

TP 2

HTML 5 (suite) et introduction aux CSS

Internet et Outils (IO2)

Février 2013

Assurez-vous que vous déclarez bien l'encodage UTF-8 dans l'en-tête de vos fichiers et que vos éditeurs sont réglés en conséquence (voir section 7 du TP1).

Créez un sous-répertoire IO2-TP2 dans votre repertoire public_html dans lequel vous placerez les fichiers de ce TP. Pensez également à vérifier systématiquement la validité de vos pages web à l'aide du validateur du W3C à l'adresse <http://validator.w3.org>.

1 Les balises html

Un aide-mémoire contenant une liste de balises HTML expliquées est disponible sur l'aide-mémoire du cours <http://didode.fr/wp/aide-memoire-html/>.

1.1 Echauffement et balises sémantiques

Exercice 1 Créez un fichier cv.html dans lequel vous recréerez le CV fictif de la figure 1.

Exercice 2 Pour faciliter l'analyse des données par des outils automatiques (moteurs de recherche par exemple), il faut structurer le document. C'est à cela que servent les balises sémantiques introduites par HTML 5 (cf. aide-mémoire).

Structurez votre fichier cv.html à l'aide de telles balises de sorte que les coordonnées soient dans un header et que chaque section du document soit bien indiquée comme telle.

Ajoutez également, en bas du document, une petite mention en footer indiquant que l'auteur du document n'est pas Paul Durand mais vous, ainsi que les coordonnées auxquelles quelqu'un qui voudrait faire appel à vos services pour avoir un CV similaire pourrait vous joindre.

1.2 Navigation

Exercice 3 Créez un nouveau fichier l Ipsum.html qui contiendra 7 sections (appelez-les Lundi, Mardi, Mercredi, etc.) que vous remplirez chacune avec un ou deux paragraphes de texte automatique¹.

Au début de votre page ajoutez une table des matières où chaque titre de section sera cliquable et renverra à la section correspondante.

Devant chaque section, ajoutez une double flèche vers le haut (2uparrow.png dans le repertoire TP2 sur DDEL), cliquable elle aussi, et qui permet de retourner à la table des matières. Ajoutez également devant chaque section une flèche simple (uparrow.png, disponible sur DDEL) qui renvoie à la section précédente.

Pensez à utiliser des balises sémantiques pour structurer votre document.

1. Vous en trouverez à l'adresse <http://fr.lipsum.com/>.

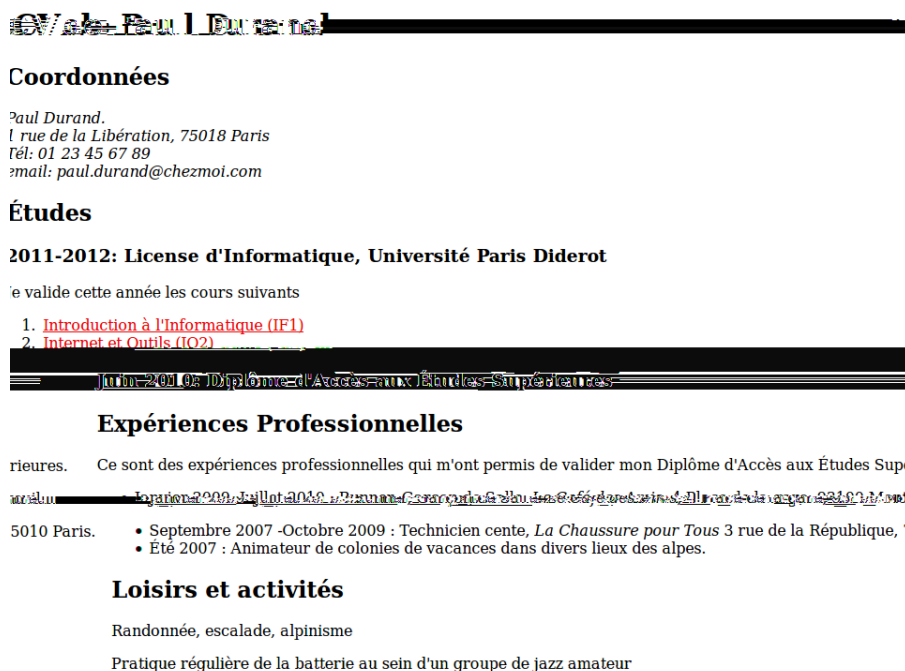


FIGURE 1 – CV pour l'exercice 1

Exercice 4 — Pour aller plus loin. . . Récupérez l'image d'Astérix, Obélix et Idéfix (asterix.jpg) et insérez-la dans votre page en utilisant la balise ``. Cette balise s'utilise avec des attributs, dont :

- `src` (attribut obligatoire) l'URL de l'image,
- `alt` (attribut obligatoire) le texte à afficher si le navigateur n'arrive pas à charger l'image,
- `title` un texte qui s'affiche sous forme de petit pop-up lorsque la souris survole l'image.

