3I017 - TECHNOLOGIES DU WEB

Les bases HTML

Wednesday 20th February, 2019

Laure Soulier

LIP6

MLIA Machine Learning & Deep Learning for Information Access

Présentation HTML

Les langages HTML et CSS

SCIENCES SORBONNE UNIVERSITÉ

Production partie visuelle du site

Langages HTML et CSS

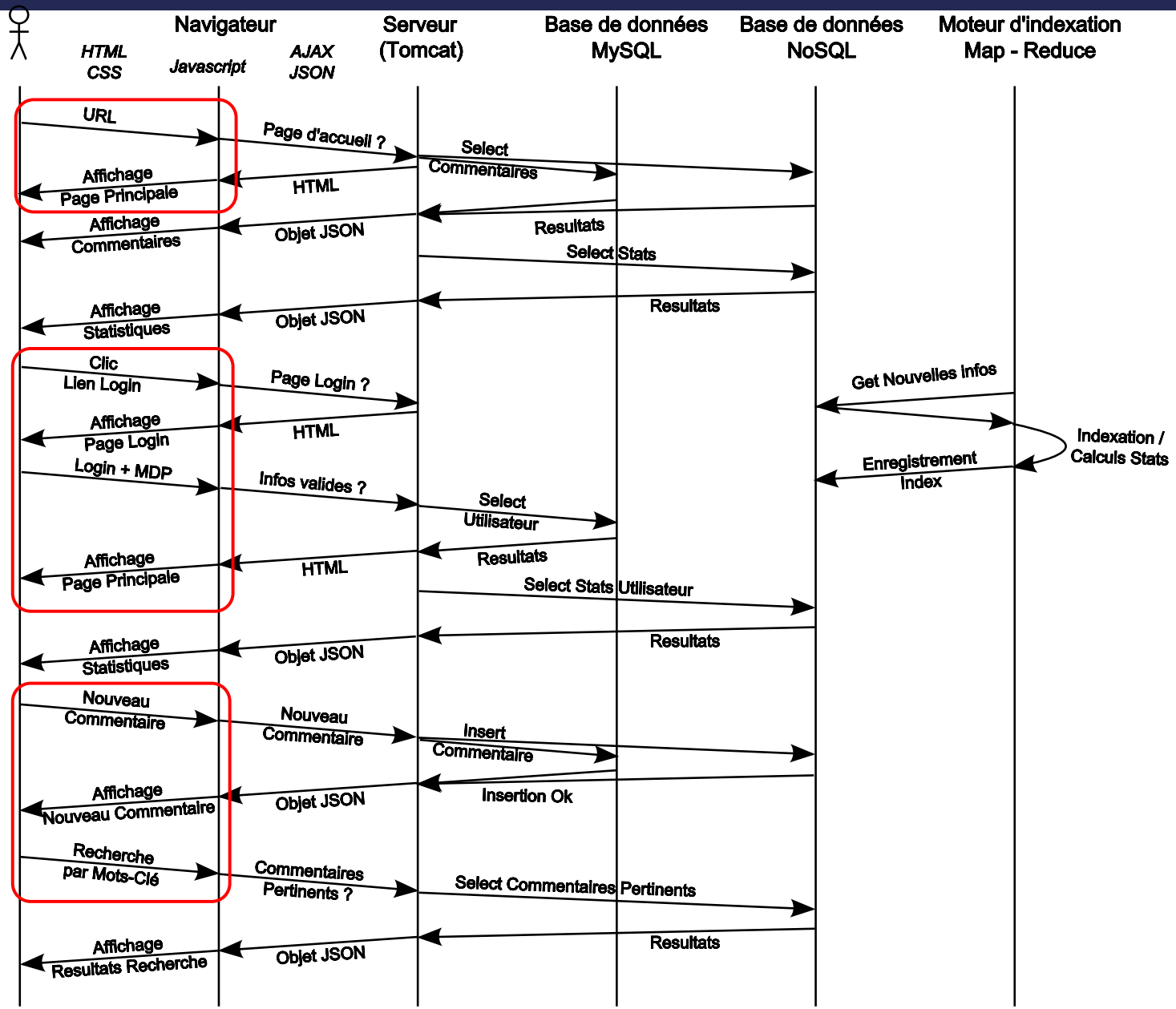
Traduction par l'ordinateur

Résultat visible à l'écran

Les langages HTML et CSS

- Interprétation côté client
- Pris en charge par le navigateur Web
  - Firefox
  - Internet Explorer
  - Google Chrome
  - etc...
- Correspond à la vue du modèle MVC

La Vue : Interface utilisateur



Deux langages complémentaires

- HTML (Hypertext Markup Language : langage de balisage d'hypertexte)
  - Définition des éléments graphiques de la page
  - Création d'hyperliens (liens entre pages Web)
  - Ensemble de balises <HTML> ... </HTML>

- CSS (Cascading Style Sheets : feuilles de style en cascade)
  - Propriétés visuelles des éléments de la page (agencement, positionnement, décoration, couleur, taille du texte...)
  - Ensemble de règles d'association propriété - valeur

