

Les boîtes qui ont un nom

On pourrait tout construire avec des `div`. HTML propose pourtant des boîtes qui disent ce qu'elles contiennent — et ça n'a rien d'un détail.

4TTR

 Découverte

Techniquement, une page entière peut être bâtie avec des `<div>`. Ça fonctionne, le navigateur ne proteste pas.

Mais une page pleine de `<div>` ne dit rien de ce qu'elle contient — ni à Google, ni à un lecteur d'écran, ni à toi dans trois semaines. HTML fournit donc des conteneurs qui portent un nom.



Objectifs

À la fin de ce cours, tu seras capable de :

1. Utiliser `<header>`, `<nav>`, `<main>` et `<footer>` au bon endroit.
2. Écrire la structure type d'une page.
3. Construire un menu de navigation.
4. Habiller un en-tête.

Quatre conteneurs, quatre rôles

Balise	Ce qu'elle contient
<code><header></code>	L'en-tête : le titre du site, un logo, une accroche
<code><nav></code>	Le menu de navigation
<code><main></code>	Le contenu principal — une seule fois par page
<code><footer></code>	Le pied de page : mentions, contact, date

Elles se comportent **exactement** comme des `<div>` : ce sont des éléments bloc, elles prennent toute la largeur, elles acceptent les mêmes propriétés CSS. La seule différence est le **sens** qu'elles portent.

On les appelle des balises **sémantiques** : leur nom décrit le rôle du contenu, pas son apparence.

🧠 **À RETENIR** : un lecteur d'écran permet à une personne aveugle de sauter directement au `<main>` pour éviter de réécouter le menu à chaque page. Avec des `<div>`, ce raccourci n'existe pas.

La structure type

```
1  <body>
2
3  <header>
4    <h1>Les types de virus</h1>
5  </header>
6
7  <nav>
8    <a href="#">Introduction</a>
9    <a href="#">Les virus</a>
10   <a href="#">Phishing</a>
11 </nav>
12
13 <main>
14   <p>On appelle « virus » à peu près tout ce qui infecte un ordinateur.</p>
15 </main>
16
17 <footer>
18   <p>Mini-site cybersécurité – 2026</p>
19 </footer>
20
21 </body>
```



Cette organisation en quatre zones est celle de l'immense majorité des sites. Une fois que tu la reconnais, tu la vois partout.

Le menu de navigation

Un menu, ce n'est rien d'autre qu'une **liste de liens** placée dans un `<nav>`.

```

1 | <nav>
2 |   <a href="#">Introduction</a>
3 |   <a href="#">Les virus</a>
4 |   <a href="#">Phishing</a>
5 | </nav>

```

Petit rappel : `<a href="...">` crée un lien, et l'attribut `href` porte la destination.

i `href="#"` est un lien qui ne mène nulle part. C'est pratique pour construire un menu avant que les vraies pages existent : le lien se comporte comme un lien, mais ne provoque aucun déplacement.

Les trois liens s'affichent l'un à côté de l'autre — les balises `<a>` sont des éléments **en ligne** — mais collés, séparés par une simple espace. On verra comment les répartir proprement au cours suivant.

Habiller l'en-tête

Comme `<header>` est une boîte ordinaire, elle accepte tout ce que tu sais déjà faire :

```

1 | header {
2 |   background-color: MidnightBlue;
3 |   color: white;
4 |   padding: 20px;
5 |   text-align: center;
6 | }

```

Quatre lignes, et l'en-tête ressemble enfin à un en-tête : un bandeau sombre, du texte blanc centré, de l'air autour.

Remarque le sélecteur : `header`, sans point et sans chevrons. C'est un **sélecteur d'élément** — il vise la balise elle-même, comme `p` ou `h1`. Pas besoin de classe ici : il n'y a qu'un seul `<header>` dans la page.

⚠ ATTENTION : si ton bandeau ne touche pas les bords de la fenêtre, c'est la marge par défaut du `<body>` qui traîne. Ajoute `body { margin: 0; }`.

Et les div, alors ?

Ils ne disparaissent pas. Les balises sémantiques couvrent les **grandes zones** de la page ; à l'intérieur, quand tu regroupes trois cartes ou deux paragraphes, `<div>` reste l'outil normal.

La règle est simple : s'il existe une balise qui décrit ce que tu emballes, utilise-la. Sinon, prends un `<div>`.

Vérifications

1. Quelle différence de comportement y a-t-il entre `<header>` et `<div>` ?
2. Combien de `<main>` une page doit-elle contenir ?
3. À quoi sert `href="#"` ?
4. Ton bandeau d'en-tête laisse un liseré blanc sur les bords. D'où vient-il ?

À retenir

- `<header>`, `<nav>`, `<main>`, `<footer>` sont des `<div>` **qui portent un sens**.
- Un seul `<main>` par page.
- Un menu = un `<nav>` contenant des liens `<a>`.
- Le sélecteur CSS d'une balise s'écrit sans point : `header { ... }`.
- S'il existe une balise qui décrit ton contenu, préfère-la au `<div>`.

Suite

Ton menu existe, mais ses liens sont collés les uns aux autres. Il est temps de découvrir l'outil qui a changé la mise en page du web : Flexbox.