

Initiation aux langages du Web :

HTML / XHTML

Table des matières

<u>1.Introduction</u>	4
<u>2.Définitions</u>	4
<u>a)HTML</u>	4
<u>b)XHTML</u>	4
<u>3.Conception de pages web (outils)</u>	4
<u>a)Windows</u>	4
<u>b)Linux</u>	5
<u>4.Navigateurs</u>	5
<u>5.Structure d'un document HTML</u>	6
<u>6.Balises</u>	6
<u>a)Généralités sur les balises</u>	6
<u>b)Les balises de type bloc</u>	6
<u>Position</u>	7
<u>Taille</u>	7
<u>Liste des balises</u>	7
<u>c)Les balises de type en ligne</u>	7
<u>Position</u>	7
<u>Taille</u>	7
<u>Liste des balises</u>	7
<u>d)Les balises de type auto-fermante</u>	7
<u>Liste des balises</u>	8
<u>7.Attributs</u>	8
<u>8.Règles de syntaxe</u>	8
<u>9.Structure d'un document XHTML</u>	9
<u>a)Aperçu</u>	9
<u>b)L'élément « DOCTYPE »</u>	10
<u>DTD 1.0 Frameset</u>	10
<u>DTD 1.0 Transitional</u>	10
<u>DTD 1.0 Strict</u>	10
<u>c)A proscrire !</u>	11
<u>d)Les balises « meta »</u>	12
<u>Attribut « name »</u>	12
<u>Attribut « http-equiv »</u>	12
<u>10.Mise en forme, sémantique et style</u>	13
<u>a)Mise en forme</u>	13
<u>Le Texte</u>	13
<u>Les listes</u>	13

b)Sémantique.....	14
c)Style.....	15
La balise « font ».....	15
L'attribut « align ».....	15
d)Les caractères spéciaux.....	15
11.Les images.....	16
a)Les formats.....	16
b)Les attributs.....	16
12.Liens hypertexte.....	17
a)Aperçu.....	17
b)Attributs.....	17
c)Exemples.....	17
13.Les tableaux.....	18
a)Aperçu.....	18
b)Utilisation des groupes de cellules.....	18
14.Les frames.....	19
a)Aperçu.....	19
b)Les différents éléments.....	19
c)Segmentation.....	20
d)Fonctionnalités supplémentaires.....	21
Support des navigateurs ancien.....	21
Ajout d'un ascenseur.....	21
Utilisation de liens hypertexte.....	21
15.Les formulaires.....	22
a)Aperçu.....	22
b)Utilisation des différents éléments.....	23
c)Exemple.....	24
Annexes.....	25
1.Index des exemples.....	25
2.Index des illustrations.....	25

1. Définitions

a) HTML

L'*Hypertext Markup Language*, généralement abrégé **HTML**, est le format de données conçu pour représenter les pages web. C'est un langage de balisage qui permet d'écrire de l'hypertexte, [...] de structurer sémantiquement et de mettre en forme le contenu des pages.

b) XHTML

eXtensible HyperText Markup Language, dit XHTML, est un langage de balisage servant à écrire des pages pour le World Wide Web. Conçu à l'origine comme le successeur d'HTML, XHTML se fonde sur la syntaxe définie par XML, plus récente, mais plus exigeante que celle définie par SGML sur laquelle repose HTML.

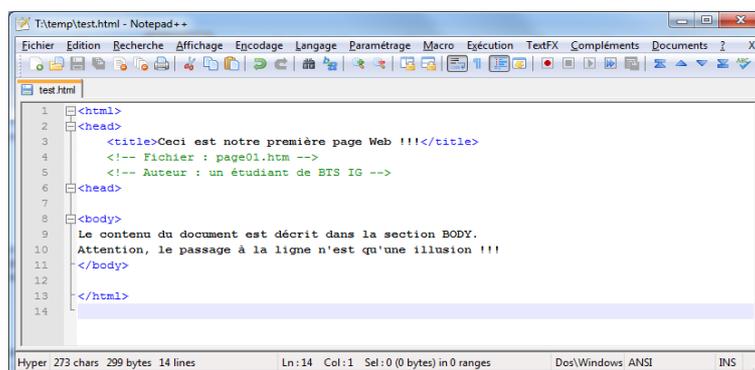
2. Conception de pages web (outils)

Vous pouvez vous simplifier la tâche en utilisant la coloration syntaxique fournie par certains éditeur de texte.