Les langages HTML et CSS

HTML
(pas de CSS)

HTML + CSS

Les langages HTML et CSS

- Deux langages pour :
  - Séparer la structure de la page de son style de présentation
  - Décliner les styles selon les clients
  - Permettre des combinaisons (cascades) de styles

⇒ Développement en deux temps
  - HTML pour créer les éléments à afficher...
  - ... Puis CSS pour améliorer le rendu visuel

Hypertext Markup Language

SCIENCES SORBONNE UNIVERSITÉ

HTML

- Une des trois inventions à l'origine du World Wide Web :
  - Hypertext Markup Language (HTML)
  - Hypertext Transfer Protocol (HTTP)
  - Adressage web (URL)
- Origine : 1990
  - Simple structuration du texte en titres, sous-titres, listes ou texte brut
- Aujourd'hui : HTML 5
  - Prise en charge de nombreux medias / technologies riches
- Régi par le World Wide Web Consortium (W3C)

HTML : Ensemble de balises

- Une page HTML est un ensemble de balises
  - ⇒ Déclarer les éléments constituants de la page
  - ⇒ En définir la structure
- Une balise est de la forme :
  $< nom\ attribut_1 = \text{“}valeur\text{”}\ attribut_2 = \text{“}valeur\text{”} ... >$
- Exemple : $< a\ href = \text{“}www.lip6.fr\text{”} >$
- Certaines balises, $< balise > ... < /balise >$, peuvent englober des éléments...
  D'autres, $< balise\ / >$, non.
- Toutes balise doit être fermée, que ce soit par $< /balise >$ ou par $< balise/ >$.

Code HTML

```
<HTML>
    <HEAD>
        <TITLE>Titre de la page</TITLE>
    </HEAD>

    <BODY>
        Contenu de la page
    </BODY>
</HTML>
```

Document Type Definition

- Différents types de balises
  - ⇒ Chaque balise à un rôle / comportement qui lui est propre
  - ⇒ Définies dans la DTD (Document Type Definition) utilisée
- Exemple de DTD :
  http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd
- Déclaration DTD utilisée
  - ⇒ Pour validation W3C
  - ⇒ Par navigateur pour distinguer le type de page HTML

Déclaration DTD

- Différentes DTD selon la version HTML
  - DTD déclarée en première ligne du fichier HTML (Attention : déclaration sensible à la casse).

