

Initiation au développement Web – Partie 0

Introduction

Laure Petrucci

Laure.Petrucci@univ-paris13.fr

<http://www.lipn.fr/~petrucci/M1106/>

Version : 6 janvier 2021 (diapositives à trous)



Compétences visées

- Être capable de formater et de modifier des documents dans un langage normalisé de description de contenus
- Être capable de mettre en œuvre des feuilles de styles
- Être capable de concevoir un site Web adaptatif

Objectifs

- Découvrir le contenu et le contenant d'une **page Web** qui présente des informations sur Internet pour **PC, tablette et téléphone intelligent**
- Découvrir les notions de base sur le fonctionnement du **modèle client-serveur**
- Savoir **écrire et modifier des pages Web** dans un langage normalisé de description de contenus
- Savoir modifier les **attributs de présentation** d'une page Web
- Savoir **adapter** un contenu Web pour **navigateur mobile**

Organisation du module

Volume horaire

Cours	2×1h30
Travaux pratiques	4×3h
Examen sur table	3h

Vos enseignants en travaux pratiques

- L. Petrucci
- M. Sow

Évaluation

- Tous travaux pratiques notés
- Examen final sur table

Organisation pratique

- Support de cours : diapositives
- Sujets de TP et diapositives (avec les trous!) accessibles **sous forme électronique en ligne**

Adresse des ressources :

<http://www.lipn.fr/~petrucci/M1106/>

Licence

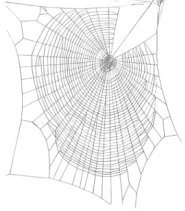
Plan du cours

- 1 Architecture Client-serveur
- 2 HTML
- 3 CSS
- 4 Sites Web adaptatifs

Source des images utilisées (1/1)



Titre : Zygiella web
Auteur : Laura Bassett
Source : https://commons.wikimedia.org/wiki/File:Zygiella_web.jpg
Licence : attribution



Initiation au développement Web – Partie 1

Architecture client-serveur

Laure Petrucci

Laure.Petrucci@univ-paris13.fr

<http://www.lipn.fr/~petrucci/M1106/>

Version : 6 janvier 2021 (diapositives à trous)



Plan: Architecture client-serveur

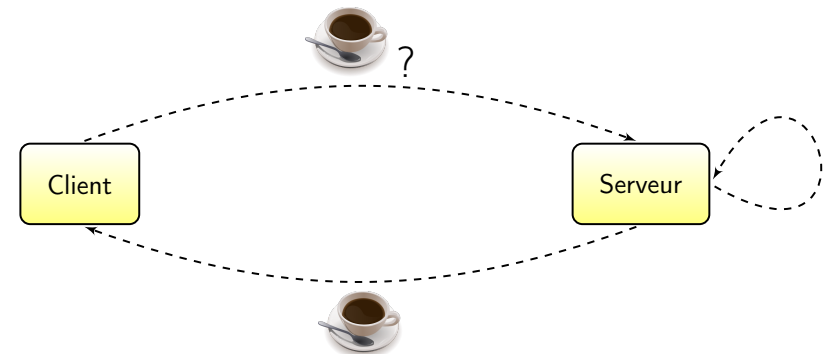
- 1 Architecture client-serveur
- 2 Client-serveur et Web
- 3 Une autre architecture

Partie 1: Architecture client-serveur – Plan

- 1 Architecture client-serveur
- 2 Client-serveur et Web
- 3 Une autre architecture

L'analogie du café

Client : « Bonjour, je voudrais un café. »
(préparation du café)
Serveur : « Bonjour, voici votre café. »



Définition

Modèle apparu dans les années 1970 (Xerox PARC)

Définition (Serveur)

Un **serveur** est un ordinateur (et/ou un programme informatique) offrant un **service** ou une **ressource** sur un réseau.

Définition (Client)

Un **client** est un programme informatique contactant un serveur via un réseau, afin de bénéficier d'un **service** ou d'une **ressource**.

Applications du modèle client-serveur

- Serveur de **courrier électronique**
 - ▶ Protocoles courants :

- Serveurs de **pages Web**
 - ▶ Protocole : généralement

Communication client-serveur

Le client et le serveur communiquent via un

Les protocoles clients-serveurs se situent au niveau de la couche

Plusieurs clients peuvent se connecter à un même serveur.



Plan: Client-serveur et Web

- 1 Architecture client-serveur
- 2 Client-serveur et Web
- 3 Une autre architecture

Serveurs Web : matériel

Serveur Web : ordinateur affecté au stockage des pages Web et au traitement des requêtes provenant des clients



Le premier serveur Web



Salle de serveurs (2009)

Pour les petits sites Web : un serveur **partagé** (ou mutualisé), affecté à plusieurs sites Web

Pour les grands sites Web : un ou plusieurs serveurs **dédiés** à ce site

- Google reposerait sur près d'**un million de serveurs** dans le monde

Serveurs Web : logiciel

Systèmes d'exploitation courants :

- Variantes de **Linux** (67%)
- Windows Server (Microsoft)
- Plus rarement : OSX (Apple)

Serveur HTTP courant :

- Apache

Logiciels courants :

- MySQL (base de données), PHP (pages dynamiques)

Un exemple de client : le

Le client est le

- Il supporte au minimum le protocole

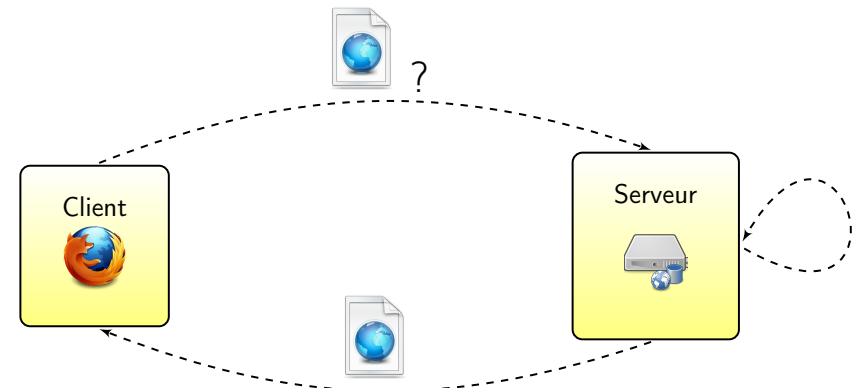
Clients les plus courants aujourd'hui

Nom	Depuis	Licence
Android (navigateur)	2008	GNU GPL
Chromium	2008	BSD (etc.)
Google Chrome	2008	propriétaire
Internet Explorer (et Edge)	1995	propriétaire
Mozilla Firefox	2002	MPL
Opera	1995	propriétaire
Safari	2003	propriétaire

Requête client-serveur

Client : « Bonjour, je voudrais `www.univ-paris13.fr` »
(préparation de la page Web)

Serveur : « Bonjour, voici la page Web »



Protocole HTTP et URL

Protocole

- *Hypertext Transfer Protocol*
- Port
- Couche du modèle Internet

Basé sur les adresses Web ou URL

- *Uniform Resource Locator*
- Transformées en par un système de résolution de noms
 - ▶ `www.univ-paris13.fr` ⇒ `194.254.164.6`

Plan: Une autre architecture

- 1 Architecture client-serveur
- 2 Client-serveur et Web
- 3 Une autre architecture

Avantages et inconvénients du modèle client-serveur

- ☺ Calculs à la charge du serveur
 - ▶ Création de pages dynamiques (sauf Javascript)
 - ▶ Recherche d'information dans des bases de données immenses (moteurs de recherche)
- ☺ Client léger possible
 - ▶ Fonctionnant sur téléphone ou tablette
- ☹ Disponibilité du serveur : **point critique**
 - ▶ « Comment faire si l'on veut redémarrer le serveur correspondant à `www.google.fr` ? »
 - ▶ Exemple : OVH garantit une disponibilité de 99,9%
 - ★ Ce qui fait tout de même près de (Imaginons les conséquences d'une telle panne le samedi avant Noël pour un site de vente en ligne)
 - Cf. double panne géante d'OVH du 9 novembre 2017

Une autre architecture : réseaux pair à pair (1/3)

Principe

Dans un réseau pair à pair (*peer to peer*), chaque client peut également jouer le rôle de **serveur**.

