

HTML

Developpement web



Association Acos Parla Html

Objectifs



- Prise en main des outils
- Créer une page HTML
- découper une page HTML en utilisant balises de base et de section
- Valider vos pages web
- Créer un formulaire avec HTML

HTML

PLAN



- Introduction
- Les balises
- Balises fondamentales pour structure minimale
- Organiser son texte
- Illustrer vos contenus
- Créer des liens
- Tableau
- Découper votre page web (HTML5)
- Exercice: créer votre C.V
- Validation
- Formulaire (HTML5)
- Audio & Video

HTML

INTRODUCTION



Présentation

- Langage informatique **descriptif** (≠ langage de programmation, de requête)
- Utilisé pour structurer le contenu de page web
- Concept Hypertext : liens entre les pages
 - **H**YPER
 - **T**EXT
 - **M**ARKUP
 - **L**ANGUAGE

Page H.T.M.L

```
<!DOCTYPE html>  
<html>  
  <head>  
    .....  
  </head>  
  <body>  
    .....  
  </body>  
</html>
```

Comment écrire du html ?

- Un fichier HTML se définit avec l'extension .html dans lequel l'information contenu dans la page, sera entouré dans des balises.

Les balises donnent des instructions à l'ordinateur:

"aller à la ligne", "afficher une image"

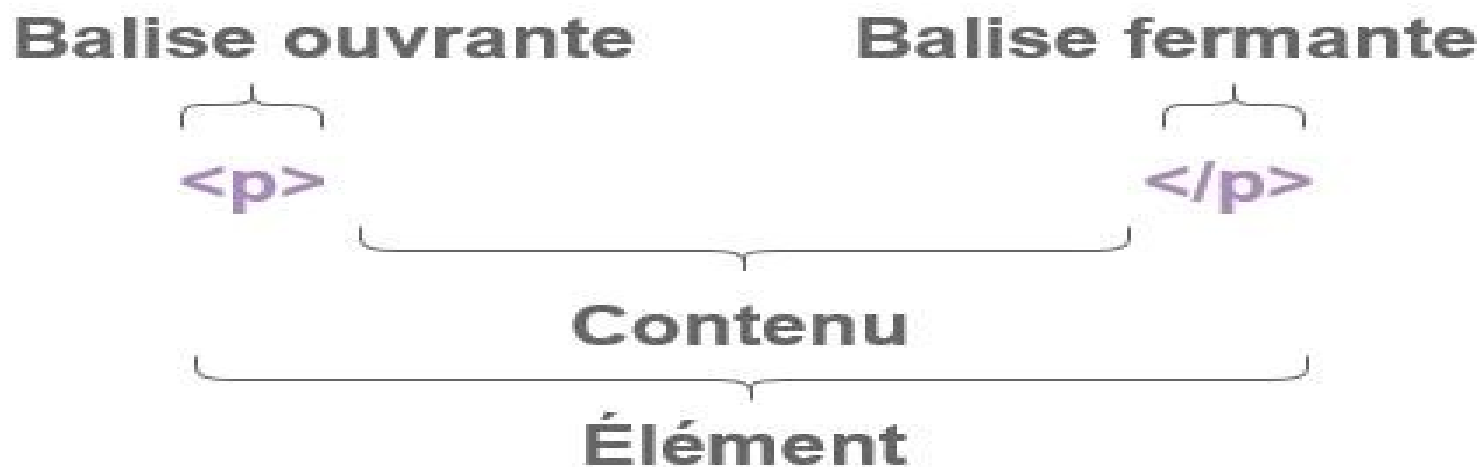
Langage à Balises

Le HTML est un langage à balise

Des balises assemblés autour du contenu

Exemple:

Pour indiquer que le contenu appartient à un paragraphe, il est placé entre balises ici `<p>` : l'ensemble se nomme élément



Sémantique/structuration

Le langage HTML utilise des balises prédéfinies -
Toutes ont une utilité et un sens: **sémantique**.

La sémantique décrit la structure d'un document
Respecter la sémantique des balises, c'est
encadrer un contenu avec des balises adaptés.

- Conséquence :
 - le document est structuré correctement
 - accessibilité et référencement sont améliorés

Accessibilité

L'accessibilité, c'est construire des sites web qui fonctionnent quelques soient le navigateur et le système d'exploitation que vous utilisez, et quel que soit le handicap que vous pourriez avoir. Il s'agit de respecter les besoins et préférences personnelles différents de personnes différentes. Tout le monde n'utilise pas le web de la même manière, ni avec le même équipement. **Roger**

Johansson Swedish web professional

Si les moteurs de recherche arriveront à lire le contenu de votre site (textes, images, liens) facilement , ils vous référenceront plus facilement.

Outils: éditeur de page H.T.M.L

Simple éditeurs de texte :

- Notepad++
- SublimeText
- TextMate



I.D.E :

- Adobe Dreamweaver
- WebStorm



Framework:

- Visual studio 2015
- Eclipse



Navigateurs

- Programme informatique
- Reçoit et affiche les pages du Web
- Utilise des hyperliens pour naviguer de page en page.

3 principaux:

Internet Explorer :



Mozilla Firefox :



Google Chrome :



Mise à jour du H.T.M.L

Comme de nombreux autres langages de nouvelles versions apparaissent:

2 groupes s'intéressent à l'évolution du HTML et des technologies associées :

- W3C



- WHATWG



HTML Living Standard

En mai 2007, W3C adopte le HTML5 de WHATWG comme point de départ de leur travail.

- Depuis, W3C et WHATWG travaillent ensemble sur le développement de HTML5.
- HTML5 est aussi nommé:
 - HTML Living Standard
 - Plus de nouvelle version (pas de HTML6).
Juste amélioration de HTML Living Standard.

De HTML à XHTML



X.H.T.M.L acronyme pour

EXtensible HyperText Markup Language

- XHTML évolution du langage HTML
- XHTML est plus strict que HTML :
évite erreurs d'interprétation des navigateurs
- XHTML supporté par la plupart des navigateurs
- Respecte la syntaxe du langage X.M.L (eXtensible Markup Language)

X.M.L

- en X.M.L, les balises indiquent la structure du document et non sa présentation qui reste le rôle de la feuille de style
- Plus stricte que le h.t.m.l:
 - Présence de toutes les balises qui définissent la structure(nt html, body...)
 - Fermeture de toutes les balises
 - Nom de balise et attribut toujours écrit en minuscules.
 - Valeurs des attributs sont entre guillemets
- Le langage HTML n'a pas à vocation à être enrichi mais le XML permet de créer son propre vocabulaire de balises personnalisables.

Exemple d'utilisation: fichiers de configurations pour application web (j.e.e / j.s.f / descripteur de déploiement...)

```
<welcome-file-list>  
  <welcome-file>index.html</welcome-file>  
  <welcome-file>index.jsp</welcome-file>  
</welcome-file-list>
```


Exemple XHTML

Pour convertir H.T.M.L vers X.H.T.M.L

- Ajouter XHTML <!DOCTYPE> en début de page H.T.M.L:

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

- Ajouter l'attribut **xmlns** à la balise html:

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

HTML

BALISES



Association Acos Parla H.T.M.L

Utilisation

- Les balises se délimitent par `<` et `>`
`<tag_name>`
- Toutes les balises doivent être:
 - fermées
 - correctement imbriquées:
`<p>Imbrication de balise</p>`
- Les noms des balises s'écrivent en minuscule
- Eviter balises ou d'attributs obsolètes qui ne sont plus interprétées par les navigateurs.
 - remplaçables par des équivalents CSS.

classification de balises:

- **Par Paire:**
s'ouvrent, contiennent du texte, et se ferment
`<p>mon paragraphe</p>`
- **Orpheline:** (self-closing tag)
insère un élément à un endroit précis
Pas de balise fin se termine avec /
``
`<input type="button"/>`
- **Autre classification:**
 - inline:** exemple liens `<a>`.
 - block:** exemple paragraphes `<p></p>`.