HTML4

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

HTML5

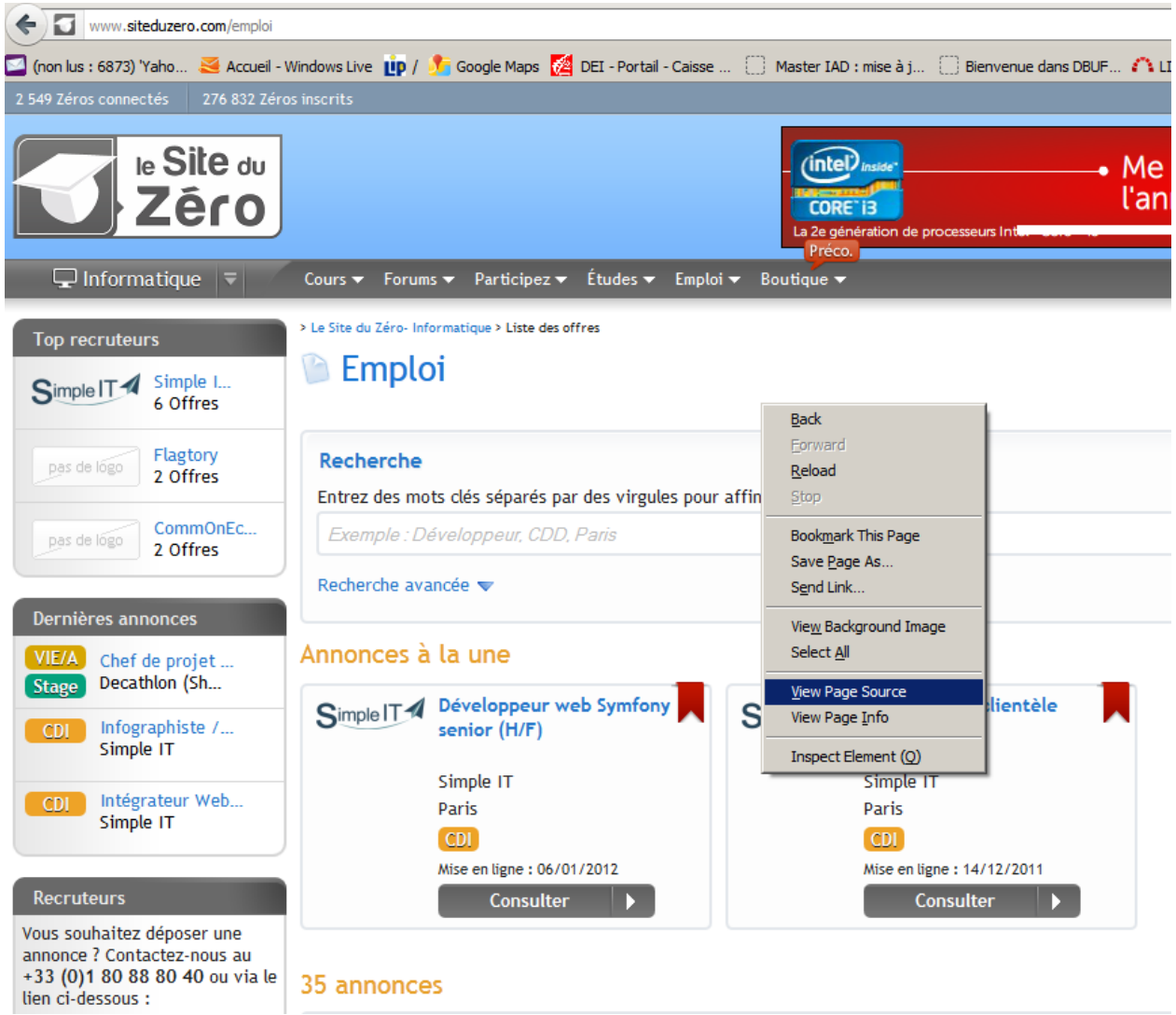
```
<!DOCTYPE html>
```

XHTML1

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

⇒ Pour bien choisir sa DTD : http://www.alsacreations.com/article/lire/560-dtd-doctype-html-xhtml-comment-choisir.html

Visualisation de code HTML



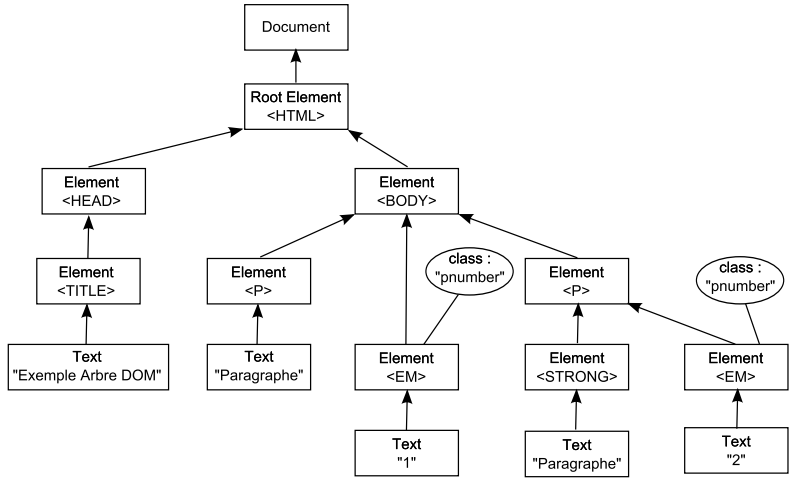
Exemple de code source

Arbre DOM

- L'ensemble de balises d'un fichier HTML peut se représenter sous forme d'arbre

⇒ L'arbre DOM (Document Object Model)

- Noeud, racine, feuille



Code HTML

```
<HTML>
 <HEAD>
  <TITLE>Exemple Arbre DOM</TITLE>
 </HEAD>
 <BODY>
  <p> Paragraphe <em class="pnumber">1 </em> </p>
  <p> <strong> Paragraphe </strong> <em class="pnumber">2 </em> </p>
 </BODY>
</HTML>
```

Catalogue des balises

- Différents types de balises
  - Balises de premier niveau (html, head, body)
  - Balises d'en-tête (title, meta, link, etc...)
  - Balises de contenu (p, a, br, img, etc...)
  - Balises de liste (ul, li, ol, etc...)
  - Balises de tableau (table, td, tr, etc...)
  - Balises de formulaire (form, input, button, etc...)
  - Balises de structure (frameset, div, span, header, etc...)

HTML : Balises de premier niveau

- Balises de premier niveau
  - $<html>...</html>$: Balise principale qui marque le début du document. Englobe l'ensemble du code (sauf déclaration doctype).
  - $<head>...</head>$: En-tête de page. Regroupe les balises d'entête (déclaration titre, mots-clé, fichiers de style, etc...).
  - $<body>...</body>$: Corps de la page. Contient l'ensemble du contenu à afficher.

HTML : Balises d'en-tête

- Balises d'en-tête
  - *< title >...< /title >* : Titre de la page
  - *< style >...< /style >* : Permet d'inclure du code CSS
  - *< link / >* : Permet d'indiquer certaines informations
    - ⇒ Inclure une page de style :
      < link rel = "stylesheet" href = "style.css" / >
    - ⇒ Déclarer un icône pour le site :
      < link rel = "shortcut icon" type = "image/x – icon" href = "icon.ico" / >
    - ⇒ Déclaration de fils RSS, page accueil site, fichier d'aide, etc...
  - *< script >...< /script >* : Permet d'inclure un script
    - ⇒ Du code :
      < script type = " text/javascript" >...< /script >
    - ⇒ Un fichier script :
      < script type = " text/javascript" src = "script.js" >< /script >