a) Windows

Sous Windows, le plus connu est certainement « Notepad++ » et vous pouvez le télécharger à l'adresse suivante :

<http://notepad-plus-plus.org/fr/download/v6.1.5.html>



```
1 <html>
2 <head>
3   <title>Ceci est notre première page Web !!!</title>
4   <!-- Fichier : page01.htm -->
5   <!-- Auteur : un étudiant de BTS IG -->
6 </head>
7
8 <body>
9   Le contenu du document est décrit dans la section BODY.
10  Attention, le passage à la ligne n'est qu'une illusion !!!
11 </body>
12
13 </html>
14
```

Illustration 1: Notepad++

b) Linux

Sous Linux, vous pouvez simplement utiliser l'éditeur de texte de Gnome : « Gedit ».

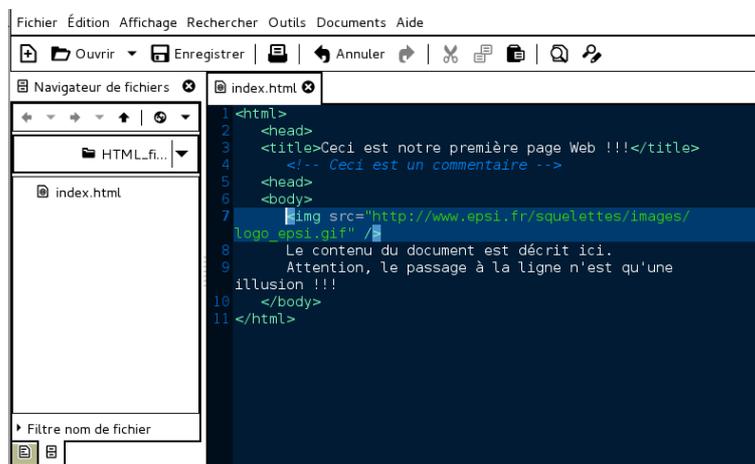


Illustration 2: Gedit

3. Navigateurs

Le travail du navigateur, c'est de lire le code que vous avez écrit (XHTML, CSS, Javascript, etc...) et d'afficher une représentation graphique.

Il existe de nombreux navigateurs :

³⁵/₁₇ Mozilla Firefox ;

³⁵/₁₇ Internet Explorer ;

³⁵/₁₇ Google Chrome ;

³⁵/₁₇ Safari ;

³⁵/₁₇ Opera ;

³⁵/₁₇ ...

4. Structure d'un document HTML

Veillez observer le document HTML ci-dessous :

```
<html>
<head>
  <title>Ceci est notre première page Web !!!</title>
  <!-- Ceci est un commentaire -->
</head>
<body>
  
  Le contenu du document est afficher ici, à droite de l'image.
  Attention, le passage à la ligne n'est qu'une illusion !!!
  <div>Ceci est une div</div>
  <div>Ceci est une autre div</div>
</body>
</html>
```

Exemple 1: Page HTML

On peut remarquer :

- que certaines balises sont ouvrantes `<XXX>` et d'autre fermantes `<XXX/>` ;
- une entête, délimitée par les balises « *head* » ;
- un titre, contenu dans l'entête et délimité par les balises « *title* » ;
- un contenu, également appelé corps de la page, délimité par les balises « *body* » ;
- que l'ensemble du contenu est délimité par les balises « *html* » ;
- que les commentaires sont entourés par les balises `<!--` et `-->`.

5. Balises

a) Généralités sur les balises

Les balises portent toutes un nom. Dans l'exemple précédent nous avons vu les balises « *html* », « *head* », « *title* », « *body* », « *div* » et « *img* ».

Elles peuvent être :

- ouvrante (`<html>`, `<head>`, ...) ;
- fermante (`</html>`, `</head>`, ...) ;
- auto-fermante (``).

Il existe 3 types de balises : bloc, en ligne et auto-fermante.

b) Les balises de type bloc

Les balises de type *bloc* sont des boites d'éléments qui peuvent contenir des éléments orphelins comme du texte, des balises de type *en ligne*, des balises de type *auto-fermante*

et même d'autres balises de type *bloc*.

Position

Elle se positionnent toujours les unes en dessous des autres comme les balises « div ».

Taille

Leurs largeurs est celle de l'élément qui les contient, ici l'élément « body ».

