

Support de cours Module XHTML/CSS

2eme Edition – 2014

Ce support de cours contient quelques explications sur certains concepts de HTML en vigueur au programme en 1ère année BTS option IDA en Côte d'Ivoire. Il a été élaboré sur la base de supports de cours existants et de sources trouvées sur Internet.

TABLE DES MATIERES

1^{ère} PARTIE : LE LANGAGE HTML

INTRODUCTION	- 3 -
Chapitre I – NOTION DE BALISES.....	- 5 -
Chapitre II – LES BALISES EN BLOC ET LES BALISES EN LIGNE	- 11 -
Chapitre IV – LES BALISES DE TABLEAU.....	- 16 -
Chapitre V – LES BALISES D’IMAGE ET D’IMAGES-MAPS	- 20 -
Chapitre VI – LES BALISES DE FORMULAIRE	- 23 -
Chapitre VII – LES BALISES DE CADRE.....	Error! Bookmark not defined.
Chapitre VIII – LA BALISE D’HYPERLIEN	- 32 -
Chapitre IX – LES BALISES DE LISTE	- 34 -
Chapitre X – LES BALISES DE CONTENU EXECUTABLE	- 37 -

2^{ème} PARTIE : LES FEUILLES DE STYLE OU CSS

Chapitre XI – LES FEUILLES DE STYLE (CSS).....	- 39 -
Mini-PROJET:	- 47 -

INTRODUCTION

Historique du langage HTML

HTML (*Hypertext Markup Language*) est un langage de balisage qui permet de structurer le contenu des pages web.

Conçu par Tim Berners-Lee en 1991, HTML¹ est un langage basé sur le SGML (*Standard Generalized Markup Language*) : il s'agit en effet de présenter un contenu affichable non seulement par les ordinateurs classiques, mais également sans trop de dégradation par des terminaux mobiles (PDA : Personal Digital Assistant, Smartphones, etc) bien moins puissants. (Wikipedia, 2011).

Les versions successives de HTML sont HTML 1 en 1991, HTML 2 en 1994, HTML 3 en 1996 et HTML 4 en 1998. En 2000, le W3C lance la version XHTML qui est une variante plus rigoureuse du langage.

Les premières spécifications du XHTML ont vus le jour en 2000 sous l'appellation XHTML 1.0. Un an plus tard, elle fut remplacée par la version 1.1. En 2009, le W3C lance une étude pour la version HTML 5.

Le langage de styles CSS

Parallèlement au HTML, le langage de mise en page et de mise en forme des pages web fut développé. Les différentes versions sont CSS 1 en 1996. Ensuite suivies les versions CSS 2.1 en 1998 et enfin la dernière version CSS3.

L'environnement de travail

Pour créer des pages web avec XHTML et CSS, il faut disposer d'un environnement de travail adapté.

Si en théorie, un simple éditeur de texte (tel que le Bloc Notes de Windows ou Emacs de Linux) peut suffire, en pratique, il existe des éditeurs plus élaborés permettant de gagner du temps.

Les éditeurs visuels

Avec ces éditeurs vous avez la possibilité de travailler graphiquement sur la page sans taper de code qui est généré automatiquement. L'on peut citer le plus connu est Dreamweaver, FrontPage (payant ou en version d'essai) et Nvu (libre et gratuit).

Leur inconvénient est qu'ils rendent le développeur trop passif et le code généré est difficilement maîtrisable (souvent problème de compatibilité avec les nouvelles spécifications).

Les éditeurs classiques

Ils offrent des fonctions d'aide à la saisie. Souvent gratuits en téléchargement, ils servent surtout à la phase de création de la structure de base du document. On peut citer entre autres EditPlus, NotePad ++, PSPad, JEdit et bien d'autres pour Windows ; Espresso, Smultron, TextWrangler pour Macs et gEdit, Vim, Emacs, Kate. Vous trouverez une variété d'éditeur en faisant une recherche sur Google.

¹ Le langage HTML, qui a maintenant évolué vers le XHTML, fut inventé en 1991 par Tim Berners-Lee lorsqu'il travaillait au CERN de Genève (Centre Européen de Recherche Nucléaire). L'objectif était alors d'afficher des pages d'information ayant les propriétés suivantes :

- les pages étaient reliées entre elles par des liens hypertexte (ces liens sur lesquels, aujourd'hui, nous n'arrêtons pas de cliquer !)
- ces documents devaient être lisibles sur tous les ordinateurs, quel que soit leur système d'exploitation : Windows, Unix, Mac OS, etc. (Drailard, 2008)

Les navigateurs web

Les navigateurs permettent de visualiser le contenu des pages web. Ils traduisent les codes HTML et CSS et présentent le résultat visuel à l'écran. Il existe de nombreux navigateurs parmi lesquels on peut citer :

- Google Chrome
- Mozilla FireFox
- Internet Explorer
- Safari
- Opera

Il existe des variantes de ces navigateurs pour les appareils mobiles comme les Smartphones. On peut citer :

- Safari Mobile sur iPhone d'Apple
- Chrome Mobile pour les appareils sous Android
- Internet Explorer Mobile pour les appareils sous Windows Mobile

Il est important de noter qu'il faut disposer de plusieurs navigateurs dans leurs dernières versions sur notre ordinateur afin de tester la conformité du code HTML et CSS des pages avec les dernières spécifications du langage.

Fonctionnement d'un site web

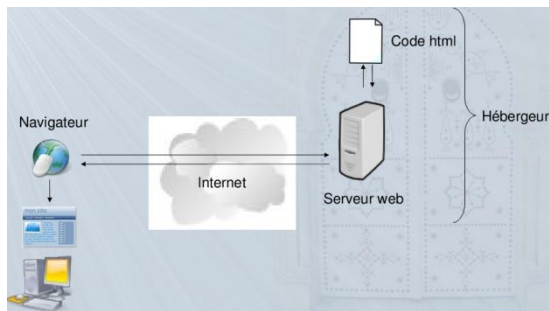


Figure 1 - Fonctionnement d'un site web statique

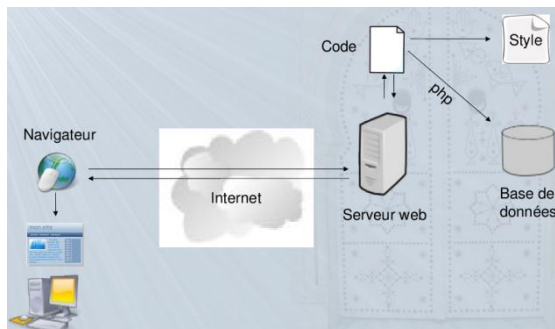


Figure 2 - Fonctionnement d'un site web dynamique

CHAPITRE I – NOTION DE BALISES

Les différentes parties d'une page web (la structure) sont représentées dans le code HTML sous forme d'**éléments** constitués de **balises**. On parle alors de langage de balisage.

