

Les langages du web

1

Les langages du web

- Le Web fonctionne sur le protocole HTTP
- Plusieurs langages sont utilisés pour amener des pages personnelles aux utilisateurs

2

Le premier outil du Web : HTML

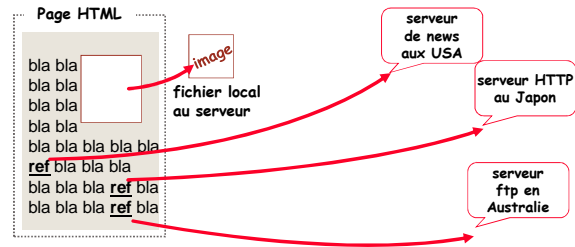
Programmation client web

3

Principes de HTML

Structuration d'un texte à l'aide de balises

Ici, 4 liens hyper-textes



4

HTML

- *Hyper Text Markup Language*
 - un langage de description permettant de structurer et d'afficher différents objets sur un écran (textes, tableaux, images, vidéos, ...)
 - une implémentation simple de SGML (*Standard Generalized Markup Language*)
 - un document HTML doit pouvoir s'afficher dans n'importe quel navigateur **MAIS**
 - les affichages ne sont pas identiques selon le contexte et le navigateur utilisé
- Pourquoi apprendre le HTML ?

5

Faut-il connaître le HTML ?

- HTML est une base indispensable
- Le code généré par les logiciels spécialisés doit presque toujours être complété ou amélioré :
 - les infographistes le modifient souvent pour ajouter des effets, assurer la compatibilité entre des éléments, gérer les erreurs, ...)
- La création de sites web dynamiques nécessite la connaissance des bases du HTML

6

Sites WEB dynamiques

- Ce sont des pages interactives
 - L'accès à certaines pages déclenche l'exécution d'un programme sur le serveur ou le navigateur.
- Les programmes (php, asp, ...) s'exécutent sur le serveur WEB ou sont hébergées les pages
 - Ils récupèrent des données sur le navigateur du poste client (ex. contenu de formulaires), effectuent des requêtes dans des bases de données sur le serveur, et renvoient les informations résultat en HTML au poste client
- D'autres programmes (scripts javascript, applets java...) sont exécutés localement: on les utilise généralement pour effectuer des contrôles sur le poste Client.

Pour générer du HTML *par programme*,
il faut connaître ce langage !

7

HTML : présentation générale

- Langage de description des **éléments** qui composent un document :
 - Entête, titre, auteur, paragraphe, image, lien
- Chaque élément peut lui-même être composé d'autres éléments :
 - Une **entête** possède un **titre** et une série de **paragraphes**
 - Un **tableau** possède des **lignes**
 - Une **ligne** est composée de **cases**
- Il s'agit d'une description logique et non pas *par points* comme pour les images

8

HTML

- Définit la structure logique d'un document WEB
- Composé d'un ensemble de commandes de formatage
- Basé sur la notion d'**environnement** possédant un début et une fin
 - → **délimiteurs** : balises ou marqueurs
- Pour mettre en œuvre du HTML, il suffit de
 - un éditeur de texte
 - un navigateur WEB

9

HTML

- Les balises sont définies entre <>
 - <BALISE>...</BALISE>
- La plupart des environnements peuvent être imbriqués selon des règles bien définies
 - <H1>Mon titre</H1>
- Il n'est pas permis de faire chevaucher des environnements
 - <H1>Mon titre</H1>
- Les minuscules/majuscules n'interviennent pas dans la définition des balises

10

HTML - les bases

- Les balises dénotent des "constructions" documentaires
 - styles de paragraphe (normal, énumérations, titres...)
 - tableaux
 - styles de caractères (gras, italique, souligné...)
 - références à des images
 - références hyper-texte
 - formulaires
 - etc...
- Fichiers par défaut : Welcome.html, index.html

11

Règles de "bonne conception"

- **Mise en page et styles** : maintenir une certaine uniformité sur l'ensemble du site
- **Clarté** : éviter la confusion, présenter de façon simple avec des menus, éviter les textes longs... ; l'information doit être trouvée dès les premières secondes
- **Hyperliens** : ne pas en abuser, faire en sorte qu'ils indiquent clairement où ils mènent
- **Identité** : le visiteur doit savoir facilement sur quel site il est (quelle que soit la page visitée) et savoir à qui s'adresser → logos, @mail, date de dernière maj...
- **Réactivité** : ne pas abuser des images, exécutions de code... (temps de chargement/exécution)
- **Navigateurs** : faire attention à leurs particularités → optimiser le site pour IE et Netscape...