Deux modes :

- Via un serveur central
- Entièrement décentralisé

Une autre architecture : réseaux pair à pair (2/3)

Comparaison avec l'architecture client-serveur :

- **Client-serveur**

- ▶ Plus une ressource est populaire, et **moins elle est accessible**
 - ★ Risque de surcharge du serveur

- **Pair à pair**

- ▶ Plus une ressource est populaire, et **plus elle est accessible**
 - ★ Disponibilité grandissante de la ressource



Sources et références

Une autre architecture : réseaux pair à pair (3/3)

Applications :

- **Partage de fichiers**

- ▶ BitTorrent, eMule

- **Calcul distribué**

- ▶ Recherche de nouveaux nombres premiers de Mersenne (GIMPS, *Great Internet Mersenne Prime Search*)
- ▶ Recherche d'intelligence extra-terrestre (SETI@home)

Sources et références

Sources

- Disponibilité des serveurs OVH

https://www.ovh.com/fr/hebergement-web/hebergement_web_haute_disponibilite.xml

- Répartition des SE des serveurs

Usage Statistics and Market Share of Unix for Websites. W3Techs. November 2013.

<http://w3techs.com/technologies/details/os-unix/all/all>

Licence

Source des images utilisées I



Titre : Zygiella web
Auteur : Laura Bassett
Source : https://commons.wikimedia.org/wiki/File:Zygiella_web.jpg
Licence : attribution



Titre : An SVG rendering of cup of coffee
Auteur : Julius Schorzman / Peewack
Source : <https://commons.wikimedia.org/wiki/File:Cup-o-coffee-simple.svg>
Licence : CC BY-SA



Titre : Modèle Client-Serveur
Auteur : David Vignoni / Calimo
Source : <https://commons.wikimedia.org/wiki/File:Modèle-client-serveur.svg>
Licence : GNU LGPL



Titre : First Web Server
Auteur : Coolcaesar
Source : https://commons.wikimedia.org/wiki/File:First_Web_Server.jpg
Licence : CC BY-SA



Titre : Server room in CERN (France / Switzerland)

Source des images utilisées II

Auteur : Florian Hirzinger
Source : https://commons.wikimedia.org/wiki/File:CERN_Server_02.jpg
Licence : CC BY-SA



Titre : Logo officiel de Mozilla Firefox 3.5 à 22.0
Auteur : Mozilla Foundation
Source : https://commons.wikimedia.org/wiki/File:Mozilla_Firefox_3.5_logo_256.png
Licence : MPL / GNU GPL / GNU LGPL



Titre : Server-web-database
Auteur : RRZEicons
Source : <https://commons.wikimedia.org/wiki/File:Server-web-database.svg>
Licence : CC BY-SA



Titre : Vista Icons Toolbar
Auteur : VistalCO.com
Source : https://commons.wikimedia.org/wiki/File:1328101978_Web-page.png
Licence : CC BY



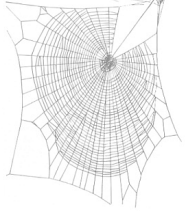
Titre : Server-based computer network
Auteur : Mauro Bieg
Source : <https://commons.wikimedia.org/wiki/File:Server-based-network.svg>

Source des images utilisées III

Licence : GNU LGPL



Titre : Peer-to-peer computer network
Auteur : Mauro Bieg
Source : <https://commons.wikimedia.org/wiki/File:P2P-network.svg>
Licence : Domaine public



Initiation au développement Web – Partie 2

HTML

Laure Petrucci

Laure.Petrucci@univ-paris13.fr

<http://www.lipn.fr/~petrucci/M1106/>

Version : 6 janvier 2021 (diapositives à trous)



Plan: Définitions

- 1 Définitions
- 2 Histoire et normalisation du HTML
- 3 Structure d'un document HTML5
- 4 Syntaxe de HTML5

Partie 2: HTML – Plan

- 1 Définitions
- 2 Histoire et normalisation du HTML
- 3 Structure d'un document HTML5
- 4 Syntaxe de HTML5

Internet

Définition (Internet)

Internet est donc un **réseau de réseaux**.

Orthographe recommandée : **Internet**
(Variantes : l'Internet, l'internet, internet, les Internets)

World Wide Web

Définition (World Wide Web)

Le **World Wide Web** est un système fonctionnant au-dessus d'Internet, et basé sur des . Il est constitué de organisées en lesquelles sont consultables grâce à un .

Autres noms :

Remarque :

- Le World Wide Web n'est qu'une application d'Internet (aux côtés d'autres comme le)

HTML : remarques

Remarque

HTML ne doit pas être confondu avec :

- **PHP** : langage dont une application possible est d'être utilisé côté serveur pour générer du code HTML
- **(My)SQL** : langage de requêtes d'accès à une base de données dont une application possible est d'être utilisé côté serveur, par exemple avec PHP, pour générer du code HTML
- **JavaScript** : langage orienté objet pouvant être utilisé côté client pour générer du code HTML
- **JQuery** : bibliothèque libre de JavaScript
- **Ajax** : architecture combinant notamment JavaScript, XML et CSS pour construire des applications Web

HTML

Définition (HTML)

HTML (HyperText Markup Language), ou langage de balisage d'hypertexte, est .

Hyperlien : permet de passer d'un document à un autre .

HTML est aujourd'hui *de facto* le seul langage de description de pages Web.

Plan: Histoire et normalisation du HTML

- 1 Définitions
- 2 Histoire et normalisation du HTML
 - Historique
 - Un langage standardisé
- 3 Structure d'un document HTML5
- 4 Syntaxe de HTML5

Au commencement : le texte

: nécessité de publier, partager et retrouver des documents

Le titre de mon document

Un paragraphe dans mon document.

Un autre paragraphe qui mentionne une autre page Web.

Les balises

1991 : proposition de **balises**

```
<TITLE>Le titre de mon document</TITLE>
```

```
Un paragraphe dans mon document.
```

```
<P>
```

```
Un autre paragraphe qui mentionne une
```

```
<A HREF="http://www.cern.fr">autre page Web</A>.
```

(Code évidemment non conforme à la norme HTML actuelle.)



Tim Berners-Lee

Les liens hypertextes

Fin des années 1980 : les **liens hypertextes**

- Permet de pointer vers une **autre page**

Le titre de mon document

Un paragraphe dans mon document.

Un autre paragraphe qui mentionne une

```
<A HREF="http://www.cern.fr">autre page Web</A>.
```

(Code évidemment non conforme à la norme HTML actuelle.)

L'influence des navigateurs

1991–1993 : en l'absence de standard, ce sont (Mosaic, Netscape) qui « décident » *de facto* de l'évolution de HTML

Ajouts notables des navigateurs :

- les images (balise ``)
- diverses balises qui seront « supprimées » par la suite (texte clignotant, centré, etc.)

```
<TITLE>Le titre de mon document</TITLE>
```

```
Un paragraphe dans mon document.
```

```
<P>
```

```
<BLINK>Du texte clignotant</BLINK>
```

```
<CENTER>Du texte centré</CENTER>
```

```
<IMG SRC="mon_image.gif">
```

(Code évidemment non conforme à la norme HTML actuelle.)

