

TD 4 : HTML5 et CSS3

21 octobre 2015

Objectif: Ce TD a pour but de mieux découvrir le standard HTML5. Pour illustrer ce dernier, l'on modifiera le site web créé la semaine dernière en utilisant quelques-unes de ses dernières spécificités, aidés par la dernière version de CSS, notamment CSS3.

1 Back to the TD

Si vous ne l'avez pas encore fait, complétez la feuille du TD 3 (vous pouvez vous passer des "tags").¹

2 HTML5 : L'ESIPE contre-attaque

2.1 Pour la petite histoire...

Deux ans après avoir inventé le « World Wide Web » (1989), Tim Berners-Lee invente la première version de l'HTML (basée sur le SGML). C'était en 1991. Depuis cette date, l'HTML n'a jamais cessé d'évoluer (en ajoutant, parfois, des "X" dans son nom).

La dernière spécification (**HTML5**) a été publiée par le consortium (organisme de standardisation) **W3C** en 2014. Cette version sera la dernière à être numérotée. Pour l'avenir, HTML sera développé par le consortium **WHATWG** dans une version en constante évolution appelée « **Living Standard** ».

2.2 Une question sémantique

On a déjà discuté sur la syntaxe du code et sur le style en CSS. Pour compléter nos études en linguistique, on va aborder maintenant la sémantique de l'HTML5.

Pendant les TDs précédents vous avez remarqué que certains éléments définissent clairement leur contenu (<table> pour un tableau, pour une image, <p> pour un paragraphe, etc.) tandis que d'autre ne le font pas (<div>, , etc.).

De plus, la plupart des pages web utilisent les mêmes identifiants pour les "boîtes" les plus courantes: <div id="header"> pour la bannière, <div id="footer"> pour le bas de page, etc.²

1. Inutile d'attendre jusqu'à 16h29: Marty va pas vous apporter les solutions...

2. Dans le TD précédent on a préféré le français, mais vous ne serez pas étonnés de découvrir que la plupart des pages web sont écrites en anglais!

Pour standardiser cela (et surtout pour permettre aux moteurs de recherche de mieux identifier le contenu de la page), des nouveaux éléments ont été introduits en HTML5 :

- `<header>`, contient l'en-tête (du document, d'une section, etc.) ;
- `<footer>`, contient le bas de page (du document, d'une section, etc.) ;
- `<nav>`, contient la barre de navigation ;
- `<main>`, contient le contenu principale du document ;
- `<section>`, contient une section du document ;
- `<article>`, similaire à `section` mais utilisé pour des contenus autonomes ;
- `<aside>`, similaire à `section` mais pour un contenu mis à part (e.g. une barre latérale) ;
- `<figure>`, contient une image ;
- `<figcaption>`, contient la légende d'une image (`figcaption` est un fils de `figure`) ;
- etc.

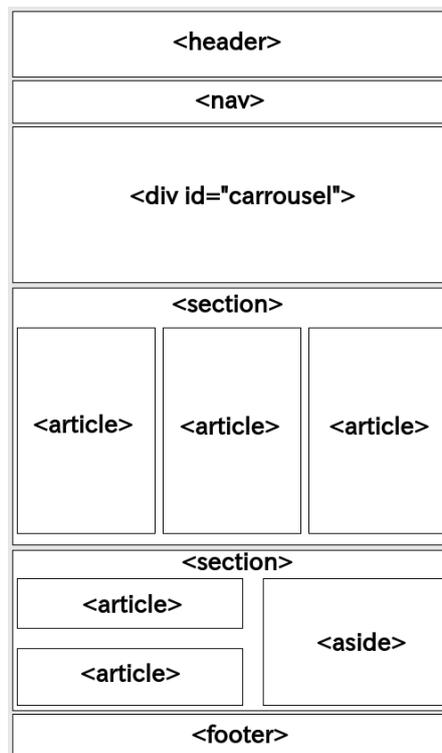


FIG. 1 – Exemple de structure d'une page web en utilisant les éléments HTML5.