Dans un langage de balisage, tout contenu, tels que les blocs de texte, les images, les tableaux, paragraphes et autres, doivent être renfermé dans un élément.

La liste des éléments utilisables est clairement définie dans la **DTD** (*Document Type Definition*) qui dépend de la version du langage. La DTD contient les spécifications du langage qui sont établies par le W3C (World Wide Web Consortium), organisme de normalisation dans le domaine du web.

I- LES BALISES

Les balises HTML structurent le contenu du document, en délimitant les blocs qui seront amenés à contenir des paragraphes, des titres, différents types de médias (images, sons, vidéos), des contrôles de formulaires ou encore des liens hypertextes.

Une balise débute par le signe « inférieur à », et finit par le signe « supérieur à » ; on parle aussi de chevrons.

Sauf exception, elles fonctionnent par paires. Et à chaque balise ouvrante correspond sa balise fermante, débutant par les caractères `</`. Ces paires délimitent la portion de texte à laquelle elles s'appliquent.

Syntaxe : `<nombalise> Portion du document </nombalise>`

Exemple de balises : `, <p>, <h1>, </h1>, `

Il peut arriver qu'une balise soit refermée immédiatement. On parle alors de « **balise vide** » ou de « **balise simple ou encore orpheline** ». Dans ce cas, pour marquer la fermeture de la balise, on utilise les caractères `</>`.

Voici quelques balises dans ce cas sont : `area`, `base`, `br`, `col`, `command`, `embed`, `hr`, `img`, `input`, `keygen`, `link`, `meta`, `param`, `source`, `wbr`.

Syntaxe : `<nombalise />`

Exemple de balise simple : ``

Un élément est l'ensemble composé d'une balise ouvrante, d'un contenu (du texte et/ou d'autres balises), et de la balise fermante correspondante.

Exemple d'élément :

` texte à mettre en gras ` ===> Toute cette ligne représente un élément

II- LES ATTRIBUTS

Les attributs ou propriétés sont des caractéristiques qui permettent de personnaliser les balises. Ils définissent des options supplémentaires pour la balise.

Les attributs sont toujours placés à l'intérieur la balise ouvrante. Leur valeur est toujours entourée d'apostrophes simples (') ou doubles (").

Syntaxe : `<nombalise nomattribut="valeur">`

Exemple :

```
<h1 id="monTitre">Bienvenue sur le site web de GOBIS TRAINING</h1>  

```

Dans l'exemple ci-dessus, l'élément 1 représente un titre de premier niveau `<h1>` possédant un attribut `id` de valeur `monTitre`.

L'élément 2 représente une image dont l'attribut `src` indique qu'il s'agit du fichier `photo.jpg` dans le dossier `images`.

NB : Certains attributs sont devenus obsolètes avec l'avènement du CSS. Ce sont pour la plupart des attributs de mise en forme tels que : `align`, `valign`, `color`, `bgcolor`, `border`, `face`, `width`, `height`.

III- IMBRICATION DES ELEMENTS

Les éléments peuvent être imbriqués, à condition de respecter une certaine hiérarchie, et d'éviter de faire se chevaucher des balises ouvrantes et fermantes qui ne se correspondent pas.

```
Bon exemple : <h2>Bienvenue chez Toto <strong>et moi</strong></h2>  
Mauvais exemple : <h2>Bienvenue chez Toto <strong>et  
moi</h2></strong>
```

Dans cet exemple, nous avons un titre de niveau 2 dans lequel la portion de texte « et moi » est renforcé (en gras).

Dans le bon exemple, la balise ouvrante `` se ferme à l'intérieur de l'élément défini par la balise `<h2>`.

Les possibilités d'imbrications doivent répondre à des critères stricts qui définissent quels éléments peuvent en contenir d'autres, et quelles sont les conséquences en termes d'affichage ou de sémantique.

Globalement, un élément de type bloc (bloc-level) peut contenir soit d'autres blocs et du contenu texte, soit uniquement des éléments de type en ligne (inline) et du contenu texte.

Un élément de type en ligne ne peut pas contenir d'éléments de type bloc, mais uniquement des éléments en ligne ou du contenu texte.

Il faut noter qu'un document HTML est une structure composée d'un ensemble d'éléments à plusieurs niveaux d'imbrication. Le premier niveau de cette hiérarchie est constitué par la balise `<html>`.

IV- QUELQUES REGLES D'ECRITURE DU HTML

Règles pour les noms des fichiers

Chaque page web est un fichier dont le nom peut comprendre des lettres, des chiffres et des tirets. Il faut éviter : les espaces, les caractères accentués et le « ç ». Son extension est généralement « **.html** ».

La première page d'un site web doit être nommé index.html (ou index.htm).

Quelques règles d'écriture des balises HTML

La norme HTML impose les règles suivantes :

- Pour respecter les règles du XHTML, il faut toujours écrire les noms de balises en minuscules
- Toute balise ouverte doit être refermée ; y compris les balises simples ou orphelines.
Dans ce dernier cas, la barre de fermeture est alors intégrée dans la balise elle-même.

Exemples :

Saut de ligne: `
` au lieu de `
`

Tracé d'une ligne horizontale : `<hr />` au lieu de `<hr>`

Image: `` au lieu de ``

- En cas d'imbrication de balises, l'ordre de fermeture est l'inverse de celui de l'ouverture
- Les valeurs des attributs des balises sont à écrire entre guillemets simples ou doubles :
``.
- Tous les attributs utilisés doivent avoir une valeur, y compris ceux dont la valeur est unique. Par exemple, il ne faut plus écrire : `<input type="checkbox" checked />` mais le code suivant : `<input type="checkbox" checked="checked"/>`
- L'attribut name qui servait à identifier certains éléments (`<a>`, `<form>`, `<input>`) doit être remplacé par l'attribut *id*.

V- STRUCTURE D'UNE PAGE HTML 5

Une page HTML ou XHTML s'écrit de la façon suivante :

```
<!DOCTYPE html>
<html>
  <head>
    <meta name="charset" content="utf-8" />
    <title>Titre</title>
  </head>
  <body>
    </body>
</html>
```

- Le reste de la page est encadré par des balises `<html>` et `</html>` qui signifient début et fin de la page HTML.
- Entre ces deux balises se trouvent deux parties :
 - l'en-tête de la page entre `<head>` et `</head>` qui contient les informations qui ne seront pas visibles sur la page sauf la balise `<titre>` qui contient le texte à afficher dans la barre de titre.

