

TP 1

Premiers pas en HTML 5

Internet et Outils (IO2)

Janvier 2013

Le but de ce TP est :

- de comprendre et écrire des documents HTML élémentaires.
- d'apprendre à se servir des ressources (logiciels, documentation) pour le développement du HTML.

L'accent est mis sur la version 5 de HTML.

0 Didel - DIDEROT En Ligne

Le cours d'IO2 est présent sur Didel. Là, vous pourrez :

- trouver les feuilles de TP et les fichiers à télécharger,
- accéder à un forum dédié au cours où poser vos questions, auxquelles les enseignants comme les étudiants pourront répondre,
- rendre vos projets,
- etc.

On vous rappelle qu'on y accède à partir de l'adresse `http://didel.script.univ-paris-diderot.fr/` avec ses identifiants ENT. Connectez-vous sur Didel, inscrivez-vous comme élève du cours et rejoignez le groupe correspondant à votre groupe de TP.

Ce document pdf ainsi que les feuilles de TP suivantes et les documents à télécharger pour faire certains exercices, seront toujours à télécharger sur le Didel du cours.

1 Analyse d'une première page HTML 5

Exercice 1 Visitez le Didel du cours IO2, "documents et liens" > "TP1", et affichez dans votre navigateur le fichier `accueil.html`. Regardez le code source de la page (on explique sur cette même page comment faire). Observez en particulier :

- les *balises* (*tag* en anglais), qui commencent par `<` et terminent par `>`, décrivent la structure de la page HTML.
- la balise de déclaration du `DOCTYPE`, obligatoire pour que le document soit reconnu en tant qu'HTML version 5¹ ;
- les balises *ouvrantes* et *fermantes* qui déterminent le début et la fin des *éléments* (comme `<h1>` et `</h1>`, ou `<p>` et `</p>`) ;
- les éléments `html`, `head`, `title` et `body`, présents dans tout document HTML ;
- l'*attribut* (*attribute* en anglais) `lang` de la balise `<html>`, qui possède la *valeur* "fr" ;
- les éléments *vides*, qui n'ont pas de balise fermante (par exemple `<meta>`). (En l'occurrence, la balise `<meta charset=...>` indique l'encodage, on en dit davantage dans un instant.)
- la balise de commentaire `<!-- ... -->` qui n'est pas lue par le navigateur.

1. Sans cela, les navigateurs modernes activent un mode de rendu qui cherche à imiter de vieilles version du navigateur Internet Explorer, mais qui n'est pas conforme à la spécification de HTML. On peut voir le mode de rendu sous firefox : voir "Informations de la page" > "Mode de rendu".

2 Ma première page HTML

On vous rappelle que le nom d'un fichier HTML prend l'extension `.html`. Le document HTML 5 minimal (conseillé) est constitué de ce qui suit :

```
<!DOCTYPE html>
<html lang="fr">
  <head>
    <meta charset="utf-8">
    <title></title>
  </head>
  <body>
  </body>
</html>
```

Les balises telles que celles qui indiquent les titres (`<h1>...</h1>`, `<h2>...</h2>`, `<h3>...</h3>`, etc.) ou un paragraphe (`<p>...</p>`), ou similaire, doivent être placés entre les balises `<body>` et `</body>`.

Exercice 2 Dans votre répertoire de travail, créez un fichier html appelé `page1.html`, par exemple avec un titre (title) "Une page minimaliste" et contenant le texte "Bonjour tout le monde !" en titre de niveau 1 (`h1`). Vous pouvez également expérimenter les autres balises que vous connaissez.

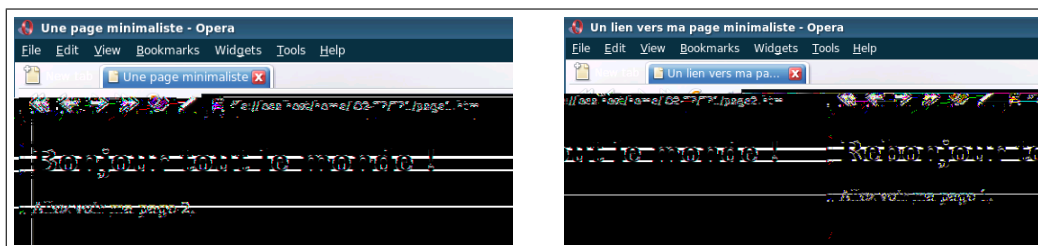
Nous vous recommandons fortement d'indenter votre code, comme sur la page `accueil.html`. C'est une question de lisibilité !