12

Structure «minimale» d'un document

Indentation !

```
<HTML>
  <HEAD>
    <TITLE>titre-fenetre</TITLE>
  </HEAD>
  <BODY>
    corps du document
  </BODY>
</HTML>
```



HEAD : pour différencier du reste du texte, contient les titres
 TITLE : affiché en haut de la fenêtre, utilisé dans les signets
 (bookmarks)
 BODY : contient le document

13

Balises et Attributs

```
...
<BODY>
  <ADDRESS>
    Jean Dupont - Tel: 04 72 72 80 80
  </ADDRESS>
</BODY>
...
```

- Exemple de balise : <ADDRESS>
 - un style "adresse", généralement affiché en italique
- Les commentaires : `<!-- ne sera pas interprété -->`
- Comme pour tout langage, RESPECTER L'IDENTATION

14

Balises et Attributs

- Attributs avec affectation d'une valeur ou non
 - <BALISE attribut1="10" attribut2="blue">
- Attributs "de cœur" (disponibles pour la plupart des balises)
 - class="name" : applique un style au contenu
 - id="name" : donne un nom unique à la balise
 - lang="langage" : spécifie la langue du contenu (ISO639)
 - des attributs liés aux événements : onclick="action", ondblclick, onkeydown, onkeypress, onkeyup, onmouseover, onmouseout, ...

15

Balise <HR> et Balise

- Balise <HR> : trait horizontal
 - <HR size="2"> (en pixels)
 - <HR width="30%">
 - <HR width="100"> (en pixels)
 - <HR align="center/left/right">
 - <HR noshade> (pas de relief)
 - pas de fermeture
- Balise
 : saut de ligne
 - passage à la ligne
 - pas de fermeture

16

Balise <P> et Balises de titres

- Balise <P>
 - début d'un nouveau paragraphe
 - fermeture optionnelle en HTML mais obligatoire en XHTML (en XML, toute balise ouverte doit être fermée)
 - attribut align="center/left/right/justify" (justify pas toujours interprété)
- Balises <Hn> ... </Hn>, n varie de 1 à 6
 - affectent la taille des caractères (taille décroissante)
 - texte en caractères gras et suivi d'un saut de ligne
 - attribut align="center/left/right"
 - <Hn> est considéré comme un paragraphe et ne doit pas être placé dans <P>...</P>

17

Balise <P>

```
<HTML><HEAD>
  <TITLE>titre-fenetre</TITLE>
</HEAD><BODY>
  <P>Ceci est un paragraphe. </P>
  <P>Ceci en est
un autre, le texte sur
plusieur lignes est
reformaté.</P>
</BODY></HTML>
```



18

Balises de titres

```
<HTML><HEAD>
  <TITLE>titre-fenetre</TITLE>
</HEAD><BODY>
  <H1>Titre "H1"</H1>
  <P>Texte sous le titre.</P>
  <H2>Titre "H2"</H2>
  <P>Texte sous le titre.</P>
  <H3>Titre "H3"</H3>
  <P>Texte sous le titre.</P>
</BODY></HTML>
```



19

Balise <PRE> - pré-formatage

- Texte pré-formaté
- Permet de conserver le formatage du texte tel qu'il a été saisi dans le fichier source HTML (respect de la mise en page précise : espacement, saut de ligne...)
- Utile par exemple pour faire un tableau "manuellement"

20

Les caractères spéciaux

- Pour être sûr qu'ils soient correctement interprétés par tous les navigateurs :
 - caractères accentués : **&lettre><accent>**;
 accent : **acute** (aigu), **grave** (grave), **circ** (^), **uml** (")
 exemples : **é**; (é), **î** (î), **ü** (ü)
 - autres caractères :
ñ (ñ), **ç** (ç), **ß** (ß), **©** (©)
æ (æ), **&** (&), ** ** (espace), **"** (")
> (>), **<** (<)

21

Balises - listes

- Non numérotées
 - : Unnumbered List
 - ... : List Item
 -
 - attribut `type="square/circle/disc"` pour ou (**priorité au tag interne en HTML!**)
- Numérotées
 -
 - ...
 -
 - attribut `type="1/A/a/I/i"` pour
 - attribut `start="valeur"` pour (valeur de départ)
 - attribut `value="valeur"` pour (réinitialise le séquençement à la nouvelle valeur)

22

Balises et

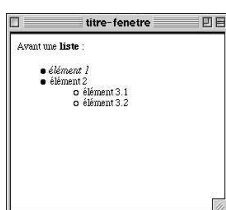
```
...
<UL>
  <LI>&eacute;&eacute;ment 1</LI>
  <LI>&eacute;&eacute;ment 2</LI>
</UL>
  <LI>&eacute;&eacute;ment 3.1</LI>
  <LI>&eacute;&eacute;ment 3.2</LI>
</UL>
</UL>
...
```



23

Balises de formatage de texte

```
<HTML><HEAD>
  <TITLE>titre-fenetre</TITLE>
</HEAD><BODY>
  <P>Avant une <B>liste</B> :</P>
  <UL><LI><I>&eacute;&eacute;ment 1</I></LI>
  <LI>&eacute;&eacute;ment 2</LI>
  <UL><LI>&eacute;&eacute;ment 3.1</LI>
  <LI>&eacute;&eacute;ment 3.2</LI>
  </UL></UL>
</BODY></HTML>
```



24

Balises de formatage de texte

- `<CENTER>...</CENTER>` - centrer le contenu
- Taille, police et couleur de caractères
`...`
 - taille comprise entre 1 et 7
 - par défaut : 3 (équivalent à 12 pt)
 - taille absolue
 - `TRES GROS`
 - `très petit`
 - taille relative
 - `plus gros`

25

O
b
s
o
l
è
t
e

Balises de formatage de texte

- `<CENTER>...</CENTER>` est remplacé par
`<DIV>...</DIV>` qui permet en plus un alignement à droite ou à gauche
 - attribut `align="left/center/right"`
 - attribut `nowrap` : le contenu n'est pas renvoyé à la ligne automatiquement --> ascenseur
- `...` est remplacé
 - par les feuilles de styles (CSS)
 - ou par `<SMALL>`, `<BIG>`, ...