- et le contenu (le corps) de la page entre <body> et </body>.

VI- LES ELEMENTS DE BASE

La déclaration DOCTYPE

```
<!DOCTYPE html >
```

Sur la première ligne, la balise <!DOCTYPE> indique au navigateur le type de document qui va être créé.

Elle est obligatoire dans tout document.

L'élément racine <html>

```
<html>
```

Il représente le parent. Il contient la racine de tout le document HTML. De ce fait, les balises <html>, </html> encadre tous les éléments.

L'entête du document : L'élément <head>

Cette section donne quelques informations sur la page ; telles que le titre, l'encodage des caractères spéciaux, les codes externes utilisés.

Parmi ses éléments enfants, on peut citer : <link>, <meta>, <script>, <style> et <title>.

Seul le contenu de l'élément <title> est visible dans la barre de titre de la fenêtre.

Parmi ces éléments seuls les éléments <title> et <base> ne peuvent être utilisés qu'une seule fois. Les autres peuvent être utilisés de façon illimitée.

Les meta-informations : l'élément <meta />

Les informations contenues dans les balises <meta /> sont destinées au serveur web, au navigateur et aux moteurs de recherche.

Chaque information est identifiée par un nom et un contenu. Le nom est défini par l'attribut **name** et le contenu par l'attribut **content** sous la forme qui suit :

```
<meta name="nom" content="valeur" />
```

Leur rôle diffère selon la valeur de l'attribut name. Ainsi, on peut définir :

L'auteur de la page : **name="author"**

```
<meta name="author" content="Gogbé Benjamin" />
```

Les mots clés : **name="keywords"**

```
<meta name="keywords" content="Cours XHTML, Les balises, ..." />
```

Une description de la page : **name="description"**

```
<meta name="description" content="Cours de développement web pour le cycle BTS IDA 1" />
```

Le temps de rechargement automatique de la page : **http-equiv="refresh"**

```
<meta http-equiv="refresh" content="30" />, la page sera rafraichie chaque 30 secondes.
```

Encodage des caractères : **charset**


```
<meta charset="utf-8" />
```

Permet de définir le jeu de caractères utilisé dans la page. Il existe plusieurs jeux de caractères dont les plus utilisés sont:

1. **iso-8859-1** : encodage classique pour les langues d'Europe occidentale
2. **iso-8859-15** : complète le précédent avec les caractères € et œ.
3. **utf-8** : utilise les caractères de la majorité des langues du monde. Il permet d'utiliser les caractères spéciaux et les accents.

Exercice pratique :

Créer une page contenant un texte vous décrivant en quelques mots (au plus 2 lignes).

NB :

1. le texte doit contenir des accents et des caractères spéciaux
2. définir les mots-clés et une description afin que votre page soit visible par les moteurs de recherche.

Les commentaires

Ils permettent une meilleure compréhension, une réutilisation facile et une meilleure lisibilité du code lors des mises à jour. On utilise les symboles : <!-- et -->.

```
Exemple : <!-- Menu principal du site -->
```

Le corps du document : l'élément <body>

C'est le conteneur de l'ensemble des éléments textuels et graphiques de la page web. Le corps du document est délimité par les balises <body> et </body>.

```
<!DOCTYPE html >
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <meta charset="utf-8" />
  <title>Première page XHTML</title>
</head>
<body>
<!-- Contenu du document -->
</body>
</html>
```

Exercices

Exercice 1 : À quel emplacement est défini la DTD utilisée dans le document ?

Exercice 2 : Quels sont les éléments enfants de l'élément <html> ?

Exercice 3 : À quoi sert l'élément <head> ? Quels peuvent être ses éléments enfants ? Peuvent-ils être employés plusieurs fois dans le même document ?

Exercice 4 : Quel élément est obligatoire dans l'élément <head> ? À quoi sert-il ?

Exercice 5 : Comment déclarer les mots-clés associés à la page ? Quelle est l'utilité de cette déclaration ? Écrivez-en un exemple.

Exercice 6 : Peut-on écrire le code suivant dans une page ?

```
<body>
  <title>Bienvenue dans notre site </title>
  <h1>Le site du XHTML et de CSS2</h1>
</body>
```

Exercice 7 : Peut-on inclure les éléments , <form> et dans l'élément <body> ?

Exercice 8 : Écrire le code HTML complet d'une page web comportant les éléments suivants :

1. *Titre : Ma bienvenue sur ma page personnelle*
2. *Mots clés : GOGBE, Benjamin, Informaticien, Ingénieur, Enseignant*
3. *Rechargement chaque 10 minutes*
4. *Auteur : Benjamin GOGBE*
5. *Description : Ma page personnelle*
6. *Jeu de caractères : utf-8*

Insérer des commentaires à chaque niveau pour expliquer chaque étape du code HTML.

CHAPITRE II – LES BALISES EN BLOC ET LES BALISES EN LIGNE

Le langage HTML distingue deux grandes catégories de balises :

1. les balises en bloc (block)
2. et les balises en ligne (inline).

I- LES ELEMENTS DE STRUCTURE OU DE BLOC

1- Définition

Ce sont les éléments qui se placent *automatiquement les uns en dessous des autres*.

Le rendu de ces balises occupe par défaut toute la largeur de la page.

Généralement, le navigateur insère automatiquement un espace au-dessus et en dessous du contenu de la balise.

Les blocs distinguent des sections entières de la page comme des titres, des paragraphes, des listes, des citations, etc.

```
Exemples d'éléments de type bloc
<h1>...</h1> <!-- titre de niveau 1 -->
<p>...</p> <!-- paragraphe -->
```

Les blocs peuvent contenir d'autres blocs et des éléments en ligne, à l'exception des paragraphes `<p>` et des titres `<h1>`, `<h2>`,...`<h6>` qui ne peuvent inclure d'autres éléments de bloc.