Liste des balises

address, blockquote, body, dd, div, dl, dt, fieldset, form, h1, h2 , h3, h4, h5, h6, head, html, hr, li, map, noscript, ol, p, pre, script, style, table, tbody, td, tfoot, th, thead, tr, ul.

c) Les balises de type en ligne

Les balises de type en ligne ne peuvent contenir que des éléments orphelins comme du texte, d'autres balises en ligne, mais **pas** de balises de type bloc.

Position

Elle se positionnent toujours les unes à côté des autres comme les balises « div ».

Taille

Les balises de type en ligne ne possèdent aucune dimensions.

Liste des balises

a, abbr, acronym, area, bdo, br, button, caption, cite, code, del, dfn, em, img, input, ins, kbd, label, legend, link, meta, object, optgroup, option, param, q, samp, select, span, strong, sub, sup, textarea, title, var.

d) Les balises de type auto-fermante

Les balises de type auto-fermante sont des balises qui sont ouvrantes et fermantes en même temps. Elles ne contiennent pas de contenu et se ferment en ajoutant un espace et un slash (/) à la fin de la balise.

Les balises auto-fermante sont soit des balises de type bloc soit des balise de type en ligne.

Liste des balises

area, br, hr, img, input, link, meta, param.

6. Attributs

Un attribut est une instruction contenue à l'intérieur d'une balise et dont le but est de fournir une information supplémentaire sur la manière dont cette balise doit être interprétée.

Les attributs varient en fonction des balises et certaines balises possèdent des attributs obligatoires. Par exemple la balise « img » possède deux attributs obligatoires : **src** et **alt**.

7. Règles de syntaxe

- Toute balise ouvrante doit obligatoirement être **fermée**.

Juste : `<div>Ceci est une div</div>`

Faux : `<div>Ceci est une div`

- Balises et attributs doivent être écrits en **minuscule**.

Juste : ``

Faux : ``

- Les valeurs des attributs doivent être **entre guillemet ou apostrophe**.

Juste : ``

Faux : ``

- Chaque attribut doit obligatoirement avoir **une valeur**.

Juste : ``

Faux : ``

- Chaque élément doit être **imbriqué correctement**.

Juste :<head><title>...</title></head>

Faux :<head><title>...</head></title>

8. Structure d'un document xHTML

a) Aperçu

Veillez observer le document xHTML ci-dessous :

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" lang="fr">
<head>
  <title>Ceci est notre première page Web !!!</title>
  <!-- Ceci est un commentaire -->
  <meta name="keywords" lang="fr" content="motcle1,motcle2" />
  <meta name="description" content="Description de ma page web." />
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
  <meta http-equiv="Content-Language" content="fr" />
  <meta http-equiv="Content-Script-Type" content="text/javascript" />
</head>
<body>
  