HTML : Balises d'en-tête

SCIENCES SORBONNE UNIVERSITÉ

- < *meta* / > permet de :
  - ⇒ Définir différentes propriétés

    Auteur de la page :
    < meta name = "author" content = "Sylvain Lamprier" / >

    Description de la page :
    < meta name = "description" content = "Cours Technos Web" / >

    Mots-clé de la page :
    < meta name = "keywords" content = "Web, HTML, CSS, Javascript, AJAX, Tomcat" / >

    Adresse de contact :
    < meta name = "reply – to" content = "Sylvain.Lamprier@lip6.fr" / >

  - ⇒ Spécifier l'encodage des caractères :

    < meta charset = "utf – 8" >

  - ⇒ Empêcher la mise en cache de la page :

    < meta http – equiv = "pragma" content = "no – cache" / >

  - ⇒ Forcer le rafraîchissement régulier de la page :

    < meta http – equiv = "refresh" content = "10; URL = http : //www.lip6.fr" / >

HTML : Balises de contenu

- Balises de contenu
  - $<p>...</p>$: Paragraphe
  - $<pre>...</pre>$: Paragraphe affiché tel qu'il a été tapé dans le code
  - $<h\$>...</h\$>$: Titre de niveau \$ (avec $\$ \in \{1..6\}$)
  - $<a\ href="\$">...$: Hyperlien vers une page (adresse \$)
  - $<a\ name="\$">...$: Ancre nommée \$ (accessible par un lien pointant sur #\$)
  - $<br\ />$: Passage à la ligne
  - $<hr\ />$: Ligne de séparation horizontale
  - $<img\ />$: Insertion d'une image

    $<img\ src="images/image1.png"\ alt="Image1"\ />$
  - $<iframe\ />$: insertion d'une page html externe
  - $^{...}$: Mise en exposant
  - $_{...}$: Mise en indice
  - Nombreuses autres balises de mise en forme $<strong>$, $<em>$, $<cite>$, $<blockquote>$, etc...

HTML : Balises de contenu

Code HTML

```
<!doctype html>
<html>
      <head>
            <title> Exemple de page HTML </title>
      </head>
      <body>
            <h1> Titre de niveau 1 </h1>
            <h2> Titre de niveau 2 </h2>
            <h3> Titre de niveau 3 </h3>
            <h4> Titre de niveau 4 </h4>
            <h5> Titre de niveau 5 </h5>
            <h6> Titre de niveau 6 </h6>
            Ici les        multiples espaces et

            retours    la ligne ne sont pas pris en compte. <br /> Pour passer
               la ligne, utiliser &lt; br /&gt;
            <p> ou placer dans un paragraphe. </p>
            <pre> Ici par contre,                les
            espaces et retours    la ligne
             sont          pris en compte car on est dans
             un bloc &lt; pre &gt;. </pre>
             <hr />
             Enfin, voici un <a href="Exemple_balises_contenu2.html"> hyperlien

       </body>
</html>
```

HTML : Balises de contenu

Titre de niveau 1

Titre de niveau 2

Titre de niveau 3

Titre de niveau 4

Titre de niveau 5

Titre de niveau 6

Ici les multiples espaces et retours à la ligne ne sont pas pris en compte.
Pour passer à la ligne, utiliser < br />

ou placer dans un paragraphe.

```
Ici par contre,               les
           espaces et retours à la ligne
            sont         pris en compte car on est dans
            un bloc < pre >.
```

Enfin, voici un hyperlien

Caractères spéciaux

- Le standard HTML respecte le codage des caractères ASCII 7 bits
- Pas de caractères accentués → codage particulier
- Code hexadécimal : &xxxx;

Caractères généraux

caractère	code texte	code numérique	commentaire
			espace insécable
			tiret de césure optionnelle (permet au navigateur de couper le mot au bon endroit si besoin de passer à la ligne)
	‎	‎	marque gauche-à-droite
	‏	‏	marque droite-à-gauche
"	"	"	guillemet anglais, guillemet droit (quote)
«	«	«	guillemet français ouvrant
»	»	»	guillemet français fermant
‹	‹	‹	guillemet français simple ouvrant
›	›	›	guillemet français simple fermant
“	“	“	guillemet double ouvrant, guillemet-apostrophe double culbuté
”	”	”	guillemet double fermant, guillemet-apostrophe double
„	„	„	guillemet double fermant bas, guillemet-virgule double inférieur
'	'	'	guillemet simple droit
‘	‘	‘	guillemet simple ouvrant, guillemet-apostrophe culbuté
’	’	’	guillemet simple fermant, guillemet-apostrophe
‚	‚	‚	guillemet simple fermant bas, guillemet-virgule inférieur

Figure 1: https://alexandre.alapetite.fr/doc-alex/alx_special.html

HTML : Balises de listes

- Balises de listes
  - $
...
$: Liste à puces non numérotée
  - $
...
$: Liste à puces numérotée
  - $
...
$: Item de liste à puces
  - $<dl>...</dl>$: Liste de définitions
  - $<dt>...</dt>$: Terme à définir
  - $<dd>...</dd>$: Définition