2- Les principaux éléments de type bloc

(Goetter, 2005) P 29

Balise	Commentaires	Exemple d'utilisation
<code><blockquote></code>	Introduit des citations longues. Par défaut, certains navigateurs prévoient une marge gauche aux blocs de citation, qu'on pourra bien sûr modifier en CSS.	<code><p>Comme le dit toujours ma grand-mère :</p></code> <code><blockquote><p>il ne faut pas vendre la peau de l'ours avant de l'avoir tué</p></code> <code><p>non non, il ne faut pas !</p></blockquote></code>
<code><div></code>	Conteneur générique de type bloc. Cette balise n'apporte pas de sens spécifique, à l'instar de son équivalent en ligne <code></code> . Elle sert à regrouper d'autres balises bloc ou en ligne.	<code><div><p>Voici un texte important.</p></code> <code><p>Voici un autre texte</p>.</div></code>
<code><dl></code>	Liste de définitions, utile pour structurer les éléments associant des définitions ou contenus à des termes ou à des titres. Ces listes distinguent les titres (<code><dt></code>) des éléments de définition (<code><dd></code>).	<code><dl></code> <code><dt>Titre de l'élément</dt></code> <code><dd>description de l'élément</dd></code> <code><dd>Suite de la description</dd></code> <code></dl></code>

Balise	Commentaires	Exemple d'utilisation
<form>	Balise délimitant un formulaire interactif. Elle contient généralement des éléments d'interface (champs de texte, boutons de validation, cases à cocher, etc.) Cet élément doit renfermer immédiatement d'autres éléments de type bloc.	<pre><form action="pagesuivante.php" method="get"> <p> <input type="text" name="recherche" /> <input type="submit" value="ok" /> </p> </form></pre>
<h1>, <h2>, ... <h6>	HTML prévoit six niveaux de titres, hiérarchiquement placés sous le titre principal (<h1>). Rappel : ces éléments constituent une exception à la règle des blocs ; ils ne peuvent pas contenir d'autres blocs.	<pre><h1>Un titre principal</h1></pre>
, 	Ces deux balises désignent des listes ordonnées () ou à puces simples (). Elles comportent exclusivement les éléments d'objet de liste ().	<pre> premier objet de la liste second objet de la liste </pre>
<p>	Balise désignant un paragraphe de texte. Cet élément constitue une exception à la règle des blocs car il ne peut en contenir d'autres.	<pre><p>Un paragraphe de texte</p></pre>
<table>	Tableau contenant des données. Les cellules du tableau sont d'abord rassemblées sous forme de rangées ou lignes (<tr>).	<pre><table> <tr> <td>cellule</td> <td>autre cellule</td> </tr> </table></pre>

Exercice pratique : TP BALISES DE BLOC 1

Réaliser une page web en utilisant ci-dessous :

```
<!DOCTYPE html >
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <!--Exemples d'utilisation des différentes variantes de l'élément
<meta> -->
    <meta charset="utf-8" />
    <meta name="author" content="Gogbé Benjamin" />
    <meta name="keywords" content="Cours XHTML, Les balises,
Exemples du cours" />
    <meta name="description" content="Cours de développement
web pour le cycle BTS IDA 1" />
    <meta name="refresh" content="10" />
    <title>Exemples Cours HTML 5</title>
  </head>
```

```

<body>
  <!--Utilisation des balises <blockquote> et <ul> (les puces)-->
  <!--Dans un premier temps ne pas utiliser les éléments de mise en
forme ou en ligne-->
  <!-- Utiliser les éléments progressivement -->
  <p>Paroles d'or du père de la Nation ivoirienne <strong>Président
Houphouet Boigny</strong>:</p>
  <blockquote>
    <ul>
      <li>La paix n'est pas un vain mot. C'est un
comportement. Alors faisons en sorte de graver la paix dans chacun de
nos actions.</li>
      <li>Le vrai bonheur on ne l'aperçoit que lorsqu'on
l'a perdu. Faisons en sorte de jamais la perdre</li>
    </ul>
  </blockquote>
  <!-- Utilisation de <div>-->
  <div>
    <h3>Les fondamentaux du XHTML et
du CSS</h3>
    <dl>
      <dt>Les balises de bloc</dt>
      <dd>Toutes les balises
permettant de réaliser la structure du
document</dd>
      <dd>Il s'agit également de
choisir la bonne imbrication des balises</dd>
    </dl>
  </div>
  <!--utilisation de <table> -->
  <div>
    <h5>Tableau de quelques éléments
XHTML</h5>
    <table>
      <thead><th>Balise</th><th>Commentaires</th></thead>
      <tbody>
        <tr><td>body</td><td>Corps du
document</td></tr>
        <tr><td>head</td><td>Entête du
document</td></tr>
      </tbody>
    </table>
  </div>
</body>
</html>

```

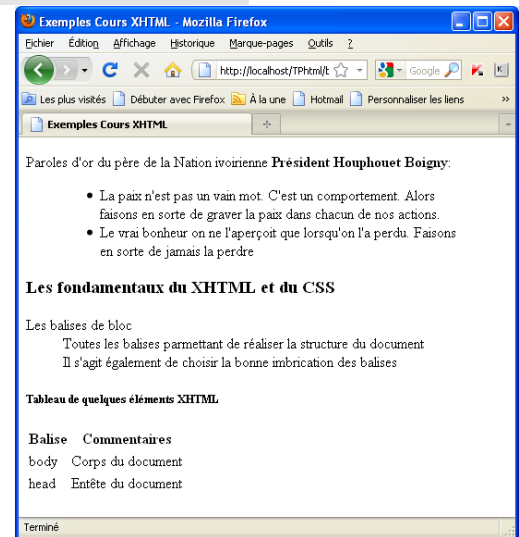


Figure 3 : Aperçu TP 1 Balises de blocs

II- LES BALISES EN LIGNE

1- Définition

Contrairement aux balises de bloc, les balises en ligne se placent toujours les unes à côté des autres.

Les éléments en ligne sont prévus pour enrichir localement des portions de texte (lien hypertexte, emphase, renforcement, etc.).

Par exemple : le renforcement d'une partie de texte à l'aide de la balise ``

Les éléments en ligne ne peuvent contenir que d'autres éléments en ligne mais pas d'éléments de bloc.

Il existe aussi deux sortes d'éléments en ligne : « remplacés » et « non remplacés ».

- Seuls les éléments remplacés peuvent être redimensionnés. Ils acceptent des attributs de dimensions tels que : `height`, `width`.
Il s'agit entre autres des éléments ``, `<input>`, `<textarea>`, `<select>` et `<object>`.
- Les autres n'ont pas de dimension, et n'occupent que la place nécessaire à leur contenu. C'est le cas des éléments ``, ``, `<a>`, ``, etc.