26

Formatage de caractères

- Styles physiques
 - sont indépendants du navigateur utilisé et de sa configuration
 - `...` : gras
 - `<I>...</I>` : italique
 - `<TT>...</TT>` : machine à écrire (police largeur fixée)
 - `<STRIKE>...</STRIKE>` : texte barré
 - `<U>...</U>` : souligné
 - `_{...}` : indice
 - `^{...}` : exposant
 - `<SMALL>...</SMALL>` : petite police
 - `<BIG>...</BIG>` : grande police

27

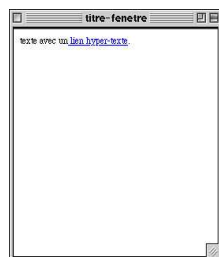
Formatage de caractères

- Styles logiques
 - peuvent dépendre du navigateur utilisé et de sa configuration
 - `` : gras
 - `` : mettre un texte en valeur (italique)
 - `<DFN>` : définition
 - `<CITE>` : citation bibliographique
 - `<CODE>` : programme informatique
 - `<BLOCKQUOTE>` : tabulation+espaces paragraphes
 - ...

28

Balises pour les liens hyper-texte

```
<HTML><HEAD>
  <TITLE>titre-fenetre</TITLE>
</HEAD><BODY>
  <P>texte avec un <A href="cible.html">
lien hyper-texte</A>.</P>
</BODY></HTML>
```



29

Balise <A> : liens et étiquettes

- Définition d'un hyperlien `text`
 - `href="nom"` : nom du document à lier
 - une URL standard : `http://www.univ-lyon1.fr`
 - une adresse mail : `mailto:toto@zozo.fr` (mais spam)
 - un chemin relatif : `Cours/index2.html`
 - un chemin absolu : / est la racine du serveur Web !
 - une étiquette : `#LABEL`
 - une combinaison URL, chemin relatif, étiquette
 - texte : texte utilisé pour décrire/représenter le lien dans le document
- Définition d'une étiquette
 - `...`
 - le texte entre `<A>` et `` n'est pas une zone cliquable

30

Balise <BODY> - paramètre la page

- Fond d'écran
 - `<BODY background="nom_image">`
 - répétition d'une image sur l'écran du navigateur
- Couleurs
 - `<BODY`
 - `bcolor="#000000"` : couleur de fond
 - `text="#000000"` : couleur du texte
 - `link="#000000"` : couleur des liens non encore visités
 - `vlink="#000000"` : couleur des liens déjà visités
 - `alink="#000000"` : couleur des liens lors du clic
 - Noms des couleurs
 - en clair : aqua, black, gray, olive...
 - en notation hexadécimale RGB

31

Balise - insertion d'images

```
<HTML><HEAD>
  <TITLE>titre-fenetre</TITLE>
</HEAD><BODY>
  <P>Texte sur l'image.</P>
  <P><IMG src="logo.jpg"></P>
  <P>Texte sous l'image.</P>
</BODY></HTML>
```



32

Balise - insertion d'images

- Insertion d'une image dans un document :
 - `<IMG src="nom"`
 - nom local ou URL distante
 - extensions : jpeg, gif, xbm ...
 - `<IMG height="hauteur" width="largeur"`
 - informe le navigateur de la hauteur et la largeur de l'image en pixels (évite au navigateur de les calculer)
 - peut aussi permettre de redimensionner l'image
 - `<IMG align="bottom/middle/top/left/right"`
 - alignement par rapport au texte (*left* et *bottom* par défaut)

33

Balise - insertion d'images

<IMG border="nb_pixels"

- affiche un cadre autour de l'image

<IMG vspace="nb_pixels" hspace="nb_pixels"

- tailles des marges (espacement image/texte)

<IMG alt="ceci est une info-bulle"

- définit un commentaire pour l'image
- affiché à la place de l'image pendant le téléchargement
- affiche une info-bulle quand la souris passe sur l'image (certains navigateurs seulement)
- donne des infos aux indexeurs
- L'attribut **alt** est aussi utilisable dans d'autres balises

34

Accélérer le chargement des pages ?