Balise inline

- Formatage des données
- Affichage sur un axe horizontal
- se trouve obligatoirement à l'intérieur d'une balise block.
- ne crée pas de retour à la ligne
 - le contenu de la balise suivante s'écrit à la suite du texte précédent, sur la même ligne d'où le nom balise «en ligne ».

Exemples: ``, ``, etc...

Balise block

- Affiche du contenu
 - sur un axe vertical
 - crée automatiquement un retour à la ligne
- Permet de structurer la page web avec une succession de blocs les uns en dessous des autres
- Exemples: `<p>`, `<h1>` , `<title>` , ...

Exemple: inline/block

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>Page Title</title>
```

```
</head>
```

```
<body>
```

```
<h1>My First Heading</h1>
```

```
<p>My first paragraph<a href="#">Lien Inline</a>
```

```
</p>
```

```
</body>
```

```
</html>
```



My First Heading

My first paragraph [Lien Inline](#)

Les balises universelles

Ce sont des balises qui n'ont aucun sens particulier (contrairement à `<p>` qui signifie " paragraphe"....)

Le principal intérêt de ces balises est que l'on peut leur appliquer une class/un id) (voire cours CSS) quand aucune autre balise ne convient.

Il existe 2 balises génériques avec pour différence l'une est **inline** et l'autre est **block** :

- ``(inline) ;
- `<div></div>`(block).

Les attributs

Les attributs complètent les balises en donnant des informations supplémentaires.

L'attribut se place après le nom de la balise ouvrante et a le plus souvent une valeur

Toute valeur affectée à un attribut doit être entre guillemets :

```
<balise attribut="valeur">
```

```
<image nom="photo.jpg" />
```

```
<p align="left" </p>
```

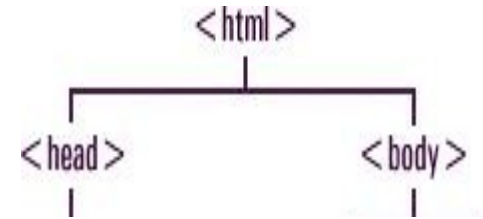


HTML

BALISES FONDAMENTALES POUR STRUCTURE MINIMALE

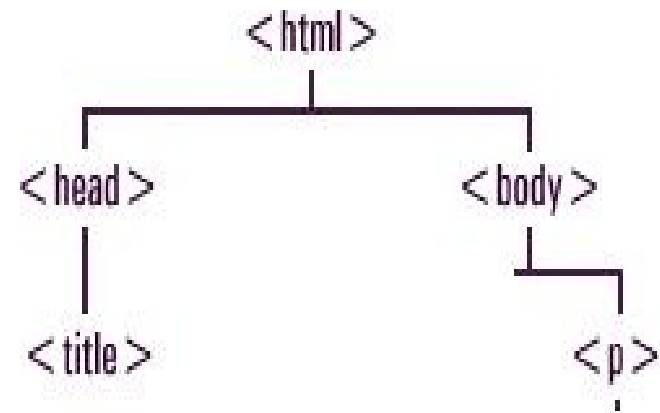
Présentation

- La balise `<!DOCTYPE html>`
 - Définit le type de document
- La balise `<html>`
 - Encapsule tous les autres balises, élément racine
- La balise `<head>` inclus dans `<html>`
 - Définit les propriétés de la page
- La balise `<body>` inclus dans `<html>`
 - Encapsule tous les éléments visibles.



Exemple

```
<!DOCTYPE html>
<html>
  <head>
    <title>Page Title</title>
  </head>
  <body>
    <p>Bonjour tout le monde </p>
    .....
  </body>
</html>
```



Balise `<html>`

```
<!DOCTYPE html>
```

Signale au navigateur que nous écrivons en HTML

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

- *Block* `<html>`
 - Racine du document
 - Contient les balises `<head>` et `<body>`

Balise `<head>`

```
<head>
...
</head>
```

Définit les propriétés de la page :

- `<title>...</title>` Titre de la Page
- `<style>...</style>` Inclut du code C.S.S utilisable sur cette seule page
- `<link href="style.css" />`
- `<script>` Inclut du code JavaScript
- `<meta charset="utf-8" />` encodage

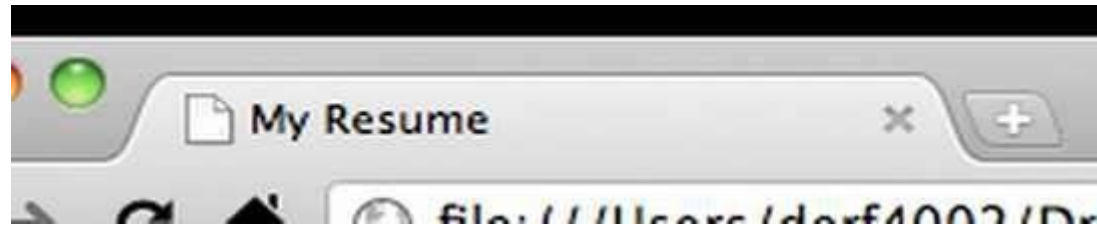
Utf-8 affiche tous les symboles de toutes les langues

Balise `<head>`

- Titre de la page dans la barre du navigateur

```
<!DOCTYPE html>
<html>
  <head>
    <title>My resume</title>
  </head>
  <body>

  </body>
</html>
```



Balise `<body>`

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

Balise (type Block) marque le début et la fin du corps du document. Elle contient tous les éléments décrivant le contenu du document:

- Paragraphs
- Images

Liste des joueurs enregistrés

Rafael Nadal

Novak Djokovic

Gaël Monfils

Sérèna Williams

Kiki Bertens

HTML

ORGANISER SON TEXTE

PLAN



- rédiger des paragraphes ;
- donner de l'importance à certains mots de son texte;
- structurer sa page avec les titres ;
- organiser les informations sous forme de listes.

Paragraphe

`<p>...</p>`

- Balises définissent un paragraphe
 - retour à la ligne,
 - supprime les espaces

Exemple:

`<p>`This paragraph contains a lot of lines
and spaces in the source code, but the
browser ignores it.`</p>`



This paragraph contains a lot of lines and spaces in the source code, but the browser ignores it.

Les espaces ne sont pas interprétés
pour préserver les espaces utilisez ` `;
(Non Breakable Space)

Exemple :`<p>`

This paragraph contains ` ` a lot
of lines and spaces `</p>`



This paragraph contains a lot of lines and spaces

break row

`
`

- Utilisé pour retourner à la ligne :
- obligatoirement dans un paragraphe.