HTML : Balises de listes

Code HTML

```
<!doctype html>
<html>
      <head>
            <title> Exemple de page HTML </title>
      </head>
      <body>
            <h3> Liste Non Numerot e </h3>
            <ul>
              <li> Element 1 </li>
              <li> Element 2 </li>
              <li> Element 3 </li>
            </ul>

            <h3> Liste Numerot e </h3>
            <ol>
              <li> Element 1 </li>
              <li> Element 2 </li>
              <li> Element 3 </li>
            </ol>

            <h3> Liste D finitions </h3>
            <dl>
              <dt> Terme 1 </dt>
              <dd> Definition terme 1 </dd>
              <dt> Terme 2 </dt>
              <dd> Definition terme 2 </dd>
            </dl>
       </body>
</html>
```

Liste Non Numerotée

- Element 1
- Element 2
- Element 3

Liste Numerotée

1. Element 1
2. Element 2
3. Element 3

Liste Définitions

Terme 1
    Definition terme 1
Terme 2
    Definition terme 2

HTML : Balises de tableau

- Balises de tableau
  - $<table>...</table>$: Déclare un tableau
  - $<caption>...</caption>$: Titre du tableau
  - $<tr>...</tr>$: Ligne du tableau
  - $<td>...</td>$: Case du tableau
    Attribut colspan=n : la case occupe n colonnes
    Attribut rowspan=n : la case occupe n lignes
  - $<th>...</th>$: Case en-tête

HTML : Balises de tableau

Code HTML

```
<!doctype html>
<html>
      <head>
            <title> Exemple de page HTML </title>
      </head>
      <body>
           <table>
                    <caption>Tableau </caption>
                    <tr>
                        <th>Colonne 1</th>
                        <th>Colonne 2</th>
                        <th>Colonne 3</th>
                    </tr>

                    <tr>
                        <td> Case 1,1 </td>
                        <td> Case 1,2 </td>
                        <td> Case 1,3 </td>
                    </tr>

                    <tr>
                        <td> Case 2,1 </td>
                        <td> Case 2,2 </td>
                        <td> Case 2,3 </td>
                    </tr>
           </table>
       </body>
</html>
```

Tableau

Colonne 1	Colonne 2	Colonne 3
Case 1,1	Case 1,2	Case 1,3
Case 2,1	Case 2,2	Case 2,3

HTML : Balises de formulaire

SCIENCES SORBONNE UNIVERSITÉ

- Balises de formulaire
  - < *form* > ... < / *form* > : Déclare un tableau
  - < *label* > ... < / *label* > : Titre d'un élément de formulaire
  - < *input* / > : Champ de formulaire
    - ⇒ Zone de texte d'une ligne < *input type* = "*text*" / >
    - ⇒ Mot de passe < *input type* = "*password*" / >
    - ⇒ Case à cocher < *input type* = "*checkbox*" / >
    - ⇒ Envoi de fichier < *input type* = "*file*" / >
    - ⇒ Bouton d'option < *input type* = "*radio*" / >
    - ⇒ Bouton d'envoi < *input type* = "*submit*" / >
    - ⇒ Bouton de remise à zéro < *input type* = "*reset*" / >
    - ⇒ Champ caché < *input type* = "*hidden*" / >
  - < *textarea* > ... < / *textarea* > : Zone de saisie multiligne
  - < *select* > ... < / *select* > : Liste déroulante
  - < *option* > ... < / *option* > : Element de liste déroulante

