

HTML5 & Structure du Web

Un cours complet pour maîtriser HTML5, les éléments sémantiques et la structuration correcte de vos pages web.

Niveau : Débutant • Durée : ~12h • 8 chapitres • Certificat inclus

1. Introduction au HTML5 & histoire du Web

HTML (HyperText Markup Language) est le langage de balisage utilisé pour créer et structurer le contenu de toutes les pages web. Ce n'est pas un langage de programmation : il ne contient pas de logique ou de calculs, mais décrit la structure d'un document (titres, paragraphes, listes, images, liens...).

HTML5 est la 5e révision majeure du standard, officiellement stabilisée en 2014 par le W3C puis maintenue en continu par le WHATWG (Web Hypertext Application Technology Working Group) sous forme de « standard vivant ». Elle a apporté des éléments sémantiques, la vidéo/audio natifs, de nouveaux types de champs de formulaire et plusieurs APIs JavaScript (géolocalisation, stockage local, etc.).

Le trio du Web

Technologie	Rôle
HTML	Structure et contenu de la page
CSS	Présentation visuelle (couleurs, mise en page)
JavaScript	Comportement et interactivité

■ À retenir : HTML structure, CSS habille, JavaScript anime. Les trois sont complémentaires et ne doivent jamais être confondus.

2. Structure de base d'un document HTML5

Chaque page HTML5 suit un squelette standard. Voici le document minimal valide :

```
<!DOCTYPE html>

<html
  lang
  =
  "fr"
>

<head>

<meta
  charset
  =
  "UTF-8"
>

<meta
  name
  =
  "viewport"
  content
  =
  "width=device-width, initial-scale=1.0"
>

<title>
Ma première page
</title>

</head>

<body>

<h1>
Bonjour HTML5 !
</h1>

</body>

</html>
```

Rôle de chaque partie

- `<!DOCTYPE html>` — indique au navigateur d'utiliser le mode standard HTML5
- `<html lang="fr">` — racine du document ; l'attribut lang aide les lecteurs d'écran et le SEO
- `<head>` — métadonnées invisibles (encodage, titre, liens CSS, favicon)
- `<meta charset="UTF-8">` — indispensable pour afficher correctement les accents
- `<meta name="viewport">` — indispensable pour un affichage responsive sur mobile
- `<body>` — tout le contenu visible de la page

■■ Erreur fréquente : oublier `<meta charset="UTF-8">` provoque l'affichage de caractères bizarres (é devient Ã©).

3. Éléments de texte et sémantique HTML5

HTML5 a introduit des balises sémantiques qui décrivent le rôle du contenu plutôt que son apparence. Elles remplacent l'usage abusif de `<div>` partout, améliorent l'accessibilité et le référencement (SEO).

```
<header>
...
</header>

<!-- en-tête de page ou de section -->

<nav>
...
</nav>

<!-- bloc de navigation -->

<main>
...
</main>

<!-- contenu principal, unique par page -->

<article>
...
</article>

<!-- contenu autonome (article, post) -->

<section>
...
</section>

<!-- section thématique -->

<aside>
...
</aside>

<!-- contenu secondaire (sidebar) -->

<footer>
...
</footer>

<!-- pied de page -->
```

Titres et paragraphes

Les titres vont de `<h1>` (le plus important) à `<h6>`. Une page ne devrait avoir qu'un seul `<h1>`, et les niveaux doivent suivre une hiérarchie logique sans en sauter.

```
<h1>
Titre principal
</h1>

<h2>
Sous-titre
</h2>

<p>
Un paragraphe de texte normal.
</p>

<strong>
Texte important (gras sémantique)
</strong>

<em>
Texte insisté (italique sémantique)
</em>

<br>

<!-- retour à la ligne -->
```

■ Sémantique ≠ apparence : n'utilisez pas `<strong>` juste pour mettre en gras visuellement (utilisez CSS pour ça) — réservez-le à ce qui est réellement important pour le sens.