2.1 Mes premiers liens

Tout l'intérêt du web, c'est qu'on peut se déplacer en cliquant sur des liens. Pour insérer un lien, on utilise la balise `<a>` avec les attributs `href` et `title`. La valeur de `href` donne l'adresse cible (URL, pour *Uniform Resource Locator*). Exemple :

```
<a href="http://www.example.com">Ce lien est un exemple</a>
```

Exercice 3 Créez une deuxième page similaire à la première dans un fichier appelé `page2.html` dans le même répertoire de travail. Changez le titre (title) en : "Un lien vers ma page minimaliste". Changez "Bonjour" en "Rebonjour". Ajoutez un paragraphe "Allez voir ma page 1." Insérez un lien vers votre première page sur les mots "ma page 1". Ajoutez de la même manière un lien de votre page 1 vers votre page 2.



2.2 Ma première page web

Le SCRIPT dispose d'un serveur web sur lequel vous allez publier les pages ci-dessus. Votre site sera alors accessible à l'adresse : `http://pams.script.univ-paris-diderot.fr/~votre_login`. Ce site n'est accessible que depuis le réseau interne de l'université, il est malheureusement impossible de le voir depuis l'extérieur.

Ce qui est accessible sur votre serveur correspond au contenu du répertoire `public_html` qui se trouve à la racine de votre répertoire principal.

Exercice 4 — Utiliser le serveur web du SCRIPT

1. Il faut tout d'abord s'assurer que le répertoire `public_html` est accessible : créez-le s'il n'existe pas, et donnez les droits d'accès (`chmod a+x ~/public_html`) à ce répertoire ainsi qu'à votre répertoire principal.
2. Mettez-y `page1.html` et `page2.html`, et affichez ces pages dans votre navigateur, à partir de l'adresse en `http` du serveur du SCRIPT.

3 Outils et ressources sur Internet

3.1 Documentation

Outre l'aide-mémoire rudimentaire du cours sur <http://didiode.fr/wp/aide-memoire-html/>, on pourra se référer (y compris durant les TP) aux sources suivantes :

- Le manuel de référence à destination des développeurs web du W3C² : <http://dev.w3.org/html5/spec-author-view/>.³
- Le site <http://www.w3schools.com/html5/> contient des tutoriaux (en anglais), des exemples, ainsi qu'une applet qui permet de tester du code HTML directement dans le navigateur. Cependant, comme d'autres sites Internet, c'est quelquefois à prendre avec des pincettes.

3.2 Le validateur

Le W3C fournit un validateur de pages HTML que l'on trouve à l'adresse <http://validator.w3.org/>. On peut y entrer une URL, ce qui ne fonctionnera pas avec le serveur du SCRIPT qui est isolé de l'extérieur.

On peut aussi copier-coller une page HTML sous l'onglet "Validate by Direct Input", ou télécharger un fichier à partir du disque sous l'onglet "Validate by File Upload". Choisissez la solution que vous préférez.

Une page est valide si le validateur affiche "This document was successfully checked as HTML5 !". (Il est normal qu'il y ait des "warnings".)

Exercice 5 — Validation Vérifiez la validité de la page `accueil.html` de l'exercice 1 en copiant-collant le code source de la page.

1. La page est-elle valide ?
2. Que se passe-t-il si on échange l'ordre des balises `` et `` juste avant **Indice**, sans changer l'ordre des balises fermantes ? Le document HTML ainsi modifié reste-t-il valide ? Pourquoi ?
3. Vérifier que la plus petite page HTML 5 valide est la suivante :

```
<!DOCTYPE html>
<html>
  <head>
    <title></title>
  </head>
</html>
```

c'est-à-dire qu'on ne peut rien y enlever.