- Les visiteurs ne patientent pas plus de 10 secondes pour charger une page..
- Données Multimédia réellement utiles ?
- Ré-utilisation des images par ex. (de index.html)
- **Réduire** la taille des images
- Utiliser les attributs *height* et *width*, pour permettre au navigateur de fixer le texte le plus tôt possible
- Découper les longues pages



35

Balise <EMBED>

- Permet
 - d'inclure des fichiers multimédias : sons, vidéos
 - de les jouer automatiquement
- Attributs


```
<EMBED src="nom_fichier" autostart="true | false"
  hidden="true | false" width="145" height="60"
  loop="n | infinite">
```

 - **autostart** permet le lancement automatique du fichier
 - **hidden** affiche ou non la boîte de dialogue associée au fichier (lecteur multimédia) avec les dimensions **width** et **height**
- Attention à la lenteur du chargement du fichier

36

Tableaux

- Complètement définis et normalisés dans HTML 4.0
- Permettent de positionner précisément les objets dans le navigateur
- Les cases (cellules) peuvent contenir des données multimédia (textes, images, liens...)
- Les tableaux sont très utilisés en particulier pour gérer la mise en page

37

Balises <TABLE> <TR> <TH> <TD>

- <TABLE>...</TABLE> définit un tableau dans un nouveau paragraphe
- Spécification d'une ligne
 <TR>...</TR>
 marquent le début et la fin d'une ligne du tableau
- Spécification des cellules
 - Cellules d'en-tête (centré, en gras)
 <TH>...</TH>
 - Cellules de données
 <TD>...</TD>

38

Balise <TABLE>

- Accélérer l'affichage des tableaux (HTML 4.0)
 - fixer le nombre de colonnes
 - ex : <TABLE cols="nb">
 - fixer la taille du tableau
 - par défaut, la taille est calculée automatiquement par le navigateur en fonction du contenu des cellules et de la taille de la fenêtre
 - largeur fixée
 <TABLE width="nb_pixels ou %"> : taille en pixels ou % par rapport à la fenêtre
 - hauteur fixée
 <TABLE height="nb_pixels ou %">

39

Balise <TABLE>

- Épaisseur des bordures du tableau
 - encadrement du tableau : attribut **border**
`<TABLE border>` : équivalent à `border="1"`
 - épaisseur des traits de bordure
`<TABLE border="nb_pixels">`
- Espacements
 - distance en pixels entre les bords et le contenu des cellules
`<TABLE cellpadding="nb_pixels">`
 - distance entre les cellules du tableau en pixels
`<TABLE cellspacing="nb_pixels">`
- Positionnement du tableau dans la page
 - `<TABLE align="left | center | right">`

40

Balise <TABLE>

- Image d'arrière plan
 - `<TABLE background="nom_image">`
- Couleurs du fond et des bords
 - `<TABLE bgcolor="couleur">`
 - `<TABLE bordercolor="couleur">`
- Ombrage
 - `<TABLE bordercolordark="couleur">` : définit la couleur la plus sombre de l'ombrage
 - `<TABLE bordercolorlight="couleur">` : définit la couleur la plus claire de l'ombrage

41

Balises <TR>, <TH> et <TD>

- Attributs communs aux lignes et cellules
 - alignement horizontal du contenu dans une cellule ou une ligne
`<TR align="left | center | right">...</TR>`
`<TD align="left | center | right">...</TD>`
 - alignement vertical du contenu dans une cellule ou une ligne
`<TR valign="top | middle | bottom">...</TR>`
`<TD valign="top | middle | bottom">...</TD>`
 - Couleurs d'arrière plan d'une cellule ou d'une ligne
`<TR bgcolor="couleur ou #RRVVB">...</TR>`
`<TD bgcolor="couleur ou #RRVVB">...</TD>`

42

Balises <TH> et <TD> - cellules

- Largeur et hauteur d'une cellule
 - La hauteur pour une cellule est appliquée à toutes les cellules de la ligne
 - La largeur d'une cellule est appliquée à toutes les cellules de la colonne.
 - il suffit de formater une seule cellule
 - les tailles sont prises en compte si non contraires à la taille du tableau, sinon le navigateur essaie d'harmoniser

<TH **width**="nb_pixels | %" **height**="nb_pixels | %">

- Empêcher le passage automatique à la ligne dans la cellule
<TD **nowrap**>

43

Fusion de cellules - Balise <CAPTION>

- Fusions de cellules
 - définit le nombre de colonnes sur lesquelles s'étend une cellule
<TH **colspan**="nombre">...</TH> : nombre de cellules à fusionner sur une ligne
 - définit le nombre de lignes sur lesquelles s'étend une cellule
<TD **rowspan**="nombre">...</TD> : nombre de cellules à fusionner sur une colonne
 - ces fusions doivent être cohérentes par rapport au lignes et colonnes du tableau
- Balise <CAPTION align="left | center | right"> : permet de donner un titre au tableau

44

Les images sensibles

45

Images sensibles ou cliquables

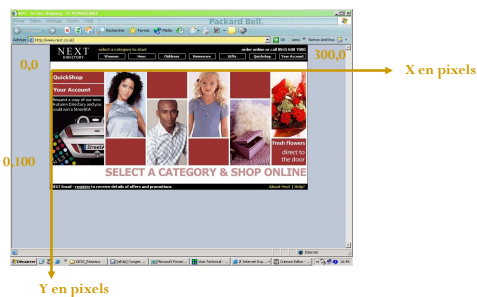
- Images et liens
 - `` permet d'afficher une image "mini" et en cliquant dessus de voir l'image agrandie
 - > 1 image = 1 lien
- Principe des images cliquables
 - l'image est découpée en zones
 - on peut associer un lien différent à chaque zone de l'image
 - très souvent utilisé pour la page de garde des sites
 - facile à maintenir (découpage des zones cliquables dans un fichier séparé)

46



Le système de coordonnées

- Il faut commencer par définir les zones en pixels !
- les zones peuvent être des rectangles, cercles, ...



