***Structurer sa page***

* Plusieurs balises ont été introduites avec HTML5 pour délimiter les différentes zones qui constituent la page web :  
  • **<header>** : en-tête ;  
  • **<footer>** : pied de page ;  
  • **<nav>** : liens principaux de navigation ;  
  • **<section>** : section de page ;  
  • **<aside>** : informations complémentaires ;  
  • **<article>** : article indépendant.
* Ces balises peuvent être imbriquées les unes dans les autres. Ainsi, une section peut avoir son propre en-tête. Ces balises ne s’occupent pas de la mise en page. Elles servent seulement à indiquer à l’ordinateur le sens du texte qu’elles contiennent. On pourrait très bien placer l’en-tête en bas de la page si on le souhaite.

**Exemple**

***Code Html***

<!DOCTYPE html>  
<html>  
     <head>  
         <meta charset="utf-8" />  
         <link rel="stylesheet" href="css/styles.css" />  
         <title>Mon site avec les balises HTML5</title>  
     </head>  
     <body>  
         <header>  
               <h1>Mon site Web</h1>  
         </header>  
         <nav>  
                <ul>  
                       <li><a href="#">Accueil</a></li>  
                       <li><a href="#">Blog</a></li>  
                       <li><a href="#">CV</a></li>  
                </ul>  
          </nav>  
          <section>  
                <aside>  
                         <h1>À propos de l’auteur</h1>  
                         <p></p>  
                </aside>  
                <article>  
                         <p>Bonjour et bienvenue sur mon site !</p>  
                         <p>Pour le moment, mon site est en <em>construction</em>. Patientez encore un peu !</p>  
               </article>  
          </section>  
          <footer>  
                  <p>Copyright  - Tous droits réservés<br />  
                           <a href="#">Me contacter !</a></p>  
         </footer>  
     </body>  
</html>

**Exercice n°4 sur HTML5 et CSS3****Instructions**

* + Intégrer les balises (**<header>, <footer>, <nav>, <section>,  <aside>,  <article>**) dans les pages html que vous avez codées dans les exercices n°1, 2a, 2b et 3.

***Le modèle des boîtes***

* On distingue deux principaux types de balises en HTML :  
  • Le type block (**<p>**, **<h1>**...) : ces balises créent un retour à la ligne et occupent par défaut toute la largeur disponible. Elles se suivent de haut en bas.  
  • Le type inline (**<a>**, **<strong>**...) : ces balises délimitent du texte au milieu d’une ligne. Elles se suivent de gauche à droite.
* On peut modifier la taille d’une balise de type block avec les propriétés CSS **width** (largeur) et **height** (hauteur). On peut définir des minima et maxima autorisés pour la largeur et la hauteur : **min-width**, **max-width**, **min-height**, **max-height**.
* Les éléments de la page disposent chacun de marges intérieures (**padding**) et extérieures (**margin**).
* S’il y a trop de texte à l’intérieur d’un bloc de dimensions fixes, il y a un risque de débordement. Dans ce cas, il peut être judicieux de rajouter des barres de défilement avec la propriété **overflow** ou de forcer la césure avec **word-wrap**.

***Le positionnement en CSS***

* La mise en page d'un site web s'effectue en CSS. Plusieurs techniques sont à notre disposition.
* Le positionnement flottant (avec la propriété **float)** est l'un des plus utilisés à l'heure actuelle. Il permet par exemple de placer un menu à gauche ou à droite de la page. Néanmoins, cette propriété n'a pas été initialement conçue pour cela et il est préférable, si possible, d'éviter cette technique.
* Le positionnement *inline-block* consiste à affecter un type *inline-block* à nos éléments grâce à la propriété **display**. Ils se comporteront comme des inlines (placement de gauche à droite) mais pourront être redimensionnés comme des blocs (avec **width** et **height**). Cette technique est à préférer au positionnement flottant.
* Le positionnement absolu permet de placer un élément où l'on souhaite sur la page, au pixel près.
* Le positionnement fixe est identique au positionnement absolu mais l'élément restera toujours visible même si on descend plus bas dans la page.
* Le positionnement relatif permet de décaler un bloc par rapport à sa position normale.
* Un élément A positionné en absolu à l'intérieur d'un autre élément B (lui-même positionné en absolu, fixe ou relatif) se positionnera par rapport à l'élément B, et non par rapport au coin en haut à gauche de la page.