Exercice 6 — Invalidation

1. Vérifiez la validité de vos pages `page1.html` et `page2.html`.
2. Corrigez votre code HTML jusqu'à ce qu'elles soient valides. Vous pouvez vous aider des indications d'erreur que vous fournit le validateur, des sources ci-dessus, de google, etc.

2. Le World Wide Web Consortium est l'organisme chargé de développer les standards pour le web.

3. Remarquer que la recommandation HTML 5 est encore en développement, donc le manuel est encore au stade de brouillon (*draft*). Cependant, "Le W3C vise la clôture des ajouts de fonctionnalités le 22 mai 2011 et une finalisation de la spécification en 2014, et encourage les développeurs Web à utiliser HTML 5 dès maintenant." (Wikipédia)

3.3 Inspection du DOM avec Firebug

Pour l'instant, on a aperçu nos documents HTML de deux façons :

- comme un fichier texte, à travers le code source de la page, lorsqu'on l'écrit à la main ou qu'on le génère ;
- comme une page web telle que rendue par le navigateur ;

En réalité, le navigateur se représente la page HTML de façon bien plus complexe, conformément au Document Object Model ou DOM. Le DOM décrit la structure (arborescente) de la page, mais aussi la façon dont on peut interagir avec la page pour la rendre dynamique (à l'aide d'un langage tel que le JavaScript).⁴ On peut accéder à cette représentation interne grâce à l'outil Firebug. Au SCRIPT, Firebug n'est utilisable qu'avec le navigateur Firefox, mais ailleurs, Firebug est disponible dans sa version Lite pour les principaux navigateurs.

Firebug est un outil extrêmement puissant pour aider les développeurs web : il permet entre autre d'inspecter l'arbre, d'aider à la mise au point des feuilles de style (dont on parle plus loin), de déboguer du code JavaScript et d'analyser les performances d'un site.

Exercice 7 — Mes premiers pas avec Firebug

- Visualisez dans Firefox la page accueil.html et lancez Firebug (menu "Outils"). Dans le cadre de gauche, vérifiez que vous êtes bien dans l'onglet HTML. Regardez rapidement comment le navigateur interprète le fichier HTML sous forme d'arbre et modifiez-le (changez une partie du texte, supprimez une branche de l'arbre, etc.). Vérifiez que le changement se fait immédiatement sur la page web.
- Rafraichissez ensuite votre navigateur. Que se passe-t-il ? Pourquoi ?
- Allez maintenant sur un site un peu plus imposant (comme <http://www.lemonde.fr>), jetez un coup d'œil à l'arbre.

4 Éléments de style

HTML applique le principe de séparation du *style* et du contenu. Ce que l'on a vu pour l'instant décrit le sens d'une page web, mais pas la façon dont la page applique un style à chaque élément. Pour décrire le style, ou la façon dont les éléments sont dessinés, on a recours à un autre langage (CSS), souvent dans un fichier séparé.

L'une des raisons de cette séparation est qu'on peut très bien vouloir qu'une même page s'affiche de plusieurs façon différentes, surtout de nos jours où on consulte internet à partir d'écrans de toutes tailles. Sans parler des visiteurs spécifiques tels que les mal-voyants, pour qui la bonne structuration du HTML est plus importante que le style...

Les navigateurs utilisent des styles par défaut, par exemple en écrivant les titres en plus gros, le contenu de la balise `<kbd>` avec des caractères de largeur constante, ou celui de `` en gras. Nous verrons dans quelques semaines comment faire nos propres styles.

Cette semaine le but est de bien comprendre qu'une même page web, avec le même code source peut s'afficher de deux façons bien différentes si on lui applique deux style différents.

Exercice 8 — Attacher une feuille de style

- Visualisez la page accueil.html à partir de Didel (rafraichissez la page si vous aviez fait des modifications lors de l'exercice précédent). À l'aide de Firebug, rajoutez dans l'entête (entre les balises `<head>`) la balise `<link>` suivante :

```
<link rel="stylesheet" href="style1.css">
```

Pour cela on peut se servir de la fonctionnalité "Éditer le code HTML" de Firebug (menu du clic droit).

- Essayez maintenant avec `style2.css`.

