HTML INTRODUCTION

LI288 – Web et développement web

WWW = World Wide Web

- Créé en 1989 au CERN par Tim Berners-Lee
  - Objectif initial : mettre en ligne de la documentation (initialement technique pour physiciens)
- 1993 : Mosaic 1er navigateur graphique (NCSA)
- 1994 : World Wide Web Consortium (W3C)
  - créé par le CERN et le MIT
  - organisme de standardisation du Web
  - administré par le MIT et l'INRIA
- 1994 : Netscape Mozilla navigateur (Win, Unix, Mac)
  - développé notamment par Marc Andreessen (un des auteurs de Mosaic)
- 1996 : Microsoft Internet Explorer
- 1996 : 1ère version d'Opera en téléchargement (1994 Telenor R&D, Norway)
- 2003 : création de la Mozilla Foundation (Firefox, Thunderbird)

WWW = World Wide Web

- Services de l'Internet
  - www
  - messagerie
  - transfert de fichiers (FTP)
  - forum de nouvelles (news), de discussion (chat)
  - peer-to-peer (Gnutella, Freenet, ...), diffusion de flux audio et vidéo (voip, vod, streaming)

- Principes du Web
  - hypertexte : HTML
  - client/serveur : HTTP
  - schéma de désignation : URL

World Wide Web - Hypertexte

- Texte "normal" :
  - Organisation linéaire.
  - Eventuellement renvois sous forme de sommaires, index, notes de bas de page.

- Hypertexte :
  - Organisation pas forcément linéaire.
  - Texte enrichi de liens :
    - renvoi vers un document.
    - renvoi vers une partie du même document.
    - renvoi vers une partie d'un autre document.

World Wide Web - Client/serveur

- Client : le navigateur (Internet Explorer, Firefox, Chrome, Opera, Safari...)
- Serveur : le serveur Web (Apache, Microsoft IIS...)

- 1. Le client émet une requête
- 2. Le serveur répond en fournissant le document demandé ou un message d'erreur si le document n'existe pas

World Wide Web - URL

- Schéma de désignation
- Uniform Resource Locator (URL)
  - Permet de désigner une page Web
  - Chaque page a un nom unique : pas d'ambiguïté possible
    - protocole://serveur/page
    - http://www.lip6.fr/index.html
- Organisation hiérarchique possible pour les pages
  - protocole://serveur/répertoire/.../page
  - http://www.lip6.fr/congres/2002/index.html

World Wide Web - standards

- HTTP
  - 0.9 : version de base avec requête/réponse
  - 1.0 : version standardisée IETF (RFC 1945)
  - 1.1 : version étendue (connexions persistantes)
- HTML
  - 1.0 : version initiale
  - 2.0 : version standardisée W3C
  - 3.2 : version étendue (tableaux, images cliquables, applets)
  - 4.0 : version étendue (frames, feuilles de style) : version basique et stricte
  - 5.0 : en cours de normalisation (déjà utilisé par la plupart des navigateurs)
- URL
  - format stable depuis 1989

(x)HTML

LI288 – Web et développement web

HyperText Markup Language

- HTML : langage de balisage issu de SGML
- Document HTML contient :
  - du texte
  - des balises (ou tags) : structuration ou mise en forme
    - Ce texte est en italique
    -
  - les balises peuvent être imbriquées
    - texte en italique !

HyperText Markup Language

- Structure d'un document HTML :
  - Déclaration de la version HTML utilisée
  - En-tête
  - Corps du document.
- HTML Strict : tout est obligatoire.
- Absence de corps = document vide.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-
    strict.dtd">


<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
  <title>Première page web</title>
</head>