HTML 2.0 : la première spécification

- 1995 : spécification de HTML 2.0 par l'**IETF**
 - ▶ *Internet Engineering Task Force* (Détachement d'ingénierie d'Internet), qui élabore les standards d'Internet
- 1996 : prise en main par le **W3C** (World Wide Web Consortium)
 - ▶ Consortium (421 organisations membres en 2016) qui élabore les standards du Web
- 1997 : publication de **HTML 3.2** par le W3C (sous forme de recommandation)

HTML5 : le nouveau standard

- 2008 : publication par le W3C d'un **premier brouillon** (*draft*) de la spécification de
 - ▶ Début de la création de pages Web conformes à la pré-spécification HTML5
- 2011 : **dernier appel** pour contribuer à la spécification
- 28 octobre 2014 : publication du **standard HTML5** comme recommandation W3C

Ce cours se concentre sur **HTML5**.

HTML 4.01 : la stabilité

- 1997–1998 : publication de **HTML 4.01** par le W3C (sous forme de recommandation)
- 3 variations
 - ▶ *Strict* : éléments obsolètes interdits
 - ▶ *Transitional* (transitoire) : éléments obsolètes autorisés
 - ▶ *Frameset* (cadre) : assemblage d'éléments de type cadre
- Les **éléments obsolètes** sont des éléments autorisés par les navigateurs Web de l'époque, mais destinés à être remplacés par les feuilles de style
- 1999 : publication par le W3C de **HTML 4.01**, plus récente version du standard

L'intérêt du respect du standard

- **Interopérabilité, compatibilité** entre navigateurs
- **Meilleur référencement**
 - ▶ Google favoriserait le référencement des sites Web dont le code HTML est conforme aux standards

Limite du respect du standard

Limites de l'interopérabilité

Une page Web peut ne pas s'afficher de la même façon selon les navigateurs, même si elle respecte entièrement le standard HTML5, et même si les navigateurs respectent eux aussi le standard.

Il convient donc d'en **tester l'affichage** sur les principaux navigateurs.

Néanmoins, respecter le standard limite les différences potentielles d'affichage !

Valdateur du W3C

Un outil indispensable

- le **valdateur** du W3C et de la fondation Mozilla

<http://validator.w3.org/>

The screenshot shows the W3C Markup Validation Service interface. At the top, it says "W3C Markup Validation Service" and "Check the markup HTML, XHTML, ... of Web documents". Below that, there are links for "Jump To: Notes and Potential Issues" and "Congratulations - Icons". A green banner states "This document was successfully checked as HTML5!". The "Result:" section shows "Passed, 2 warning(s)". The "Source:" section displays the HTML code:


```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Le titre de mon document</title>
  </head>
  <body>
    Ce document est constitué d'une seule phrase.
  </body>
</html>
```

 The "Encoding:" is set to "utf-8", "Doctype:" is "HTML5", and "Root Element:" is "html". At the bottom, there is a logo for "W3C VALIDATOR Suite" and a note: "Try now the W3C Validator Suite™ premium service that checks your entire website and evaluates its conformance with W3C open standards to quickly identify those portions of your website that need your attention."

Validation du code HTML

À la différence de langages de programmation, un non-respect de la syntaxe de HTML n'empêchera pas l'affichage sur le navigateur.

Plus généralement, le navigateur affichera quelque chose (de non spécifié) ou n'affichera rien, mais **n'affichera jamais d'erreur**.

Validation

Il faut impérativement **valider** son code HTML avant de le publier.

Plan: Structure d'un document HTML5

- 1 Définitions
- 2 Histoire et normalisation du HTML
- 3 Structure d'un document HTML5
 - Principe des balises
 - En-tête
- 4 Syntaxe de HTML5

Un langage à base de balises

HTML repose sur la notion de *(tags)*

Deux types de balises :

- ❶ Les balises qui sont ouvertes puis fermées, et encadrent du **contenu**
 - ▶ Exemple : `Un peu d'italique`
- ❷ Les balises qui s'ouvrent et se ferment en même temps
 - ▶ Exemple : ``

```
<h1>Un exemple de titre</h1>
<hr />
<div>
  <p>
    Du texte <b>gras</b>, puis <em>italique</em>,
    puis <b><em>gras et italique</em></b>.
  </p>
  
</div>
```

Balises et attributs

Certaines balises peuvent posséder des **attributs**.

Exemples :

- `<meta charset="utf-8" />`
- ``
- `<p style="text-align: right">
 Du texte aligné à droite
</p>`

Balises et hiérarchie

Les balises sont structurées de façon **hiérarchique**

Chaque balise **hérite** des propriétés de la balise de niveau supérieur – à moins qu'elles ne les redéfinissent.

```
<p>Ce texte est <b>seulement en gras</b>.</p>
<p>Ce texte est <b>seulement en gras <i>et
  cette partie est en gras et en italique</i></b>.</p>
```

Ordre de fermeture

Il est impératif de fermer les balises dans l'ordre

Commentaires

Les commentaires sont du code HTML qui n'est pas exécuté, donc

Syntaxe : `<!-- Texte entre commentaires -->`

```
<h1>Un titre (qui va s'afficher)</h1>
<p>Cette phrase va s'afficher sur le navigateur.
  <!-- En revanche, cette phrase ne s'affichera pas -->
</p>
```

Attention

Les commentaires restent visibles dans la **source** de la page Web, qui est accessible depuis le client.

- Attention à ne pas y laisser d'informations confidentielles!

Structuration d'un document HTML

Deux grandes parties :

- **En-tête** : définition du titre, du codage, des mots-clés, etc.
 - ▶ Information destinée aux machines (navigateur, robots, etc.)
- **Corps** : contenu de la page Web
 - ▶ Information destinée à l'humain (et aux machines)

```
<!DOCTYPE html>
<html>
  <head>
    <!-- Ici, l'en-tête -->
  </head>

  <body>
    <!-- Ici, le corps de la page -->
  </body>
</html>
```

Un exemple minimal (syntaxe conforme à HTML5)

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>Le titre de mon document</title>
  </head>

  <body>
    <p>Ce document est constitué d'une seule phrase.</p>
  </body>
</html>
```

Document Type Definition

Un document conforme à la syntaxe de `<!DOCTYPE html>` commence nécessairement par

```
<!DOCTYPE html>
```

Il s'agit de la **définition du type du document** (DTD), qui décrit un modèle de document XML ou SGML.

Valeurs pour d'anciennes versions de (X)HTML :

- HTML 4.01 strict

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
  "http://www.w3.org/TR/html4/strict.dtd">
```

- XHTML 1.1

```
<!DOCTYPE html PUBLIC
  "-//W3C//DTD XHTML 1.1//EN"
  "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
```

La balise `title` (1/2)

Située dans l'**en-tête** du document

- Entre les balises `<head>` et `</head>`

Seule **balise obligatoire** de l'en-tête

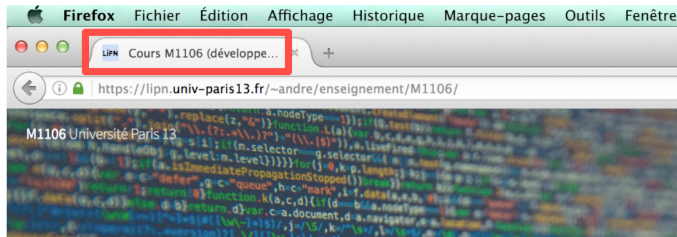
Syntaxe :

```
...
  <head>
    <title>Le titre de ma page Web</title>
  </head>
...
```

La balise `title` (2/2)

Son contenu définit

- Auparavant systématiquement affiché dans le **titre de la fenêtre du navigateur** ou de l'onglet
- De moins en moins aujourd'hui
- Améliore grandement le **référencement**



Les balises `meta`

Situées dans l'**en-tête** du document

- Entre les balises

Ordre non spécifié

Permettent de donner des informations au **navigateur** et aux **moteurs de recherche**

- Codage des caractères
- Auteur, description, mots-clés de la page

Ne s'affichent jamais directement sur la page

Codage des caractères

Syntaxe :

```
<meta charset="NOM_DU_CODAGE" />
```

Valeurs possibles

: unicode (**recommandé**)

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

: alphabet latin

- et bien d'autres

La présence de cette balise n'est pas obligatoire selon le W3C, mais **très très fortement recommandée** (comprendre : obligatoire).