Dans la feuille de style on remplacera les `id` par les nouveaux éléments. Par exemple, à la place de :

```
div#banniere, div#bas_de_page, div#menu, div#contenu, {  
    background-color: pink;  
    margin: auto;  
}
```

on écrira :

```
header, footer, nav, article {
    background-color: pink;
    margin: auto;
}
```

Comme vous pouvez le remarquer, à la différence des identifiants, on peut utiliser le même élément plusieurs fois. De plus, il est possible d'encapsuler un élément à l'intérieur d'un autre. Par exemple un élément `section` peut avoir comme fils plusieurs éléments `article`, et ces derniers peuvent être pères d'autres `section`.

► **Exercice 1** : Modifiez la page web créée dans le TD précédent en remplaçant certains des `div` avec des éléments d'HTML5.

2.3 Explorer, Chrome, Firefox, Colomb, de Gama et Magellan

Tous les navires ne sont pas égaux. Et on peut en dire de même pour les navigateurs (browsers).

Certains browsers (surtout les plus anciens) ne supportent pas HTML5. Pour résoudre ce problème il faut leur spécifier comment reconnaître les nouveaux éléments HTML.

Pour les versions d'Internet Explorer antérieures à IE9, on peut utiliser la "formule magique" (en fait, un script *JavaScript*³) « **the shiv** », créée par Sjoerd Visscher :

```
<!--[if lt IE 9]>
  <script src="http://html5shiv.googlecode.com/svn/trunk/html5.js">
  </script>
<![endif]-->
```

Remarquez que la ligne de code "magique" ci-dessus est interprétée comme un commentaire par tous les navigateurs sauf les versions antérieures à IE9 d'Internet Explorer.

Un autre problème classique avec les anciens navigateurs est qu'ils affichent nos nouveaux éléments en ligne (même quand ce n'est pas l'effet souhaité). Pour éviter cela, on donne à la propriété `display` la valeur `block` dans notre feuille de style :

```
header, nav, main, footer, aside, section, article [...] {
    display: block;
}
```

► **Exercice 2** : Modifiez votre page pour permettre la retrocompatibilité avec les anciennes versions des navigateurs (en particulier d'Internet Explorer).

2.4 Il est temps de danser !

Avec HTML5 ajouter des contenus audiovisuels n'a jamais été si facile !

3. Pas de panique ! On va expliquer ça mieux la semaine prochaine.

```
<audio controls>
  <source src="ma_musique_preferee.mp3" type="audio/mpeg">
  Nous sommes désolés mais votre navigateur ne supporte pas
  l'élément audio.
</audio>
```

L'attribut `control` permet d'afficher les commandes audio (bouton "play / pause", barre de volume, ...).

► **Exercice 3 :** Ajoutez dans votre page la musique `ievan_polkka.mp3`. En ajoutant l'attribut `autoplay`, assurez-vous que la musique démarre automatiquement quand on ouvre la page⁴.

Pour ajouter une vidéo, la syntaxe est presque la même que pour les fichiers audio :

```
<video controls width="320" height="240">
  <source src="mon_film_prefere.mp4" type="video/mp4">
  Nous sommes désolés mais votre navigateur ne supporte pas
  l'élément video.
</video>
```

Pour des raisons d'espace, on préfère ne pas stocker des vidéos sur notre serveur (ou dans notre cas, sur le serveur de l'Université). Une possibilité est donnée par les sites d'hébergement de vidéos comme, par exemple, YouTube.

Imaginons que l'on veuille insérer sur notre page la vidéo YouTube ayant comme id `9r8373gcYB8`. Pour cela, on utilisera des *iFrames*, comme dans l'exemple suivant :

```
<iframe width="500" height="350"
src="http://www.youtube.com/embed/9r8373gcYB8">
</iframe>
```

Une autre méthode consiste à utiliser l'élément `object` (moins intrusif mais non supporté par Internet Explorer), comme dans l'exemple suivant :

```
<object width="500" height="350"
data="http://www.youtube.com/embed/9r8373gcYB8">
</object>
```

► **Exercice 4 :** Cherchez sur YouTube un court métrage réalisé par des anciens étudiants IMAC et insérez-le dans votre page web à l'aide d'un `iFrame`.

3 One, two, CSS3, viv' la feuille de style !

Comme HTML, le standard CSS continue à évoluer. Par la suite, on va découvrir certaines nouvelles fonctionnalités qu'on peut ajouter dans nos feuilles de style.