48

La méthode ancienne - principe

- /cgi-bin/imapmap et ISMAP
- transmission via l'URL de la coordonnée du click au serveur
- un programme sur le serveur récupère la position du click dans l'image et retourne la page associée à la coordonnée du click
- méthode ancienne mais utilisable sur de nombreux navigateurs
- non *standalone* (aller/retour entre le client et le serveur)
- nécessite l'installation sur le serveur du programme spécifique...
- à utiliser s'il y a trop de zones dans l'image

49

La méthode ancienne - exemple

- Créer un fichier /map/3zones.map sur le serveur
default enter.html
rect internet.html 5,5 21,14
poly css.html 38,6 50,6 48,18 45,15 40,21 35,24
circle http://www.univ-lyon1.fr 52,32 45,32
- Ajouter une entrée dans imagemap.conf (serveur)
3z: /map/3zones.map
- Utilisation dans la page HTML cliente

50

La méthode USEMAP

- Géré entièrement par le client : tout est inclus dans la page HTML qui comporte l'image cliquable
 - il est préférable de mettre la définition des zones dans un fichier séparé
- Le client détermine à partir des coordonnées du click de la souris le lien qu'il doit exécuter
- Pas d'interaction avec le serveur
 - minimisation des accès réseau
 - peut être testé dans un navigateur local sans faire appel à un serveur HTTP

51

Utilisation de USEMAP

- Définition d'une image sensible
``
- Définition d'une carte (map)
`<map name="3zones"> ... </map>`
- Définition d'une zone dans la carte
`<area shape="fig" href="url" coords="x0,y0,x1,y1,...">`
 - fig peut prendre les valeurs
 - rect=rectangle (angles supérieur gauche et inférieur bas)
 - poly=polygones (coords de chaque extrémité du vecteur)
 - circle=cercle (centre et rayon)
 - attributs **title**="infobulleIE" **alt**="infobulleNetscape"

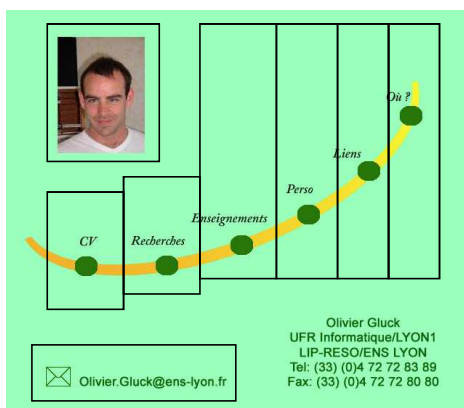
52

Un exemple avec USEMAP

```
<!-- index.html -->
<html><head>
</head><body>
...
<map name="choix">
<area shape="rect" coords="80,50,180,180" href="cv.html">
<area shape="rect" coords="0,230,160,460" href="cv.html">
<area shape="rect" coords="160,230,250,460" href="recherches.html">
<area shape="rect" coords="250,0,370,380" href="enseignements.html">
<area shape="rect" coords="370,0,445,380" href="perso.html">
<area shape="rect" coords="445,0,520,380" href="liens.html">
<area shape="rect" coords="520,0,640,380" href="where.html">
<area shape="rect" coords="0,460,280,560"
      href="mailto:Olivier.Gluck@ens-lyon.fr">
</map>

</body></html>
```

53

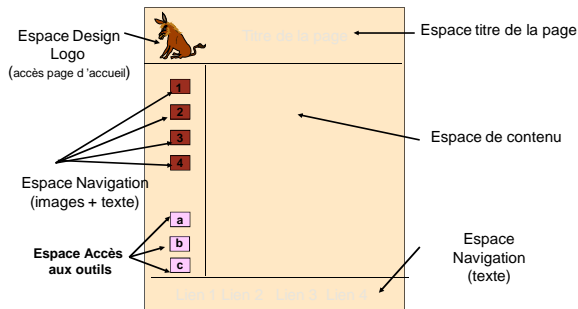


54

Les cadres (frames) en HTML

55

Organiser sa page HTML



56

Les cadres ou "frames"