```
...  
<body>  
  <p>First Paragraph</p>  
  <p>Second <br />  
  Paragraph</p>  
</body>  
...
```



First Paragraph

Second
Paragraph

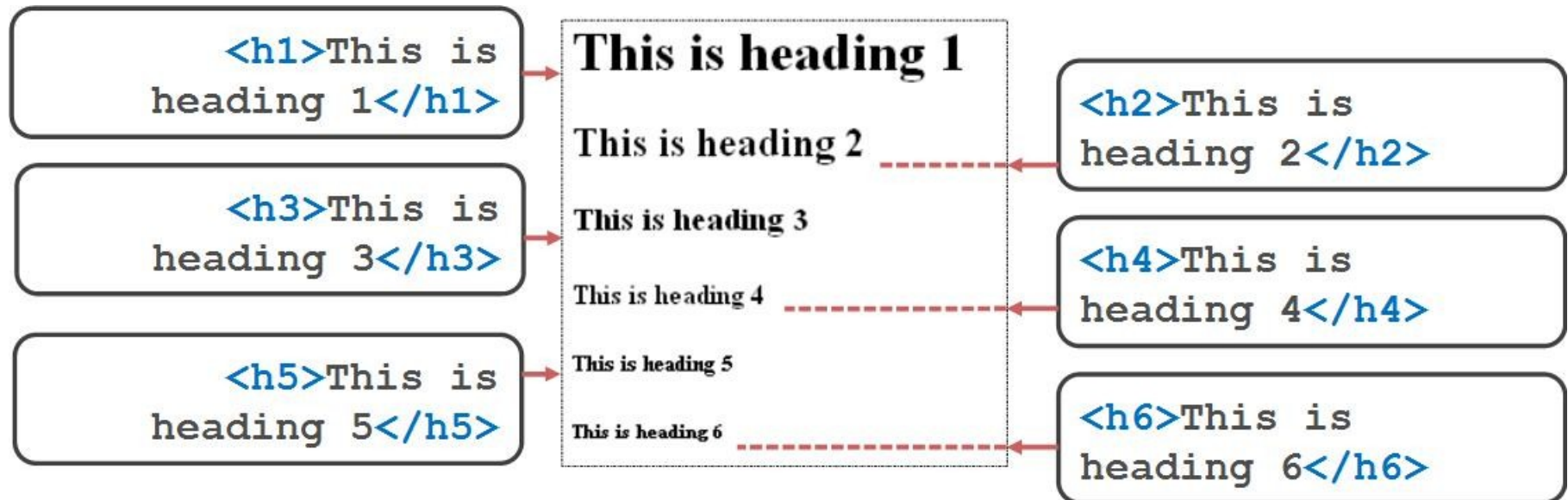
*Bonjour, je voudrais un pack de
 s'il vous plaît.*



Titres dans la page

`<hn>...</hn>` $n=\{1;2;3;4;5;6\}$

- 6 balises définissent les titres



Ne confondez pas avec la balise `<title>` les titres sont affichés dans la page web.

Mettre en valeur

Avec la balise `..`

→ signifie important

La plupart des navigateurs choisissent d'afficher en gras les mots importants pour les mettre en valeur mais il pourrait aussi bien l'afficher en rouge.

``ce texte est important``

La ligne de code s'interprète comme important et non comme mettre en gras

Liste `` ``

- `...`

Définit une liste Ordonnée 1, 2,..... **Front end language**

```
<h3>Front end language</h3>
<ol>
  <li>HTML</li>
  <li>CSS</li>
  <li>Java Script</li>
</ol>
```

1. HTML
2. CSS
3. Java Script

- `...`

Définit une liste non ordonnée : Unordered

```
<h3>Back end language</h3>
<ul>
  <li>PHP</li>
  <li>Ruby</li>
  <li>JavaScript</li>
</ul>
```

Back end language

- PHP
- Ruby
- JavaScript



- **Menu horizontale à partir d'une liste:**

```
<ul>  
  <li><a href="#home">Home</a></li>  
  <li><a href="#news">News</a></li>  
  <li><a href="#contact">Contact</a></li>  
  <li><a href="#about">About</a></li>  
</ul>
```



- Utilisation des c.s.s:



Liste déroulante: dropdownlist

- Balises:

`<select></select>`

`<optgroup></optgroup>`

`<option></option>`

- Attributs:

- disabled
- label
- selected
- value

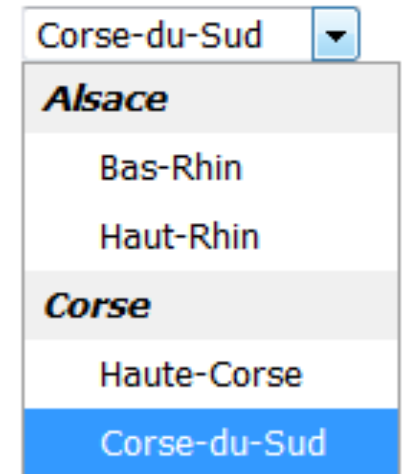


The screenshot shows a web form with a light blue background. At the top, there are two text input fields. The first field contains the text 'maradonna' and the second field contains 'diego'. Below these fields, there is a section for 'Sexe:' with two radio buttons: 'Femme' (unselected) and 'Homme' (selected). Below the 'Sexe:' section, there is a 'Nation' label and a dropdown menu. The dropdown menu is open, showing a list of countries: France (selected), Allemagne, Belgique, Cuba, Danemark, Espagne, Estonie, Etats Unis, Finlande, and France. To the left of the dropdown menu, there is a button labeled 'Enregistrer le jeu'. The bottom of the form has a dark orange background.

choix fait par l'internaute à toute sa place dans un formulaire

Liste déroulante: exemple

```
<select name="FrenchDepartments">
  <optgroup label="Alsace">
    <option value="67">Bas-Rhin</option>
    <option value="68">Haut-Rhin</option>
  </optgroup>
  <optgroup label="Corse">
    <option value="2B">Haute-Corse</option>
    <option value="2A" selected>Corse-du-Sud</option>
  </optgroup>
</select>
```



<http://acosparla.alwaysdata.net/html/Option.html>

Enregistrer des
joueurs



Enregistrer des
arbitres



Enregistrer des
courts



HTML

ILLUSTRER LES CONTENUS

Images

La balise `` permet d'afficher une image

- attributs:
 - `src="url"`
 - `alt="alternative text"` // si image ne s'affiche pas
l'attribut `alt` (pour alternatif) contient un texte descriptif de l'image à l'intention des mal-voyants, il se doit d'être le plus précis possible **référencement**)
 - `title`: info bulle au survol de l'image

```

```

Images

- les images sont des éléments « in line »



- Cliquer sur une image est possible (like a button cf next chapitre)

optimisez la taille de vos images



Copyright

Illustrer les contenus

Customisez vos propres puces

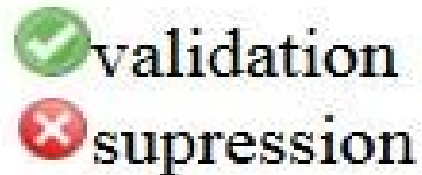
imbriquez la balise **** dans une balise ****