<body>
  <p>Ceci sera affiché dans le navigateur</p>
</body>
</html>
```

Règles de balisage

- 2 types de balises
  - autonome < ... > (exemple : <! DOCTYPE >)
  - délimitant une zone
    - balise de début de zone < ... > (exemple : <html>)
    - balise de fin de zone </ ... > (exemple : </html>)
    - pour les balises sans contenu
      - Attention </…> est différent de <…/>
  - Toutes les balises doivent être fermées (XHTML1.0 strict)

Règles de balisage

- Identifiants de balise non sensibles à la casse :
  - <BODY> est similaire à <body>
  - Les noms de balises doivent être en minuscules

- Des attributs peuvent être associés aux balises :
  - Un attribut à un identifiant et une valeur id="valeur"
    -
  - Les noms d'attributs doivent être en minuscules
  - Les valeurs d'attributs doivent être entre guillemets
  - Tous les attributs doivent avoir une valeur
    - Pas de <input checked>

Règles de balisage

- Commentaires
  - <!-- ceci est un commentaire -->

- Tous les caractères spéciaux peuvent/doivent être encodés :
  - format : "&xxx;" avec xxx variable
  - < > & : < > &
  - caractères accentués : & lettre accent ;
    - acute, grave, circ, uml, tilde, ring
  - é à ô : é à ô

STRUCTURE D'UN DOCUMENT

En-tête des documents HTML

- Langage et version utilisés par le document :
  - les navigateurs savent s'ils pourront afficher le document.
  - <! DOCTYPE langage statut version >
  - http://validator.w3.org

- Exemple
  - <!DOCTYPE HTML PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd>
  - <!DOCTYPE html> pour HTML5

Balise html

- La balise html englobe tout le document (Sauf doctype)
  - Uniquement 2 balises filles possibles : head et body
- Attributs possibles :
  - xmlns : http://www.w3.org/1999/xhtml
  - dir : rtl ou ltr – indique le sens de lecture du document
  - xml:lang : indique le langage utilisé dans le document.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
   "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">
<head>... </head>

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

Balise head

- Informations sur le document :
  - Titre, auteur, description, mots clefs
  - Inclusion d'autres fichiers (javascript, feuilles de style, …)

- <title>un titre</title>
  - Titre du document
    - Apparaît notamment dans la barre supérieur du navigateur.
    - Utilisé par les moteurs de recherche.
  - Obligatoire !

```
<head>
  <title>Ma page web</title>
  ...
</head>
```

Balise head – les metas

- Informations supplémentaires sur la page
  - Propriété du document / non affichée.
  - <meta name="propriété" content="valeur">
    - Content est obligatoire
  - Valeurs possibles
    - author, description, keywords, generator, revised
    - Autres valeurs possibles, potentiellement utilisées mais pas normalisées :
      - Robots : indexer la page ? suivre les liens ?

```
<head>
  <title>Ma page web</title>
  <meta http-equiv="content-type" content="text/html; charset=utf-8" />
  <meta name="author" content="moi" />
  <meta name="description" content="cours web" />
  <meta name="mise a jour" content="tous les ans" />
  <meta name="robots" content="index, follow" />
</head>
```

Balise head – les link

- Créer un lien avec un autre document :
  - Surtout utilisé pour inclure une feuille de style.
  - En général on utilise :
    - href : localisation du fichier
    - rel : définition du type de lien défini – stylesheet, home, …
      - Pas forcément utilisé par les navigateurs
    - type : type (MIME) du fichier
    - media : indique si l'inclusion doit être faite pour certains media

```
<head>
  <title>Ma page web</title>
  <link rel="stylesheet" type="text/css" href="x.css" />
</head>
```

Balise head – les scripts

- Définition d'un script exécuté en local :
  - type : définition du type de script (obligatoire)
  - src : url du script si défini dans un fichier externe
  - Attention à fermer avec </script>