- Permettent de créer des cadres indépendants à l'intérieur d'une fenêtre du navigateur
 - développés par Netscape
 - existent depuis la version 3.0
- Permettent d'accélérer le chargement des pages
 - seule le cadre qui change est chargé
- Souvent utilisés pour avoir
 - un petite cadre pour un index fixe
 - un grand cadre pour les données à afficher
- Décrits par certains concepteurs
 - certains moteurs de recherche ont du mal à analyser les sites à base de cadres --> entrée sans *frame*
 - ascenseur par cadre pénible à utiliser
 - un cadre n'a pas d'URL --> pas de *bookmark* possible

57

Définition des cadres - <FRAMESET>

- Utilisation de la balise **<FRAMESET>**
 - <FRAMESET>...</FRAMESET>
 - contient la description de chacun des cadres présents
 - pas de balise <body> !!!**

```
<!-- fichier index.html -->
<html><head>
<title>Mise en place de frames</title>
</head>
<frameset cols="20%,80%">
  <frame src="menu.html" name="menu">
  <frame src="contenu.html" name="contenu">
</frameset>
</html>
```

58



Attributs de <FRAMESET>

- ROWS** - Division en **zones horizontales**
 - ROWS="h1,h2,...,hn"
 - hi peut être
 - hi=n : hauteur du cadre en pixels
 - hi=n% : hauteur du cadre en % de la taille de la zone mère
 - hi=* : le cadre prend toute la place restante disponible
 - ex : 2 frames horizontales de taille identique
 - <frameset rows="50%,50%">
- COLS** - Division en **zones verticales**
 - COLS="l1,l2,...,ln"
- FRAMEBORDER** - Frontière avec effet 3D
 - valeurs yes (1) ou no (0) - 1 par défaut
- BORDERCOLOR** - Couleur des bordures
- BORDER** - largeur de la bordure entre cadres
 - BORDER="0" --> cadres invisibles

59

Portabilité des cadres - <NOFRAMES>

- Si les cadres ne sont pas supportés par le navigateur


```
<NOFRAMES>
  texte d'erreur à afficher
</NOFRAMES>
```
- A l'intérieur d'une balise <FRAMESET>, on ne peut mettre que les balises suivantes :
 - <FRAMESET>...</FRAMESET>
 - <FRAME>
 - <NOFRAMES>...</NOFRAMES>

60

Description d'un cadre - <FRAME>

- Utilisation de la balise <FRAME>
- Attributs **src** et **name** de <FRAME>
 - contenu du cadre
 - **src**="URL de la page à afficher dans le cadre"
 - nom du cadre
 - **name**="nom du cadre"
 - sert à référencer le cadre pour qu'il devienne la cible de n'importe quel autre lien hypertexte

61

Les attributs de <FRAME>

- **MARGINWIDTH**="n" (15 par défaut)
 - nombre de pixels entre les frontières gauche et droite de la zone et le document HTML à afficher dans la zone
- **MARGINHEIGHT**="n"
 - nombre de pixels entre les frontières haute et basse de la zone et le document HTML à afficher dans la zone
- **NORESIZE** - pas de valeur
 - le navigateur empêche le redimensionnement de la zone à l'aide de la souris

62

Les attributs de <FRAME>

- **SCROLLING**="yes | no | auto"
 - indique si la zone doit posséder une barre de défilement
 - yes - affiche une barre même si le document est plus petit que le cadre
 - no - n'affiche jamais la barre - des parties du document peuvent ne pas être atteinte
 - auto - la barre apparaît si nécessaire (valeur par défaut)

63

Utilisation des cadres - attribut target

- Attribut **target**="nom du cadre" de <A> et <FORM>
 - désigne le cadre cible du lien
 - nom réservé ou nom local au frameset

```
<!-- fichier index.html -->
<html><head>
<title>Mise en place de
frames</title>
</head>
<frameset cols="20%,80%">
<frame src="menu.html"
name="menu">
<frame src="contenu.html"
name="contenu">
</frameset>
</html>
```

```
<!-- fichier menu.html -->
<html><head>
<title>Mise en place de frames</title>
</head><body>
Menu<br>
<a href="1.html" target="contenu">Cours1</a>
<br>
<a href="2.html" target="contenu">Cours2</a>
<br>
<a href="3.html" target="contenu">Cours3</a>
<br>
</body></html>
```

84



Utilisation des cadres - attribut target

- Il faut maintenant créer contenu.html, 1.html...

```
<!-- contenu.html -->
<html><head>
<title>Mise en place de frames (2)</title>
</head><body>
ici apparaîtra le contenu<br>
</body>
</html>
```

```
<!-- 1.html -->
<html><head>
<title>Cours 1</title>
</head><body>
Internet...<br>
</body>
</html>
```

85



Les liens et les cadres - attribut target

- L'attribut **target**
 - target contient le nom d'un cadre existant
 - la cible est affichée dans le cadre nommé
 - target n'est pas renseigné
 - la cible est affichée dans le cadre courant
 - target contient le nom d'un cadre inexistant
 - la cible est affichée dans une nouvelle fenêtre
 - target contient une valeur réservée
 - _self** : la cible est affichée dans le cadre où est défini le lien
 - _parent** : la cible est affichée dans le cadre "père" (celui qui l'a créé)
 - _blank** : la cible est affichée dans une autre fenêtre sans structure de cadre
 - _top** : la cible affichée remplace la fenêtre courante --> supprime tous les cadres
- utilisez **_top** dès qu'un lien pointe sur une page extérieure