```
<ol>
```

```
  <li>validation</li>
```

```
  <li>supression</li>
```

```
</ol>
```



HTML

CRÉER DES LIENS



Présentation

`<a>...`

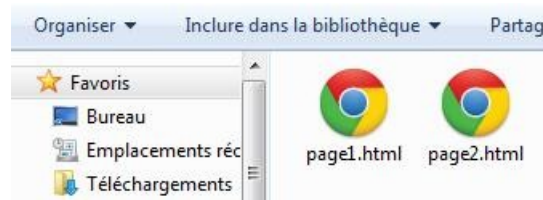
- Utilisé pour créer:
 - Un lien hypertexte avec une ressource :
 - un fichier à télécharger
 - un élément sur la même page
 - une autre page
 - ...

Lien externe : exemple

- lien chemin relatif pour accéder à la ressource:

page1.html

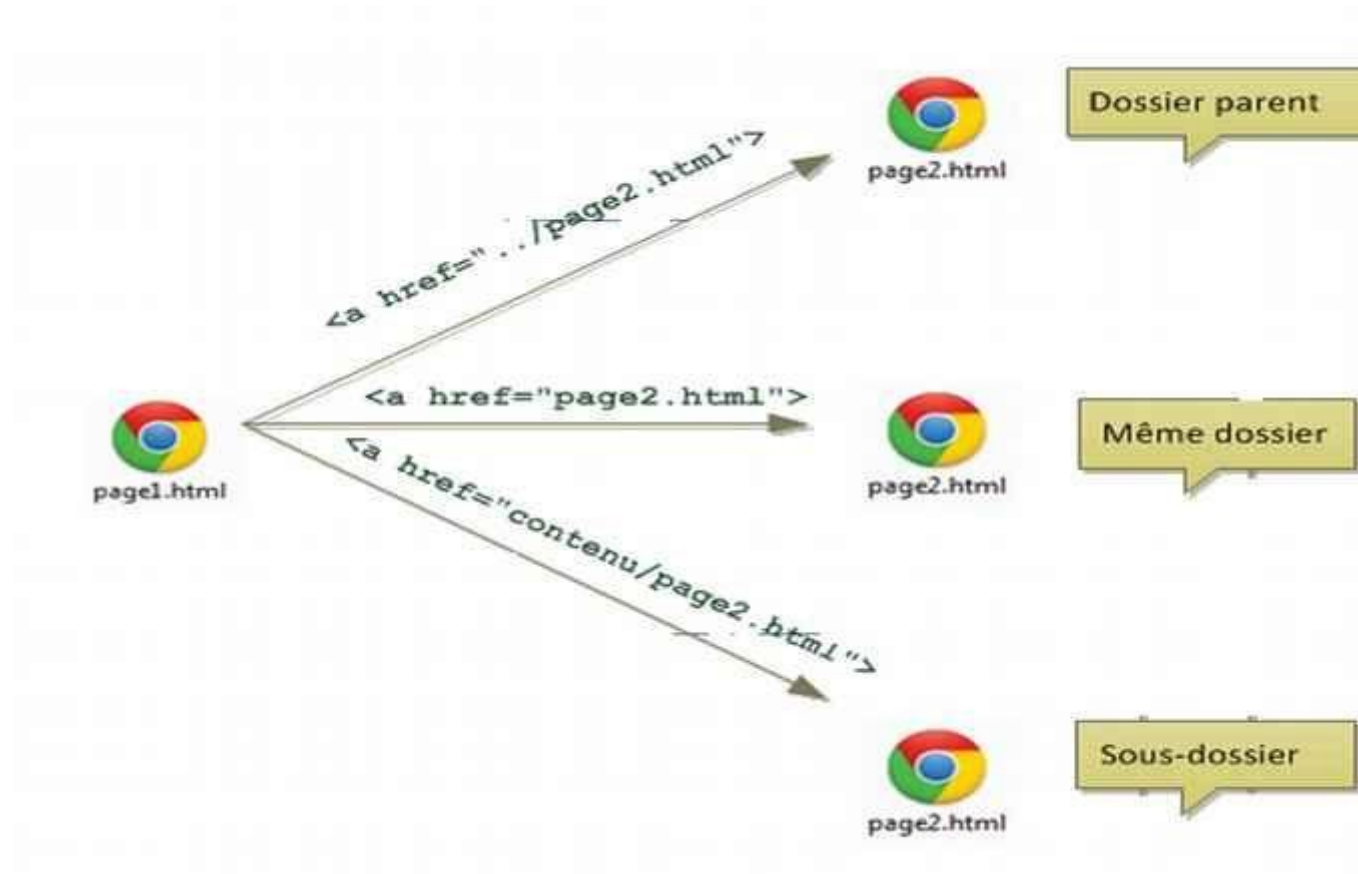
```
<a href="page2.html">lien vers page 2 </a>
```



- lien chemin absolu (pour accéder à une ressource externe)

```
<a href="www.website.com/file.html">My link</a>
```

Liens relatif variante



Lien vers une Ancre

Saut direct

- sur la même page :

```
<a href="#myAnchor">My link</a>
```

- sur une page différente :

```
<a href="file.html#myAnchor">My link</a>
```

à une section spécifique du document :

```
<h3 id="myAnchor">Bas de page</h3>
```

Ancre exemple:vue d'ensemble (1/3)

1. [Programming Languages](#)
2. [Operating Systems](#)
3. [Foreign Languages](#)

en cliquant sur un lien la page se déplace sur une section spécifique du document



Programming Languages

Operating Systems

Foreign Languages

Ancre exemple: vue ecran (2/3)

1. [Programming Languages](#)
2. [Operating Systems](#)
3. [Foreign Languages](#)

Click sur le lien



Ancre exemple: vue ecran 3/3



Défilement
de la page

Programming Languages

Operating Systems

Les limites de l'utilisation d'une ancre

Ergonomie: what you see is what you get

- Il n'est pas conseillé d'avoir peu de pages avec beaucoup de texte: les informations présentées dépassent l'écran et il faut se servir de ces liens et ancres pour naviguer dans la page.
- Mieux vaut avoir plus de pages dont chacune en dimension ne dépasse la taille de l'écran....

Site optimisé avec: Mozilla firefox ver 51.0.1 Résolution écran:1680*1050

La page contient des liens vers d'autres pages.

Lien sur d'autres pages web

- `target="_blank"`

permet d'afficher la page web sur un
nouvel onglet dans votre navigateur

```
<a href="http://acosparla.toile-libre.org/"  
target="_blank">  
    Site acos parla  
</a>
```

D'autres liens

- **lien pour envoyer un e-mail de la page web vers le webmaster**

en utilisant des liens de type **mailto**.

`Envoyez - moi un e- mail`

- **lien pour télécharger un fichier**

`Download file `

HTML

LES TABLEAUX



Balises pour coder un tableau

`<table> . . . </table>`

Divisé en rows (Table Rows):

`<tr> . . . </tr>`

- Chaque row peut contenir:
 - Headers (Table Headers): `<th> . . . </th>`
 - Datas (Table Data): `<td> . . . </td>`
- Les balises **td** et **th** génèrent des cellules!

Example

```
<table>
  <tr>
    <th>Line 1, Header 1</th>
    <th>Line 1, Header 2</th>
  </tr>
  <tr>
    <td>Line 2, Cell 1</td>
    <td>Line 2, Cell 2</td>
  </tr>
  <tr>
    <td>Line 3, Cell 1</td>
    <td>Line 3, Cell 2</td>
  </tr>
</table>
```

Line 1, Header 1	Line 1, Header 2
Line 2, Cell 1	Line 2, Cell 2
Line 3, Cell 1	Line 3, Cell 2

Bordure

- En utilisant CSS vous définissez un style de bordure à votre tableau

Line 1, Header 1	Line 1, Header 2
Line 2, Cell 1	Line 2, Cell 2
Line 3, Cell 1	Line 3, Cell 2

- Ou en utilisant, dans le fichier H.T.M.L, l'attribut **border** de la balise **table**:

- Déprécié !