```
<head>
  <title>Ma page web</title>
  <script type="text/javascript">
    alert("Bonjour !")
  </script>
  <script type="text/javascript" src="menu.js"></script>
</head>
```

Corps des documents HTML

- <body attr1="val1" ... attrn="valn">
  - Attributs possibles :
    - bgcolor : couleur de fond.
    - text : couleur du texte.
    - background : URL de l'image de fond d'écran.
    - link : couleur des liens non encore visités.
    - vlink : couleur des liens visités.
    - ...
  - Normalement on utilise les CSS, mais en attendant…

Attributs

- Les différentes balises peuvent avoir :
  - Un identifiant : id="monid"
  - Un nom : name="monnom"
    - Majoritairement pour les formulaires
  - Une ou plusieurs classes : class="classe1 classe2".

- Un seul ID par fichier HTML :
  - Permet d'identifier un objet en particulier
    - Par exemple le menu.

- Plusieurs objets peuvent avoir la même classe :
  - Utilisées pour dire que plusieurs objets sont similaires.
    - Principalement utilisé pour le rendu visuel (CSS).

Autres attributs classiques

- title :
  - Donne un titre à la balise concernée.
  - Peut-être utilisé ou pas.
- style :
  - Donne un style à la balise concernée.
  - Plus utilisé, on fait des feuilles de style à la place.
- xml:lang :
  - Langue du contenu de la balise.
- dir :
  - Sens dans lequel la balise va être lue.

Les types de balises

- Deux types de balise :
  - Bloc (peut contenir des balises blocs et en-ligne, sauf exception)
    - Exemples : div, p, h1, h2, ul, table
  - En-ligne (ne contient que des balises en-ligne, sauf exception) :
    - Exemples : span, em, strong, img, br, input
  - En HTML5 c'est moins simple

- Principe de base :
  - Les blocs sont les uns aux dessus des autres.
  - Les éléments en-ligne sont les uns à coté des autres.
  - Les éléments sont placés les uns à la suite des autres dans la page (le flux).

- On peut modifier le type des balises.

BALISES EXISTANTES

Balises classiques

- Titres :
  - <h1>Titre</h1>
  - <h2> à <h6> pour les niveaux inférieurs
  - Attribut possible align="left|center|right"
- Paragraphes :
  - contenu du paragraphe
- Autres :
  -
 passage à la ligne
  - <hr /> trait horizontal
  - <b> = gras, <i> = italique, <u> = souligné, ...

Liens hypertextes

- Portion de texte permettant d'atteindre un document désigné par une URL
  - texte du lien
- URL absolue
  - <a href="http://www.lip6.fr/index.html">LIP6</A>
- URL relative <a href="sommaire.html">sommaire</A>
  - <a href="../divers/plan.html">plan d'accès</A>
  - <a href="/index.html">accueil</A>
- Désignation de signets à l'aide du caractère # dans les URLs
  - <a href="#conclusion">conclusion du rapport</A>
  - <a href="rapport.html#conclusion">conclusion du rapport</A>
  - Crée un lien vers la balise dont l'id est conclusion dans le fichier mentionné.

Compléments sur les URLs

- Eléments constitutifs (optionnels) d'une URL
  - protocole:// protocoles utilisables : http, ftp, mailto, file, news
  - Login:motdepasse@ pour les pages protégées
  - nom ou adresse IP (ex. : 163.136.27.6) du serveur
  - :numéro de port TCP du serveur
  - /répertoire chemin d'accès dans la hiérarchie de répertoires
  - fichier nom du document à atteindre
  - #signet signet dans le document à atteindre
  - ?options information transmise au serveur (ex. : formulaire)

- Exemples
  - http://bob:motdepassedebob@12.7.6.1:8080/pub/cv.html#diplomes
  - ftp://anonymous@ftp.lip6.fr
  - http://www.altavista.com/query.html?jussieu

Images

- Insertion d'une image dans un document :

- Attributs possibles
  - height="99" / width : hauteur/largeur en pixels (ou en pourcentage) de l'image
  - alt=" ... " : légende associée à l'image, utilisé en remplacement de l'image si elle n'est pas affichée.
  - hspace="99" / vspace : espacement autour de l'image
  - align="bottom|middle|top|left|right" : alignement de l'image / au texte
- Le navigateur redimensionne l'image si les tailles indiquées ne correspondent pas.

Listes

- Listes numérotées (
) ou non (
), et définitions (<dl><dt><dd>)
- Attributs possibles
  - <ul type="disc|circle|square"> : type de puce
  - <ol type="1|I|i|A|a" : type de numérotation
  - start="99"> : départ de la numérotation
  - <li type="disc|circle|square"> : type de puce (liste ul)
  - <li type="1|I|i|A|a "> : type de numérotation (liste ol)

```
<ul>
  <li>rouge</li>
  <li>vert</li>
</ul>

<dl>
  <dt>Titre</dt>
  <dd>définition</dd>
</dl>
```

Listes

- Imbrication possible des listes :

```
<ol start="3">
  <li type="1"> élément 1,1 </li>
  <li> élément 1,2 </li>
  <li><ul type="a">
    <li type="square"> élément 2,1 </li>
    <li type="circle"> élément 2,2 </li>
  </ul></li>
  <li>élément 1,3 </li>
</ol>
```

Tableaux

- Tableaux à 2 dimensions (<table> ... </table>)

```
<table>                  <!-- balise de début de tableau -->
<tr>                     <!-- balise de début de ligne -->
  <td> cellule </td>     <!-- balise de cellule -->
  ...
</tr>                    <!-- balise de fin de ligne -->
<tr>
  <td>
  ...
</tr>
</table>                 <!-- balise de fin de tableau -->
```

Tableaux

- Possibilité de laisser des cellules vides
  - A ne pas encourager.

