

Chapitre I : LES INSTRUCTIONS DE BASES DU LANGUAGE HTML.

I) Introduction

Le **HTML** (HyperText Markup Language) est un format de présentation de données permettant de créer des pages web pouvant être lues dans des navigateurs.

II) Définition

HTML C'est un langage de description de données, et non un langage de programmation. Il est figé c'est-à-dire qu'une fois le document chargé dans le navigateur, il ne répond à aucune action de l'utilisateur sur le contenu de la page.

Ce langage est pourvu d'un système de balisage qui va permettre de structurer notre document.

III) La Balise HTML

Une balise HTML est un élément que l'on va ajouter au texte de départ pour dire au navigateur de quelle manière l'afficher. Elle n'est pas affichée telle quelle est écrite dans le navigateur, elle est interprétée par celui-ci.

Elle est toujours délimitée par les signes < et >.

Exemple de balise :

`<html>`

Une balise peut en plus comporter de zéro à plusieurs attributs. Les attributs sont des informations complémentaires qui la caractérisent. Ils se présentent sous la forme **nom attribut="valeur"**. Attention à ne pas confondre un attribut HTML et une propriété CSS !

Exemple de balise et son attribut

`<html lang="fr">`

On distingue deux types de balises :

Les balises Simples ou Orphelines et les balises Doubles ou Paires.

1. Balises Simples ou orphelines

Ce sont des balises qui sont dites "vides", c'est-à-dire qu'elles ne vont contenir aucune autre balise HTML. Ces balises n'ont pas besoin d'être fermées.

Exemple de balise simple

``

2. Balises Doubles ou paires

Les balises doubles sont dites ouvrantes/fermantes, c'est-à-dire qu'elles nécessitent deux balises, une ouvrante et une fermante dans lesquelles on va pouvoir mettre d'autres balises ou du texte.

La balise fermante est identique à la balise ouvrante, à la différence qu'elle contient un "/" pour indiquer à quel endroit on la ferme.

Exemple de balise double

Sélectionnez

`<p>Ici du texte ou tout autre balise.</p>`

3. balises Imbriquées

Lorsque l'on cumule l'écriture de plusieurs balises ouvrantes (forcément inévitable), il est impératif de respecter la hiérarchie des balises, c'est-à-dire que la première ouverte sera la dernière à être fermée.

Le chevauchement des balises est une grave erreur.

Exemple de balises imbriquées

Sélectionnez

`<u>Ici les balises sont correctement imbriquées.</u>`

Exemple de balises chevauchées

Sélectionnez

`<u>Ici les balises ne sont pas correctement imbriquées, c'est une erreur.</u>`

Chapitre II : LA STRUCTURE D'UN DOCUMENT HTML

I. Introduction

Avant de commencer tout développement, il est important de connaître la structure d'une page HTML.

Commençons donc par voir celle-ci avec la structure minimale obligatoire d'une page qui sera notre base

`<html>`

`<head>`

```
<title>Titre de la page</title>
<meta http-equiv="content-type" content="text/html;
charset=utf-8">
</head>
<body>

<!-- Ici votre site -->

</body>
</html
```

II. Le DOCTYPE

Il sert d'une part à déclarer quel "langage", "grammaire", ou norme on applique à notre page. Cette déclaration nous permet de soumettre nos pages au [validateur](#) du [W3C](#) (World Wide Web Consortium) qui vérifie que notre code HTML ne comporte aucune erreur de syntaxe (balises mal fermées, utilisation d'attributs inexistants...) en fonction du DOCTYPE, donc de la [DTD](#) (Document Type Definition) choisie. D'autre part, il indique au navigateur de quelle manière interpréter telle balise, telle propriété CSS. L'absence de cette balise est appelée le mode Quirks, ou non conforme, alors qu'au contraire, la présence de cette balise est appelée le mode Standard, ou conforme.

III. Les commentaires

Les commentaires permettent de laisser des informations ou des descriptions du code et sont utilisables à n'importe quel endroit de votre document. Cependant, ce ne sont pas ces commentaires-là qu'il faudra utiliser pour les langages JavaScript et CSS qui possèdent leur propre système de commentaires.