Liste complète des valeurs :

<https://www.iana.org/assignments/character-sets/character-sets.xhtml>

Description, mots-clés, auteur

Syntaxe :

```
<meta name="description" content="Introduction
au développement Web" />
<meta name="keywords" content="cours ,HTML, IUT de
Villetaneuse, M1106" />
<meta name="author" content="Le prof" />
```

Important

Remplir ces champs est important, non pour le client, mais pour le référencement dans les moteurs de recherche.

Un exemple d'en-tête

```

<!DOCTYPE html>
<html lang="fr">
  <head>
    <meta charset="utf-8" />
    <meta name="description" content="Une page Web
      vraiment intéressante" />
    <meta name="keywords" content="cours ,HTML, IUT de
      Villetaneuse ,M1106" />
    <meta name="author" content="Le prof" />
    <title>Le titre de ma page Web</title>
  </head>

  <body>
    ...
  </body>
</html>

```

En-tête : bonnes pratiques

Bonnes pratiques

Il est très fortement conseillé de toujours spécifier, en plus de la balise `<title>` :

- le codage des caractères,
- l'auteur,
- la description,
- les mots-clés.

Plan: Syntaxe de HTML5

- 1 Définitions
- 2 Histoire et normalisation du HTML
- 3 Structure d'un document HTML5
- 4 Syntaxe de HTML5
 - Structure de la page
 - Styles de texte
 - Listes et énumérations
 - Tableaux
 - Formulaires
 - Médias

Sections

Balise `<div>` : élément structurel (ou section)

Bonnes pratiques

Il est d'usage d'organiser la page Web en `<div>` imbriquées.

```

<div>
  <div>
    <!-- Une première partie -->
  </div>
  <div>
    <!-- Une seconde partie -->
  </div>
</div>

```

Paragraphes

Balise `<p>` : paragraphe

Balise `
` : saut de ligne

Bonne pratique

Il est d'usage de créer un second paragraphe (`<p>`) plutôt qu'un saut de ligne (`
`).

```
<div>
  <p>Un premier paragraphe.</p>
  <p>Un second paragraphe.</p>
</div>
```

Espaces

Les espaces affichées en HTML s'affichent également sur le navigateur.

- Attention donc à ne pas multiplier les espaces !

En revanche, les sauts de ligne en HTML deviennent des espaces sur le navigateur.

- Pour afficher un saut de ligne, utiliser

```
<p>Cette phrase va s'afficher
sur une seule ligne.<br />Cette phrase sera dessous.</p>
```

Cette phrase va s'afficher sur une seule ligne.

Espaces insécables

Les **espaces insécables** s'affichent en utilisant

- Pour « *non-breakable blank space* »

Usage en français :

- Après «
- Avant ? ! ; : »

« Comment allez-vous ? », demanda-t-elle.

Quelques caractères spéciaux

Certains caractères sont **interdits** dans le code HTML (en dehors des balises)

Il faut donc utiliser un codage du caractère :

```
: &lt; ;
: &gt; ;
: &amp; ;
```

Exemple :

Le DUT R&T le reste du monde

Le DUT R T le reste du monde

Les titres

Syntaxe :

: titre de premier niveau

: titre de second niveau

- ...

: titre de sixième niveau

Bonnes pratiques

Il est d'usage de n'avoir qu'un seul titre dans le niveau courant, et de respecter

Les titres : exemple

```
<h1>Le titre général</h1>
<h2>Une première grande section</h2>
<p>Un texte d'introduction</p>
<h3>Un sous-titre dans la première section</h3>
<h3>Un autre sous-titre dans la première section</h3>
<h2>Une seconde grande section</h2>
```

affichera

Le titre général
Une première grande section
 Un texte d'introduction
Un sous-titre dans la première section
Un autre sous-titre dans la première section
Une seconde grande section

Les liens hypertextes externes

Syntaxe :

Affiche

[Le texte à afficher](#)

Deux types de liens :

- Liens **relatifs**

- ▶ Chemin local

- ▶ `Texte`

Pointe sur la page [index.html](#) dans le répertoire courant

- Liens **absolus**

- ▶ Chemin absolu

- ▶ `Texte`

Les liens hypertextes internes

On peut effectuer un lien vers **une autre partie du même document**

Syntaxe pour définir l'endroit dans le document (destination)

- À l'aide de `id`
- Exemple :

Syntaxe pour définir le lien

Affiche

Mes films préférés
 ...
[Remonter aux films préférés](#)

Les liens hypertextes vers une adresse électronique

On peut effectuer un lien vers **une adresse électronique**

Syntaxe :

Affiche

[Écrivez-moi](#)

Attention

Cette utilisation est fortement déconseillée, du fait de la présence sur Internet de nombreux **robots malveillants** qui récupèrent les adresses électroniques en clair pour envoyer des **pourriels**.

Un exemple un peu plus fourni

```

<!DOCTYPE html>
<html>
  <head>
    <title>Le titre de mon document</title>
    <meta charset="utf-8" />
    <meta name="author" content="L'auteur de la page" />
  </head>

  <body>
    <h1>Le titre de la page</h1>
    <p>Une phrase avec <a href="index.html">un lien</a>.</p>
    <h2>Une section</h2>
    <p>Une autre phrase.</p>
    <h2>Une seconde section</h2>
    <p>Encore une autre phrase.</p>
  </body>
</html>

```

Nouvelles balises HTML5

Peuvent être vues comme des variations **sémantiques** de

- Pas d'incidence sur l'affichage ou la mise en page
- Peuvent être **imbriquées**

Syntaxe :

- : un bloc de contenu cohérent
- : un article au contenu cohérent
- : contenu accessoire
- : en-tête (de page, de section, etc.)
- : pied (de page, de section, etc.)
- : informations de navigation (liens)

Nouvelles balises HTML5 : exemple

```

...
<header>Le haut de la page</header>
<article>
  <h2>Le titre de mon article</h2>
  <header>Un chapeau pour mon article</header>
  <p>Premier paragraphe.....</p>
  <p>Second paragraphe.....</p>
  <footer>Infos finales sur mon article</footer>
  <aside>Voir un autre article sur le même sujet</aside>
</article>
<footer>
  <nav>
    <a href="index.html">Retour à l'accueil</a>
  </nav>
</footer>
...

```

Nouvelles balises HTML5 : attention

En-tête : attention aux confusions

Il convient de bien différencier :

- ❶ l'**en-tête syntaxique** du document HTML (balise `<head>` unique, contenu non affiché), et
- ❷ les **en-têtes sémantiques** dans le corps (`<body>`) du document (balises `<header>` éventuellement multiples).

Plan: Syntaxe de HTML5

- ❶ Définitions
- ❷ Histoire et normalisation du HTML
- ❸ Structure d'un document HTML5
- ❹ Syntaxe de HTML5
 - Structure de la page
 - Styles de texte
 - Listes et énumérations
 - Tableaux
 - Formulaires
 - Médias

Gras, italique, souligné

Syntaxe :

(*bold*) : **gras**

(*italique*) : *italique*

(*emphasis*) : *emphasis* (s'affiche en général comme italique)

(*underline*) : souligné

Bonne pratique

Le soulignage ne devrait jamais être utilisé en informatique, à l'exception (éventuelle) des liens hypertextes (qui tend elle-même à disparaître).