```
<table border="1">
```

- Mais tant que vous ne connaissez pas les CSS 😊

Fusion de cellules

- 2 attributs:
 - **colspan**: largeur sur le nombre de cellules
 - **rowspan**: hauteur sur le nombre de cellules

```
<table border="1">
  <tr>
    <th colspan="2">Header</th>
  </tr>
  <tr>
    <td>1</td>
    <td rowspan="2">Merging two lines</td>
  </tr>
  <tr>
    <td>2</td>
  </tr>
  <tr>
    <td colspan="2">
      Fusionning two cells
    </td>
  </tr>
</table>
```

Header	
1	Merging two lines
2	
Fusionning two cells	

bonne pratique de développement

Les outils de structuration du contenu d'un document doivent être utilisés en relation avec leur sémantique :

Les listes ne sont pas des tableaux et inversement ...

Les tableaux ne sont pas un moyen de placement dans une page XHTML, ils ne servent qu'à structurer des éléments tabulaires

réserver la balise `<table>` aux tableaux.

Pour structurer la page en zones graphiques irrégulières, utiliser la balise :

`<div></div>`

voire suite du cours

HTML

WHAT ELSE ?



What else ?

Commentaires

- Invisible aux navigateurs Web
- But:
 - Reprise du projet après une longue période
 - Reprise du projet par un autre développeur web
travail en équipe

```
<!-- This is a comment that can help you in the future -->
```

What else ?

Caractères spéciaux

Syntaxe:

& <HTML Code or “#” + représentation unicode> ;

€	€	€
©	©	©

`<p> Web ' n ' Dev © </p>`

Web'n'Dev ©

What else ?

Les accents

Syntaxe: '&' + code + ';'

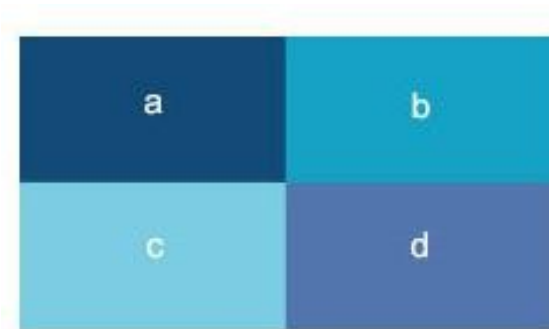
Exemples:

Code	Display
&grave;	è
&acute;	é
&acirc;	â
&ccedil;	ç
&uuml;	ü
&tilde;	~

Association AcosParla La Canebi**`**re 13001 Marseille

Association Acos Parla H.T.M.L

DÉCOUPER UNE PAGE H.T.M.L



Balises <FRAME> <IFRAME>

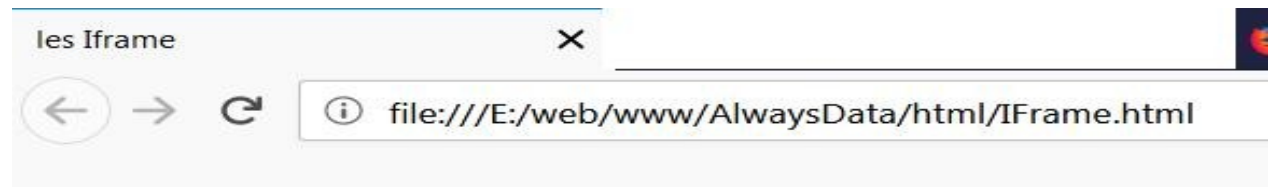
<Frame> Obsolète 

- Cette fonctionnalité a été supprimée des standards du Web, à la place utiliser <iframe>
- La balise <iframe> permet d'intégrer une page HTML dans la page courante.
- Permet de conserver un fond identique lors de l'affichage de différentes pages.

Découper une page HTML

intégrer une page HTML dans la page courante

```
<p>
  <iframe src="IncludeIFrame.html" align="left" width="400"
height="100" scrolling="yes"></iframe>
</p>
file: includeIFrame.html
<body bgcolor="blue">
  <h5>Développeur Logiciel</h5>
  <p>Détails de la formation :</p>
  <p>BLABLABLABLABLABLABLABLABLABLA</p>
  <p>BLABLABLABLABLABLABLABLABLABLA</p>
  <p>....</p>
</body>
</html>
```



Formations en informatique AFPA



Balise générique DIV

- **aucune valeur** sémantique
- Un seul but
 - regrouper d'autres éléments HTML pour:
leur affecter un style C.S.S commun
(voire cours CSS)
interagir avec eux via le D.O.M

balises sémantiques

Les balises : `section`, `article`, `nav`, `aside`, `header`, `footer` segmentent des portions du document qui possèdent une valeur sémantique particulière.

Ne se limitent pas à un simple découpage pour regrouper des éléments HTML dans le but de leur affecter un style CSS commun, mais bien dans le but d'une meilleure classification sémantique de l'information.

La balise Section

- La plus générique : utilisée par défaut si aucun autre élément de section n'a pu lui être préféré.
- La section regroupe un ensemble:
 - de contenu d'une même thématique ensemble
 - d'éléments offrant une fonctionnalité spécifique dans une application web.
- Ne pas confondre avec l'élément **div**

La balise Section

Il contient naturellement un titre d'introduction

Exemple : une chapitre dans la page web

```
<section>
  <h1>Chapter 2 : Basic HTML tags</h1>
  <p>
    HTML markup consists of several key components,
    including elements [...], character-based data types,
    character references and entity references
  </p>
</section>
```

La balise Article

- Article est une spécialisation de section indépendante, possédant une plus forte valeur sémantique.
- Fréquent de présenter plusieurs articles dans un même document.

```
<article>  
  <h1>Titre de l'article</h1>  
  <p>Contenu de l'article</p>  
</article>
```

Balise Aside

destiné à un contenu indirectement lié à l'article lui-même

Exemple: définir un terme, une biographie de l'auteur de l'article, préciser des sources, une liste de lien en relation avec l'article...

```
<aside>
  <h4>Sources de l'article</h4>
  <ul>
    <li><a href="#">Lien 1</a></li>
    <li><a href="#">Lien 2</a></li>
    <li><a href="#">Lien 3</a></li>
  </ul>
</aside>
```

Balise Nav

L'élément `<nav>` représente une section de liens de navigation:

```
<nav>
  <h1>Navigation</h1>
  <ul>
    <li><a href="/">Home</a></li>
    <li><a href="/about.html">About</a></li>
  </ul>
</nav>
```

Balise Header

représente l'en-tête d'une section (section, article...) ou d'une page entière.

peut contenir un logo, un ou plusieurs titres, d'autres informations d'introduction, une navigation.