Ces commentaires ne seront pas affichés dans le navigateur mais ils resteront présents dans le code source, donc il faut à tout prix éviter de mettre des informations confidentielles dedans.

Ils sont délimités par `<!--` et `-->` et peuvent être multilignes.

Par souci de compatibilité entre les navigateurs et de mal interprétation de ceux-ci, il est fortement déconseillé d'écrire deux tirets d'union (ou plus) consécutifs.

Commentaires HTML

Sélectionnez

```
<!-- Voici un premier commentaire sur une ligne. -->
```

```
<!-- Voici un deuxième commentaire.  
      Celui-ci est multilignes. -->
```

```
<!------- Cette méthode est déconseillée
```

IV. Le corps du document –BODY

C'est dans la balise `<body>` que nous allons mettre tout le texte et toutes les images qui apparaîtront lorsqu'un visiteur viendra sur notre page.

V. Les niveaux de titres

Il existe différents niveaux de titres, ils vont de 1 à 6 et ont chacun leur importance. La balise représentant ces niveaux de titre est `<hX>` où X représente le niveau.

Exemple :

```
<h1>Titre de ma page</h1>
```

Il est important de respecter la hiérarchie des différents niveaux de titres, sachant que l'élément le plus haut est le `<h1>`. En effet, un titre `<h2>` ne se retrouvera jamais au-dessus d'un `<h1>` dans un document.

Le titre `<h1>` étant le plus important, il convient également de ne le faire apparaître qu'une seule fois dans un document.

Vous ne devez pas choisir un titre pour le rendu qu'il a dans le navigateur (différent selon les navigateurs et pouvant être modifié à l'aide de CSS), mais bien pour le niveau hiérarchique qu'il représente.

VI. Les paragraphes de texte

Comme vous tapez du texte dans un traitement de texte, vous segmentez votre texte en paragraphes. En HTML c'est la même chose, sauf qu'il faut préciser nous-même où commence un paragraphe et où il se termine. C'est ce à quoi sert la balise `<p>`.

Les paragraphes de texte en HTML

`<p>Voici un premier paragraphe.</p>`

`<p>Voici un deuxième paragraphe.</p>`

VII. La mise en forme de texte (texte en Gras, italique, etc.)

Les différentes techniques citées ci-dessous pour mettre en forme du texte sont **obsolètes**. Si elles fonctionnent encore en HTML 4.01, il vaut mieux s'en séparer et les remplacer par un langage fait pour ça et beaucoup plus puissant : le CSS

1. Le texte en gras

Pour mettre le texte en gras on l'encadre de la balise `` :

Texte en gras

`Ce texte s'affichera en gras.`

2. Le texte en italique

Pour mettre le texte en italique on l'encadre de la balise `<i>` :

Texte en italique

`<i>Ce texte s'affichera en italique.</i>`

3. Le texte souligné

Pour souligner le texte on l'encadre de la balise `<u>` :

Texte souligné

`<u>Ce texte sera souligné.</u>`

4. Le texte barré

Pour barrer le texte on l'encadre de la balise `<s>` :

Texte barré

Sélectionnez

`<s>Ce texte sera barré.</s>`

5. Le texte en exposant

Pour placer le texte en exposant on l'encadre de la balise `<sup>` :

Texte en exposant

```
<sup>Ce texte sera en exposant.</sup>
```

6. Le texte en indice

Pour placer le texte en indice on l'encadre de la balise `<sub>` :

Texte en indice

```
<sub>Ce texte sera en indice.</sub>
```

7. Modification de la couleur du texte

Pour modifier la couleur du texte on utilise l'attribut `color` de la balise `` :

Texte en rouge

```
<font color="#ff0000">Ce texte sera en rouge.</font>
```

Les couleurs peuvent être écrites de deux manières :

En hexadécimal de type RVB et précédées d'un dièse (#) ; Exemples :
#ff0000 => rouge, #00ff00 => vert, #0000ff => bleu.

Textuelles en anglais US ; Exemples : red, yellow, pink.