4. Le DOM est une recommandation du W3C, intégrée à la recommandation HTML depuis la version 5.

5 Quelques balises

Un certain nombre d'éléments HTML figurent dans l'aide mémoire (<http://didiode.fr/wp/aide-memoire-html/>). Il sera régulièrement mis à jour avec les nouvelles balises que nous allons rencontrer au cours des TP.

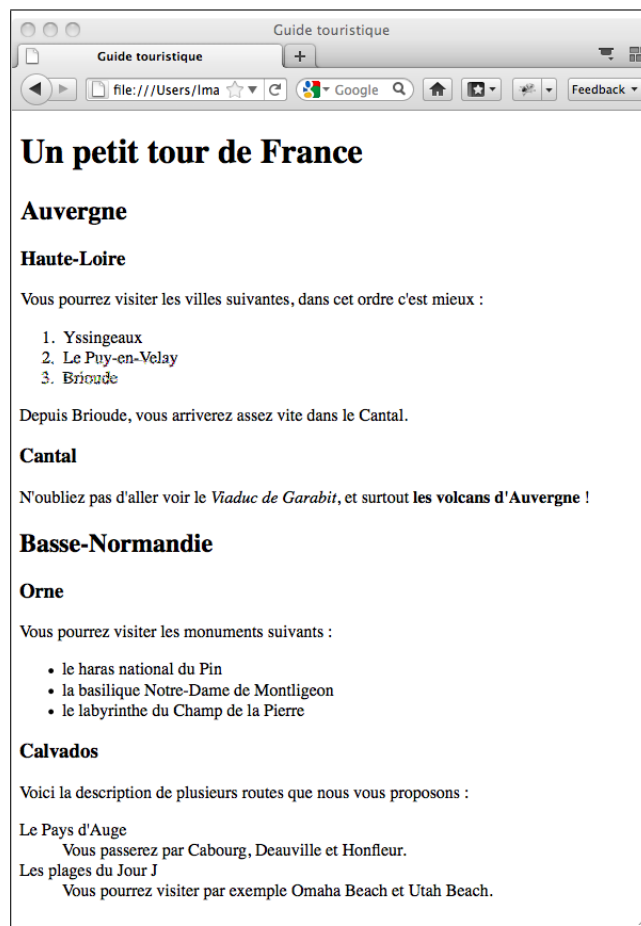
Encore une fois, les balises servent à **structurer** la page web en fonction du **sens** qu'ont les différents éléments. Pour la mise en page, on utilisera les feuilles de style CSS.

Pensez aux personnes aveugles quand vous écrivez une page web. Leur interface vocale ne leur décrit pas la mise en page, mais la structure du document.

Pensez à la façon dont les moteurs de recherche vont vous référencer, le jour où vous écrierez de vraies page web. Là aussi, c'est la structure du document qui prime sur l'affichage...

En plus des éléments d'en-tête, de lien et de paragraphes déjà rencontrés dans les exercices précédents, nous allons utiliser ici les différents éléments de listes, de mise en valeur, et d'insistance.

Exercice 9 Reproduisez la page web ci-dessous dans un fichier appelé `tourisme.html`, en respectant la structure du document (Indice : allez regarder les balises de liste). Vous pouvez aussi lier une feuille de style à l'aide de la balise `<link>` comme dans l'exercice précédent. Assurez-vous que `tourisme.html` est valide grâce au validateur.



Balises sémantiques Une page internet, outre le contenu que cherche le lecteur (l'article de blog ou de journal, etc.), comporte habituellement une en-tête avec un titre ou un logo, un menu de navigation, des publicités, etc.

HTML 5 introduit des balises dites *sémantiques* pour identifier ces différentes parties récurrentes d'un site internet. Par exemple, cela aidera les agents automatisés (le robot Google, le site <http://www.readability.com>, les outils pour aveugles...) à comprendre le contenu de la page.

- `<section>` identifie une partie de la page,
- `<header>` identifie une en-tête de page ou de section,
- `<nav>` un élément dédié à la navigation,
- `<footer>` un pied-de-page (ces trucs qu'on ne lit jamais),
- `<article>`, `<figure>`, `<aside>`, et d'autres encore.