```
<table border="1">
<tr>
  <td> case 1,1 </td>
</tr>
<tr>
  <td> case 2,1 </td>
  <td> case 2,2 </td>
</tr>
<tr>
  <td></td>
  <td> case 3,2 </td>
</tr>
</table>
```

Tableaux

- border="99" :
  - Epaisseur en pixels des bordures du tableaux.
- align="left|right" (obsolète)
  - Alignement du tableau par rapport au texte.
- width="99|99%" :
  - Largeur du tableau en pixels ou en % de la largeur de la fenêtre.
- cellspacing="99" :
  - Espacement en pixels entre les cellules.
- cellpadding="99" :
  - Marge intérieure en pixels des cellules.

Tableaux

- Attributs possibles pour <tr>
  - align="left|right|center|justify" : alignement horizontal du texte (≠ <table>)
  - valign="middle|top|bottom" : alignement vertical du texte
- Attributs possibles pour <td>
  - align/valign : idem <tr>, supplante <tr align/valign >
  - width="99|99%" : largeur de la colonne en pixels | en % largeur tableau (obsolète)
- Balise <caption> légende associée au tableau
  - align="top|bottom" : placement de la légende

Affichage des bordures

- Attribut frame de table : traits autour du tableau
  - void : aucun.
  - above, below, lhs, rhs : haut, bas gauche droite.
  - hsides : traits horizontaux.
  - vsides : traits verticaux.
  - Box, border : tous les cotés.

- Attribut rules : traits dans le tableau
  - none : aucun.
  - groups : entre les groupes de lignes (thead, tfoot, tbody).
  - rows : entre les lignes uniquement – traits horizontaux.
  - cols : entre les colonnes uniquement – traits verticaux.
  - all : tous.

Regroupement de cellules

- On peut grouper des cellules :
  - colspan : sur une même ligne
  - rowspan : sur une même colonne

```
<table border="1">
<caption>légende</caption>
<tr>
  <td colspan=2>x</td>
</tr>
<tr>
  <th rowspan=2>x</th>
  <td>x</td>
</tr>
<tr>
  <td>x</td>
</tr>
</table>
```

Un exemple complet

- Tableaux complets : thead, tbody, tfoot

```
<table cellpadding="6" summary="valeurs par jour">
<thead>
  <tr><th>jour</th><th>date</th><th>nombre</th></tr>
</thead>
<tbody>
  <tr><td>lundi</td><td>09/11</td><td>639</td></tr>
  <tr><td>mardi</td><td>09/12</td><td>596</td></tr>
  <tr><td>mercerdi</td><td>09/13</td><td>1135</td></tr>
</tbody>
<tfoot>
  <tr><th colspan="2">total</th><th>4923</th></tr>
</tfoot>
</table>
```

Tableaux - remarque finale

- Les tableaux servent à :
  - Représenter des données tabulées.
- Ils ne devraient pas servir à :
  - Faire de la mise en page de données non tabulées.
    - Par exemple pour créer un menu il faut plutôt utiliser une liste d'objets
- En pratique on se sert trop des tableaux !
  - Pas durant le module

Blocs génériques

- Pour agencer des éléments on utilise les blocs génériques :
  - : balise de type bloc générique
  - : balise de type en-ligne générique
- Uniquement de la structuration

```
<ul id="menu">
...
</ul>
<div id="main">
  <p>Ceci est très <span class="important">important</span></p>
</div>
<div id="footer">
...
</div>
```

Autres balises

- A tester dans un navigateur pour voir le rendu :
  - En général on utilise plutôt des CSS.

```
<em>Texte mis en avant</em><br />
<strong>Texte important</strong><br />
<dfn>Définition</dfn><br />
<code>Code source</code><br />
<samp>Exemple de code source</samp><br />
<kbd>Texte clavier</kbd><br />
<var>Variable</var><br />
<cite>Citation</cite><br />
<sub>Indice</sub><br />
<sup>Exposant</sup>
```

Autres balises

- Balises supprimées/déconseillées dans HTML 5 :
  - Mise en forme pure.
  - Éviter de les utiliser même si ça fonctionne.