  Le contenu du document est afficher ici, à droite de l'image.
  Attention, le passage à la ligne n'est qu'une illusion !!!
  <div>Ceci est une div</div>
  <div>Ceci est une autre div</div>
</body>
</html>
```

Exemple 2: Page xHTML

On peut remarquer :

- l'utilisation de la balise « DOCTYPE » ;
- que la balise « html » s'est dotée d'attributs ;
- que le contenu de l'élément « head » est enrichi par des élément « meta » ;

b) L'élément « DOCTYPE »

L'élément « DOCTYPE » contient la DTD (Déclaration de Type de Document) de la page web. La DTD indique au navigateur quelles sont les règles syntaxiques suivies par la page web.

C'est au développeur web de choisir la DTD utilisée parmi « Frameset », « Transitional » et « Strict » en fonction de ses objectifs et besoins.

DTD 1.0 Frameset

Voici l'élément « DOCTYPE » à utiliser :

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

Exemple 3: DTD « Frameset »

Ci-dessous sont énoncées les particularités de cette DTD :

- il faut respecter les 5 règles énoncées au paragraphe 8 ;
- elle accepte les balise agissant sur le style ;
- elle accepte d'utiliser la balise « iframe » ;
- l'attribut « target » est accepté pour les balise « a » ;
- la balise « body » est remplacé par « frame ».

DTD 1.0 Transitional

Voici l'élément « DOCTYPE » à utiliser :

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

Exemple 4: DTD « Transitional »

Ci-dessous sont énoncées les particularités de cette DTD :

- il faut respecter les 5 règles énoncées au paragraphe 8 ;
- elle accepte les balise agissant sur le style ;
- elle accepte d'utiliser la balise « iframe » ;
- l'attribut « target » est accepté pour les balise « a » ;

DTD 1.0 Strict

Voici l'élément « DOCTYPE » à utiliser :

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

Exemple 5: DTD « Strict »

Ci-dessous sont énoncées les particularités de cette DTD :

- il faut respecter les 5 règles énoncées au paragraphe 8 ;
- elle **n'accepte pas** les balise agissant sur le style ;
- elle **n'accepte pas** l'utilisation la balise « iframe » ;
- l'attribut « target » **n'est pas accepté** pour les balise « a » ;

c) A proscrire !

La DTD Strict n'acceptant pas les balises de présentation, comme par exemple « font » ou « center », celles-ci ne doivent plus être utilisées. Il est vivement conseillé d'utiliser une feuille de styles CSS, plus propre et plus pratique.

En ce qui concerne la DTD Transitional, ces balises sont dépréciées. Certaines de ces balises sont même complètement obsolètes et n'appartiennent à aucune norme et sont condamnées, à l'avenir, à disparaître.

Voici la liste des balises dépréciées : « basefont », « center », « font », « s », « strike », « u ».

Voici, par balise, la liste des attributs dépréciés :

Balises	Attributs
a	target
body	alink, background, bgcolor, link, vlink, text
caption	bgcolor, height, nowrap, width
hr	clear, noshade, size, width
img	border, hspace, vspace
li	compact, type
object	border, hspace, vspace
ol	Star, value
pre	width
table	bgcolor, height, nowrap, width
tbody	bgcolor, height, nowrap, width
td	bgcolor, height, nowrap, width
tfoot	bgcolor, height, nowrap, width
thead	bgcolor, height, nowrap, width
th	bgcolor, height, nowrap, width
tr	bgcolor, height, nowrap, width

d) Les balises « meta »

Les balises « meta » contiennent des méta-données et permettent ainsi de spécifier une description, des mots-clés, un auteur, des règles spécifiques destinées aux robots des moteurs de recherches, une langue et bien plus encore.

Cette balise possède deux attributs :

- « name » ou « http-equiv » pour spécifier le nom de la méta-données ;
- « content » pour spécifier son contenu.

Attribut « name »

Voici quelques valeurs que peut prendre l'attribut « name » ainsi que la signification de la méta-donnée :

- « author » spécifie l'auteur de la page ;
- « category » spécifie une catégorie / rubrique ;
- « copyright » spécifie un copyright ;
- « description » permet d'ajouter une description ;
- « keywords » permet de spécifier les mots-clés par langue en utilisant l'attribut « lang » ;
- « language » permet de spécifier la langue (déprécier, cf. CONTENT-LANGUAGE) ;
- « robot » autorise ou interdit l'indexation par des robots ;
- ...

Attribut « http-equiv »

Voici quelques valeurs que peut prendre l'attribut « http-equiv » ainsi que la signification de la méta-donnée :

- « Content-Type » définit la table de caractères (*charset*) à utiliser. Généralement "text/html; charset=iso-8859-1" ;
- « Content-Language » définit la langue utilisée ;
- « Content-Script-Type » autorise l'utilisation de gestionnaire d'événement.
- ...

9. Mise en forme, sémantique et style

a) Mise en forme

Le Texte

La mise en forme du texte se fait par l'intermédiaire des balises suivantes :

- « b » pour mettre en gras ;
- « bdo » pour le sens de lecture ;
- « br » pour un saut de ligne ;
- « em » pour une mise en exergue générique ;
- « i » pour mettre en italique ;
- « p » pour un paragraphe ;
- « pre » pour insérer un bloc de code ;
- « del » pour barrer le texte ;
- « strong » pour une mise en exergue importante ;
- « sub » pour la mise en indice ;
- « sup » pour la mise en exposant ;
- « ins » pour souligner le texte .

Les balises « em » et « strong » auront des interprétations différentes en fonction du navigateur utilisé.

Les listes

Liste non-ordonné

Les listes non-ordonnée sont définie grâce à l'utilisation de la balise « ul » (unordered list) :