En utilisant la balise `<map>`, faites en sorte que :

1. si l'on laisse sa souris sur l'un des personnages, son nom soit affiché ;
2. si l'on clique sur l'un des personnages, on soit renvoyé sur la page wikipédia correspondante.

Notez que pour que le code fonctionne, il faut identifier le map à utiliser grâce à un attribut `name`.

Pour générer le code html nécessaire à rendre actives certaines zones de l'image, utilisez le plugin image Map de l'Editeur d'image GIMP. Après avoir ouvert l'image asterix.jpg avec GIMP, cliquez avec le bouton droit de la souris au milieu de l'image, pour obtenir le menu des différentes options. Cherchez Filtres, Web puis Image cliquable Web. Pour obtenir le code source à utiliser ensuite, dans le menu cherchez Affichage et puis Source

Exercice 5 Que se passe-t-il si dans un map deux area s'intersectent ?

Balises à regarder : `<tabl e>`–`</tabl e>`, `<tr>`–`</tr>`, `<td>`–`</td>` et `<th>`–`</th>`.
Balises sémantiques : `<thead>`–`</thead>`, `<tfoot>`–`</tfoot>` et `<tbody>`–`</tbody>`.

```
<style>
table {
    border: 1px solid black;
}
</style>
```

Découpez logiquement votre tableau en utilisant les balises sémantiques. Ce découpage logique sera utilisé ensuite pour la mise en page CSS.

Mois	Bénéfices
Janvier	100€
Février	120€
Mars	42€
Avril	60€

Exercice 7 — Pour aller plus loin... Reproduisez le tableau de la figure 3.

Un tableau inutile mais plus compliqué !

	Une colonne	Deux colonnes	Et de trois !
Une ligne	<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>	<p>Vestibulum gravida tristisque odio. Curabitur et pede. Vivamus adipiscing. Praesent figula libero, adipiscing non, hendrerit ac, eleifend sed, lectus. In ornare lacus ut nulla. Morbi a lorem.</p>	<p>Sed et erat. Ut class aptent taciti sociosqu ad litora torquent per</p>

3

2 Rudiments de css

2.1 Séparer le fond de la forme

L'intérêt (et l'efficacité) des langages à balises, c'est la possibilité de séparer le fond de la forme. Lorsque vous écrivez `<h1>Bonjour</h1>` en HTML, vous spécifiez que "Bonjour" a valeur de titre dans votre page. Mais vous n'indiquez absolument pas quelle doit être son apparence (gras, souligné, police, taille, etc.). Ces éléments complémentaires de présentations seront les mêmes pour tous les titres `<h1>--</h1>` et peuvent être stipulés dans une feuille de style, ou CSS en anglais (pour Cascading Style Sheets) :

1. soit en début de document entre les balises `<style>--</style>` (voir Exercice 8) ;
2. soit dans un fichier séparé, auquel cas il faudra alors préciser dans votre document HTML où se trouve cette CSS (voir Exercice 9).

Si aucun style n'est spécifié, le navigateur utilise des styles par défaut qui lui sont spécifiques (c'est ce que nous avons fait jusqu'à présent).

Une spécification de style est une déclaration de la forme :

```
sélecteur
{
    propriété1: valeur1;    /* commentaire */
    propriété2: valeur2;
    ...
    propriétém: valeurm
}
```

où sélecteur est souvent un élément simple HTML (comme `body` ou `h1`), propriété est une propriété de style (qui n'a rien à voir avec un attribut d'un élément) permise pour cet élément, et valeur est une valeur possible pour la propriété concernée. Ces déclarations vont spécifier le style lors de l'affichage des éléments. Par exemple, si on écrit dans l'en-tête d'un document HTML :

```
<style>
  body
  {
    background-color: green
  }
  h1
  {
    background-color: blue;
    text-align: right
  }
</style>
```

On obtient une page (horrible) sur fond vert et dont les titres de niveau 1 sont sur fond bleu et alignés à droite.

2.2 Utiliser la balise `<style>--</style>`

Exercice 8 Ouvrez le fichier `home.html` (dans le repertoire "Fichiers" sur le Didel) et la fenêtre `fi rebug` sur votre navigateur. Dans l'arborescence `html`, déroulez le bloc `head` et tout à la fin, sélectionnez puis éditez le bloc `style`. Amusez-vous à changer la présentation de la page. Vous pourrez trouver des idées sur <http://www.w3.org/TR/CSS/#properties>. Essayez par exemple de :

- changer la couleur de fond ;
- aligner tous vos paragraphes à droite ;
- mettre le contenu des paragraphes en rouge.