3.1 Arrondissons les angles

Pour des raisons esthétiques ainsi que diplomatiques, on pourrait vouloir arrondir les angles d'un élément HTML. Avec CSS3, on peut le faire en utilisant la propriété `border-radius`.

⁴. Mais ne dérangez pas trop vos camarades...

Par exemple :

```
.mon_element_arrondi {  
    border-radius: 20px;  
}
```

Que se passe-t-il si on donne deux, trois ou quatre valeurs à la propriété `border-radius` (par exemple en écrivant `border-radius: 10px 0px 50px 15px;`)?

► **Exercice 5 :** Reprenez la page HTML et la feuille de style du TD précédent et modifiez-les pour arrondir les angles du menu, du carrousel, du contenu et du sous-contenu. Pour ces deux derniers, a-t-on encore besoin de `boite_haut` et `boite_bas`?

3.2 Somewhere over the rainbow

En CSS3 on a la possibilité de choisir des couleurs RGBA, où la dernière lettre indique l'opacité (ou "*alpha channel*"): de 0.0 (totalement transparent) à 1.0 (totalement opaque).

Par exemple, pour obtenir une page web avec tous les paragraphes ayant un arrière plan rouge d'opacité 0.5. on écrira dans la feuille de style :

```
p {  
    background-color: rgba(255, 0, 0, 0.5);  
}
```

En plus de cela, CSS3 permet d'afficher une transition douce entre deux (ou plusieurs) couleurs. Par exemple :

```
div.soleil_sur_mer {  
    background: linear-gradient(yellow,blue);  
}
```

donne comme résultat des transitions du jaune (en haut) au bleu (en bas) pour les éléments div de classe « `soleil_sur_mer` ».

► **Exercice 6 :** Définissez une classe « `arc_en_ciel` » pour les éléments div qui permet d'afficher dans l'ordre, et avec une transition horizontale, les couleurs: rouge, orange, jaune, vert, bleu, indigo et violet (avant les couleurs pensez à ajouter "`to right`").

3.3 N'ayez pas peur de votre ombre

Avec CSS3 on peut aussi jouer avec les textes, par exemple en ajoutant des ombres ou en changeant la police de caractère.

Pour le premier effet, on utilisera la propriété `text-shadow` ayant comme valeurs le décalage horizontale et le décalage verticale.

Par exemple,

```
h1.ombre {  
    text-shadow: 10px 5px;  
}
```

permet d'afficher une ombre décalée de 10 pixels à droite et 5 pixels en bas par rapport au texte.

Il est possible d'ajouter aussi un effet de flou et de changer le couleur de l'ombre par rapport au texte, comme dans l'exemple suivant :

```
h1.ombre_coloree {
    text-shadow: 10px 5px 2.2px red;
}
```

3.4 Des photos un peu en désordre

Une autre famille d'effets sympatiques qu'on peut obtenir avec CSS3 est celle des transformations : décalage, rotation, mise à l'échelle, etc. Par exemple, le code :

```
div.pivote {
    transform: rotate(30deg);
}
```

permet de pivoter l'élément `div` de classe « `pivote` » de 30° dans le sens des aiguilles d'une montre.

► **Exercice 7 :** À l'aide de http://www.w3schools.com/css/css3_2dtransforms.asp définissez la classe « `loupe` » pour les éléments `div` qui augmente de trois fois (horizontalement et verticalement) la taille.

► **Exercice 8 :**

1. Modifiez la galerie d'images que vous avez créée dans le TD précédent en utilisant les nouveaux éléments HTML5 (notamment `figure` et `figcaption`).
2. En utilisant la pseudo-classe “`:nth-child()`”, donnez des angles de rotations “pseudo-aléatoire” aux images de la galerie.
3. En utilisant la pseudo-classe “`:hover`”, faites en sorte qu'au survol, les images s'agrandissent.
4. En utilisant la pseudo-classe “`:target`” et la propriété “`display`”, trouvez une solution alternative aux `iFrames` pour obtenir le même résultat. Dans le code HTML vous devriez avoir quelque chose comme :

```
<figure>
  <a href="#radeau"></a>
</figure>
<figure>
  <a href="#beach"></a>
</figure>
<figure>
  <a href="#journal"></a>
</figure>
<figure>
  <a href="#groupe"></a>
</figure>

[...]

<figure id="grand_figure">
  
  
  
  
</figure>
```