8. Modification de la police du texte

Pour modifier la police du texte on utilise l'attribut `face` de la balise `` :

Texte en verdana

```
<font face="verdana">Ce texte sera en verdana.</font>
```

On a tendance à écrire des familles de polices plutôt qu'une seule police. En effet, il se peut que sur un système différent du vôtre la police choisie ne soit pas installée et il faut qu'il puisse alors afficher le texte dans une police proche de celle que vous vouliez avoir.

On sépare alors les différentes polices par une virgule dans l'ordre de sélection le cas où la précédente ne peut pas être affichée.

Texte en verdana ou sans-serif

``Ce texte sera en verdana ou en sans-serif si verdana n'est pas installée.``

9. Modification de la taille du texte

Pour modifier la taille du texte on utilise l'attribut **size** de la balise `` :

Texte en taille 5

``Ce texte sera en taille 5.``

Par défaut, la valeur de l'attribut **size** vaut "3".

Il y a deux manières de modifier la valeur de cet attribut : et les valeurs possibles sont les entiers de "1" à "7".

En absolu : Les valeurs possibles sont les entiers de "1" à "7".

En relatif : Les valeurs possibles les entiers de "-2" à "+4".

Et il est bien évidemment possible de renseigner ces trois attributs (**color**, **face**, **size**) à la fois dans la même balise ``.

10. Le déplacement du texte

Pour déplacer un texte (d'un endroit à un autre), on utilise la balise `<marquee>`.

Cette balise à pour attributs :

- 1) **Behavior** : il définit le comportement du texte qui se déplace ;il contient trois(3) valeurs :
 - **Alternate** : le déplacement du texte du coin au coin ;
 - **Slide** : le déplacement du texte avec des arrêts ;
 - **Scroll** : le déplacement progressif du texte.
- 2) **Scrollamount** : définit la vitesse du déplacement du texte et cette vitesse va de 1 à 10.
- 3) **Loop** : il indique le nombre de tour qu'effectue le texte
- 4) **Direction** : indique la direction du texte. Il a pour valeurs :
 - **Up** : direction vers le haut ;
 - **Down** : direction vers le bas ;
 - **Left** : direction vers la gauche ;
 - **Right** : direction vers la droite.

- 5) L'attribut **bgcolor** permet de changer la couleur du fond (de l'arrière-plan).

Exemple :`<marquee bgcolor="#00adec" Behavior="alternate" >`
PAGE1`</marquee>`

Chapitre III : Les images et zones réactives

I- Les images

Pour afficher une image en HTML, on utilise la balise ``. Elle admet différents attributs, dont les plus importants sont `src` et `alt`.

L'attribut `src` permet de renseigner le chemin de l'image, qui peut être [absolu](#) HTTP ou [relatif](#), très utile pour pouvoir l'afficher.

L'attribut `alt` affiche un texte alternatif à la place de l'image dans le cas où, pour une raison ou autre, celle-ci ne peut pas être chargée. Il aide de plus les moteurs de recherche à référencer l'image et les déficients visuels à savoir ce que l'image représente.

I.1) Affichage d'une image :

```

```

L'attribut `alt` n'est pas censé créer une infobulle lors du survol de l'image, qui est le rôle de l'attribut `title`, cependant sur Internet Explorer l'`alt` affiche cette infobulle.

Les différents formats d'image supportés par cette balise sont :

- GIF
- JPG
- PNG
- BMP (à éviter)

L'attribut align : il spécifie l'alignement de l'image par rapport au texte adjacent. Il peut prendre plusieurs valeurs : top, center ...

I.2) Affichage d'une image en arrière-plan

Pour afficher une image en arrière-plan, on utilise l'attribut **background** de la balise `body`.

Exemple : `<body background = "k.jpg">`

Attributs	Valeurs	Résultats
Align	Top Center ou middle Left ou Justify Right Bottom	Alignement au-dessus (en haut) Alignement au centre Alignement à gauche Alignement à droite Alignement au-dessous (en bas)
Alt		Texte alternatif au cas où l'image ne s'affiche pas
Border	Nombre entier	Nombre de pixels de la bordure. Par défaut l'attribut border vaut 1 ; ce qui crée un petit cadre autour de l'image. Pour éviter le cadre, placer zéro (0).
height	Nombre entier	hauteur de l'image (en pixels ou %)
Hspace	Nombre entier	Nombre de pixels d'ajustement entre l'image et le texte adjacent (horizontalement)
Width	Nombre entier	largeur de l'image (en pixels ou %)
name		Permet de définir un nom pour l'image
title		Texte alternatif au cas où l'image.
Usemap		url ou nom de l'ancre définissant l'image réactive.