2.3 Ma première feuille de styles

Exercice 9 Dans votre répertoire de travail, créez un fichier `messtyles.css` et copiez dedans toutes les spécifications que vous aviez ajoutées entre les balises `<style>–</style>` sous `fi rebug`. Ajoutez dans l'en-tête de votre fichier `cv. html` la ligne

```
<link rel="stylesheet" href="messtyles.css">
```

Rechargez la page et observez le résultat.

2.4 Attributs et css

Si certaines propriétés de style sont communes à toutes les balises (comme `background` par exemple, ou encore `color`), certaines propriétés ne peuvent être utilisées qu'avec certaines balises.

Ainsi, la propriété `list-style` ne peut être utilisée qu'avec des balises de liste et peut prendre des valeurs différentes selon qu'il s'agit d'une liste ordonnée ou non. Par exemple, on peut énumérer une liste ordonnée à l'aide des lettres de l'alphabet ou des chiffres romains et remplacer les puces rondes d'une liste non-ordonnée par des petits carrés.

Pour votre information, un certain nombre de ces propriétés étaient autrefois des attributs des balises (en HTML4) et ont été converties en propriété lors du passage à HTML 5 pour refléter le fait qu'elle participaient plus de la forme que de la structure du document.

Exercice 10 Modifiez sous `fi rebug` la feuille de style précédemment créée de façon à ce que :

- la liste ordonnée soit "numérotée" par les lettres de l'alphabet,
- la liste à puce utilise des petits carrés.

Essayez de faire utiliser des petits carrés à la liste ordonnée pour voir ce qui se passe.

3 Formulaires élémentaires

Un formulaire sert à transmettre des données à un serveur. Il peut contenir divers types d'entrée (champs textuels, menus déroulants, boutons à cliquer, etc.). Nous ne verrons pas ici comment les données transmises au serveur sont traitées par celui-ci (cela viendra plus tard), mais nous constaterons que des renseignements passent dans les requêtes HTTP.

Un formulaire se trouve toujours entre les balises `<form>–</form>`. Une entrée correspond à la balise `<input>`, l'attribut `type` permettant de spécifier le type d'entrée.

Par exemple :

Un champ textuel correspond à l'attribut `text` : `<input type="text" name="reponse">`

Un bouton de soumission correspond à l'attribut `submit` : `<input type="submit" value="ok">`

L'attribut `name` permet de récupérer la valeur donnée à l'attribut par l'utilisateur.

La description des principaux types d'entrée se trouve sur http://www.w3school.s.com/html/html_forms.asp.

Exercice 11 Écrivez une page `formulaire.html` contenant votre premier formulaire qui ressemblera à celui de la figure 4. Remplissez le champ textuel et soumettez. Regardez la barre d'adresse de votre navigateur. Si vous n'aviez pas mis d'attribut `name`, ajoutez-le et recommencez. Que remarquez-vous ?



FIGURE 4 – Un formulaire très simple

Ajouter une série de boutons radios à votre formulaire et observez à nouveau la barre d'adresse de votre navigateur lorsque vous soumettez ce formulaire.

La chaîne de caractères qui s'affiche dans la barre d'adresse, s'appelle une URL (Uniform Resource Locator).

Exercice 12 HTML 5 a introduit de nouveaux types d'entrée. Ils sont explicités ici : http://www.w3schools.com/html5/html5_form_input_types.asp.

Écrivez un formulaire qui contient un champ de type email et observez ce qui se passe si votre entrée ne ressemble pas à une adresse électronique. Regardez de même le type url .

Entrez une adresse électronique valide et observez l'URL obtenue par validation du formulaire. Le caractère @ a été transformé en %40 car une URL ne peut contenir que des caractères ASCII ; les autres caractères sont remplacés par un % suivi de deux chiffres.

Essayez maintenant de spécifier directement une adresse électronique non valide, par exemple ne contenant pas le caractère @, à travers l'adresse qui apparaît dans la barre du navigateur. Cela pose-t-il un problème ?

Exercice 13 A l'aide du type d'entrée datalist dont la documentation se trouve là : http://www.w3schools.com/html5/html5_form_elements.asp, écrivez un formulaire dans lequel on peut introduire une url correspondant en priorité à un des sites web de vos cours.

4 Caractères particuliers en HTML

Indépendamment des problèmes de codage de caractères déjà évoqués, certains caractères sont réservés en HTML. Cela signifie qu'ils ont une signification spéciale. Par exemple le caractère < signale le début d'une balise. On peut cependant souhaiter écrire ce caractère dans une page html ; on utilise alors un codage spéciale, soit sous forme de nom, soit sous forme de nombre. De même, si un caractère n'apparaît pas sur le clavier, on peut l'expliciter de manière codée. La table 1 donne certains codages par nom.

(espace insécable)	
<	<
>	>
≥	≥
&	&
€	€
É	É

TABLE 1 – Codage de certains caractères en HTML