde depuis plus de 10 ans pour partager ses découvertes.

Articles récents

- [Le Japon entre tradition et modernité](#)
- [Sur les traces des Incas au Pérou](#)

Comment Procéder ?

1. **Observez la page dans son ensemble.** Ne vous perdez pas dans les détails tout de suite.
2. **Identifiez les grandes régions :** Où se trouve l'en-tête principal ? La navigation ? Le contenu central ? La colonne latérale ? Le pied de page global ?

Ouvrez draw.io et notez la balise qui vous semble la plus appropriée pour chaque bloc. Nous corrigerons ensuite ensemble.

Voyage en Islande : Terre de Glace et de Feu

Publié par Alex le 15 mars 2025



La fameuse cascade Seljalandsfoss où l'on peut marcher derrière.

Main

Le Cercle d'Or, un classique incontournable

Notre première étape fut le fameux Cercle d'Or. C'est un **incontournable** pour tout premier voyage en Islande. Nous avons été émerveillés par le geyser Strokkur, qui jaillit avec une régularité impressionnante toutes les quelques minutes.

La magie des aurores boréales

Le clou du spectacle fut sans conteste la chasse aux aurores boréales. Loin de la pollution lumineuse des villes, le ciel nocturne s'est transformé en une toile vivante, peinte de rubans verts et violets.

C'est un spectacle que des mots ne peuvent décrire, une expérience humble face à la puissance de la nature.

— Citation tirée de mon *Journal de bord personnel*

Catégories : [Europe](#), [Nature](#)

À propos de l'auteur

Alex, photographe et voyageur passionné, explore le monde depuis plus de 10 ans pour partager ses découvertes.

Articles récents

- [Le Japon entre tradition et modernité](#)
- [Sur les traces des Incas au Pérou](#)

Aside

3. **Zoomez sur les contenus** : Quelle partie est l'article principal ? Est-il lui-même divisé en sections thématiques ?
4. **Repérez les éléments spécifiques** : Y a-t-il une image avec une légende ? Une citation mise en évidence ? Une date ?