Couleurs

Syntaxe :

- Utilisé avec la balise `` au sein d'un paragraphe
 - ▶ Le grand `bleu`
- Peut également être utilisé avec
 - ▶ `<p style="color: green">Le rayon vert</p>`

Valeurs des couleurs :

- **Prédéfinies** :
 - ▶ Voir liste complète par exemple sur <http://www.standardista.com/css3/cssnamed-hsl-and-rgb-colors/>
- **Hexadécimales** : format `#RRVVBB` (rouge vert bleu)
 - ▶ Exemple : `#FFFF00` =

Styles de texte : CSS

Bonne pratique

En général, le formatage du texte est géré non par HTML mais par une feuille de style CSS.
(voir plus loin)

Plan: Syntaxe de HTML5

- 1 Définitions
- 2 Histoire et normalisation du HTML
- 3 Structure d'un document HTML5
- 4 **Syntaxe de HTML5**
 - Structure de la page
 - Styles de texte
 - **Listes et énumérations**
 - Tableaux
 - Formulaires
 - Médias

Listes à puces

Syntaxe :

```
<p>J' aime bien&nbsp;  ;</p>
<ul>
  <li>Le HTML</li>
  <li>Le CSS</li>
</ul>
```

Affichage par défaut :

J' aime bien :

- Le HTML
- Le CSS

(Le style des puces par défaut peut différer selon le navigateur.)

Listes à puces : remarques

Remarques

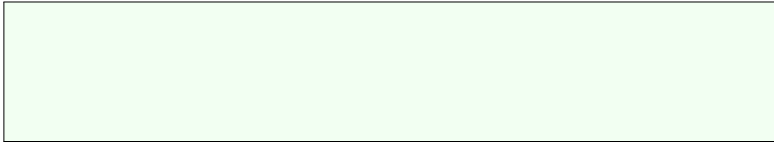
- 1 Les listes à puces peuvent être **imbriquées**.
- 2 L'affichage des listes à puces peut être considérablement personnalisé par les **feuilles de style**.
- 3 Il est d'usage d'utiliser l'environnement `` pour la création des **menus** (même si ceux-ci n'ont pas l'apparence d'une liste à puces).

Énumérations

Syntaxe :

```
<p>J'aime bien ;</p>
<ol>
  <li>Le HTML</li>
  <li>Le CSS</li>
</ol>
```

Affichage par défaut :



(Le style des puces par défaut peut différer selon le navigateur.)

Définitions

Syntaxe :

```
<dl>
  <dt>HTML</dt>
  <dd>Langage de description de pages Web</dd>
  <dt>CSS</dt>
  <dd>Langage décrivant la présentation de pages HTML</dd>
</dl>
```

Affichage par défaut :

HTML Langage de description de pages Web CSS Langage décrivant la présentation de pages HTML

Plan: Syntaxe de HTML5

- 1 Définitions
- 2 Histoire et normalisation du HTML
- 3 Structure d'un document HTML5
- 4 **Syntaxe de HTML5**
 - Structure de la page
 - Styles de texte
 - Listes et énumérations
 - **Tableaux**
 - Formulaires
 - Médias

Tableaux : exemple (1/2)

Syntaxe :

```
<table border="1">
  <tr>
    <th>Titre du film</th>
    <th>Année de sortie</th>
  </tr>
  <tr>
    <td><em>Drive</em></td>
    <td>2011</td>
  </tr>
  <tr>
    <td><em>The Neon Demon</em></td>
    <td>2016</td>
  </tr>
</table>
```

Tableaux : exemple (2/2)

Affichage :

Titre du film	Année de sortie
<i>Drive</i>	2011
<i>The Neon Demon</i>	2016

Un tableau plus complexe : syntaxe (2/2)

```
<tr>
  <td><em>Chungking Express</em></td>
</tr>
<tr>
  <td>2000</td>
  <td><em>In the Mood for Love</em></td>
</tr>
</tbody>
</table>
```

Un tableau plus complexe : syntaxe (1/2)

Syntaxe :

```
<table border="1">
  <thead>
    <tr>
      <th colspan="2">Films de Wong Kar-Wai</th>
    </tr>
    <tr>
      <th>Année</th>
      <th>Titre</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td rowspan="2">1994</td>
      <td><em>Les Cendres du temps</em></td>
    </tr>
```

Un tableau plus complexe : affichage

Affichage :

Films de Wong Kar-Wai	
Année	Titre
1994	<i>Les Cendres du temps</i> <i>Chungking Express</i>
2000	<i>In the Mood for Love</i>

Tableaux : bonne pratique

Bonne pratique

Il est d'usage de n'utiliser l'environnement `<table>` que pour les **tableaux**.

Pour aligner des éléments lors de la mise en page, on utilise les environnements `<div>` et des feuilles de style.

Plan: Syntaxe de HTML5

- 1 Définitions
- 2 Histoire et normalisation du HTML
- 3 Structure d'un document HTML5
- 4 **Syntaxe de HTML5**
 - Structure de la page
 - Styles de texte
 - Listes et énumérations
 - Tableaux
 - **Formulaires**
 - Médias

Principe des formulaires

Objectif

Transmettre des informations saisies sur la page Web courante à une **autre page Web** (qui, en général, est dynamique)

Exemples :

Un exemple de formulaire

```
<form action="traitement.php">
  <fieldset>
    <legend>Vos données personnelles</legend>
    <p><label for="nom">Nom</label>
      <input type="text" name="nom" id="nom" placeholder="
        Votre nom"></p>

    <p><label for="ddn">Date de naissance</label>
      <input type="date" name="ddn" id="ddn"></p>

    <p><label for="mdp">Mot de passe</label>
      <input type="password" name="mdp" id="mdp" placeholder="
        Votre mot de passe"></p>

    <p><button type="submit" role="button" aria-disabled="
      false">Envoyer</button></p>

  </fieldset>
</form>
```

Un exemple de formulaire : rendu

Rendu dans Mozilla Firefox 30 (sans feuille de style)

Formulaires : remarques

- Il est conseillé d'ajouter à chaque champ un `label` (étiquette décrivant le champ)
- L'identifiant `for` du `label` doit se référer à l'identifiant `id` du champ correspondant
- Ces identifiants ne doivent pas contenir d'espace, et ne seront jamais affichés
- L'attribut `placeholder` contient le texte affiché par défaut (et peut, à l'inverse, contenir tout type de caractères)
- La balise `<fieldset>` permet un rendu agréable par défaut

Formulaires : larges possibilités

Nombreux types de champs possibles :

- Cases à cocher (`checkbox`), boutons radio (`radio`), zones de texte (`textarea`), sélections (`select`), barres de boutons (`toolbar`), fichiers à téléverser (`file`), etc.
- Variantes de `text` : `tel`, `email`, `url`, `date`, `color`, etc.

Voir par exemple :

<http://dmouronval.developpez.com/> (CC BY)

Attention, certains nouveaux champs (`tel`, `date`, `color`, etc.) ne s'affichent pas toujours bien sur tous les navigateurs.

Plan: Syntaxe de HTML5

- 1 Définitions
- 2 Histoire et normalisation du HTML
- 3 Structure d'un document HTML5
- 4 Syntaxe de HTML5
 - Structure de la page
 - Styles de texte
 - Listes et énumérations
 - Tableaux
 - Formulaires
 - Médias

Images

Syntaxe :

```

```

La plupart des extensions courantes sont prises en charge par HTML5

Exemple :

```

```

Attention : texte de remplacement

Il est affiché par le navigateur si l'image n'est pas trouvée (fichier manquant), ou dans certains cas spécifiques (navigateur Web pour non-voyants).

Audio

Syntaxe :

```
<audio controls>
  <source src="yumeji_theme.ogg" type="audio/ogg">
  <source src="yumeji_theme.mp3" type="audio/mpeg">
  Votre navigateur ne prend pas en charge les formats
  audio proposés.
</audio>
```

Résultat (dans Mozilla Firefox 24) :



Les différents formats sont essayés dans l'ordre descendant, jusqu'au premier format pris en charge par le navigateur.

Audio et navigateurs

Attention

Tous les navigateurs ne prennent pas tous les formats en charge.