```
<tt>texte teletype</tt>
<i>texte en italique</i>
<big>Texte en gros</big>

<b>Texte en gras</b>
<small>Texte en petit</small>
```

Couleurs

- 2 solutions :
  - Identificateurs prédéfinis : green, yellow, purple, blue, red, ...
  - Nombre de 6 chiffres hexadécimaux (ex. : #1A7FC0) codant les intensités de rouge/vert/bleu.
- Equivalence entre les 2 :
  - blue ≡ #0000FF
  - purple ≡ #800080
  - Mais très peu de couleurs avec identificateurs.
- Utiliser des palettes de couleurs pour trouver les codes hexa.

FORMULAIRES

LI288 – Web et développement web

Formulaires

- HTML 1.0 essentiellement "mono"-directionnel :
  - Informations fournies par le serveur (suite à une commande client)
- Utilisation professionnelle
  - Nécessité de flux d'information bi-directionnels (client ↔serveur)
  - HTML 2.0 introduit les formulaires.
  - Permettent aux clients de saisir des informations à envoyer au serveur.

- Un formulaire est défini en HTML et peut contenir :
  - Zones de saisie de texte.
  - Boîtes à cocher.
  - Boutons radio.
  - Menus déroulants.
  - Boutons.

Formulaires

- Déclaration d'un formulaire :
  - Balises <form> et </form>
- Attributs principaux
  - <form action="…" method="…" … >
  - action : url vers laquelle envoyer les données saisies.
  - method : commande http (get ou post) à utiliser pour effectuer l'envoi.
- Toutes les balises HTML sont permises entre <form> </form>
  - Images, tableaux, … peuvent être inclus dans un formulaire.
  - Les formulaires peuvent être imbriqués.

Formulaire – champs input

- Déclaration des champs de saisie
  - exclusivement entre <form> </form>

- name = nom du champ de saisie (unique dans un formulaire)
  - Permet de récupérer les valeurs soumises
- type = type du champ de saisie :
  - text : zone de saisie texte (type par défaut en cas d'omission de type)
    - size : taille apparente maxlength : taille max
  - radio : bouton radio
    - tous les boutons ayant même nom (name) ∈ au même groupe
    - dans ce cas, les attributs (value) permettent de les différencier
  - checkbox : boîte à cocher
  - submit : bouton d'envoi des données au serveur
  - reset : bouton d'effacement du formulaire

Exemple

```
...
<form action="prog.php" method="get">
  <p>nom <input name="client" type="text" size="4" /></p>
  <p>rue <input name="rue" type="text" size="40" /></p>
  <p>ville <input name="ville" type="text" size="20" /></p>
  <p>code postal <input name="cp" type="text" size="5" /></p>
  <p>carte de crédit no <input name="carte" type="text" size="10" /></p>
  <p>expire <input name="expire" type="text" size="4" /></p>
  <p>m/c <input name="cc" type="radio" value="mc" checked="checked" />
     visa <input name="cc" type="radio" value="vis" /></p>
  <p>contre remboursement <input name="cr" type="checkbox" /></p>
  <p><input type="submit" value="envoi" /></p>
  <p><input type="reset" value="remise à zéro" /></p>
</form>
...
```

Labels

- Intitulé associé à un champ de formulaire
  - Le clic sur l'intitulé donne la main sur le champ

```
<label for="id">Label :</label>
<input id="id" type="text" />

<label for="id">Label :
  <input id="id" type="text" />
</label>
```

Envoi des données au serveur

- Lorsque l'utilisateur appuie sur le bouton SUBMIT, le navigateur :
  - Construit une chaîne de caractères contenant toutes les données du formulaire.
  - Envoie cette chaîne au serveur.

- Chaîne :
  - Ensemble de couples séparées par le caractère &
  - Chaque couple est de la forme nom de champ = valeur saisie
  - Les espaces sont remplacés par le caractère + (ou %20)
  - Les caractères + & = sont encodés %2B %26 %3D

- Exemple :
  - client=Jean+Vier&ville=Paris&cp=75001&carte=0123456789&cc=vis&cr=on
  - Pour les boîtes à cocher : « on » si cochée, rien sinon

Autres types

- password zone de saisie d'un mot de passe + attribut enctype
  - les caractères saisis sont remplacés par des *
- button : un bouton simple (association avec un traitement JavaScript)

```
<form action="prog.php" method="post" enctype="multipart/form-
   data">
  <p>passe <input type="password" name="passe" size="16" /></p>
  <p><input type="submit" value="envoi" /></p>
  <p><input type="reset" value="remise à zéro" /></p>
</form>
```

Autres types

- hidden : champ caché (ne provoque aucun affichage)
  - Permet de transmettre des informations supplémentaires.
  - Ne rien cacher dedans, on peut voir le contenu en affichant la source
- file sélection d'un fichier à envoyer :
  - Création :
    - D'un champ de saisie du nom du fichier
    - D'un bouton permettant de sélectionner un fichier, via une fenêtre de parcours du disque.