2- Les principaux éléments en ligne

Balise	Commentaires	Exemple d'utilisation
<code><a></code>	Désigne un lien hypertexte. Elle s'accompagne de l'attribut <code>href</code> , qui renferme la cible du lien (son contenu représentant le texte à cliquer pour activer le lien).	<code>allez en page 2</code>
<code></code>	Met en emphase une portion de texte. Quand la police utilisée est droite, la plupart des navigateurs graphiques la traduisent comme une mise en italique.	<code><p>Voici un texte important.</p></code>
<code></code>	Inclut une image dans le document. Cette balise s'accompagne des attributs <code>alt</code> (texte alternatif pour malvoyants ou navigateurs en mode texte) et <code>src</code> (qui indique le chemin vers l'image). Le format de fichier de l'image, en théorie libre, est souvent limité à ce que comprennent les navigateurs – c'est-à-dire GIF, JPEG, et PNG. Remarquons que <code></code> est une balise sans contenu (« auto-fermante », ou encore « balise vide »). Ceci se traduit par son <code>/></code> final.	<code></code>

Balise	Commentaires	Exemple d'utilisation
<q>	Balise utilisée pour les citations courtes, en ligne. On utilisera la balise bloc <blockquote> pour des citations plus longues.	<p>Comme le dit toujours ma grand-mère : <q>il ne faut pas vendre la peau de l'ours avant de l'avoir tué </q>.</p>
	Conteneur en ligne générique, dépourvu d'un sens précis mais qui peut servir à regrouper d'autres éléments au fil du texte. Son équivalent est l'élément <div>.	un texte
	Indique un renforcement, généralement représenté en gras dans les navigateurs graphiques.	<p>Comme le dit toujours ma grand-mère : <q>il ne faut surtout pas vendre la peau de l'ours avant de l'avoir tué</q>.</p>

Exercice pratique : TP 2 BALISES DE BLOC

Utiliser la page web du TP précédent et effectuer les modifications afin d'obtenir la page ci-dessous :

Paroles d'or du père de la Nation ivoirienne **Président [Houphouët Boigny](#)**:

- “**La paix** n'est pas un vain mot. C'est un *comportement*. Alors faisons en sorte de graver la paix dans chacun de nos actions.”
- “Le vrai bonheur on ne l'aperçoit que lorsqu'on l'a perdu. Faisons en sorte de jamais la perdre.”

Les fondamentaux du XHTML et du CSS

Les balises de bloc
Toutes les balises permettant de réaliser la structure du document
Il s'agit également de choisir la bonne imbrication des balises

Tableau de quelques éléments XHTML

Balise	Commentaires
body	Corps du document
head	Entête du document

Figure 4 : Aperçu TP 2 Balises de bloc

Evaluation N°1

CHAPITRE IV – LES BALISES DE TABLEAU

Un tableau est composé de lignes et de colonnes. L'intersection entre une ligne et une colonne forme une cellule. La balise permettant de créer un tableau est `<table>`.

Ainsi, on peut créer des tableaux simples et des tableaux plus complexes.

I- TABLEAU SIMPLE

Syntaxe d'un tableau de 2 lignes et 2 colonnes:

```
<table>
  <tr>
    <td>cellule 1</td>
    <td> cellule 2</td>
  </tr>
  <tr>
    <td> cellule 3</td>
    <td> cellule 4</td>
  </tr>
</table>
```

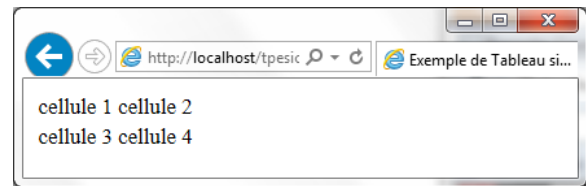


Figure 5 - Exemple tableau simple

L'élément `<table>` `</table>` marque le début et la fin du tableau.

L'élément `<tr>` `</tr>` désigne le début et la fin d'une ligne.

L'élément `<td>` `</td>` indique le début et la fin d'une cellule.

II- LA LIGNE D'ENTETE

Pour créer une entête de tableau, il suffit d'utiliser l'élément `<th>` `</th>` à la place de l'élément `<td>` `</td>` dans la première ligne du tableau. En général, le contenu de l'élément `<th>` `</th>` apparaît en gras. On obtient la syntaxe suivante :

```
<table>
<tr>
  <th>Entête 1</th>
  <th> Entête 2</th>
</tr>
<tr>
  <td>cellule 1</td>
  <td> cellule 2</td>
</tr>
<tr>
  <td> cellule 3</td>
  <td> cellule 4</td>
</tr>
</table>
```

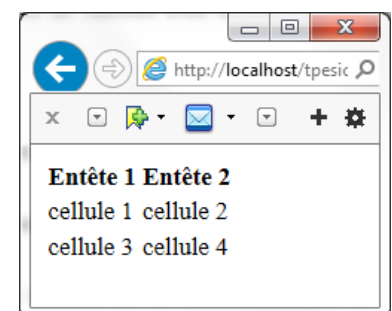


Figure 6- Tableau avec entête

III- LE TITRE DU TABLEAU

On utilise l'élément `<caption></caption>` au début du tableau avant le code de l'entête.

Exercice d'application

Ecrire le code HTML du tableau ci-dessous :

```
<table>
  <caption>LISTE DES CLIENTS</caption>
  <tr>
    <th>NOM ET PRENOMS</th>
    <th>VILLE</th>
    <th>CONTACT</th>
  </tr>
  <tr>
    <td>KONE MOHAMED</td>
    <td>BOUAKE</td>
    <td>05 45 76 23</td>
  </tr>
  <tr>
    <td>KOUAKOU JOEL</td>
    <td>ABIDJAN</td>
    <td>09 47 28 47</td>
  </tr>
  <tr>
    <td>AGOULE BENOIT</td>
    <td>DIVO</td>
    <td>04 57 65 48</td>
  </tr>
</table>
```

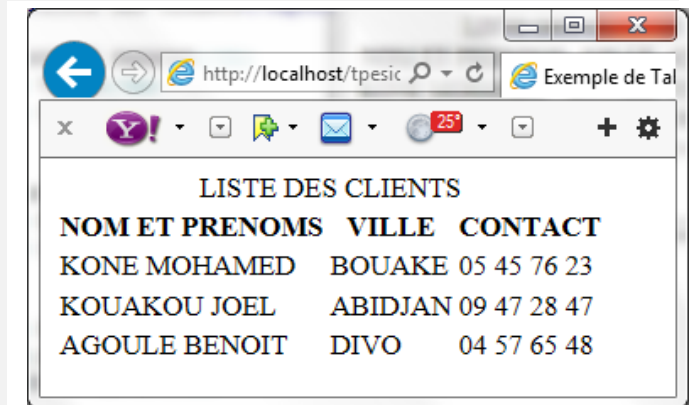


Figure 7-Aperçu de la Liste des clients sous IE 9

IV- LES TABLEAUX STRUCTURES

Dans le cas de longs tableaux, il importe de le diviser en 3 grandes parties :