HTML : Balises de formulaire

Code HTML

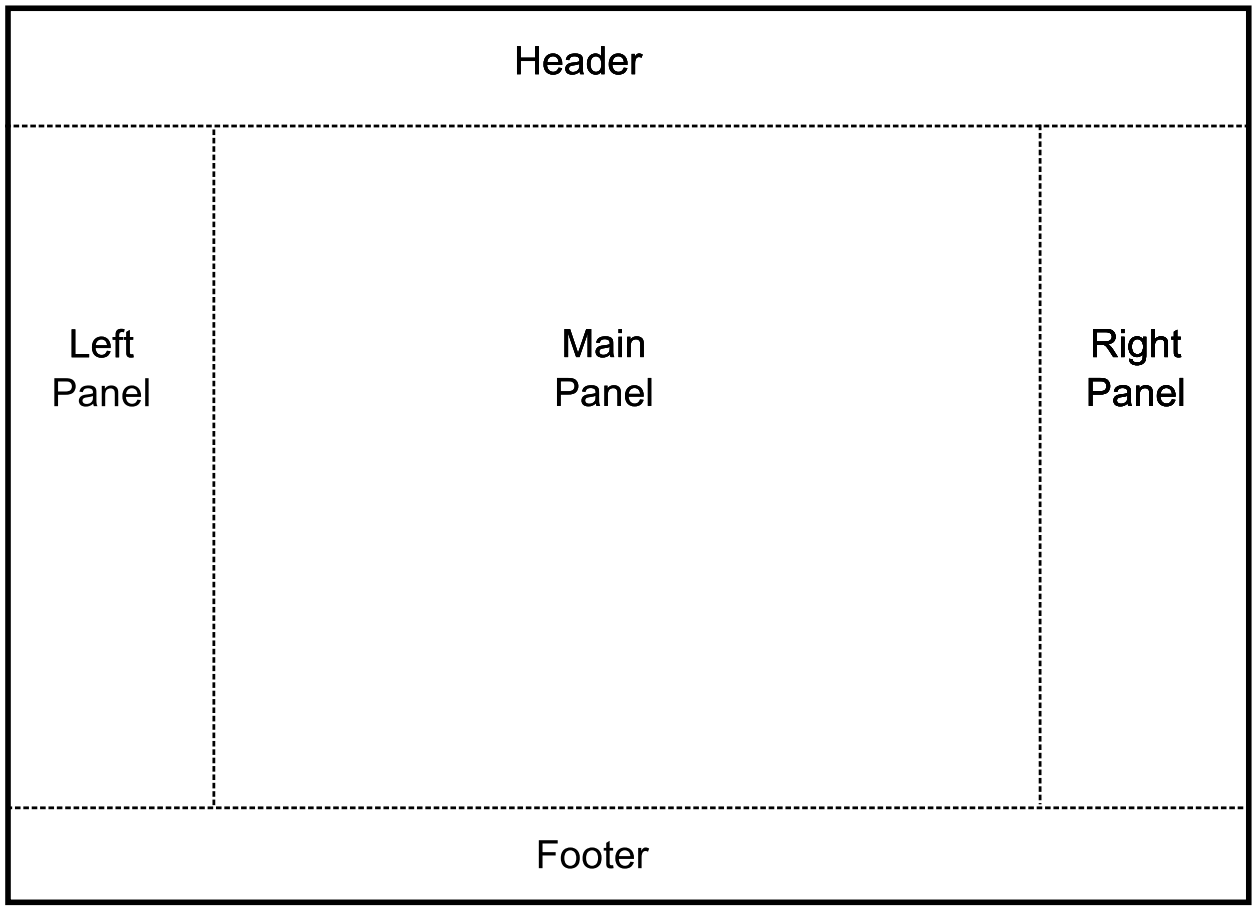
```
<html>
    <body>
    <FORM method=post action="cgi-bin/Enregistrement.pl">
      <TABLE BORDER=0>
      <TR><TD>Nom</TD><TD><INPUT type=text name="nom"></TD></TR>
      <TR><TD>Mot de Passe</TD>
              <TD><INPUT type="password" name="pass"></TD></TR>
      <TR><TD>Sexe</TD><TD>
      Homme : <INPUT type=radio name="sexe" checked value="M"> <br/>
      Femme : <INPUT type=radio name="sexe" value="F"></TD></TR>
      <TR><TD>Fonction</TD>
              <TD><SELECT name="fonction">
                      <OPTION VALUE="enseignant">Enseignant</OPTION>
                      <OPTION VALUE="etudiant">Etudiant</OPTION>
                      </SELECT></TD></TR>
      <TR><TD>Commentaires</TD><TD>
              <TEXTAREA rows="3" name="commentaires">Tapez ici vos commentaires
              </TEXTAREA></TD></TR>
      <TR><TD COLSPAN=2>
              <INPUT type="submit" value="Envoyer">
              <INPUT type="reset"  value="R tablir ">
             </TD></TR>
      </TABLE>
      <INPUT type="checkbox" name="rester" value="true"/> Rester connect
      </FORM>
  </body>
</html>
```

Balises de structures

HTML : Balises de structure

- Balises de structure
  - Servent à structurer les pages
  - Disposition des éléments les uns par rapport aux autres
- Différentes balises
  - < *frameset* >, < *frame*> pour une structure par cadres
  - < *div* >, < *span* > : boîtes génériques permettant de regrouper des éléments
  - Balises structurantes < *header* >, < *footer* >, < *aside* >, etc...

Structuration de pages Web



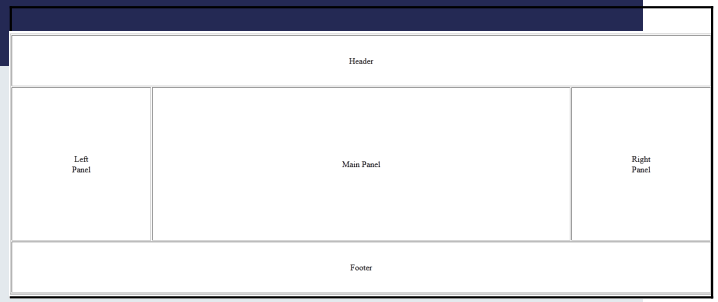
Structuration de pages Web

SCIENCES SORBONNE UNIVERSITÉ

- Différentes possibilités
  - Structuration par tableau
  - Utilisation de cadres (frames)
  - Utilisation du "flux" naturel
  - Usage de balises structurantes
  - Positionnement / Structuration CSS