```
<form action="prog.php" method="post">
  <p><input type="hidden" name="cache" value="test" /></p>
  <p><input type="button" value="cliquez" /></p>
  <p>select file <input type="file" name="fichier" /></p>
  <p><input type="submit" value="envoi" /></p>
  <p><input type="reset" value="remise à zéro" /></p>
</form>
```

Autres types

- textarea : zone de saisie d'un texte sur plusieurs lignes :
  - Spécification du nombre de lignes et de colonnes.
- select : définition d'un menu déroulant :
  - les choix se font avec la balise option.

```
<form action="prog.php" method="post">
  <textarea name="zone" rows="3" cols="40">texte</textarea>
  <select name="musictypes">
    <option>valeur 1</option>
    <option selected="selected">valeur 2</option>
    <option>valeur 3</option>
  </select>
  <input type="submit" value="envoi" />
</form>
```

Autres types

- select multiple :
  - Menu déroulant à choix multiples

```
<form action="prog.php" method="post">
  <select multiple="multiple" name="musictypes">
    <option>valeur 1</option>
    <option selected="selected">valeur 2</option>
    <option>valeur 3</option>
  </select>
  <input type="submit" value="envoi" />
</form>
```

HTML
AUTRES FONCTIONNALITÉS

LI288 – Web et développement web

Frames

- Permettent de scinder la fenêtre du navigateur en plusieurs parties (frames)
  - Nécessite une dtd particulière.
  - Disparait dans HTML 5

- Principe :
  - Un découpage (<frameset> ... </frameset>) défini dans un fichier HTML.
  - Chaque frame du frameset est associée à une balise <frame>.
  - Autant de fichiers html qu'il y a de frames définies dans le frameset.

- Attributs possibles pour la balise <frameset> :
  - rows="99|99%,.." : Liste des tailles de ligne du découpage.
  - cols="99|99%,.." : Liste des tailles de colonne du découpage.

- Attributs possibles pour la balise <frame> :
  - src="url" : url du fichier à afficher dans la frame.
  - noresize : interdit le redimensionnement de la frame.
  - scrolling="auto|yes|no" : barres de défilement verticale et horizontale.

Frames

- Attributs rows/cols
  - * : taille restante
  - Σ > 100% | taille fenêtre en pixels ! règle de 3
  - Σ < 100% | taille fenêtre ! dernière taille ignorée

```
<frameset rows="50%,50%" cols="70%,*">
  <frame src="frame1.html" noresize="noresize">
  <frame src="frame2.html">
  <frame src="frame3.html" scrolling="yes">
  <frame src="frame4.html" scrolling="no">
</frameset>
```

Frames

- Imbrication possible des définitions de framesets.

- On peut associer un nom aux frames :
  - <frame src="..." name="framedegauche">
  - les liens peuvent référencer cette frame à l'aide de l'attribut target (cible)
  - <a href="url" target="framededroite">

- Identificateurs de target réservés :
  - target="_self" : affichage du document dans la frame du lien
  - target="_blank" : affichage du document dans une nouvelle fenêtre
  - target="_parent" : affichage dans la fenêtre contenant le frameset
  - target="_top" : affichage dans la fenêtre du lien
  - Les cibles peuvent aussi servir pour les liens classiques

Images cliquables

- Définition d'un lien (<a href="...">) autour d'une image (<img src="...">)
  - L'image acquiert un cadre (bleu)
    - <img border="0" ...> enlève le cadre
  - Un clic sur l'image provoque le chargement du lien

- Image cliquable sous forme de plan
  - Définition d'un plan (<map> ... </map>)
  - Chaque zone du plan (<area ... >)
    - a une forme : rectangle, cercle ou polygone
    - est associée à une URL
  - Association entre le plan et l'image (<img usemap="..." ... >)

- Formes :
  - Rectangles : coordonnées des coins supérieur gauche et inférieur droit.
  - Cercles : coordonnées du centre et rayon.
  - Polygones : coordonnées de tous les coins.

Images cliquables

- Images cliquables (map.gif) et les fichiers (r.html, …)
  - Rectangle : coordonnées du coin supérieur gauche et inférieur droit.
  - Cercle : coordonnées du centre et rayon.
  - Polygone : coordonnées des sommets du polygone.