- L'entête définit par les balises `<thead> </thead>`
- Le corps du tableau définit par les balises `<tbody></tbody>`
- Le pied de tableau définit par les balises `<tfoot></tfoot>`

Toutefois, il est conseillé d'utiliser l'ordre suivant lors de la création du tableau :

1. `<thead>` le haut du tableau (l'entête)
2. `<tfoot>` le bas du tableau (le pied de page)
3. `<tbody>` le corps du tableau

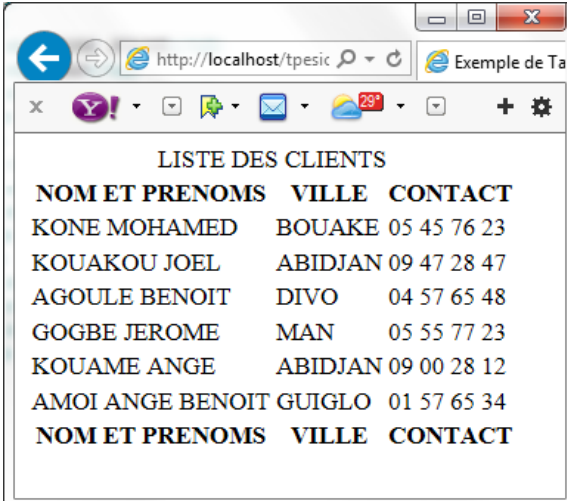
En reprenant l'exemple précédent, on pourra écrire une longue liste de clients avec le code suivant :

```
<table>
  <caption>LISTE DES CLIENTS</caption>
  <thead>
    <tr>
      <th>NOM ET PRENOMS</th>
      <th>VILLE</th>
      <th>CONTACT</th>
```

```

</tr>
</thead>
<tfoot>
<tr>
<th>NOM ET PRENOMS</th>
<th>VILLE</th>
<th>CONTACT</th>
</tr>
</tfoot>
<tbody>
<tr>
<td>KONE MOHAMED</td>
<td>BOUAKE</td>
<td>05 45 76 23</td>
</tr>
<tr>
<td>KOUAKOU JOEL</td>
<td>ABIDJAN</td>
<td>09 47 28 47</td>
</tr>
<tr>
<td>AGOULE BENOIT</td>
<td>DIVO</td>
<td>04 57 65 48</td>
</tr>
<tr>
<td>GOGBE JEROME</td>
<td>MAN</td>
<td>05 55 77 23</td>
</tr>
<tr>
<td>KOUAME ANGE</td>
<td>ABIDJAN</td>
<td>09 00 28 12</td>
</tr>
<tr>
<td>AMOI ANGE BENOIT</td>
<td>GUIGLO</td>
<td>01 57 65 34</td>
</tr>
</tbody>
</table>

```



LISTE DES CLIENTS		
NOM ET PRENOMS	VILLE	CONTACT
KONE MOHAMED	BOUAKE	05 45 76 23
KOUAKOU JOEL	ABIDJAN	09 47 28 47
AGOULE BENOIT	DIVO	04 57 65 48
GOGBE JEROME	MAN	05 55 77 23
KOUAME ANGE	ABIDJAN	09 00 28 12
AMOI ANGE BENOIT	GUIGLO	01 57 65 34
NOM ET PRENOMS	VILLE	CONTACT

Figure 8-Exemple de Tableau structuré

V- FUSION DE LIGNES OU DE COLONNES

1. Fusion de lignes

Pour fusionner des lignes, on utilise l'attribut `rowspan=x` dans la balise `<td>` de la cellule fusionnée ; où `x` représente le nombre de lignes à fusionner verticalement.

Par exemple pour réaliser le tableau ci-dessous :

1	2
3	4
	5

On utilisera le code suivant :

```
<table>
  <tr>
    <td>1</td>
    <td>2</td>
  </tr>
  <tr>
    <td rowspan="2">3</td>
    <td>4</td>
  </tr>
  <tr>
    <td>5</td>
  </tr>
</table>
```

2. Fusion de colonnes

Pour fusionner plusieurs colonnes, on dispose de l'attribut `colspan=x` où `x` est le nombre de colonnes à fusionner horizontalement.

Par exemple, pour réaliser le tableau ci-dessous :

1	
3	4
	5

On écrira le code suivant :

```
<table>
  <tr>
    <td colspan="2">1</td>
  </tr>
  <tr>
    <td>2</td>
    <td>3</td>
  </tr>
  <tr>
    <td>4</td>
    <td>5</td>
  </tr>
</table>
```

CHAPITRE V – LES BALISES D'IMAGE ET D'IMAGES-MAPS

I- LES DIFFERENTS FORMATS D'IMAGES

Les formats d'images les plus répandus actuellement sont :

- Le format GIF : utilisé pour les petits éléments graphiques, les icônes, les petites images animées avec un seul niveau de transparence.
- Le format JPEG : utilisé pour les photographies. Il ne supporte pas la transparence
- Le format PNG : utilisé pour les images de haute qualité avec plusieurs niveaux de transparence.

II- LES BALISES D'IMAGES

Pour afficher une image dans une page HTML, on utilise la balise . Elle possède plusieurs attributs dont les plus utilisés sont : src, alt et title.

Syntaxe :

```

```

- **L'attribut src** : chemin d'accès du fichier image. Ce chemin peut être relatif ou absolu selon les cas.
 - **Cas de chemin relatif** : le chemin fait référence à l'emplacement courant du fichier image.
Exemple : `src="images/photo.jpg"`
 - **Cas de chemin absolu** : le chemin fait référence à la racine du site web. Il est utilisé généralement pour faire référence à une image située sur un autre site web.
Exemple : `src="http://www.galeriedumonde.com/photos/montnimba.png"`
- **L'attribut alt** : texte alternatif qui s'affiche lorsque l'attribut **src** n'est pas défini ou lorsque l'image n'arrive pas à s'afficher pour une raison quelconque.
- **L'attribut title** : texte d'infobulle qui apparaît lorsqu'on pointe la souris sur l'image.

III- LES BALISES D'IMAGES-MAPS ET SES ATTRIBUTS

La balise <map> permet de créer des images réactives. L'image réactive contient plusieurs zones cliquables amenant chacune vers un lien déterminé.