Structuration par tableau

Code HTML




```
<!doctype html>
<html>
  <head>
    <title> Structuration par tableaux </title>
  </head>
  <body>
      <TABLE BORDER=1  style="width:100%;height:500px">
      <TR ALIGN="center" style="height:20%"><TD colspan="3">  Header </TD>  </TR>
      <TR ALIGN="center" style="height:60%">
           <TD style="width:20%"> Left <br /> Panel </TD>
           <TD style="width:60%">Main Panel</TD>
           <TD style="width:20%"> Right <br /> Panel </TD></TR>
      <TR ALIGN="center" style="height:20%"><TD colspan="3"> Footer </TD> </TR>
      </TABLE>
  </body>
</html>
```

- Possibilité d'imbriquer des tableaux pour structures plus complexes
- Structuration par tableau à éviter
  - ⇒ Structure figée (difficilement modifiable, problèmes de portabilité)
  - ⇒ Coûteux en ressources machine
  - ⇒ Code complexe

(Autre exemple sur slide "HTML: Balises de formulaire")

Structuration par Frames

- Code page distribué sur différents fichiers
  - Un fichier principal décrivant la structure
  - Des fichiers à afficher dans les différents cadres

- Deux balises :
  - $<frameset>...</frameset>$: défini la manière de disposer les cadres

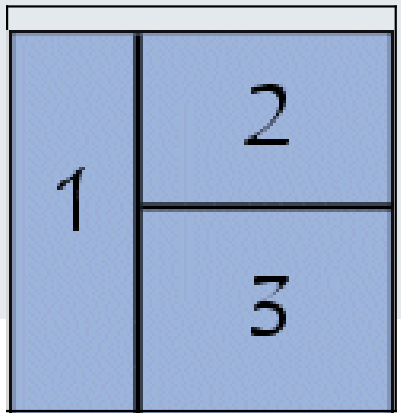
    Attribut rows="$x_1, x_2, ..., x_n$" : Dispose les cadres les uns en dessous des autres (les valeurs x correspondent aux hauteurs des différents cadres)
    Attribut cols="$x_1, x_2, ..., x_n$" : Dispose les cadres les uns à côté des autres (les valeurs x correspondent aux largeurs des différents cadres)
  - $<frame/>$: Correspond à un cadre L'attribut src permet de spécifier la fichier html à charger dans le cadre
    L'attribut name permet de donner un nom au cadre

- Si un cadre correspond à un menu ses liens doivent spécifier le cadre cible :
  $<a\ href="accueil.html"\ target="main">$, où main correspond au cadre principal

Structuration par Frames

Exemple de structuration par Frames

```
<frameset cols="30%,*">
  <frame src="menu.html">
  <frameset rows="40%,*">
    <frame src="top.html">
    <frame src="main.html">
  </frameset>
</frameset>
```



- Structuration par frames à éviter
  - Rendu peu esthétique / démodé
  - Pages mal indexées par moteurs de recherche
  - Problèmes d'impression, de navigation, d'accessibilité
- Exemples de sites avec Frames :
  - www.angelfire.com/super/badwebs/
  - www4.tripnet.se/~slarti/FrameEx/Bad/Uk/Top/frameset.htm

Structuration selon le flux naturel

- Flux naturel => Comportement par défaut des navigateurs
- En flux naturel, deux facteurs sont considérés pour positionner les éléments :
  - L'ordre d'apparition des balises
    - ⇒ Détermine l'ordre d'apparition des éléments
  - Le type des balises
    - ⇒ Détermine la manière dont s'enchainent les éléments (horizontalement ou verticalement)
- Quatre grands types de balises :
  - inline
  - block
  - inline-block
  - none
- Possibilité de changer le type d'une balise par la propriété CSS "diplay"

Structuration selon le flux naturel

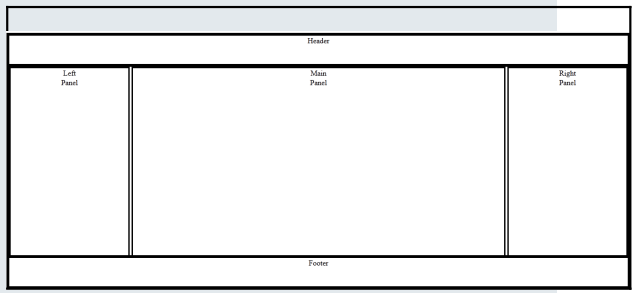
- Éléments inline :
  - Se placent les uns à côté des autres
  - Occupent uniquement la place du texte contenu
  - ⇒ Balises de contenu , , <a>, <img />, etc...
  - ⇒ Balises <input>, <label>, <textarea>, <legend>
  - ⇒ **Balise inline générique **
- Éléments block :
  - Se placent les uns en-dessous des autres
  - Occupent tout l'espace horizontal disponible
  - Peuvent être redimensionnés
  - ⇒ Balises de contenu <h1>, .., <h6>, <p>, <pre>, etc...
  - ⇒ Balises <ul>, <ol>, <dl>, <dd>,<dt>, <form>, <option>, <table>, <header>, <aside>, etc...
  - ⇒ **Balise block générique **
- Éléments inline-block :
  - Se placent les uns à côté des autres
  - Peuvent être redimensionnés
  - ⇒ Balises de formulaire <select> et <input>
- Éléments none :
  - Non affichés, n'occupent aucune place
  - ⇒ Balises <head> et balises d'en-tête