```
<img src="map.gif" usemap="#plan">
<map name="plan">
  <area shape="rect" coords="0,0,50,50" href="r.html">
  <area shape="circle" coords="100,100,50" href="c.html">
  <area shape="polygon" coords="60,10,60,50,60,10"
   href="p.html">
</map>
```

Balises obsolètes ou presque

- De nombreuses balises et propriétés sont à éviter :
  - font, basefont, center, s, u, b, i, …
  - align, width, height, size, color, border, background, bgcolor, border, face, name, target, …
  - En clair, tout ce qui décrit la forme plus que le fond
    - On utilise les feuilles de style pour ça.

```
<p>
  <font
    size="10"
    color="green">
  <b>texte</b>
  </font>
</p>
```

```
<p>texte</p>


p {
  font-size: 20px;
  color: red;
  text-transform: uppercase;
}
```

CONCLUSION

LI288 – Web et développement web

Conclusion

- De nombreuses balises pour faire ce qu'on veut :
  - Utiliser les balises dans le bon objectif
    - Une table et une liste n'ont pas le même sens.
- La présentation se fait ensuite avec les feuilles de style.
- Ecrire du code valide :
  - Doc : http://www.w3schools.com/tags/default.asp
  - Validateur : http://validator.w3.org/

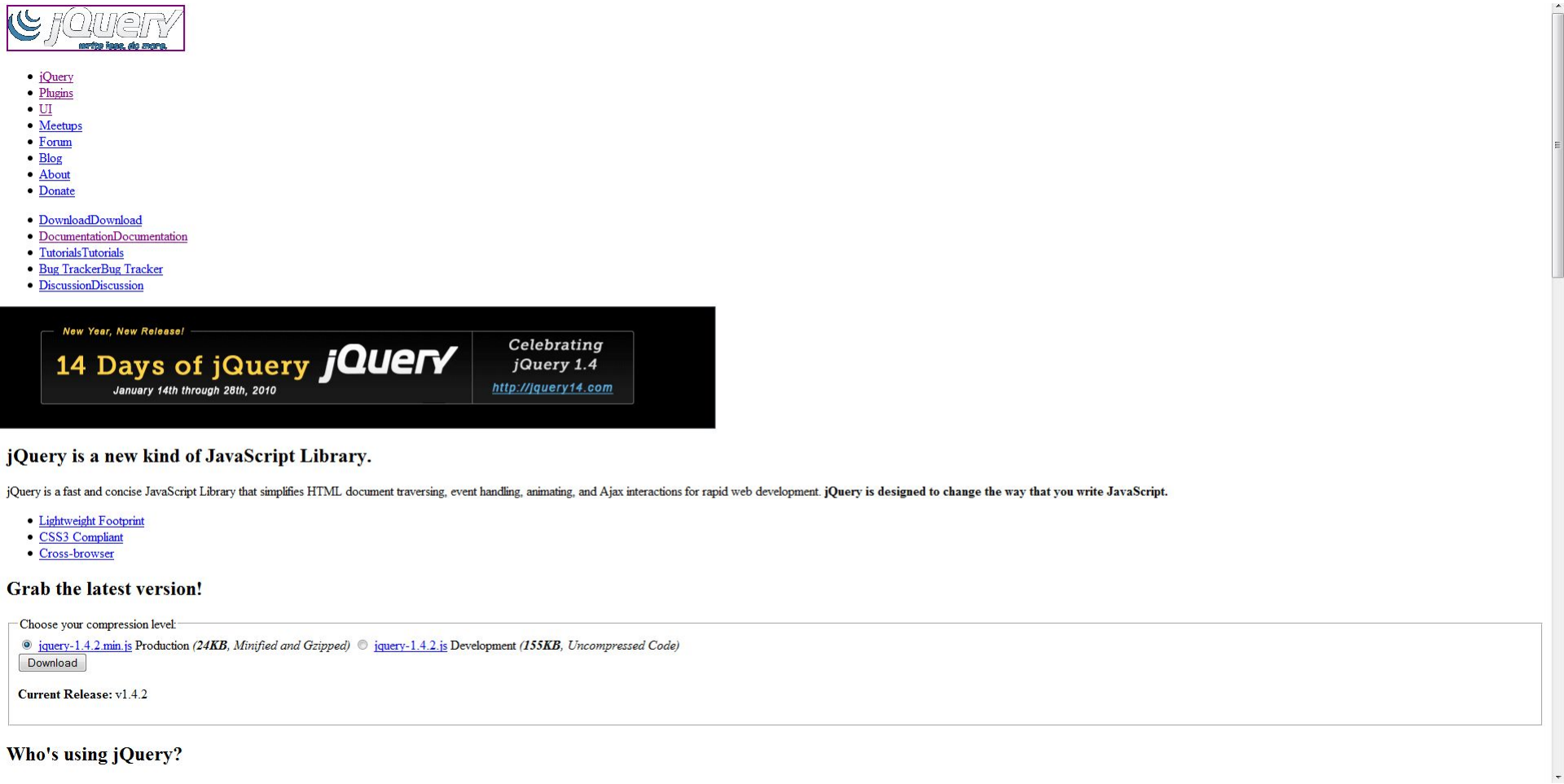
Et HTML 5 ?

- En cours de normalisation, support variable selon les navigateurs :
  - http://caniuse.com/
  - Plutôt tester sur Chrome ou Opéra

- Nouveautés :
  - Code light / moins strict
  - Nouvelles balises sémantiques / multimédia / dessin
  - Formulaires améliorés
  - Drag and drop et édition de contenu
  - Notifications
  - Géolocalisation
  - Création de site hors ligne (avec base de données)
  - …

Version avec CSS

version sans CSS



Markup Validation Service

Check the markup (HTML, XHTML, ...) of Web documents

Jump To: Validation Output

Errors found while checking this document as XHTML 1.0 Transitional!

Result:	1788 Errors, 3479 warning(s)	
Address :	http://www.fnac.com/	
Encoding :	utf-8	(detect automatically)
Doctype :	XHTML 1.0 Transitional	(detect automatically)
Root Element:	html	
Root Namespace:	http://www.w3.org/1999/xhtml	

Options

☐ Show Source ☐ Show Outline ◉ List Messages Sequentially ○ Group Error Messages by Type

☐ Validate error pages ☐ Verbose Output ☐ Clean up Markup with HTML-Tidy

Help on the options is available.

Revalidate

↑ TOP

Markup Validation Service

Check the markup (HTML, XHTML, ...) of Web documents

