

TD3 — Structuration de pages et mise en boîte

Dans ce TD, nous allons voir comment mieux structurer les différentes parties à l'intérieur d'une page web. La plupart des pages sur Internet ont la même structure :

- Une bannière en haut avec le titre principal (à ne pas confondre avec le titre du site, renseigné dans les balises `<head>...</head>`).
- Parfois menu horizontal en haut.
- Parfois un menu vertical sur l'un des côtés.
- Le contenu principal au milieu.
- Éventuellement des informations annexes sur le côté.
- Un pied de page, avec des informations de contact, des mentions légales, etc.

HTML5 fournit des balises pour organiser l'ensemble de manière sémantique¹ :



```
1  <!DOCTYPE html>
2  <html lang="fr">
3      <head>...</head>
4      <body>
5          <header>
6              ...
7              <nav>...</nav>
8          </header>
9          <main>
10             <article>...</article>
11             <article>...</article>
12             <aside>...</aside>
13          </main>
14          <footer>
15              ...
16          </footer>
17      </body>
18  </html>
```

On peut en outre utiliser la paire de balises `<section>...</section>` pour séparer les différentes parties d'un bloc donné (par exemple, les différentes parties d'un article).

1 Structuration des pages

Créez un répertoire TD3 et les habituels sous-dossiers `html`, `css`. Au besoin, vous pourrez en créer de nouveaux. Ouvrez et configurez QuickPHP.

Nous allons créer deux pages utilisant le même *template* (modèle) : la classique `index.html` et une autre, `contact.html`.

1. Image de [Romain LEBRETON](#) sous licence CC BY-SA.

- Exercice 1 (Lorem ipsum).** 1) Reprenez le modèle fourni dans l'introduction et copiez-le dans `index.html`.
- 2) Nous allons compléter la page en écrivant des articles. Pour gagner du temps, nous utiliserons des textes de test, sans les écrire nous-mêmes. Le plus utilisé dans l'imprimerie est *lorem ipsum*. Rendez-vous sur la page <https://www.faux-texte.com/> et générez quelques paragraphes pour chaque `<article>...</article>`
- 3) Compléter les balises `<header></header>` avec un titre pour la page.
- 4) Dans les balises `<nav>`, créez une liste non numérotée qui contiendra deux liens :
- un lien nommé « Accueil » vers `index.html` ;
 - un autre nommé « Contact » vers la future page `contact.html`.
- 5) Remplissez le pied de page avec des informations quelconques (adresse de contact, votre nom, etc.).
- 6) Dans ce TD, nous n'utiliserons pas les balises `<aside>...</aside>` : vous pouvez les supprimer.
- 7) Vérifiez que votre code HTML est valide à l'aide du validateur W3C.

Exercice 2 (La page de contact). 1) Construire la page `contact.html` ayant la même *template* que `index.html` :

- elle reprend les même `<header>` et `<footer>` ;
 - elle partage la même feuille de style CSS ;
 - en revanche, chaque page a son propre `<title>`.
- 2) Dans la partie `<main>`, ajouter :
- a. Un titre « Adresse ». Quel niveau de titre allez-vous choisir ?
 - b. Une image de votre choix.
 - c. L'adresse de l'IUT, en utilisant la balise `<address>` (cf. [la documentation](#)).
- 3) Vérifiez la validité de votre code.

2 Mise en boîte !

2.1 Un peu de théorie

Cette partie est inspirée de [cette page](#).

Pour le moment, tout ce travail ne change rien visuellement. Mais cela améliore la structuration des pages, ce qui aide les navigateurs Internet à mieux afficher vos pages, et les moteurs de recherche à les analyser. Cela va aussi faciliter l'écriture du code CSS, puisque toutes les pages suivent la même organisation.

L'ensemble des balises de structure définissent des *boîtes* : on dit que ce sont des balises HTML de type block. Tous les blocks ont les propriétés CSS suivantes :

margin Marge à l'extérieur de la bordure, entre cette boîte et la suivante, et/ou entre cette boîte et son parent. La zone couverte par la marge est de la même couleur que son parent.

border Bordure qui entoure le contenu. Cette propriété prend trois valeurs :

- 1) **width** : épaisseur de la bordure.
- 2) **style** : pointillés, ligne, points, etc.
- 3) **color** : c'est logiquement la couleur.

padding Marge à l'intérieur. Pour bien comprendre la différence avec la propriété **margin**, faites des essais.

width largeur de la boîte.

height hauteur de la boîte.

Outre les marges et les bordures, il est possible de régler la *position* des éléments de type block, avec la propriété CSS **position**, qui peut prendre l'une de quatre valeurs suivantes :

static C'est le comportement par défaut, l'élément est inséré à la suite du flux de lecture, sur une nouvelle ligne.

relative L'élément est placé relativement à l'endroit où il aurait dû être s'il était resté **static**. En revanche, le reste de la page considère que l'élément est **static**.

absolute La boîte est placée explicitement à une position, relativement à son plus propre ancêtre autre que **static**. Le reste de la page considère que le block n'existe pas.

fixed L'élément est positionné par rapport à la fenêtre d'affichage. Le reste de la page considère que le block n'existe pas. Il semble donc *fixé* lors d'un défilement.

Pour régler les décalages, on utilise les propriétés **top**, **left**, **right**, **bottom**, dont les valeurs sont des longueurs.

2.2 En pratique

Exercice 3 (Marges et bordures). Considérons le code suivant :

```
1 main {
2   padding: 20px 50px 20px 50px; /* dans l'ordre : top, right, bottom, left */
3   border: 5px solid green;
4   background-color: gold;
5 }
```

- 1) Copiez ce code dans votre fichier CSS et modifiez les valeurs des propriétés pour observer le résultat. Une fois compris, vous pouvez supprimer (ou les garder en commentaire dans votre code source) ces lignes.
- 2) Avec la propriété **width** appliquée au sélecteur **body**, fixez la largeur de la page à 800 pixels.
- 3) Limitez de même la largeur de chaque article à 650 pixels.
- 4) Intégrez le code CSS suivant :

```
1 article {
2   /* Du code est déjà écrit... */
3   margin: auto /* Ligne à rajouter */
4 }
```

Quel est l'effet de cette nouvelle ligne de code ?

- 5) Appliquez au divers blocs de votre site les couleurs, bordures, tailles et marges de votre choix pour produire un résultat harmonieux.
- 6) Vous pouvez aussi modifier les polices des titres et du contenu selon votre convenance.
- 7) Vérifiez la validité de votre code.

Exercice 4 (Position). 1) Insérer une image de votre choix dans une des deux pages du site, à l'intérieur d'un article.

- 2) Copiez et collez le code CSS ci-dessous, puis faites varier les paramètres pour comprendre son effet :

```
design.css
1 main article img {
2     display: block /* Force la balise img à se comporter comme un block */
3     position: absolute;
4     top: 20px;
5     right: 40px;
6 }
```

- 3) Modifiez ce code pour que l'image apparaisse 5 lignes avant la fin du paragraphe qui la contient, et décalée de 60 pixels vers la gauche.
- 4) Faites en sorte que l'en-tête et le menu (donc le contenu des balises `<header></header>`) soit présent en permanence en haut de l'écran, et aligné à droite avec un décalage de 10%.
- 5) Vérifiez la validité de votre code.