Navigateur	AAC	MP3	Ogg	Wav
Chrome	Oui	Oui	Oui	Oui
Chromium	Non	Oui	Oui	Oui
Firefox	Partiel	Partiel	Oui	Oui
Internet Explorer	Oui	Oui	Non	Non
Opera	Partiel	Partiel	Oui	Oui
Safari	Oui	Oui	Non	Oui

Bonne pratique

Prévoir les sons dans suffisamment de formats

Vidéo

Syntaxe :

```
<video width="640" height="480" controls>
  <source src="six_days.ogg" type="video/ogg">
  <source src="six_days.mp4" type="video/mp4">
  Votre navigateur ne prend pas en charge les formats
  vidéo proposés.
</video>
```

Les différents formats sont essayés dans l'ordre descendant, jusqu'au premier format pris en charge par le navigateur.

Avantage : **Flash est désormais inutile** pour lire les vidéos !

Vidéo et navigateurs

Attention

Tous les navigateurs ne prennent pas tous les formats en charge.

Navigateur	MP4	Ogg	WebM
Chrome	Partiel	Partiel	Oui
Chromium	Non	Partiel	Oui
Firefox	Partiel	Oui	Oui
Internet Explorer	Oui	Non	Non
Opera	Partiel	Oui	Oui
Safari	Oui	Non	Non

Bonne pratique

Prévoir les vidéos dans suffisamment de formats

Image cliquable par endroits

Syntaxe :

```


<map name="campusmap">
  <area shape="circle" coords="120,330,20"
alt="arrêt de bus" href="https://www.ratp.fr">
  <area shape="rect" coords="150,0,220,150"
alt="Galilée" href="galilee.html" >
</map>
```

Coordonnées dans `coords`

- `circle` : x (centre), y (centre), rayon
- `rect` : gauche, haut, droite, bas

Image cliquable par endroits : affichage



Sources et références

Sources et références

Références

- Documentaire « Une Contre Histoire des Internets » (J. Goetz et J.M. Manach)
<http://lesinternets.arte.tv/>
- Spécification de HTML5 (recommandation du 28/10/2014)
<http://www.w3.org/TR/html5/>

Sources

- Histoire de HTML : Wikipédia > HTML
<http://fr.wikipedia.org/wiki/HTML>

Licence

Source des images utilisées (1/2)



Titre : Zygiella web
Auteur : Laura Bassett
Source : https://commons.wikimedia.org/wiki/File:Zygiella_web.jpg
Licence : attribution



Titre : Tim Berners-Lee
Auteur : John S. and James L. Knight Foundation
Source : https://commons.wikimedia.org/wiki/File:Tim_Berners-Lee-Knight-crop.jpg
Licence : CC BY-SA 2.0



Titre : Titre d'onglet (Mozilla Firefox 49.0.1 sous Mac OS)
Auteur : Étienne André
Source : Own work
Licence : CC BY-SA 3.0



Titre : Fenêtre de validation du W3C
Auteur : Étienne André
Source : Own work
Licence : CC BY-SA 3.0



Titre : Formulaire depuis Mozilla Firefox 30
Auteur : Étienne André
Source : Own work
Licence : CC BY-SA 3.0

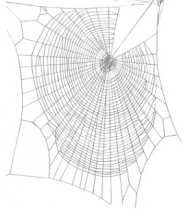
Source des images utilisées (2/2)



Titre : Lecture audio depuis Mozilla Firefox 24
Auteur : Étienne André
Source : Own work
Licence : CC BY-SA 3.0



Titre : Campus de Villetaneuse (2016)
Auteur : OpenStreetMap / cropped by Étienne André
Source : <http://www.openstreetmap.org/>
Licence : CC BY-SA 2.0



Initiation au développement Web – Partie 3

CSS

Laure Petrucci

Laure.Petrucci@univ-paris13.fr

<http://www.lipn.fr/~petrucci/M1106/>

Version : 6 janvier 2021 (diapositives à trous)



Partie 3: CSS – Plan

- 1 Définition et historique
- 2 Syntaxe de CSS3
- 3 Sites Web adaptatifs

Plan: Définition et historique

- 1 Définition et historique
- 2 Syntaxe de CSS3
- 3 Sites Web adaptatifs

CSS

Définition (CSS)

CSS (*Cascading Style Sheets*, ou feuilles de style en cascade) est

Standard développé par le

CSS est incrémenté par **niveaux** et non par versions

- Le niveau 3 réutilise et étend les niveaux 1 et 2
- Différent de HTML !

Principe : séparation de la forme et du fond

Principe

Le principe est de séparer le fond de la forme

Une feuille de style CSS va donc **mettre en forme** la page HTML.

Bonne pratique

Un site HTML bien conçu

Bref historique

- Début des années 1990 : concept de feuille de style
- 1995 : démonstration au congrès WWW
- 1996 : **CSS1**
- 1997 : **CSS2**
- 1999 : premiers brouillons pour CSS3
- 2001 : **CSS2.1** (correction de CSS2)
- 2010 : début de la réflexion pour **CSS4**
- 2011–12 : début des recommandations officielles du W3C pour des modules de **CSS3**

CSS3 est donc toujours

CSS3

CSS3 est divisé en

- chaque module étend CSS2
- 4 modules publiés en tant que recommandation formelle du W3C

Exemples de modules :

- `css3-background`
- `css3-color`
- `css3-selectors`

Plan: Syntaxe de CSS3

- 1 Définition et historique
- 2 **Syntaxe de CSS3**
 - Principes
 - Affichage du texte
 - Mise en forme de la page
 - Un exemple
- 3 Sites Web adaptatifs

Exemple simple

Page HTML :

```
...
<h1>Happy Together</h1>
<p>Film de Wong Kar-Wai
réalisé en 1997</p>
...
```

Feuille de style :

```
body {
  color: #000080;
}

h1 {
  font-weight: bold;
  font-style: italic;
  color: red;
  font-size: 24px;
}
```

Affichage :

Happy Together
Film de Wong Kar-Wai réalisé en 1997

Syntaxe générale d'une feuille CSS

```
Sélecteur
h1 {
  font-weight: bold;
  font-family: Times New Roman, Times, serif;
}
```

Propriété Valeur

Propriété Valeurs

Remarques :

- Extension
- Pas d'en-tête ou de pied dans un fichier CSS
- Syntaxe insensible aux espaces et retours à la ligne.
 - ▶ Néanmoins, l'**indentation est fortement recommandée** !

Style inclus ou séparé ? (1/2)

Moche :

```
<h1 style="color:red">Happy Together</h1>
```

Vaguement mieux :

```
<head>
...
<style type="text/CSS" >
  h1 {color:red;}
</style>
...
</head>
<body>
  <h1>Happy Together</h1>
</body>
```

Style inclus ou séparé ? (2/2)

Bien :

```
<head>
...
<link rel="stylesheet" type="text/css"
      media="screen" href="mafeuille.css" />
...
</head>
<body>
  <h1>Happy Together</h1>
</body>
```

...où `mafeuille.css` est un `mafeuille.css` définissant les propriétés de `<h1>` (et des autres balises).

Commentaires

Les commentaires sont du code CSS qui n'est pas exécuté, donc

Syntaxe :

- Attention : la syntaxe (à la C) « // » n'est pas reconnue !

```
body {
color: #000080; /* la couleur */
/* text-decoration: line-through; */
}
```

Attention

Les commentaires restent visibles dans la **source** de la feuille de style, qui est accessible depuis le client.

- Attention à ne pas y laisser d'informations confidentielles !