La balise <map> contient un attribut **name** permettant de définir le nom du **map**.

Pour définir les zones cliquables, on utilise la balise <area> dont les principaux attributs sont :

1. **shape** : qui définit la forme de la zone. Ses valeurs possibles sont **rect** (rectangle), **circle** (cercle), **poly** (polygone),
2. **coords** : qui indique les coordonnées en pixels des points de la zone en fonction de la forme. Ainsi, nous aurons :
 - Pour un rectangle, les coordonnées (abscisse, ordonnée) du coin supérieur gauche et celles du coin inférieur droit ;
 - Pour un polygone, les coordonnées de chaque sommet ;
 - Pour un cercle, les coordonnées du centre et le rayon ;

3. **href** : qui indique l'URL de la page de destination du lien

SYNTAXE GENERALE

Insertion de l'image

```
<img src = 'chemin d'accès image' usemap= "#nom du map">
```

Définition de la structure du map

```
<map name="nom du map">
```

```
  <area shape="..." coords="..." href="#..." />
```

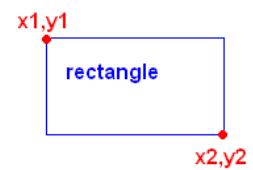
```
  .....
```

```
  <area shape="..." coords="..." href="#..." />
```

```
</map>
```

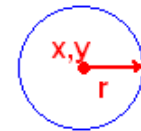
Cas 1 : Zone rectangulaire :

`<area shape="rect" coords="x1,y1,x2,y2" href="#adresse de destination"/>` où (x1,y1) et (x2,y2) représentent respectivement les points sur l'image ci-contre.



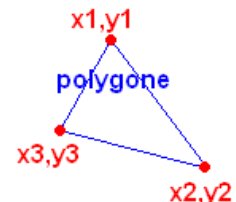
Cas 2 : Zone réactive circulaire :

`<area shape="circle" coords="x,y,r" href="#adresse de destination"/>` où (x,y) représente les coordonnées du centre du cercle et r le rayon du cercle.



Cas 3 : Zone polygonale

`<area shape="rect" coords="x1, y1, x2, y2, x3, y3" href="#adresse de destination"/>` où (x1,y1), (x2,y2), (x3,y3) représente les coordonnées des différents points du polygone dans l'ordre.



NB :

- Pour obtenir les coordonnées d'un point sur une image, il suffit d'ouvrir le fichier image avec les logiciels **PAINT** ou encore **Photoshop**. Passer la souris sur le point et regarder en bas sur la barre de statut.
- Les balises ``, `<map>` sont des balises en ligne. De ce fait elles doivent toujours être utilisées dans une balise de bloc telle que `<p></p>` ou `<div></div>`.

EXERCICE D'APPLICATION N°1

Soit l'image *casemappe.jpg* ci-contre :

Créez deux zones réactives sur cette image permettant d'indiquer le toit et le mur de la maison.

Pour la première zone réactive, nous retiendrons un polygone dont les points représentent la toiture.

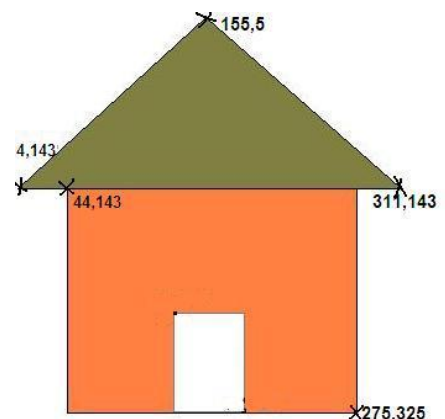


Figure 9 : Image Case mappée

Pour la seconde zone réactive, nous représenterons le mur de la maison en forme rectangulaire.
Ecrire le code HTML permettant d'insérer l'image et de créer les 2 zones réactives.

Proposition de solution

```

<map name="case">
  <area shape="poly" href="toiture.html" title="Toiture" coords="4,143,155,5,311,143"/>
  <area shape="rect" href="mur.html" title="Le mur" coords="44,143,275,325"/>
</map>
```

Dans notre cas le clic sur les zones n'auront pas d'effet puisque les valeurs de **href** n'ont pas été définies.

EXERCICE D'APPLICATION N°2

Soit l'image relative à la photo d'ensemble d'étudiants ci-après. L'on veut créer des images réactives sur cette photo, de sorte à pouvoir ouvrir une page concernant chaque étudiant en cliquant sur sa tête.

Nom du fichier image : **etudiant.jpg**

URL de l'image : **images/etudiant.jpg**

Coordonnées des centres des cercles d'images réactives :

A(58,39,39)

B(123,41,30)

C(167,48,25)

D(228,46,46)

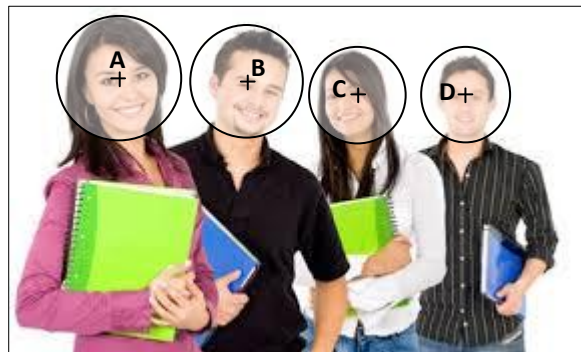


Figure 10 : Exercice Zones réactives

Travail à faire :

Créer le code HTML permettant d'insérer l'image etudiant.jpg et de créer les zones réactives circulaires sur le visage de chaque étudiant avec les coordonnées ci-dessous.

Proposition de solution :

```
<p>
  
  <map name="mapetudiant">
    <area title="Etudiante 1" href="fille1.html" shape="circle" coords="58,39,39"/>
    <area title="Etudiant 2" href="garcon1.html" shape="circle" coords="123,41,30"/>
    <area title="Etudiante 3" href="fille2.html" shape="circle" coords="167,48,25"/>
    <area title="Etudiant 4" href="garcon2.html" shape="circle" coords="228,46,46"/>
  </map>
</p>
```

CHAPITRE VI – LES BALISES DE FORMULAIRE

Les formulaires permettent d'accroître l'interactivité de l'utilisateur avec le serveur web. En effet, ils servent à saisir un texte, choisir une option, valider une action, etc.

Les usages des formulaires sur les sites web sont diverses ; entre autres, on peut citer : effectuer des inscriptions en ligne, créer un compte utilisateur, modifier des profils, envoyer des commentaires, etc.