86

Imbrication de cadres

```
<!-- fichier index.html -->
<html><head>
<title>Imbrication de cadres</title>
</head>
<frameset rows="25%,75%" frameborder="yes">
  <frame src="haut.html" name="haut">
  <frameset cols="*,2*" border="10">
    <frame src="basgauche.html" name="basgauche">
    <frame src="basdroit.html" name="basdroit">
  </frameset>
</frameset>
</html>
```

Mettre frameborder="no"



67

Imbrication de documents

```
<!-- fichier index.html -->
<html><head>
<title>Imbrication de documents</title>
</head>
<frameset rows="25%,75%">
  <frame src="vide.html" name="haut">
  <frame src="fils.html" name="bas">
</frameset>
</html>

```

Un dernier exemple

```
<!-- descend.html -->
<html><head>
<title>Imbrication</title>
</head>
<frameset rows="*,*">
  <frame src="plushaut.html">
  <frame src="plusbas.html">
</frameset>
</html>

<!-- adroite.html -->
<html><head>
<title>adroite</title>
</head>
<frameset cols="*,*">
  <frame src="plushaut.html">
  <frame src="plusbas.html">
</frameset>
</html>

<!-- plushaut.html -->
<html><head>
<title>plushaut</title>
</head><body>
<a href="plusbas.html" target="_parent">
Plus haut</a> (plusbas.html _parent)<br>
<a href="plusbas.html" target="_top">
Au sommet</a> (plusbas.html _top)<br>
<a href="adroite.html" target="_parent">
Tourner</a> (adroite.html _parent)<br>
</body></html>

<!-- plusbas.html -->
<html><head>
<title>plusbas</title>
</head><body>
<a href="descend.html">Plus bas</a>
</body></html>
```

70



La balise <IFRAME>

- Permet de définir des cadres "inline" (dans le corps du document à l'intérieur de <body>)

```
<!-- iframe.html -->
<html><head>
</head><body>
avant iframe
<iframe src="vide.html" width="400" height="500" frameborder="1">
</iframe>
apres iframe
</body></html>

<!-- vide.html -->
<html><head>
</head><body>
je suis l'iframe
</body></html>
```

71



Référencer son site

72

Référencer son site

- Un site n'a un intérêt que s'il est connu et facilement atteignable
 - -> enregistrer son site manuellement sur les principaux moteurs de recherche (google, yahoo, altavista, hotbot, voila...)
 - -> utiliser des produits commerciaux ou gratuit qui font le référencement automatiquement...
 - -> faire inscrire des liens vers son propre site dans des sites déjà référencés...
 - Fichier « robots.txt »
- Faut-il encore ensuite que le site soit correctement référencé...
 - -> il faut renseigner dans sa page les balises META et utiliser l'attribut ALT pour les images

73

La balise <META>

- Fournit des informations sur le document HTML qui sont utilisées par les robots d'indexation
- Introduite à partir de HTML 2.0
- Ces informations ne sont pas visibles à l'écran
- Placée entre <HEAD>...</HEAD> de la page d'accueil
- Utilisation classique avec attributs NAME et CONTENT


```
<META NAME="info" CONTENT="valeur de info">
```

 - des valeurs particulières de l'attribut NAME sont utilisées par les robots

74

La balise <META> - exemples

- `<META NAME="description" CONTENT="...">`
 - description du site sous la forme d'un texte (200 caractères max.)
 - ce texte sera affiché par les moteurs en guise de présentation du site
 - exemple : `CONTENT="Page personnelle de ..."`
- `<META NAME="keywords" CONTENT="...">`
 - définit une liste de mots-clés qui permet d'identifier le contenu du site (500 caractères max)
 - exemple : `CONTENT="enseignement, recherche, réseaux, outils internet, photographie"`
- `<META NAME="author" CONTENT="...">`

75

La balise <META> - exemples

- `<META NAME="robots" CONTENT="valeur">`
 - donne des indications aux robots d'indexation sur
 - l'indexation de la page en cours
 - le suivi des liens présents sur la page
 - valeur=all | none | index | noindex, follow | nofollow (all par défaut)
 - exemple : `CONTENT="index, follow"`
- Autres valeurs pour l'attribut NAME

Identifier-URL : URL du site

Date-Creation-yyyymmdd : date de création du site

Date-Revision-yyyymmdd : date de dernière modification

Reply-to : adresse mail

Revisit-after : indique la ré-indexation du site par le robot après n jours

76

La balise <META>

- L'attribut LANG
 - spécifie la langue du document (également utilisé par les robots d'indexation) ou du contenu de la balise
 - valeur normalisée ISO
 - exemples :
 - `<META NAME="description" LANG="fr"...>`
 - `<META NAME="keywords" LANG="en" ...>`
 - `<META NAME="description" LANG="en-us" ...>`
 - `<META NAME="keywords" LANG="de" ...>`