Plan: Syntaxe de CSS3

- 1 Définition et historique
- 2 Syntaxe de CSS3
 - Principes
 - Affichage du texte
 - Mise en forme de la page
 - Un exemple
- 3 Sites Web adaptatifs

Propriétés : styles

- Propriété **font-size** :
 - ▶ Valeurs prédéfinies absolues : `xx-small`, `x-small`, `medium`, `large`, `x-large`, `xx-large`
 - ▶ Valeurs prédéfinies relatives : `smaller`, `larger`
 - ▶ Valeur numérique : x unités, où unité $\in \{\text{pt}, \text{px}, \text{em}, \%\}$
- Propriété **font-weight** :
 - ▶ Valeurs prédéfinies : `lighter`, `normal`, `bold`, `bolder`
 - ▶ Valeurs numérique : $x \in [0; 1000]$ (1000 étant le plus gras)
- Propriété **font-style** :
 - ▶ Valeurs prédéfinies : `normal`, `italic`, `oblique`
- Propriété **text-decoration** :
 - ▶ Valeurs prédéfinies : `underline`, `overline`, `line-through`, `none`

Propriétés : police

- Propriété **font-family** : famille de police de caractères
 - ▶ Grandes familles : serif, sans-serif, mono, monospace
 - ▶ Nom de police : `Arial`, `Courier New`, `Courier`, `Helvetica`, `Times New Roman`, `Times`, et bien d'autres

vif zéphyr | vif zéphyr | vif zéphyr
serif sans-serif monospace

Bonnes pratiques typographiques

- 1 Les titres sont souvent en `serif`, et le corps en `sans-serif`
- 2 Il convient d'éviter de mélanger outre mesure les familles de polices dans un même document.

Propriétés : capitales et petites capitales (1/2)

- Propriété `font-transform` :
 - ▶ Valeurs : `capitalize` (première lettre), `uppercase` (capitales), `lowercase` (minuscules), `none` (normal)
- Propriété `font-variant` :
 - ▶ Valeurs : `normal`, `small-caps`

Propriétés : couleurs

- Propriété `color` :
 - ▶ Couleurs prédéfinies : `red`, `blue`, `green`, `orange`, etc.
 - ★ Voir par exemple web-color.aliasdmc.fr/
 - ▶ Valeurs utilisateur :
 - ★ Hexadécimal : `#RRVVBB`
 - ★ RVB : `rgb(r, g, b)` avec valeurs de 0 à 255
 - ★ RVB : `rgb(x%, g%, b%)` avec valeurs de 0 à 100
 - ★ RVB + transparence : `rgba(r, g, b, a)` avec valeurs de 0 à 255, et transparence (`a`) entre 0 et 1
 - ★ HSL (teinte / saturation / lumière) : `hsl` et `hsla`

Propriétés : capitales et petites capitales (2/2)

```
<p>Un bout de texte en
  <span style="font-transform:uppercase">capitales</span> et
  en <span style="font-variant:small-caps">Petites
  Capitales</span>.</p>
```

Affiche :

Un bout de texte en CAPITALES et en Petites Capitales.

Bonne pratique

On ne doit jamais écrire de code HTML directement en majuscules, mais plutôt utiliser les attributs CSS correspondants.

Plan: Syntaxe de CSS3

- 1 Définition et historique
- 2 Syntaxe de CSS3
 - Principes
 - Affichage du texte
 - Mise en forme de la page
 - Un exemple
- 3 Sites Web adaptatifs

Identifiants et classes

Identifiants

- Syntaxe HTML : `<h1 id="titre">`
- Syntaxe CSS : `h1 #titre { ... }`
- Utilisation **unique** (un seul `id=titre` dans la page HTML)

Classes

- Syntaxe HTML : `<h1 class="titre">`
- Syntaxe CSS : `h1 .titre { ... }`
- Utilisation **multiple** (plusieurs `class=titre` autorisés dans la page HTML)

Identifiants et classes : exemple (affichage)

As Tears Go By
Nos années sauvages
Les Cendres du temps
Chungking Express
Les Anges déchus

Identifiants et classes : exemple

Page HTML :

```
<h2 id="premier">As Tears Go By</h2>
<h2>Nos années sauvages</h2>
<h2 class="bleu">Les Cendres du temps</h2>
<h2>Chungking Express</h2>
<h2 class="bleu">Les Anges déchus</h2>
```

Feuille de style :

```
h2 {
  font-style: italic;
}
#premier {
  color: red;
}
.bleu {
  color: blue;
}
```

Héritage

Les propriétés CSS **héritent** des propriétés parentes.

Héritage par raffinement de propriété :

- `h2 #premier` hérite de `h2`
- `h2 .bleu` hérite de `h2` (mais pas de `h2 #premier`)

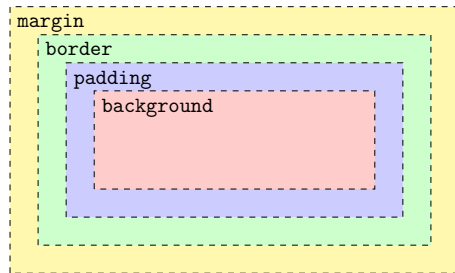
Héritage par propriété sémantiquement parente :

- `h2` hérite de `body`

Marges et bordures

Tout élément HTML est considéré comme contenu dans une **boîte**, dont on peut définir :

- l'aire intérieure
- l'espace intérieur
- la bordure
- la marge extérieure



Marges et bordures : exemple

```
<p class="boiteorange">Ce petit exemple est tiré du Wikibook sur CSS (Cascading Style Sheets).</p>
```

```
.boiteorange {
  border: 1px dotted black;
  padding-top: 1em;
  padding-right: 2em;
  padding-bottom: 3em;
  padding-left: 4em;
  background-color: #fc9;
  width: 10em;
  text-align: justify
}
```

Ce petit exemple est tiré du Wikibook sur CSS (Cascading Style Sheets).

Le validateur du W3C

Comme pour HTML, le W3C et la fondation Mozilla mettent à disposition un validateur de code CSS3.

<http://jigsaw.w3.org/css-validator/>

Plan: Syntaxe de CSS3

- 1 Définition et historique
- 2 Syntaxe de CSS3
 - Principes
 - Affichage du texte
 - Mise en forme de la page
 - Un exemple
- 3 Sites Web adaptatifs

Un exemple sans CSS

Présentation de l'IUT de Villetaneuse

Menu

- Le DUT R&T
- Le DUT en apprentissage
- La licence professionnelle
- Autres départements

Le DUT R&T

Le DUT du département [réseaux et télécommunications](#) de l'IUT de Villetaneuse permet d'acquies plusieurs compétences en termes de :

1. réseaux,
2. télécommunications,
3. systèmes d'exploitation, et
4. communication et expression.

La licence professionnelle ASUR

Description

La licence professionnelle ASUR du département [réseaux et télécommunications](#) de l'IUT de Villetaneuse permet d'acquies plusieurs compétences en termes de :

- réseaux,
- sécurité et techniques d'attaques,
- administration de postes sous Linux ou Windows,
- communication et expression.

Deux versions

En formation initiale

La licence professionnelle peut être intégrée après un DUT réseaux et télécommunications, un BTS, ou d'autres formations similaires.

En alternance

Le contenu de la licence en alternance est le même que la licence en formation initiale, seules les modalités diffèrent. Dans la version en alternance, les étudiants ont le statut d'apprentis, et passent la moitié de leur temps en entreprise.

Pour en savoir plus, consultez le site Internet de la [licence professionnelle ASUR](#).

Sites intéressants

- [Wikipédia - HTML5](#)
- [Wikipédia - CSS3](#)
- [Validateur W3C - HTML](#)
- [Validateur W3C - CSS](#)

Licence

Ce contenu peut être réutilisé, republié et modifié selon les conditions de la licence [Creative Commons CC BY-NC-SA 3.0](#) ; auteur : [Romain Andrieu](#).