Structuration selon le flux naturel

Exemple de structuration selon le flux naturel

```
<!doctype html>
<html>
  <head>
    <title> Flux naturel </title>
    <style>
        *{border:solid; text-align:center;}
    </style>
  </head>
  <body>
      <div style="height:60px;"> Header </div>
      <div style="height:400px">
         <div style="display:inline-block;width:19%;height:100%">
                Left <br /> Panel </div>
         <div style="display:inline-block;width:60%;height:100%">
                Main <br /> Panel </div>
         <div style="display:inline-block;width:19%;height:100%">
                Right <br /> Panel </div>
      </div>
      <div  style="height:60px"> Footer </div>

  </body>
</html>
```



- Structuration à préférer la plupart du temps
  - Facilement modifiable et très portable
  - Presque tout se passe dans le CSS

HTML5

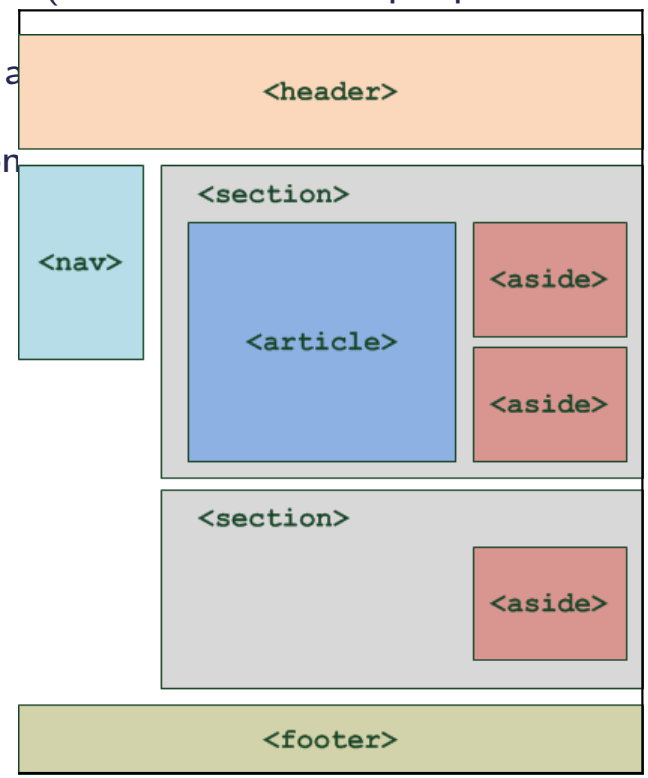
Balises structurantes

- Depuis HTML 5.0, nouvelles balises de type block :
  - < *header* > ... < /*header* > pour définir l'en-tête de la page
  - < *footer* > ... < /*footer* > pour définir le pied de page
  - < *nav* > ... < /*nav* > visant à contenir des liens de navigation
  - < *section* > ... < /*section* > pour regrouper du contenu (selon une thématique par exemple)
  - < *article* > ... < /*article* > pour englober une portion a journal par exemple)
  - < *aside* > ... < /*aside* > pour apporter des information
- Structure envisageable avec ces balises :
- Attention ! Rien n'est défini par défaut, Tout reste à positionner avec CSS
- Equivalent à utiliser des partout...

⇒ Mais meilleure lisibilité du code

⇒ Évolutions possibles des navigateurs

⇒ Structure plus riche



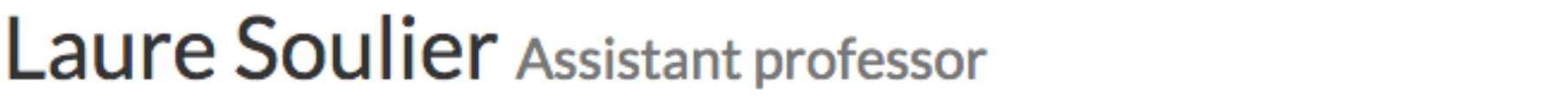
Avant / Après HTML5

Avant :

```
<div id="header">
<h1>Laure Soulier</h1>
<p class="tagline">Assistant professor</p>
</div>
```

Avec HTML5 :

```
<header>
<h1>Laure Soulier</h1>
<h2>Assistant professor</h2>
</header>
```



Avant / Après HTML5

Avant :

```
<div id="footer">
<p>Copyright Laure Soulier 2016 - Last update: July, 17th
</div>
```

Avec HTML5 :

```
<footer>
<p>Copyright Laure
</footer>
```

Copyright © Laure Soulier 2016 - Last update: July, 17th 2017

Avant / Après HTML5

Avant

```
<div     id="nav">
        <ul>
                <li><a   href="">About</a></li>
                <li><a   href="">Research</a></li>
                <li><a   href="">Publications</a></li>
                <li><a   href="">Services</a><li>
        ....
        </ul>
</div>
```

Avec HTML5

```
<nav>
    <ul>
                <li><a   href="">About</a></li>
                <li><a   href="">Research</a></li>
                <li><a   href="">Publication</a></li>
                <li><a   href="">Services</a></li>
        ...
        </ul>
</nav>
```