```
<article>
  <header>
    <h1>Titre de l'article</h1>
    <p>Auteur : Jonh Doe</p>
  </header>
  <p>Contenu de l'article</p>
</article>
```

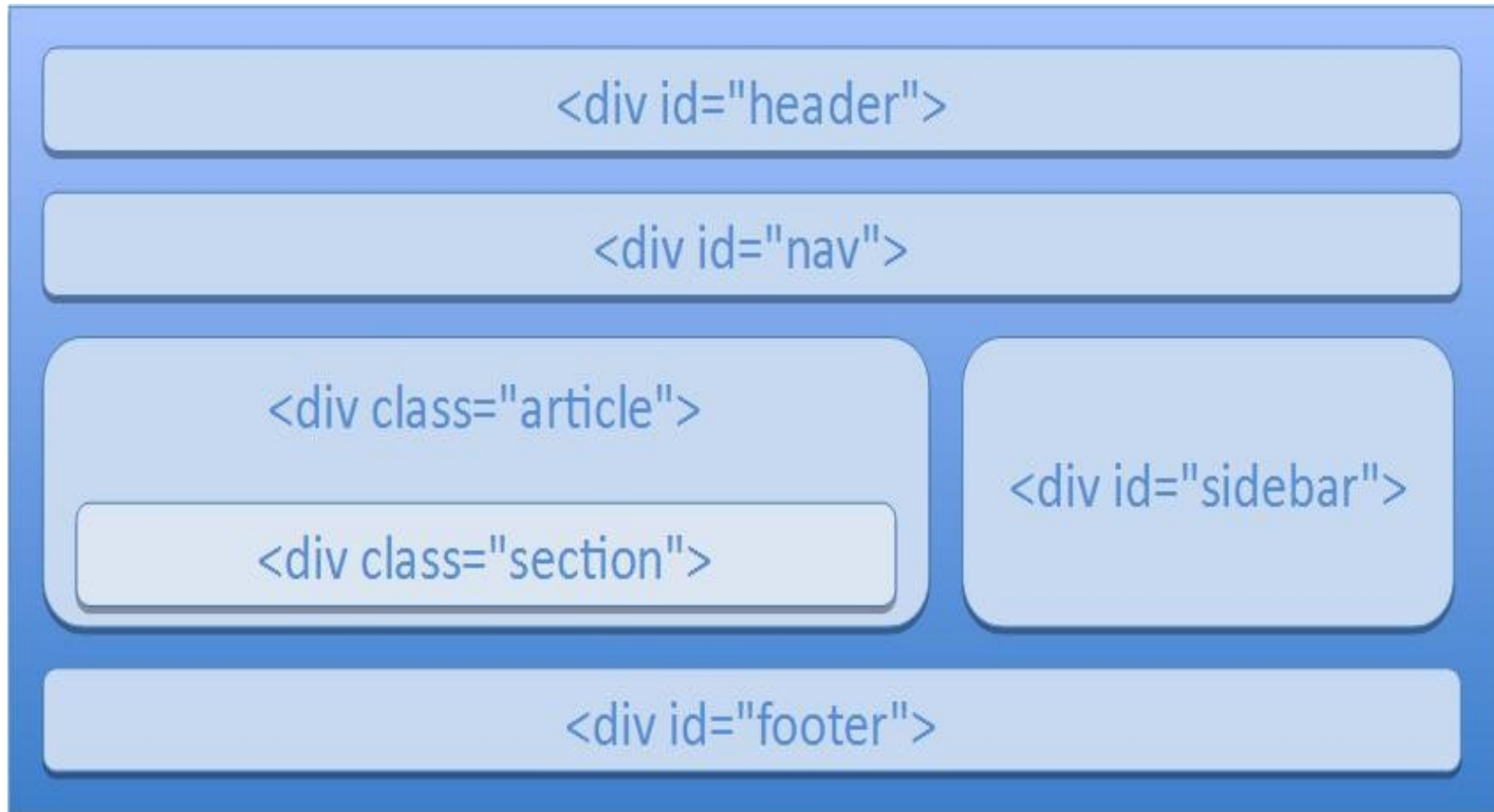
Balise Footer

représente le pied de page, la conclusion d'une section.

informations concernant l'auteur, des mentions légales, un logo de rappel, des coordonnées, des dates

```
<footer>  
  <p>© 2011 Acos Parla Atelier Informatique</p>  
</footer>
```


Ancienne Sémantique

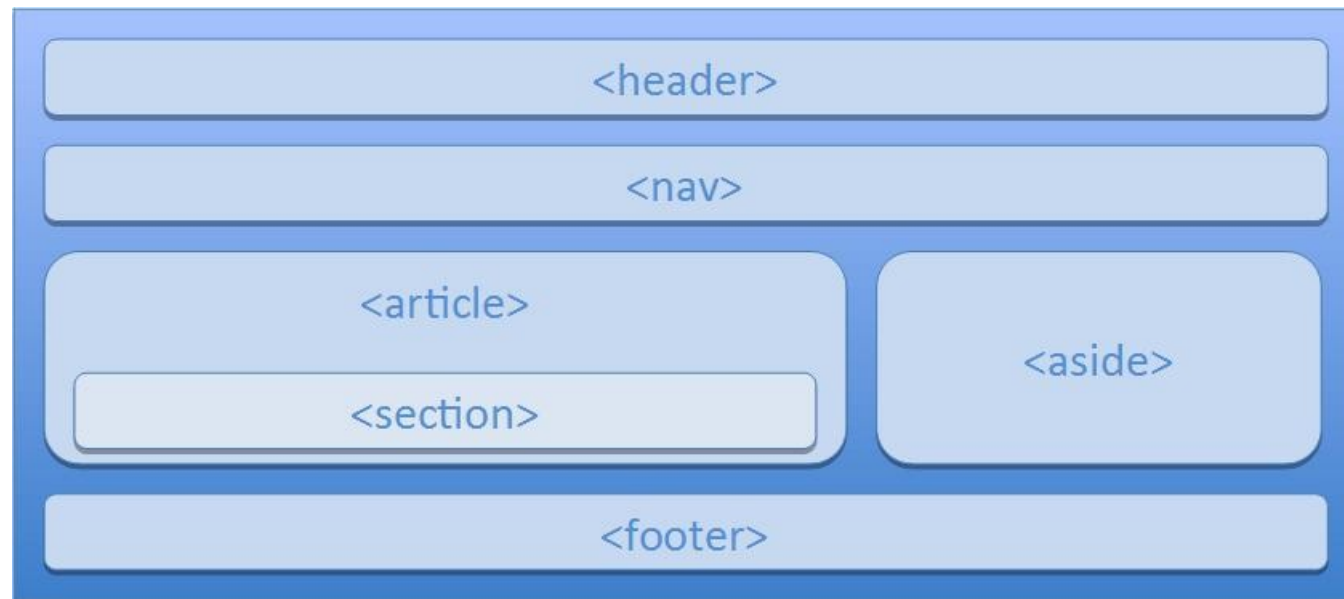


Web sémantique



Objectif: l'émergence du web sémantique en permettant de structurer la mise en page par des éléments plus sémantiques (de nouvelles balises)

Plus d'utilisation abusive de la balise *div*



Exemple de sémantique HTML 5

```
<body>
  <header><!-- Header content --></header>
  <article>
    <section>
      <!-- Section content -->
    </section>
    <section>
      <!-- Other section content -->
    </section>
  </article>
  <aside>
    <!-- Aside content -->
  </aside>
  <footer><!-- Footer content --></footer>
</body>
```

EXERCICE



Exercice : créer votre C.V

- Ecriture d'un C.V sur une page web, à partir du modèle décrit dans les prochaines diapos.
- Dans le but de découvrir les balises HTML de base
- Sans CSS pour le moment (pas de possibilité d'alignement de différentes balises sémantiques)

Exemple de C.V

The diagram illustrates the HTML structure of a resume page. The browser window shows the following elements:

- Titre de la page:** Points to the browser tab titled "My Resume".
- Titre:** Points to the name "John Doe".
- Image:** Points to a placeholder image of a person's silhouette.
- List ordonnée:** Points to a numbered list of programming languages:
 1. [Programming Languages](#)
 2. [Operating Systems](#)
 3. [Foreign Languages](#)
- Lien:** Points to the first item in the list, "Programming Languages".
- Paragraph:** Points to a block of Lorem Ipsum text.

Below the paragraph, there is a section titled "Programming Languages" with a bulleted list:

- Scala
- C#
- VB
- C/C++
- ADA
- Lisp

Exemple de CV

The screenshot shows a web browser window displaying a resume page. The browser's address bar shows the URL `http://mywebsite.com/resume.html`. The resume page contains several sections, each with a red box highlighting a specific HTML element, which is then labeled with a red box and an arrow pointing to it.

- Unordered lists:** Two lists are highlighted. The first list contains `• ADA`, `• Lisp`, and `• Scheme`. The second list contains `• MacOS X`, `• Ubuntu`, `• Debian`, `• Red Hat`, `• Windows XP`, and `• Windows7`.
- Secondary Title:** The section title `Foreign Languages` is highlighted.
- Table:** A table showing language proficiency is highlighted. The table has columns for language, skill, and level (1-5).
- Hyperlink:** A link labeled `Googleize me!` is highlighted.

The resume page also includes a section titled `More about me?` and a footer with the text `One solution: Googleize me!`.

		Level				
		1	2	3	4	5
English	Read					X
	Spoken			X		
	Written				X	
Spanish	Read			X		
	Spoken		X			
	Written	X				

HTML

VALIDATION

Association Acos Parla H.T.M.L



Pourquoi valider ?

- Html est un langage interprété par le navigateur web ...et il laisse passer beaucoup d'erreurs de codage.

La validation permet de normaliser votre page, elle est essentielle pour:

- Portabilité
- Durabilité des documents
- Référencement

Comment valider ?

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

Validez votre page web

Conseil: Si vous avez plusieurs erreurs, corriger la première, et revalider votre document.



Une fois votre page web validée :

vous pouvez ajouter un des icônes :


<https://www.w3.org/QA/Tools/Icons>

preuve de la validation de votre code: html, css...



HTML

FORMS

Pseudo:	<input type="text"/>
Mot de passe :	<input type="password"/>
<input type="button" value="Envoyer"/> 	

Pas de compte ? [Inscrivez-vous !](#)

Objectif d'un formulaire

- Interface permettant de transmettre les données saisies par l'internaute **vers le serveur web**
- Les données saisies dans des éléments **input** sont transmises avec un bouton de type **submit**.

élément <Form>

- définit un formulaire, élément conteneur au même titre qu'un élément `<div>` , `<p>` ...