```
<ul>
  <li>Un</li>
  <li>Deux</li>
  <li>Trois</li>
</ul>
<ul type="square">
  <li>Quatre</li>
  <li>Cinq</li>
</ul>
```

Exemple 6: Liste non-ordonné

On peut remarquer :

- que l'attribut « type » permet de changer le type de puce ;
- que même si la balise « ul » est de type bloc, elle n'accepte cependant que des balises « li ».

Liste ordonné

Les listes non-ordonnée sont définie grâce à l'utilisation de la balise « ol » (ordered list) :

```
<ol>
  <li>Un</li>
  <li>Deux</li>
  <li>Trois</li>
</ol>
<ol type="A">
  <li>Quatre</li>
  <li>Cinq</li>
</ol>
```

Exemple 7: Liste ordonnée

On peut remarquer :

- que l'attribut « type » permet de changer le type de numérotation ;
- que même si la balise « ol » est de type bloc, elle n'accepte cependant que des balises « li ».

b) Sémantique

La sémantique du texte se fait par l'intermédiaire des balise suivantes :

- « h1 », « h2 », ..., « h6 » pour les titres de niveau 1 jusqu'à 6 ;
- « abbr » pour spécifier une abréviation ;
- « cite » pour une citation ;
- « q » pour une citation courte ;
- « dfn » pour une définition ;

La sémantique permet d'améliorer le référencement ainsi que l'accessibilité de votre site (appareil de synthèse vocal).

c) Style

La balise et l'attribut suivants sont montrés **à titre informatif** car il est préférable d'utiliser une **feuille de style (CSS)** pour pouvoir séparer contenu et style !

Avant d'aller plus loin, n'oubliez pas de modifier votre élément « DOCTYPE » pour utiliser la DTD Transitional.

La balise « font »

Cette balise, déprécier, permet d'appliquer une police, couleur et taille sur une portion de texte.

```
<font size="5" color="#000000" face="times new roman">Grande taille</font>  
<br />  
<font size="+5">Autre manière d'agir sur la taille</font>  
<br />  
<font size="10" face="Arial">Gros mots en Arial</font>
```

Exemple 8: Balise « font »

L'attribut « align »

Cet attribut permet de spécifier l'alignement d'un élément.

```
<p align="center">Ce paragraphe est centré</p>  
<p align="right">Celui-ci est aligné à droite</p>  
<p align="justify">Un paragraphe justifié</p>
```

d) Les caractères spéciaux

Lorsque la page **ne contient pas** la balise « meta » spécifiant le jeu de caractères utilisé (*charset*), les caractères spéciaux ont tendance à être mal interprétés par le navigateur.

Par exemple, Pour pouvoir afficher les caractères « é », « à » ou encore « è », ils faut utiliser une combinaison HTML ou un code ISO qui commence par « & » et se termine par « ; ».

Ci-dessous un tableau contenant certaines de ces combinaisons :

Caractères	Code HTML	Code ISO
<	<	›
>	>	‹
&	&	&
é	é	é
è	è	è
à	à	à

espace	 	
ô	ô	ô
û	û	û

10. Les images

a) Les formats

Trois formats d'image ont été validés par la fondation W3C : *GIF*, *JPEG* et *PNG*.

Les images aux formats *GIF* et *PNG* pourront s'afficher morceau par morceau car elles sont entrelacées mais ne peuvent contenir respectivement que 256 et 24-bit de couleurs. Un autre avantage de ces deux formats est le fait que le fond peut être transparent, très pratique pour faire des montages.

En revanche, le format *JPEG* propose une palette de couleurs beaucoup plus conséquente et donc une qualité supérieure.

b) Les attributs

Comme dit précédemment, la balise « *img* » possède deux attributs obligatoires : « *src* » et « *alt* ».

La source de l'image, désignée par « *src* », peut être relative ou absolue.

L'attribut « *alt* » permet d'afficher un texte alternatif au cas où l'image ne pourrait être affichée par le navigateur.

Vous pouvez donc désigner une image :

- présente localement sur le serveur web

```

```

Exemple 9: Image avec lien relatif

- présente sur un autre site web

```

```

Exemple 10: Image avec lien absolu

Il existe d'autres attributs qui permettent de modifier le style d'une image. Cependant, comme énoncé précédemment, il est préférable d'utiliser une feuille de style (CSS) pour modifier le style d'un élément.

11. Liens hypertexte

a) Aperçu

Les liens hypertexte, sont réalisés grâce à la balise « a » (*anchor*). Les liens permettent :

- la navigation entre différentes zone d'une même page (ancres) ;
- la navigation entre différentes pages ;
- l'envoi de mail.

b) Attributs

Les deux attributs les plus intéressants de la balise « a » sont : « href » (obligatoire) et « target ».

Le premier sert à définir la destination de l'hyperlien et le deuxième à ouvrir ce lien dans un onglet ou la fenêtre courante.

La destination de l'hyperlien peut être définie, comme pour les images, de manière relative ou absolue.

Pour finir, on peut signaler l'attribut « title » qui permet d'ajouter un « *tooltip* » au passage de la souris sur le lien.

c) Exemples

Lien vers un fichier local qui utilisera la fenêtre courante :