Exercice 10 Rajoutez à `tourisme.html` un titre général ("Le site de [votre nom]...") grâce des balises `<header>` et `<h1>`. Rajoutez également une liste de deux liens vers `page1.html` et `page2.html`, au sein d'un élément `<nav>`. Le contenu initial sera identifié par une balise `<article>` qui l'entoure.

Assurez-vous que la page est valide grâce au validateur.

Observer que les balises `<header>`, `<nav>`, `<article>` ne changent rien à l'affichage de la page sous Firefox (en l'absence d'une définition explicite de leur style).

6 Images et liens

Exercice 11 — Images

Téléchargez le fichier `volcan.jpg` disponible à l'adresse Didel du cours et insérez cette image dans le fichier `tourisme.html` de l'exercice précédent, à l'endroit qui vous semble le plus approprié. Quelle balise faut-il employer ?

Faites de cette image un lien vers le site `http://www.lesvolcansdauvergne.com/`.

Comme toujours, assurez-vous que votre page est valide.

Jusqu'à présent, nous avons vu comment faire des liens vers d'autres pages web. On peut également faire un lien interne à une page :

Tout d'abord il faut donner un identifiant à l'élément que l'on souhaite lier. Pour cela il suffit de lui rajouter un attribut `id` (par exemple `id="auvergne"`), en faisant attention que deux éléments ne doivent pas avoir le même identifiant.

Alors, la balise `<a>` d'attribut `href` de valeur `#auvergne` fait un lien vers cet élément.

Exercice 12 — Liens internes Ajoutez un identifiant à chacun des titres des départements. Rajoutez au début du document, dans le `<nav>`, une liste de liens vers les différents départements. Observer l'effet des liens (quitte à réduire la taille de la fenêtre de votre navigateur). Assurez-vous que votre page est valide.

7 Encodage

Quand vous enregistrez un document, l'éditeur de texte traduit les caractères en chiffres en fonction d'un *encodage* spécifique par exemple : `iso-8859-1` (latin 1) pour les langues occidentales, `iso-8859-15` (latin 9) pour le français, `koi8-r` pour le russe. Ces différents encodages ne sont pas inter-traductibles.

Pour pallier ce problème, un jeu de caractères unifié a été défini, il s'agit d'Unicode. Unicode permet d'écrire tous les caractères actuels (lettres avec accents, chinois, arabe, hébreux, grec, etc.) mais aussi un énorme ensemble de symboles comme des notes de musiques ou ceux utilisés en mathématiques.

Unicode définit plusieurs encodage dont `utf-8` dont on va se servir par défaut, car c'est le plus répandu (retenez donc ce nom).

Malgré le progrès que constitue Unicode, un document texte n'a pas de sens si l'on ne précise pas l'encodage des caractères. Aussi, il faut faire attention quand on utilise des caractères qui ne font pas partie des 128 caractères du standard américain ASCII : accents, cédilles, signe euro, lettres dans d'autres alphabets (grecs, cyrilliques, etc.). Si l'encodage n'est pas précisé, la page web peut mal s'afficher.

Pour indiquer au navigateur quel est l'encodage utilisé, on a utilisé dans `accueil.html` la balise :

```
<meta charset="utf-8">
```

Bien sûr, il faut veiller à enregistrer le fichier avec le même encodage que celui que l'on annonce ! **Nous vous recommandons fortement de configurer votre éditeur afin qu'il utilise UTF-8 comme encodage par défaut.** Pour les utilisateurs d'emacs, utf-8 devrait être déjà configuré. Vérifiez que l'unicode est bien activé en bas à gauche de la fenêtre, où un U doit apparaître. Si ce n'est pas le cas, il suffit de rajouter dans le fichier .emacs :

```
(prefer-coding-system 'utf-8)
(set-terminal-coding-system 'utf-8)
(set-keyboard-coding-system 'utf-8)
```

Les autres éditeurs doivent être facilement configurable à partir des menus.

Exercice 13 — Réparer les accents cassés Visitez à nouveau accueil.html, et observez l'affichage en fonction de l'encodage choisi dans votre navigateur. (Avec Firefox : dans le menu Affichage > Encodage des caractères.)