II- Les zones réactives

Le HTML permet de créer des zones réactives sur les images, afin de définir plusieurs liens différents sur une seule et même image, ou de ne rendre qu'une seule partie de l'image cliquable. On appelle l'ensemble de ces zones une carte, ou map.

Les balises qui permettent de faire ces zones réactives sont `<map>` et `<area>` étroitement liées avec la balise ``. En effet, on doit forcément préciser sur quelle image ces zones vont être présentes.

L'attribut de la balise `` qui va contenir le nom de la map dont elle fait référence est `usemap`. Le nom de cette map est précédé du signe dièse (#).

Image mappée

```

```

La balise qui va contenir l'ensemble de nos zones réactives pour cette image est la balise `<map>`, qui doit obligatoirement avoir l'attribut `name` renseigné et qui aura pour valeur le nom de la map, le même renseigné dans l'attribut `usemap` de notre image.

Comme vous l'aurez sans doute deviné et comme son nom l'indique, la balise `<area>` sera une zone réactive dans notre map. Il y a une balise area par zone.

II-1. Structure d'une map

```
<map name="nomDeLaMap">
```

```
<!-- Ici nous allons mettre nos différentes zones réactives -->
```

```
<area ...>
```

<area ...>

</map>

Les différents attributs de la balise `<area>` nécessaires à son bon fonctionnement sont :

- **coords**, qui va contenir les différentes coordonnées de notre zone.
- **href**, qui va définir l'adresse de destination du lien.
- **shape**, qui va renseigner le type de zone que l'on souhaite afficher.

On distingue trois types de zones :

- Les zones rectangulaires,
- Les zones circulaires,
- Les zones polygonales.

On peut bien évidemment avoir autant de zones que l'on veut sur une map, et on peut également combiner les différents types de zones sur une seule et même image.

Attention toutefois à ne pas faire se chevaucher les zones ! Le cas échéant, c'est la zone définie la première qui s'affichera au-dessus.

II-2. Les zones rectangulaires

Pour définir une zone rectangulaire, l'attribut **shape** doit prendre pour valeur **"rectangle"** ou **"rect"**.

Les coordonnées définies dans l'attribut **coords** se réfèrent aux abscisses et ordonnées des coins supérieur gauche et inférieur droit du rectangle que l'on souhaite tracer par rapport au coin haut gauche de l'image. Ces valeurs sont séparées par une virgule et exprimées en pixels.

Syntaxe :

Zone rectangulaire

```
<area shape="rect" coords="x1,x2,y1,y2" href="adresse/de/destination">
```

Exemple de la zone rectangulaire :

```
<area shape="rect" coords="10,50,60,60" href="adresse/de/destination">
```

II-3. Les zones circulaires

Une zone circulaire est définie par la valeur **"circle"** ou **"circ"** de l'attribut **shape**. Les valeurs de l'attribut **coords** se réfèrent aux abscisse et ordonnée du centre du cercle ainsi que le rayon de ce cercle par rapport au coin haut gauche de l'image. Ces valeurs sont séparées par une virgule et exprimées en pixels.

Syntaxe :

Zone circulaire

Sélectionnez

```
<area shape="circ" coords="x,y,rayon" href="adresse/de/destination">
```

Exemple :

Zone circulaire

Sélectionnez

```
<area shape="circ" coords="40,40,10" href="adresse/de/destination">
```

II-4. Les zones polygonales

C'est la valeur "**polygon**" ou "**poly**" de l'attribut **shape** qui permet de définir une zone polygonale.

Les valeurs de l'attribut **coords** se réfèrent aux abscisse et ordonnée de chacun des points par rapport au coin haut gauche de l'image. Ces valeurs sont séparées par une virgule et exprimées en pixels.