```
<a href="page.html" title="Lien vers une page">Vers la page</a>
```

Exemple 11: Lien hypertexte local

Lien vers une page externe qui utilisera un onglet :

```
<a href="http://epsi.fr" target="_blank" />'EPSI dans un onglet</a>
```

Exemple 12: Lien hypertexte externe

Lien vers un mail, ceci ouvrira le client de messagerie local :

```
<a href="mailto:jc.forton@gmail.com">M'envoyer un mail</a>
```

Exemple 13: Lien hypertexte pour envoi de mail

Lien vers une ancre :

```
<a href="#paragraphe">Lien vers le paragraphe</a>  
<br/><br/><br/><br/><br/><br/><br/><br/>  
<br/><br/><br/><br/><br/><br/><br/><br/>  
<p id="paragraphe">bla bla</p>
```

Exemple 14: Lien hypertexte intra-page (ancres)

12. Les tableaux

a) Aperçu

La construction d'un tableau en HTML nécessite au minimum les balises suivantes :

- « table » permet de délimiter le tableau ;
- « td » permet d'ajouter un colonne ;
- « th » permet d'ajouter une cellule d'entête ;
- « tr » permet d'ajouter une ligne ;

Un tableau tout simple pourrait être le suivant :

```
<table>
  <tr>
    <th>Légumes</th>
    <td>6€</td>
  </tr>
  <tr>
    <th>Boisson</th>
    <td>4€</td>
  </tr>
</table>
```

Exemple 15: Tableau simple

b) Utilisation des groupes de cellules

Il est possible d'élaborer le tableau précédent en utilisant les groupe de cellule. Ces groupe permettent de fractionner le tableau en trois partie :

- « caption » permet d'ajouter une légende ;
- « thead » désigne l'entête du tableau ;
- « tbody » désigne le corps du tableau ;
- « tfoot » désigne le pied du tableau.

Un tableau plus évolué pourrait être le suivant :

```
<table>
  <caption>Résultat des courses</caption>
  <thead>
    <tr>
      <th>Aliment</th>
      <th>Prix</th>
    </tr>
  </thead>
  <tfoot>
    <tr>
      <th>Total</th>
      <th>10€</th>
    </tr>
  </tfoot>
  <tbody>
    <tr>
      <th>Légumes</th>
      <td>6€</td>
    </tr>
    <tr>
      <th>Boisson</th>
      <td>4€</td>
    </tr>
  </tbody>
</table>
```

Exemple 16: Tableau évolué

13. Les frames

a) Aperçu

Les frames permettent de diviser la fenêtre du navigateur en plusieurs segments, chacun pouvant afficher un contenu différent. Cela permet de diminuer l'utilisation de la bande passante en utilisant un contenu statique dans un ou plusieurs segments (eg. un menu).

Pour pouvoir utiliser les frames, il faut impérativement utiliser la DTD 1.0 Frameset décrite au paragraphe 11.

b) Les différents éléments

Les différents éléments permettant l'utilisation des frames sont les suivants :

- « framset » remplace l'élément body et permet de définir la manière dont la page est segmentée ;
- « frame » permet de définir un ou plusieurs cadres à l'intérieur de l'élément « framset ».
- « noframe » permet de contenir du HTML classique pour les navigateurs ne supportant pas les frames ;

- « iframe » permet d'inclure un autre document dans le document HTML courant.

L'insertion de contenu est spécifié grâce à l'attribut « src » de la balise « frame » :

```
<frame src="page/page.html">
```

Exemple 17: Frame avec attribut « source »

c) Segmentation

La segmentation de la page se fait grâce aux attributs « rows » et « cols » de l'élément « frameset » qui prennent comme valeur une suite de

- pourcentage ;
- valeur en pixel ;
- le signe « * » qui signifie « le reste ».

Par exemple :

- pour segmenter une page en deux verticalement