4. Listes et liens hypertextes

```
<!-- Liste non ordonnée -->
```

```
<ul>
```

```
<li>  
HTML5  
</li>
```

```
<li>  
CSS3  
</li>
```

```
</ul>
```

```
<!-- Liste ordonnée -->
```

```
<ol>
```

```
<li>  
Première étape  
</li>
```

```
<li>  
Deuxième étape  
</li>
```

```
</ol>
```

Le lien hypertexte est l'élément fondateur du Web : c'est lui qui permet de « naviguer » entre les pages.

```
<a  
  
href  
=  
"https://exemple.com"  
>  
Lien externe  
</a>
```

```
<a  
  
href  
=  
"contact.html"  
>  
Lien vers une autre page  
</a>
```

```
<a  
href  
=  
"#chap1"  
>  
Lien vers une ancre sur la même page  
</a>
```

```
<a  
href  
=  
"https://exemple.com"  
  
target  
=  
"_blank"  
  
rel  
=  
"noopener"  
>  
Ouvrir dans un nouvel onglet  
</a>
```

■■ Sécurité : quand vous utilisez `target="_blank"`, ajoutez toujours `rel="noopener"` pour empêcher la page ouverte d'accéder à `window.opener`.

5. Images, audio et vidéo

```
<img  
  
src  
=  
"logo.png"  
  
alt  
=  
"Logo de CoursInfo"  
  
width  
=  
"200"  
>
```

```
<video  
  
controls  
  
width  
=  
"480"  
>
```

```
<source  
  
src  
=  
"demo.mp4"  
  
type  
=  
"video/mp4"  
>
```

Votre navigateur ne supporte pas la vidéo HTML5.

```
</video>
```

```
<audio  
  
controls  
>
```

```
<source  
  
src  
=  
"son.mp3"  
  
type  
=  
"audio/mpeg"
```

```
>
```

```
</audio>
```

L'attribut alt n'est pas optionnel

Il décrit l'image pour les lecteurs d'écran (accessibilité) et s'affiche si l'image ne charge pas. C'est aussi un facteur de référencement (SEO) pris en compte par les moteurs de recherche.

■ Formats modernes : privilégiez .webp pour les images (plus léger que JPEG/PNG à qualité égale) et .mp4 (H.264) pour une compatibilité vidéo maximale.

6. Formulaires HTML5

Les formulaires collectent les données saisies par l'utilisateur (inscription, connexion, contact...). HTML5 a introduit de nombreux types de champs avec validation native, sans JavaScript.

```
<form  
  
  action  
  =  
  "traitement.php"  
  
  method  
  =  
  "POST"  
>
```

```
<label  
  
  for  
  =  
  "email"  
>  
Email  
</label>
```

```
<input  
  
  type  
  =  
  "email"  
  
  id  
  =  
  "email"  
  
  name  
  =  
  "email"  
  
  required  
>
```

```
<label  
  
  for  
  =  
  "age"  
>  
Âge  
</label>
```

```
<input  
  
  type  
  =
```

```
"number"
```

```
id
```

```
=
```

```
"age"
```

```
name
```

```
=
```

```
"age"
```

```
min
```

```
=
```

```
"1"
```

```
max
```

```
=
```

```
"120"
```

```
>
```

```
<input
```

```
type
```

```
=
```

```
"date"
```

```
name
```

```
=
```

```
"naissance"
```

```
>
```

```
<input
```

```
type
```

```
=
```

```
"tel"
```

```
name
```

```
=
```

```
"telephone"
```

```
placeholder
```

```
=
```

```
" +221 XX XXX XX XX"
```

```
>
```

```
<textarea
```

```
name
```

```
=
```

```
"message"
```

```
rows
```

```
=
```

```
"4"
```

```
></textarea>
```

```
<button  
  
type  
=  
"submit"  
>  
Envoyer  
</button>  
  
</form>
```

Type	Usage
email	Valide automatiquement le format d'un e-mail
number	Clavier numérique + bornes min/max
date	Sélecteur de date natif
tel	Clavier téléphone sur mobile
required	Champ obligatoire avant envoi

■■ La validation HTML n'est pas suffisante : elle est facilement contournable. Il faut toujours revalider les données côté serveur (PHP) avant de les stocker.