77

Remarques

- Les balises <META> doivent surtout être présentes dans la page d'accueil...
- La balise <TITLE> est également utilisée par les robots pour décrire le site
 - titre du site
 - limitée à 100 caractères
 - utilisée par les *bookmarks*

78

Autres utilisations de <META>

- L'attribut **HTTP-EQUIV**="info" CONTENT="valeur"
 - (info,valeur) circule dans l'en-tête HTTP du serveur vers le client quand la page est chargée - permet entre autre de paramétrer le navigateur
- quelques exemples
 - indiquer le codage des caractères à utiliser pour l'affichage de la page par le navigateur


```
<META HTTP-EQUIV="Content-Type" CONTENT="text/html; charset=ISO-8859-1">
```
 - date limite d'expiration des pages dans le cache du client


```
<META HTTP-EQUIV="expires" CONTENT="Wed, 30 Sept 2006 12:00:00 GMT">
```

79

Autres utilisations de <META>

- L'attribut **HTTP-EQUIV**="info" CONTENT="valeur"
 - quelques exemples
 - interdire la mise en cache sur le client


```
<META HTTP-EQUIV="pragma" CONTENT="no-cache">
```
 - supprimer l'apparition dans une frame


```
<META HTTP-EQUIV="Window-target" CONTENT="_top">
```
 - définir la langue du document


```
<META HTTP-EQUIV="Content-Language" CONTENT="en-GB">
```
 - plein d'autres valeurs possibles (cf. cours HTTP)
- voir <http://vancouver-webpages.com/META/>

80

Autres utilisations de <META>

- L'attribut **HTTP-EQUIV** et la valeur **refresh**
 - redirection automatique vers une autre URL


```
<META HTTP-EQUIV="refresh" CONTENT="10; URL=http://www.nouveau-site.com">
```
 - actualisation automatique de la page


```
<META HTTP-EQUIV="refresh" CONTENT="60">
```

 - utilisé par exemple pour les systèmes de caméras surveillance...


```
<!-- camera.html -->
<html><head>
<title>video-surveillance</title>
<meta http-equiv="refresh" content="5">
</head><body>
  <br>
  
</body></html>
```

81

Exemple de redirection automatique

```
<!-- redirect.html -->
<html><head>
  <title>Redirection automatique</title>
  <meta http-equiv="refresh" content="3;URL=camera.html">
</head><body>
  Veuillez noter l'@ de mon nouveau site<br>
  Vous allez être redirigé dans 3 secondes
</body></html>
```



82

Les évolutions de HTML

83

Les évolutions de HTML (1)

- 1992 : première définition de HTML (HTML 1.0)
- 1993 : ajout des formulaires et tableaux
- 1994 : HTML 2.0 normalisation des principaux éléments
- 1994 : HTML 3.0 (extension de HTML soumise en tant que draft)
- 1995 : balises non standard spécifiques à Netscape
- 1996 : guerre Netscape/IE
- 1996 : HTML 3.2 - standard basé sur les pratiques existantes

84

Les évolutions de HTML (2)

- 1997 : HTML 4.0 - apparition des feuilles de style, du *scripting* et du modèle Document Object Model (objets manipulables par les scripts)
- 1998 : XML : eXtensible Markup Language
 - langage de description de documents
 - permet de définir ses propres balises
 - forme allégée de SGML (description sémantique des documents qui sépare les problèmes de syntaxe de la structuration de l'information)
- 1999 : HTML 4.01 - modifications mineures
- 2000 : XHTML 1.0 - version XML de HTML 4.01
 - XHTML : eXtensible HTML (langage XML dédié aux documents hypertexte = HTML XML-compatible)

85

Les évolutions de HTML (3)

- Qui normalise ?
 - Initialement IETF (HTML 1.0 et 2.0)
 - Maintenant le W3C (World Wide Web Consortium)
 - <http://www.w3.org/>
 - association d'utilité publique composée de différents groupes d'intérêt : Adobe, Microsoft, IBM, Sun...
 - force de propositions et recommandations pour de nouvelles normes pour le WWW



<http://validator.w3.org>

<http://www.chami.com/html-kit/>

86

Problème syntaxe/validation

- HTML 4.01 définit une syntaxe précise et formelle
- Tout document HTML devrait respecter cette syntaxe
- Elle peut être validée automatiquement
- En pratique, très peu de documents HTML sont valides
 - les auteurs ne respectent pas totalement la norme
 - la seule validation de leur document est leur propre test dans un navigateur particulier
 - le HTML généré par des éditeurs automatiques est souvent invalide

87

Problème syntaxe/validation

- Le résultat est tout de même "acceptable"
 - les navigateurs font de leur mieux pour tolérer les erreurs
 - les erreurs de syntaxe ne sont pas signalées (exemple : définir des cadres entre `<body>...</body>`)
- Problèmes
 - différences de rendu visuel entre les navigateurs
 - les navigateurs définissent leurs propres extensions non-standard
 - de plus en plus de navigateurs différents

88