```
<frameset cols="50%,*">
  <!-- Colonne de gauche (eg. menu) -->
  <frame src="menu.html"/>
  <!-- Colonne de droite (eg. contenu) -->
  <frame src="contenu.html"/>
</frameset>
```

Exemple 18: Segmentation verticale

- pour segmenter une page horizontalement en 3 parties

```
<frameset rows="33%,33%,*">
  <!-- Ligne du haut -->
  <frame src="haut.html">
  <!-- Ligne du milieu -->
  <frame src="milieu.html"/>
  <!-- Ligne du bas -->
  <frame src="bas.html"/>
</frameset>
```

Exemple 19: Segmentation horizontale

- Les éléments « frameset » étant de type bloc, il est possible de les imbriquer pour faire des segmentations multiples. Par exemple, deux colonnes et 3 lignes :

```
<frameset cols="50%,*">
  <!-- Colonne de gauche (eg. menu) -->
  <frame src="menu.html"/>
  <frame>
    <frameset rows="33%,33%,*">
      <!-- Ligne du haut -->
      <frame src="haut.html">
      <!-- Ligne du milieu -->
```

```

        <frame src="milieu.html"/>
        <!-- Ligne du bas -->
        <frame src="bas.html"/>
    </frameset>
</frame>
</frameset>

```

Exemple 20: Imbrication des segmentations

Le site d'Oracle affichant la Javadoc est un bon exemple d'utilisation des frames :

<http://docs.oracle.com/javase/7/docs/api/>

d) Fonctionnalités supplémentaires

Support des navigateurs anciens

Les navigateurs ne supportant pas la segmentation afficheront directement le contenu des balises « noframe ».

```

...
    <frame>
        <!-- Colonne de droite -->
    </frame>
    <noframe>
        <p>
            Votre navigateur ne supporte pas la segmentation, veuillez le
            mettre a jour.
        </p>
    </noframe>
...

```

Exemple 21: Utilisation de « noframe »

Ajout d'un ascenseur

Il est possible d'ajouter un ascenseur à l'élément « frame » en ajoutant l'attribut « scrolling ».

Cette attribut peut prendre la valeur « yes », « no » ou « auto » pour permettre d'afficher un contenu plus grand dans un ou plusieurs segments de la page :

```

<frameset cols="20%,*">
    <frame>
        <!-- Colonne de gauche, affichant le menu (taille fixe) -->
    </frame>
    <frame scrolling="yes">
        <!-- Colonne de droite affichant un article (taille variable) -->
    </frame>
</frameset>

```

Exemple 22: Frame avec ascenseur

Utilisation de liens hypertexte

Avant l'utilisation de la segmentation, l'action d'un lien hypertexte remplaçait généralement

le contenu de la page courante ou ouvrait un nouvel onglet.

Maintenant que la page est segmenté, il est nécessaire de spécifier quels segments doit accueillir le contenu du lien hypertexte. Pour cela, il faut commencer par nommer les segments de manière unique en utilisant l'attribut « name » :

```
<frame scrolling="yes" name="content" src="contenu.html" />
```

Exemple 23: Frame identifiée avec l'attribut « name »

Une fois le segment identifier, il suffit d'utiliser l'attribut « target » de la balise « a » en spécifiant l'identifiant du segment :