Jump To: Notes and Potential Issues Validation Output

Errors found while checking this document as HTML5!

Result:	39 Errors, 2 warning(s)	
Address :	http://www.google.com/	
Encoding :	iso-8859-1	(detect automatically)
Doctype :	HTML5	(detect automatically)
Root Element:	html	

Options

☐ Show Source ☐ Show Outline ◉ List Messages Sequentially ○ Group Error Messages by Type

☐ Validate error pages ☐ Verbose Output ☐ Clean up Markup with HTML-Tidy

Help on the options is available.

Revalidate

Notes and Potential Issues

The following notes and warnings highlight missing or conflicting information which caused the validator to perform some guesswork prior to validation, or other things affecting the output below. If the guess or fallback is incorrect, it could make validation results entirely incoherent. It is *highly recommended* to check these potential issues, and, if necessary, fix them and re-validate the document.

Using experimental feature: *HTML5 Conformance Checker.*

Markup Validation Service

Check the markup (HTML, XHTML, ...) of Web documents

Jump To: Validation Output

Error found while checking this document as XHTML 1.0 Strict!

Result:	1 Error	
Address :	http://www.upmc.fr/	
Encoding :	utf-8	(detect automatically)
Doctype :	XHTML 1.0 Strict	(detect automatically)
Root Element:	html	
Root Namespace:	http://www.w3.org/1999/xhtml	

Options

☐ Show Source ☐ Show Outline ◉ List Messages Sequentially ○ Group Error Messages by Type

☐ Validate error pages ☐ Verbose Output ☐ Clean up Markup with HTML-Tidy

Help on the options is available.

Revalidate

↑ TOP

Markup Validation Service

Check the markup (HTML, XHTML, ...) of Web documents

Jump To: Congratulations · Icons

This document was successfully checked as XHTML 1.0 Strict!

Result:	Passed	
Address :	http://jlguillaume.free.fr/www/	
Encoding :	utf-8	(detect automatically)
Doctype :	XHTML 1.0 Strict	(detect automatically)
Root Element:	html	
Root Namespace:	http://www.w3.org/1999/xhtml	

I ♥ VALIDATOR

The W3C validators rely on community support for hosting and development.
Donate and help us build better tools for a better web.

2350 Flattr

Options

☐ Show Source ☐ Show Outline ◉ List Messages Sequentially ○ Group Error Messages by Type

☐ Validate error pages ☐ Verbose Output ☐ Clean up Markup with HTML-Tidy

Help on the options is available.

Revalidate