Le même exemple avec CSS

Présentation de l'IUT de Villetaneuse	
Menu	Le DUT R&T
Le DUT R&T	Le DUT du département réseaux et télécommunications de l'IUT de Villetaneuse permet d'acquies plusieurs compétences en termes de :
Le DUT en apprentissage	1. réseaux,
La licence professionnelle	2. télécommunications,
Autres départements	3. systèmes d'exploitation, et
	4. communication et expression.
	La licence professionnelle ASUR
	Description
	La licence professionnelle ASUR du département réseaux et télécommunications de l'IUT de Villetaneuse permet d'acquies plusieurs compétences en termes de :
	• réseaux,
	• sécurité et techniques d'attaques,
	• administration de postes sous Linux ou Windows,
	• communication et expression.
	Deux versions
	En formation initiale
	La licence professionnelle peut être intégrée après un DUT réseaux et télécommunications, un BTS, ou d'autres formations similaires.
	En alternance
	Le contenu de la licence en alternance est le même que la licence en formation initiale, seules les modalités diffèrent. Dans la version en alternance, les étudiants ont le statut d'apprentis, et passent la moitié de leur temps en entreprise.
	<i>Pour en savoir plus, consultez le site Internet de la licence professionnelle ASUR.</i>
<p>SITES INTÉRESSANTS</p> <p>WIKIPÉDIA - HTML5 WIKIPÉDIA - CSS3 VALIDATEUR W3C - HTML VALIDATEUR W3C - CSS</p> <p>LIEN</p> <p>CE CONTENU PEUT ÊTRE RÉUTILISÉ, REPRODUIT ET MODIFIÉ SELON LES CONDITIONS DE LA LICENCE CREATIVE COMMONS CC BY-NC-SA 3.0. AUTEUR : ÉTIENNE ANDRIEU.</p>	

Code HTML identique

Simple ajout d'une feuille de style

Plan: Sites Web adaptatifs

- 1 Définition et historique
- 2 Syntaxe de CSS3
- 3 Sites Web adaptatifs

Problématique

La consultation d'un site Web peut se faire depuis des appareils très divers :

- Ordinateur de bureau à écran 25 pouces
- Ordinateur portable
- Netbook
- Tablette
- Téléphone intelligent (ou non)
- Télévision
- Navigateur pour aveugles

Problème

Comment assurer un affichage optimal pour tous ?

Solution : les sites Web adaptatifs

Conception de sites Web adaptatifs

regroupe différents principes et technologies ayant pour but de faciliter la consultation d'un site Web de façon optimale quel que soit l'appareil utilisé.



Principe technique

La conception de sites Web adaptatifs repose sur l'utilisation de la règle en CSS.

Utilisation :

- Import de différentes feuilles de style en fonction du navigateur (taille de l'écran, type de système, etc.)
- Redimensionnement de blocs ou d'images
- Personnalisation de parties de la feuille de style

Utilisation de @media (1/2)

Exemple d'import conditionnel de feuilles de style dans un fichier HTML :

```
<link rel="stylesheet" media="only screen and min-device-
width:480px" href="bureau.css" type="text/css" />
<link rel="stylesheet" media="handheld, (max-device-width:480
px)" href="smartphone.css" type="text/css" />
```

- Si appareil de type **screen** et taille de l'écran supérieure à **480 px** : utilisation de **bureau.css**
- Si appareil de type **handheld** ou taille de l'écran inférieure à **480 px** : utilisation de **smartphone.css**

Utilisation de @media (2/2)

Conditions à l'intérieur d'une feuille de style :

```
header {
  width:600px;
  background-color:white;
}
@media only screen and (max-device-width:480px) {
  header {
    width:100%;
    background-color:#D0D0D0;
  }
}
```

Syntaxe de @media

Utilisation d'expressions booléennes :

- **and** : « et »
- **not** : « non »
- **,** : « ou »
- expressions imbriquées

Exemple :

```
@media screen and (max-device-width:480px) and not braille
and not projection
```

Syntaxe : types de media

Type	Description
braille	Pour aveugle
embossed	Imprimantes en braille
handheld	Tenu à la main (ex : téléphone intelligent)
print	Impression
projection	Projecteurs
screen	Écran
speech	Synthèse vocale
tty	Terminal à chasse fixe
tv	Téléviseur
all	Tous

Syntaxe : types de fonctionnalités

Propriété	Valeur	Description
color	Entier	Nombre de couleurs (bits)
color-index	Entier	Nombre d'entrées dans la table des couleurs
device-aspect-ratio	Entier/Entier	Aspect ratio
device-height	Entier	Hauteur (appareil)
device-width	Entier	Largeur (appareil)
grid	Booléen	Vrai pour un appareil à grille
height	Entier	Hauteur (surface)
max-device-width	Entier	Largeur (appareil)
monochrome	Entier	Nombre de bits (mono-chrome)
orientation	landscape / portrait	Orientation de l'écran
resolution	Résolution	Résolution
scan	progressive / interlaced	Pour télévisions
width	Entier	Longueur (surface)

Bonnes pratiques

Quelques idées à prendre en considération :

- Menus non affichés par défaut sur terminaux de petite taille (utilisation de Javascript, ou de pages séparées)
- Éviter l'utilisation du zoom ou du défilement, peu pratique pour la navigation
- Privilégier les tailles **relatives**, en % ou **em**, plutôt qu'absolues (**pt**, **px**) = taille de la police par défaut du navigateur
- Privilégier des images **vectérielles** (formats **.svg**, **.pdf**)
 - ▶ Sinon, aspect « pixelisé » en cas d'agrandissement (formats **.gif**, **.jpg**, **.png**)

Remarques

Déclaration du type

- ❶ Tous les terminaux ne se déclarent pas toujours du bon type. Par exemple, peu de navigateurs mobiles se déclarent comme
- ❷ Sur certains navigateurs (exemple : Android CyanogenMod), on peut **forcer** le type du navigateur.
 - ▶ Astucieux pour éviter d'être redirigé sur le site mobile de certains sites mal conçus !

Sources et références

Essayez !

Firefox offre des fonctionnalités très pratiques dans Outils > Développement Web

- Visualisation dynamique de la feuille de style
- Redimensionnement de la fenêtre
 - ▶ Joue sur l'attribut `max-device-width`, et donc permet de **simuler** la visualisation depuis un appareil à plus faible résolution

Sources et références

Références

- **Le langage CSS** (Wikilivres)
https://fr.wikibooks.org/wiki/Le_langage_CSS
- **Cascading Style Sheets** (Wikibooks), en anglais mais plus complet
https://en.wikibooks.org/wiki/Cascading_Style_Sheets

Sources

- **La spécification de CSS(3)** par le W3C
<http://www.w3.org/TR/CSS/>
- **Media Queries**
https://en.wikipedia.org/wiki/Media_queries

Licence

Source des images utilisées (1/2)



Titre : Zygiella web
 Auteur : Laura Bassett
 Source : https://commons.wikimedia.org/wiki/File:Zygiella_web.jpg
 Licence : attribution



Titre : Affichage HTML+CSS (boîte)
 Auteur : Étienne André
 Source : Own work
 Licence : CC BY-SA 3.0



Titre : Page HTML sans CSS
 Auteur : Étienne André
 Source : Own work
 Licence : CC BY-SA 3.0



Titre : Page HTML avec CSS
 Auteur : Étienne André
 Source : Own work
 Licence : CC BY-SA 3.0

Source des images utilisées (2/2)



Titre : Illustration for responsive web design technique
 Auteur : Muhammad Rafizeldi
 Source : https://commons.wikimedia.org/wiki/File:Responsive_Web_Design.png
 Licence : CC BY-SA 3.0

Licence de ce document

Ce support de cours peut être republié, réutilisé et modifié selon les termes de la licence Creative Commons **Attribution-NonCommercial-ShareAlike 3.0 Unported (CC BY-NC-SA 3.0)**



<https://creativecommons.org/licenses/by-nc-sa/3.0/>

Auteur : Étienne André

(Source \LaTeX disponible aux enseignants sur demande)