7. APIs HTML5 : stockage local & géolocalisation

HTML5 ne se limite pas au balisage : il définit aussi des APIs JavaScript accessibles directement dans le navigateur.

```
// Stocker une donnée qui persiste après fermeture du navigateur

localStorage.setItem(
  'theme'
,
  'sombre'
);

const
theme = localStorage.getItem(
  'theme'
);

// Géolocalisation (demande la permission à l'utilisateur)

navigator.geolocation.getCurrentPosition(
  function
  (position) {
    console.log(position.coords.latitude, position.coords.longitude);
  });
```

API	Rôle
localStorage	Stocke des données clé/valeur sans expiration
sessionStorage	Comme localStorage, mais effacé à la fermeture de l'onglet
Geolocation API	Récupère la position GPS/réseau de l'utilisateur
Canvas	Dessine des graphiques 2D dynamiquement

■■ Confidentialité : la géolocalisation nécessite HTTPS et l'accord explicite de l'utilisateur — le navigateur affiche toujours une demande de permission.

8. SEO, accessibilité et projet final

Un bon balisage HTML sert aussi le référencement naturel (SEO) et l'accessibilité (a11y) — deux aspects trop souvent négligés par les débutants.

Bonnes pratiques SEO

- Un seul `<h1>` par page, décrivant clairement le sujet
- `<title>` unique et descriptif dans `<head>`
- `<meta name="description">` pour le résumé affiché dans les résultats de recherche
- Attributs alt renseignés sur toutes les images

Bonnes pratiques d'accessibilité

- Utiliser les balises sémantiques plutôt que des `<div>` génériques
- Associer chaque `<label>` à son champ via `for/id`
- Garantir un contraste de couleur suffisant (géré en CSS)
- Permettre la navigation complète au clavier (Tab / Entrée)

■ **Projet final** : Construisez un mini-site de 3 pages en HTML5 pur (sans CSS) : une page d'accueil avec `<header>/<nav>/<main>/<footer>`, une page « À propos », et une page « Contact » avec un formulaire complet (nom, email, message, bouton d'envoi). Vérifiez la hiérarchie de vos titres et l'attribut alt de chaque image.

■ **Étape suivante** : une fois votre structure HTML solide, passez au cours CSS3 pour l'habiller visuellement.

Exercices & Quiz de vérification

Testez vos connaissances avant de passer au cours suivant. Chaque question a une correction détaillée à dérouler.

1. Que signifie le sigle HTML ?

- A. High Text Machine Language
- B. HyperText Markup Language
- C. Home Tool Markup Language

Correction : Réponse : B. HyperText Markup Language — le langage de balisage hypertexte, standard de structuration des pages web.

2. Combien de balises <h1> devrait contenir une page bien structurée ?

- A. Autant que de sections
- B. Une seule
- C. Aucune importance

Correction : Réponse : B. Une seule balise <h1> par page, qui représente le titre principal ; cela aide le SEO et l'accessibilité.

3. Quel attribut est indispensable sur une balise pour l'accessibilité ?

- A. src
- B. width
- C. alt

Correction : Réponse : C. L'attribut alt décrit l'image pour les lecteurs d'écran et s'affiche en cas d'échec de chargement.

4. Quelle balise sémantique HTML5 représente le contenu principal, unique, d'une page ?

- A. <section>
- B. <main>
- C. <div>

Correction : Réponse : B. <main> doit être unique dans le document et contenir le contenu central de la page.

Ressources supplémentaires

- [MDN Web Docs — Référence HTML](#)
- [MDN — Apprendre le développement web \(tutoriels guidés\)](#)
- [WHATWG — Spécification HTML vivante \(référence officielle\)](#)
- [Can I Use — Compatibilité des fonctionnalités HTML5 par navigateur](#)