Syntaxe :

Zone polygonale

Sélectionnez

```
<area shape="poly"
coords="x1,y1,x2,y2,x3,y3,...,xn,yn" href="adresse/de/destination">
```

Exemple :

Zone polygonale

```
<area shape="poly" coords="0,0,40,50,78,101,17,150"
href="adresse/de/destination">
```

Exemple :

```

  <map name="carte">
    <area shape="circ" coords="250px, 144px, 100" title="Papa">
    <area shape="circle" coords="567px, 205px, 100" href="page2.html"
      title="penguin 2">
  </map>
```

III- Les audio (.mp3 ; .wav ;.ogg,...)

Pour insérer de l'audio sur une page html(web), on utilise la balise

<audio>. La balise <audio> sans attributs sources ne s'achève pas ; l'attribut controls sert à afficher les boutons de contrôles, tel que les boutons de lecture, de pause et volume entre autre.

L'attribut controls est obligatoire et n'a pas besoin de valeur.

- Autoplay : permet de lancer automatiquement le fichier audio lors du téléchargement de la page.
- Loop : l'audio se répète en double ;
- Width : modifie la largeur de la barre de lecture ;
-

iv- Les liens hypertexts

Un lien hypertexte est un élément HTML permettant d'envoyer le visiteur vers une nouvelle page. On peut insérer dans cet élément toute sorte de textes, images et autres balises dites "inline" (comme les balises , , etc).

Un lien sera représenté dans le code par la balise <a>. Tout ce qui sera dans cette balise fera office de lien. L'adresse de destination doit se trouver dans l'attribut href.

L'attribut title, quant à lui, permet de renseigner un petit descriptif sur le contenu de la page cible, et est un atout non négligeable en matière de référencement et d'accessibilité.

Exemple texte :

Exemple d'un lien texte

```
<a href="adresse/de/destination.html">Texte du lien</a>
```

Exemple d'un lien image

```
<a href="adresse/de/destination.html">
```

```
 </a>
```

On distingue deux types de liens, les liens externes et les liens internes.

III-1. Les liens externes

Un lien externe est un lien hypertexte qui renvoie vers une autre page. Cette autre page peut être une page de votre site ou celle d'un site tiers.

III-2. Les liens internes

Un lien interne est un lien qui renvoie vers la même page, mais pas forcément au même endroit de la page.

III-2-1. Les ancres

Les ancres permettent de naviguer à l'intérieur d'une même page. Par exemple, ce tutoriel a en haut un sommaire avec des liens qui renvoient plus bas dans la page, ce sont des ancres.

Pour faire une ancre, il faut utiliser le signe dièse (#) suivi d'un mot-clef. Ce mot-clef sera repris dans l'attribut **id** d'une balise à laquelle on fait faire une ancre :

Exemple d'une ancre interne

```
<a href="#motClef">Texte de mon ancre</a>
```

[...]

```
<h2 id="motClef">Texte vers lequel l'ancre est renvoyée</h2>
```

Si vous voulez faire une ancre vers une page externe, il vous suffit de rajouter l'adresse de destination dans la balise de lien :

Exemple d'une ancre externe

```
<a href="adresse/de/destination.html#motClef">Texte de mon ancre</a>
```

Si vous utilisez des cadres pour votre navigations, vous aurez certainement envie que les liens de votre menu, dans le cadre nommé "cadreMenu" par exemple, pointent vers votre cadre principal que l'on appellera "cadrePrincipal".

Dans ce cas, il faut renseigner l'attribut **target** de la balise de lien :

Exemple d'un lien vers un autre cadre

```
<a href="adresse/de/destination.html" target="cadrePrincipal">Lien de mon menu</a>
```

Il existe plusieurs valeurs pour l'attribut **target** :

Valeur	Description
_self	Affichage dans le cadre courant (par défaut).
_blank	Ouvre le lien dans une nouvelle fenêtre.
_top	Affichage dans la totalité de la fenêtre courante.
_parent	Affichage dans le cadre qui enveloppe la structure courante.
NomDuCadre	Affichage dans le cadre spécifié.

Les liens e-mails

Vous pouvez permettre à vos visiteurs d'écrire sur votre adresse e-mail avec la syntaxe "**mailto:**"

Exemple d'un lien vers une adresse e-mail

`Ecrivez-moi`