```
<frameset cols="20%,*">
  <frame>
    <a href="http://epsi.fr" target="content">L'EPSI</a>
    <!-- Colonne de gauche, affichant le menu (taille fixe) -->
  </frame>
  <frame scrolling="yes" name="content">
    <!-- Colonne de droite affichant un article (taille variable) -->
  </frame>
</frameset>
```

Exemple 24: Utilisation de l'attribut « target »

L'attribut « target » peut avoir les autres valeurs suivantes :

- « _blank » pour afficher le contenu dans une nouvelle fenêtre ;
- « _top » pour afficher le contenu dans l'ensemble de la fenêtre en remplacement de tous les segments ;
- « _self » pour afficher dans le segment contenant le lien.

14. Les formulaires

a) Aperçu

Les formulaires permettent de véhiculer des données saisies par l'utilisateur vers une autre page qui, de manière général fera le traitement de ces données.

Voici les éléments utilisable dans le cadre d'un formulaire :

- « form » permet de définir le formulaire et doit **obligatoirement** posséder un attribut « action » dont la valeur correspond à la page réceptionnant les données ;
- « fieldset » permet de regrouper des champs portant sur le même sujet afin de créer des sections ;
- « legend » permet, dans le cas ou « fieldset » est utilisé, de nommer les sections ;
- « label » définit une étiquette d'un champ de formulaire ;
- « input » définit un champs de formulaire ;
- « textarea » définit un champ de texte multiligne ;
- « button » définit un bouton de formulaire.

b) Utilisation des différents éléments

Ci-dessous un tableau regroupant les attributs les plus utiles en fonction des balises :

Élément	Attribut	Description
form	action	Spécifie la page de réception des données
label	for	Fait référence à l'attribut id d'un champ de formulaire
input	type	Spécifie le type de champ : <ul style="list-style-type: none">• « text » = champ texte ;• « checkbox » = cases à cocher ;• « radio » = bouton radio ;• « file » = champ fichier ;• « image » = soumission avec image ;• « password » = champ mot de passe ;• « submit » = soumission du formulaire ;• « reset » = r.a.z du formulaire ;• « hidden » = champ caché ;• « button » = bouton générique.
textarea	rows	Indique le nombre de ligne du champ texte
	cols	Indique la largeur du champ texte

c) Exemple

Ci-dessous un exemple de formulaire qui utilise la plupart des éléments vu précédemment :

```
<form action="page_de_traitement.php">
  <fieldset>
    <legend>Informations sur vous</legend>
    <label for="nom">Nom</label> :
    <input type="text" id="nom" />
    <label for="prenom">Prénom</label> :
    <input type="text" id="prenom" />
    <hr />
    <input type="checkbox" id="newsletter" />
    <label for="newsletter">Recevoir notre newsletter</label>
    <hr />
    <input type="radio" id="sexe" value="homme" />
    <label for="homme">Homme</label>
    <input type="radio" id="sexe" value="femme"/>
    <label for="femme">Femme</label>
    <hr />
    <label for="photo">Photo</label> :
    <input type="file" id="photo" />
    <hr />
    <label for="pass">Mot de passe</label> :
    <input type="password" id="pass" />
    <hr />
    <label for="champCacher">Champ caché</label> :
    <input type="hidden" id="champCacher" />
    <hr />
    <label for="bouton">Un bouton</label> :
    <input type="button" id="bouton" value="Cliquez ici!" />
    <hr />
    <label for="initialize">Réinitialiser le formulaire</label> :
    <input type="reset" id="initialize" />
    <hr />
    <label for="image">Soumission avec une image</label> :
    <input type="image" id="image" src="image/image.jpg" />
    <hr />
    <label for="soumission">Soumission traditionnelle</label> :
    <input type="submit" id="soumission" />
  </fieldset>
</form>
```

Exemple 25: Formulaire

Nous aborderons, lors de l'étude de PHP, comment récupérer les informations de ce formulaire.

Annexes

1. Index des exemples

Index des exemples

Exemple 1: Page HTML.....	6
Exemple 2: Page XHTML.....	9
Exemple 3: DTD « Frameset ».....	10
Exemple 4: DTD « Transitional ».....	10
Exemple 5: DTD « Strict ».....	10
Exemple 6: Liste non-ordonné.....	13
Exemple 7: Liste ordonnée.....	14
Exemple 8: Balise « font ».....	15
Exemple 9: Image avec lien relatif.....	16
Exemple 10: Image avec lien absolu.....	16
Exemple 11: Lien hypertexte local.....	17
Exemple 12: Lien hypertexte externe.....	17
Exemple 13: Lien hypertexte pour envoi de mail.....	17
Exemple 14: Lien hypertexte intra-page (ancree).....	17
Exemple 15: Tableau simple.....	18
Exemple 16: Tableau évolué.....	19
Exemple 17: Frame avec attribut « source ».....	20
Exemple 18: Segmentation verticale.....	20
Exemple 19: Segmentation horizontale.....	20
Exemple 20: Imbrication des segmentations.....	20
Exemple 21: Utilisation de « noframe ».....	21
Exemple 22: Frame avec ascenseur.....	21
Exemple 23: Frame identifiée avec l'attribut « name ».....	21
Exemple 24: Utilisation de l'attribut « target ».....	22
Exemple 25: Formulaire.....	24

2. Index des illustrations

Index des exemples

Illustration 1:Notepad++	4
Illustration 2:Gedit.....	5