Il arrive parfois sur Internet d'avoir une page qui est servie avec le mauvais encodage. Vous savez maintenant comment demander à votre navigateur de le corriger.

Exercice 14 Créez un fichier accents.html contenant le titre de niveau 1 “Quelques accents et caractères spéciaux en français” et le paragraphe suivant : “On ne l'admire peut-être pas, mais elle a quelque chose d'idéal et de charmant qui force à s'en occuper. Juliane de Krüdener, *Valérie*, 1804”. Tapez les caractères accentués au clavier.

- Dans les options de votre éditeur de texte préféré, modifiez l'encodage à l'enregistrement, mettez-le en utf-8, afin d'enregistrer accents.html en utf-8. Par ailleurs, insérez la balise meta annonçant l'encodage utf-8.
- Copiez accents.html dans accents2.html. Essayez d'enregistrer accents2.html en koi8-r. Votre éditeur devrait râler, car vous allez corrompre vos données.
- Revenez maintenant à accents.html. Dans l'en-tête de votre page, modifiez l'attribut charset en remplaçant utf-8 par koi8-r. Que se passe-t-il dans le navigateur ?
- Refaites la même manipulation avec iso-8859-7 et iso-8859-15...

Vous savez maintenant faire des pages qui peuvent contenir n'importe quels caractère à l'exception de &, <, et > qui sont spéciaux en HTML. (Nous apprendrons comment faire ces caractères dans un autre TP.)

8 Pour aller plus loin : Tableaux, navigation sur un site web

Le but de cet exercice est de créer un site web sur le cinéma. Vous allez voir comment organiser votre site et comment naviguer à l'intérieur du site, ainsi que comment on fait des tableaux.

Exercice 15 — Liens relatifs Pour qu'un site soit bien organisé, on peut créer des sous-dossiers dans lesquels seront stockés les différentes pages du site et les documents (images, vidéos, sons...) qui y apparaissent.

Créez un répertoire Films-IO2.

Vous trouverez sur la page Didel du cours une archive Films-IO2.zip contenant tous les documents dont vous aurez besoin :

- Les pages du site : index.html, films.html, stars.html ;
- Les affiches des films ;
- Les photos des acteurs et actrices apparaissant dans ces films.

Organisez tous ces documents dans des sous-dossiers de votre répertoire public_html : Films et Stars. Mettez le fichier index.html dans le dossier Films-IO2.

Visualisez la page index.html dans votre navigateur, à partir de l'adresse du serveur du script. Quelle est son URL complète ?

Corrigez les liens dans les trois pages du site en tenant compte de la nouvelle arborescence.

Exercice 16 — Tableaux Complétez les tableaux contenus dans les pages films.html, stars.html en vous aidant des données contenues dans description.txt. Enfin, validez vos 3 pages.

9 Pour aller plus loin : problèmes de compatibilité.

Exemple du HTML 5 et des anciens navigateurs. Quand on écrit une page HTML, il faut avoir en tête qu'elle devra être lisible par la plupart des navigateurs en circulation.

HTML 5 est un langage récent et seules les versions les plus modernes des navigateurs vont comprendre certaines des fonctionnalités du langage (Firefox \geq 3.6, Safari \geq 4.0, Chrome \geq 4, Internet Explorer \geq 9, *etc.*).

Heureusement, la plupart des fonctionnalités que l'on a présenté assurent la compatibilité avec les anciens navigateurs.

- La balise `<meta charset="utf-8">` fonctionne avec les anciens navigateurs alors qu'elle n'existait pas avant HTML 5. Par quel concours de circonstances ?

D'autres fonctionnalités, comme les balises sémantiques, ne sont pas du tout prises en compte par les vieilles version d'Internet Explorer notamment (on ne peut pas leur y appliquer de style, par exemple, car la balise n'est tout simplement pas lue).

On trouve sur internet différentes astuces pour restaurer une certaine compatibilité au moyen de scripts JavaScript. À quoi peuvent servir les scripts suivants pour un développeur HTML ?

- Html5shiv (<http://code.google.com/p/html5shiv/>)
- Modernizr (<http://www.modernizr.com/>)

Malgré tout, il faut garder en tête qu'une proportion non nulle des internautes a le JavaScript désactivé.