
Rudiments de HTML5

I. Rappels sur le HTML et les CSS :

I.1. Les fondations :

HTML → fond (textes, images, etc., et leurs structures)

CSS → forme (décoration de l'affichage)

La différenciation du fond et de la forme est fondamentale.

I.2. Le HTML :

Un dérivé des langages de type XML utilisé pour décrire des documents *hypertextes* sur le web.

I.2.1. Il contient le fond :

- contenu
- structures (regroupements du fond)
- meta-données

I.2.2. On décrit ces éléments par :

- des balises (encadrent les structures) `<div>contenu</div>`
- des attributs (métadonnées des structures) `<div id="important">contenu</div>`

Ce qui donne une structure arborescente appelée DOM (Document Object Metadata)

I.2.3. Exercice 1 :

Tracez au brouillon l'arbre de l'exemple de CV en ligne fourni dans Moodle (la racine de l'arbre est `<html>`)

Vous pouvez utiliser <http://framindmap.org/>

I.3. Les CSS:

Des feuilles de styles qui décrivent la forme de l'affichage des structures HTML.

Grâce à :

1.3.1. un système de sélection (sélecteurs)

- Reprend les noms des balises p,div,a, etc.
- cible en fonction des attributs (métadonnées) p.premier_para, div#intro

voir aussi : <http://code.tutsplus.com/fr/tutorials/the-30-css-selectors-you-must-memorize--net-16048>

1.3.2. un système de description (bloc de règles)

Qui permet de fixer la valeur de chaque propriété de la structure sélectionnée

voir par exemple : <http://fr.openclassrooms.com/informatique/cours/apprenez-a-creer-votre-site-web-avec-html5-et-css3/memento-des-proprietes-css>

1.3.3. un système de cascade le long de l'arbre du DOM

- les propriétés des éléments pères (proches de la racine) sont héritées par les enfants (proches de feuilles)
- les parents n'héritent pas des enfants
- les dernières définitions écrasent les premières

1.3.4. Exercice 2 :

0. Créez une feuille de style vide nommée "style.css" à placer au même niveau d'arborescence que l'exemple de CV
1. Modifiez en CSS la couleur et la police de tous les titres de niveau 2
2. Modifiez la position et la forme du bloc de menu
3. Changez la taille de la police du titre du menu

1.4. Quelques défis contemporains du développement web

Il s'agit des défis que nous allons aborder dans ces séances de TP.

- faire des sites web vite (industrialiser)
- faire des sites web jolis
- faire des sites web utilisables
- faire des sites web accessibles (handicaps, ordinateurs/smartphones/tablettes, robots d'indexation, etc.)
- réutiliser les éléments déjà conçus

1.5. Quelques outils pour surmonter ces défis :

- l'anglais (pour lire, sur le web, les documentations et les Q&A → <https://stackoverflow.com/>)
- un bon éditeur de texte + zencoding (voir ZenCodingCheatSheet.pdf dans Moodle)
- le HTML5
- les Frameworks HTML/CSS (type Bootstrap)
- les outils d'analyse des (bons) navigateurs récents
- les validateurs de code (ex. : <http://validator.w3.org/> ou <https://desktop.opquast.com/fr/>)
- le javascript et ses bibliothèques

II. HTML 5

II.1. Qu'est-ce ?

Est une solution à de nombreux problèmes de développement et de qualité des pages web, mais aussi au développement d'applications web (exécutées par le navigateur).

Le web fourmille de descriptions du HTML5 et des différences avec le HTML4 (le plus utilisé auparavant).

Voir par exemple : <http://www.alsacreations.com/article/lire/750-HTML5-nouveautes.html>

II.2. On peut retenir :

- la description du langage (Doctype) est différente :

- en HTML 4 transitional :
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd" >
- en HTML5 :
<!DOCTYPE html>

C'est le doctype qui indique au navigateur à quel type de description il a à faire (XML, XHTML, HTML 1, 2 3 ou 4, etc.)

- des balises plus sémantiques pour définir les structures d'une page HTML :

- <header>
- <nav>
- <section>
- <article>
- <aside>
- <figure>
- <figcaption>
- <footer>
- <details>
- <summary>
- <mark>
- <time>

Voir : http://www.w3schools.com/html/html5_semantic_elements.asp

- une intégration multimédia plus poussée grâce à de nouvelles balises dédiées (correspondant à des API spécifiques des navigateurs) : <canvas>, <video>, <audio>

II.3. Exercice 3 :

1. faites reconnaître aux navigateurs que l'exemple de CV est désormais en HTML5
2. modifiez le menu pour qu'il utilise plutôt `<nav>`
3. Modifiez l'image pour qu'elle devienne une `<figure>`
4. Découpez la page en deux `<section>`s : "qui_suis_je" et "contact". Avec chacune un `<header>`
5. vérifiez que vos CSS sont à jour !
6. validez votre page sur <http://validator.w3.org/>

III. les compagnons de HTML5 :

CSS3 et Javascript sont souvent des compagnons implicites de HTML5.

III.1. CSS3 :

CSS3 est une version avancée de CSS2 qui permet notamment de gérer des animations CSS.

Voir par exemple :

- <http://www.alsacreations.com/actu/lire/919-ce-que-nous-reserve-css3-en-2010.html>
- http://www.w3schools.com/css/css3_intro.asp

III.2. Javascript :

C'est un langage de programmation qui est exécuté par le navigateur. Il permet notamment de manipuler les éléments du DOM pour créer de l'interaction avec l'utilisateur, sans passer par un serveur web.

IV. Quelques signets pour son auto-formation

Voici quelques idées pour travailler sur ces problématiques avec des jeux sérieux.

- Apprendre à programmer (grands débutants) avec Blockly Maze : <https://blockly-demo.appspot.com/static/apps/maze/index.html>
- Progresser plus loin en programmation (même base que Blockly) : <http://studio.code.org/s/course1>
- Le HTML5 en s'amusant : <http://www.chtemele.org/>

Mais aussi des plateformes d'e-learning :

- Kahn Academy est une plateforme d'apprentissage en ligne qui propose un parcours sur le javascript : <https://fr.khanacademy.org/computing/cs>
- Code Academy propose un site interactif d'apprentissage des technologies web : <http://www.codecademy.com/en/tracks/web>

Le web fourmille d'autres ressources (comme l'excellent <http://www.newdzign.org/> de l'un de vos collègues !)...