```
<form id="myForm" action="page.html" method="post">  
  <!-- Form codes -->  
</form>
```

- Attributs de la balise form:
 - Action**: page de destination des données
 - Id** : identifiant du formulaire
 - Méthod**: méthode de soumission **Get** / **Post**

Element <input>

- Type «text»

Affiche des champs text

```
<input type="text" name="firstname" value="textfield" />
```

Attributs supplémentaires:
size, maxlength, **placeholder**...

- place holder:
 - valeur affichée par défaut
 - s'efface au début d'une nouvelle saisie

```
<input type="text" placeholder="place holder s'efface sur saisie" size="30" />
```

Element Input

- Type « password »

Cryptage des données *****

```
<input type="password" name="pass" value="1234" />
```

- Type « checkbox »

Affiche une case à cocher

attribut optionnel: "checked" ☒ Saturday

```
<input type="checkbox" name="day" value="6" checked="checked" /> Saturday
```

Element Input

- Type «radio»

Pour choisir entre différentes options

```
<form>  
<input type="radio" name="gender" value="male" checked> Male<br>  
<input type="radio" name="gender" value="female"> Female<br>  
<input type="radio" name="gender" value="other"> Other  
</form>
```



☒ Male
☐ Female
☐ Other

Element Input

- type «reset»
 - Bouton
 - Efface toutes les valeurs dans les contrôles

```
<input type="reset" value="Reset values" />
```

Pseudo:

Mot de passe :

☒

Pseudo:

Mot de passe :

☒

Element Input

- type «submit»
 - Bouton
 - Transmet les données contenues dans la form

```
<input type="submit" value="Send contents" />
```

A rectangular button with a light gray background and a thin black border. The text "Send contents" is centered on the button in a dark gray, sans-serif font.

Element Input

- La balise input de type « button »
 - Bouton
 - Ne soumet pas le formulaire

```
<input type="button" value="No Action" />
```



Element Input

- La balise input :
 - Type « image »
 - Affiche une image
 - Fonctionne comme input type bouton

```
<a href="destination.html">  
  <input type="image" name="ButnPdf" src="pdf_icon.png" alt="Go" />  
</a>
```



Button Submit/Reset

BUTTON → BOUTON GRAPHIQUE

donne apparence et comportement d'un bouton cliquable
au code HTML : texte, image, tableau

présent entre les balise `<button ... >` `</button>`

```
<button name="send" type="submit">  
    Envoyer   
</button>  
&nbsp; &nbsp; &nbsp; &nbsp; &nbsp;  
<button name="delete" type="reset">  
    Effacer   
</button>
```



submit : input vs button

- `<input type="submit">`
- `<button type="submit"> Submit </button>`

dessineront visuellement un bouton qui effectue la même action (soumettre le formulaire).

Cependant, recommandé d'utiliser:

`<button type="submit">`

car il a une meilleure sémantique, un meilleur support ARIA et il est plus facile à styliser.

Submit Button méthode GET

- Les données sont envoyées par l'URL:
 - `<URI>?name1=value1&name2=value2`
- Inconvénients:
 - données visibles dans l'U.R.L, ne pas utiliser avec des données sensibles
 - nombre de caractères limités
variable selon les navigateurs (exemple : 255)
- Exemple:
`<form action="form2.php" method="get">`

Submit méthode GET: exemple

```
<!DOCTYPE html>
<html>
<head>
  <title>Transmission de données avec la méthode Get </title>
</head>
<body>
  <form action="form2.html" method="get">
    First name:<br>
    <input type="text" name="FirstName" value="">
    <br>
    Last name:<br>
    <input type="text" name="LastName" value="">
    <br><br>
    <input type="submit" value="Submit">
  </form>
</body>
</html>
```

First name:

Mickey

Last name:

Mouse

Submit



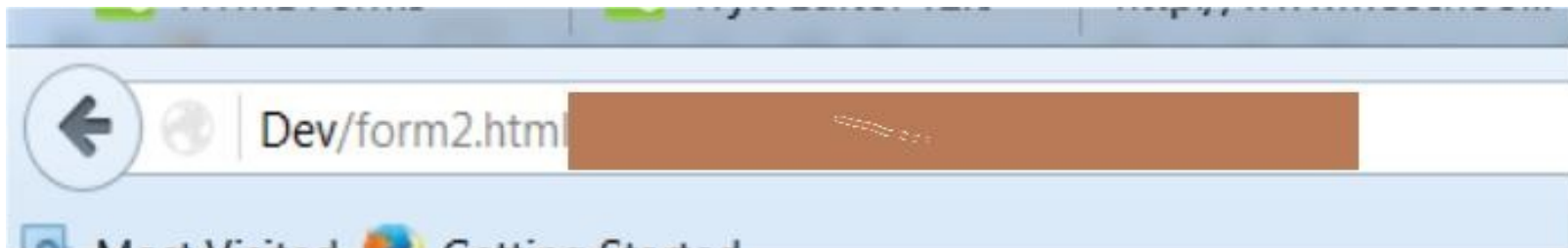
Après le click sur le bouton submit, nous transmettons à form2.html les variables firstname & lastname affectés respectivement des valeurs: Mickey & Mouse.

Submit méthode - POST

Données envoyées par HTTP

- Pas de limite sur le nombre de caractères
- Les données n'apparaissent pas dans l'U.R.L
- Un peu plus sécurisé que GET

```
<form action="form2.html" method="post">
```



Element Input

D'autres types existent, quelques exemples:

- Color
- date & datetime
- Email
- number & range
- url
- week

```
<input type=number min=0 max=3 name=nbsetgagnantbytennismatch />
```



Inclusion de données cachées : type hidden

Avec la balise `<input type="hidden">`
données non saisies transmises sur un submit

```
<form action="action_page.php">  
  First name: <input type="text" name="fname"><br>  
  <input type="hidden" name="country" value="Norway">  
  <input type="submit" value="Submit">  
</form>
```

First name:

127.0.0.1/php/action_page.php?fname=skalpa&country=Norway

Manipulation de fichiers

- La balise input Type « file »
 - Pour manipuler des fichiers
 - Méthode post et attribute *enctype* obligatoire dans la balise <form>

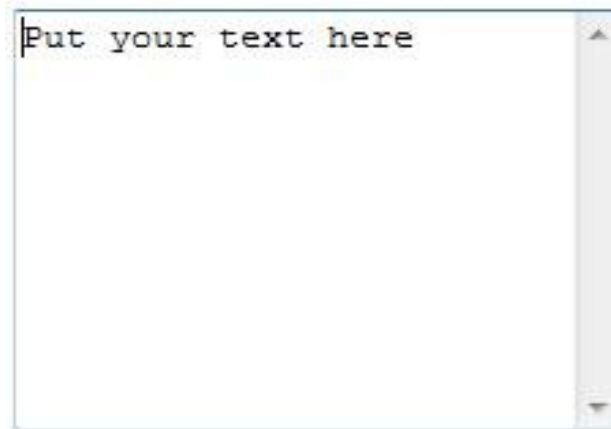
```
<form id="form1" method="post" enctype="multipart/form-data">  
  <input type="file" name="file_send" />  
</form>
```



Textarea

- Champ Text qui peut contenir plusieurs lignes
- Attributs: cols, rows

```
<textarea cols="25" rows="10" name="comments">  
    Put your text here  
</textarea>
```



Fieldset & legend

```
<fieldset>
  <legend>Informations</legend>
  <label for="last">Lastname:</label>
  <input type="text" name="last" />
  <label for="first">Firstname :</label>
  <input type="text" name="first" />
</fieldset>
```



Informations

Lastname:

Firstname :

Attributs spécifiques

readonly agit sur `<input />`, `<textarea />`
`<label>ID :</label> <input type="text" value="14" readonly>`

disabled agit sur

`<button />`, `<input />`, `<optgroup />`,
`<option />`, `<select />`, `<textarea />`
`<button type="button" disabled>Pas de Clic Possible</button>`

- hidden :

`<p hidden>` Rien ne s'affiche sur la page `</p>`

Try it Yourself »

<http://acosparla.alwaysdata.net/html/AttributSpecifique.html>

Forms HTML5 - New markups

HTML5 introduit des nouveautés sur les formulaires par rapport à HTML4 pour permettre la validation côté client.



Validation côté client

- Les forms sont annotables pour vérifier les saisies de l'utilisateur avant de soumettre (bouton submit) le formulaire.
- Permet à l'internaute de pas attendre la validation côté serveur.
- Avant HTML5, JavaScript était utilisé pour la validation côté client

Validation côté client : attribut required

- Saisie obligatoire (required)

```
<form id="formLogin" method="POST">
  <p> <label>Pseudo:</label>
    <input name="pseudo" type="text" required />
    <button name="send" type="submit">Envoyer</button>
  ....
</p>
</form>
```

Pseudo: Envoyer 

Please fill out this field.

```
<select name="cbCourt" id="cbCourt" required>
<option value=""><Selectionnez />
<c:forEach var="court" items="${listcourt }">
<option value="${court.id}">${court.nom }</option>
</c:forEach>
</select>
```

Java\Dev\JEE\Dev_Maven\tennis\src\main\webapp\WEB-INF\enregistrementmatch.jsp

Joueur 1

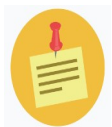
Joueur 2

Arbitre

Court

Date du match  Sélectionnez un élément dans la liste.

Enregistrer le match



l'attribut required fonctionne seulement avec b.tn type submit

Association Acos Parla H.T.M.L

Validation côté client : attribut pattern

- spécifie une RegEx (expression régulière)
 - les valeurs saisies doivent appartenir à l'expression

`<form >`

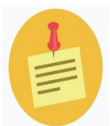
Code Postal:

```
<input type="text" name="code_postal" pattern="[0-9]{5}"  
  title="saisissez 5 valeurs numeriques">  
<input type="submit" value="validation">  
</form>
```

Code Postal: 1234

validation

Please match the requested format: saisissez 5 valeurs numeriques.



l'attribut pattern fonctionne seulement avec btn type submit

Validation côté client : champ Email

- Le type *email* valide une adresse email:

```
<p>  
  <label>E-mail: </label>  
  <input type="email" />  
</p>
```



Nouveaux types input

De Nouveaux types input sont disponibles :

- search
- tel
- url
- **email**
- datetime, date, month, week, time
- number
- range
- color
- image.....

Exemple : nouveau input type date

- Le type **date** facilite la saisie (*datepicker*)
 - interprété différemment selon navigateur
(utilisation de jquery plus fiable...)

```
<p>  
    <label>Date: </label>  
    <input type="date" >  
</p>
```

Date:

février 2017 ▼

lun.	mar.	mer.	jeu.	ven.	sam.	dim.
30	31	1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	1	2	3	4	5

Auto-completion

- L'attribut **datalist** propose une liste
 - L'attribut doit référencer l'id de la *datalist* contenant les suggestions

```
<label>  
  URL with suggestion list:  
  <input type="url" list="hpurls">  
</label>  
<datalist id="hpurls">  
  <option value="http://www.google.com/" label="Google">  
  <option value="http://www.yahoo.com/" label="Yahoo">  
  <option value="http://www.supinfo.com/" label="SUPINFO">  
</datalist>
```



URL with suggestion list:

htt	
http://www.google.com/	Google
http://www.yahoo.com/	Yahoo
http://www.supinfo.com/	SUPINFO

Submit

HTML

Audio & Video





Audio

- L'element audio represents a sound or audio stream
- Content may be provided inside the audio element
 - User agents should show this content to the user only if they don't support the *audio* element

- Exemple :

```
<audio  
src="http://images.wikia.com/starwars/images/3/38/UlicFalls.ogg"  
controls>
```

Your browser doesn't support the HTML5 audio element.

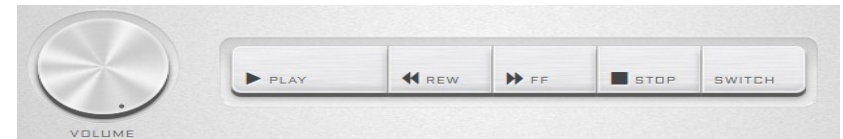
```
</audio>
```





Audio

- Customize (forget your flash code)



https://developer.mozilla.org/fr/docs/Web/Guide/Audio_and_video_delivery#customiser_votre_lecteur_media

adobe stock

abstract neon Retro Boombox Music Tape Recorder Radio Old Vintage cassette player for partys and dance occasions. Ghetto blaster with looping vibrating speakers and rolling cassette. Seamless VJ loop

left and right audio level



Intégration d'une vidéo

- Utilisation de la balise `` pour afficher des gif animés, et aussi des vidéos .avi
- The new *video* element is used for playing videos and audio files with captions
- Again, content may be provided inside the *video* element
 - User agents should show this content to the user only if they don't support the *video* element
- Example :

```
<video src="myVideo.mp4" poster="movie.jpg" controls>  
Your browser doesn't support the HTML5 video element.
```

```
</video>
```





Common attributes

- *audio* and *video* have some common attributes
 - ***preload***: to pre-download the media resource
 - ***autoplay***: to automatically begin playback of the media resource
 - ***controls***: to ask to the user agent to provide its own set of controls
 - ***loop***: to seek back to the start of the media resource upon reaching the end



Alternative sources

- The *source* element allows to specify multiple alternative of sources to a media element
- Allows the browser to choose the better source based on its media type or codec support
- Exemple:

```
<audio controls autoplay loop>  
  <source src="song.ogg" type="audio/ogg" />  
  <source src="song.mp3" type="audio/mpeg" />
```

Your browser doesn't support the HTML5 audio element.

```
</audio>
```

Pour aller plus loin avec les balises HTML



<https://developer.mozilla.org/fr/docs/Web/HTML/Element>

- D'autres sites à consulter sans modération:
 - <http://www.w3schools.com/>
 - <https://www.alsacreations.com/tuto/liste/1-html.html>



"That's all Folks!"