Les données envoyées par le biais d'un formulaire sont :

1. Soit directement affichées sur une page pour être visible par les autres visiteurs,
2. Enregistrées dans un fichier,
3. Envoyées par email à un destinataire
4. Ou encore, enregistrées dans une base de données

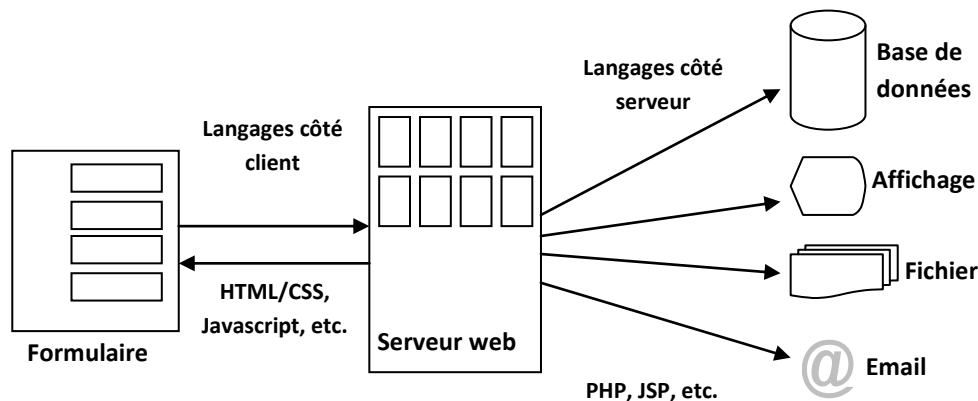


Figure 11 : Architecture d'un site web

Comme vous pouvez le constater sur le schéma, l'échange entre le serveur web et l'utilisateur se fait avec les langages côté client tels que HTML/CSS, Javascript. Alors que les transactions entre le serveur web et les serveurs de fichiers, de bases de données ou de messagerie se font avec les langages côté serveur tels que PHP, JSP, DOTNET, etc.

Notre cours s'intéressera particulièrement aux manipulations côté client du formulaire.

Le côté serveur du formulaire sera vu en 2^{ème} année de BTS avec l'étude du langage PHP.

I. LA BALISE <FORM>

La balise principale de création de formulaire est <form></form>. Il indique le début et la fin d'un formulaire.

Syntaxe :

```
<form>
  <p>Texte à l'intérieur du formulaire</p>
</form>
```

NB : il faut absolument utiliser des balises en bloc à l'intérieur du formulaire pour faire figurer du texte

A l'intérieur de la balise <form>, on trouve les attributs suivants :

1. **id** : définit le nom qui identifie le formulaire. Il ne doit pas contenir d'espace.
2. **Method** : indique le mode d'envoi des données au serveur. Il existe deux méthodes d'envoi des données :
 - **Method="get"** : cette méthode permet d'envoyer des données de petite taille (au plus 255 caractères). Les données sont envoyées par l'adresse URL de la page et sont visibles dans la barre d'adresse du navigateur.
 - **Method="post"** : c'est la méthode qui est généralement utilisée car elle permet d'envoyer une grande quantité de données sans qu'elles soient visibles dans la barre d'adresse.

3. **Action** : définit l'adresse de la page ou le script (programme) qui va traiter les informations après la validation du formulaire. Les données peuvent être envoyées par email à une adresse. Dans ce cas, on utilise le mot clé "**mailto:adresse_email_destinataire**". Pour utiliser **mailto**, il faut que la valeur de l'attribut **method** soit **post**.
4. **Enctype** : définit la manière dont les données du formulaire seront encodées lors de l'envoi au serveur.
NB : Cet attribut ne peut être utilisé que lorsque la valeur de l'attribut **method** est "**post**".

Valeurs de l'attribut enctype :

Valeur	Description
application/x-www-form-urlencoded	Par défaut. Tous les caractères sont encodés avant l'envoi (les espaces sont convertis en symboles '+' et les caractères spéciaux en valeurs hexadécimales du code ASCII).
multipart/form-data	Aucun caractère n'est encodé. Cette valeur est utilisée dans les cas où le formulaire contient un champ d'envoi de fichier au serveur.
text/plain ou text/html	Elles sont utilisées dans les cas d'envoi des données par email. Elles indiquent si les données seront envoyées sous forme de texte ou de code XHTML

Syntaxe :

```
<form id="form1" action=" valider.php" method=" valeur " enctype=" valeur ">
  <! – Contenu -->
</form>
```

II. LA BALISE CHAMP <INPUT> ET SES VARIANTES

En général, pour créer un champ dans un formulaire, on utilise la balise **<input />** et les principaux attributs de cette balise sont :

Attribut	Fonction	Exemple
type	Définir le type du champ	type="text"
Name ou id	Identifier le champ dans le formulaire afin de récupérer son contenu	id="matricule"
Size	Fixer la longueur de la zone de saisie en nombre de caractères	size ="10"
Maxlength	Limiter le nombre de caractères à saisir dans un champ de saisie	maxlength ="15"
Value	Définir la valeur par défaut du champ	value="ET-0405-00000"
Readonly	Indiquer si le champ est en lecture seule. Dans ce cas, la valeur de ce champ ne peut être modifiée.	readonly="readonly"
Placeholder	Donner une indication sur le contenu du champ. (HTML 5 seulement)	placeholder="ET-0405-00015"

Syntaxe :

```
<input type="valeur" name="nom_champ" size="N" maxlength="N" value="valeur_par_defaut"/>
```

Remarque : On peut définir un libellé pour le champ avec la balise **<label></label>**.

La balise **<input />** peut être utilisée à plusieurs fins dans un formulaire. Parmi les multiples usages, nous avons : les champs de saisie mono-ligne, les mots de passe, les champs d'envoi de fichiers et divers autres champs apparus surtout avec l'arrivée de HTML 5.

1) LE CHAMP DE SAISIE MONO-LIGNE

Dans une zone de saisie mono-ligne, le texte s'écrit sur une seule ligne sans aller à la ligne. Pour créer ce type de champ, on définit la valeur de l'attribut **type** à "text".

Exemple:

```
<label>Nom :</label><input type="text" name="nomuser" size="15"
maxlength="40" value="GOGBE" />
```

Nom :

2) LE CHAMP DE SAISIE DE MOT DE PASSE

Pour saisir un mot de passe dans un champ de saisie, on donne la valeur **password** à l'attribut **type**. Ainsi, on aura par exemple :

```
<form action="validconnexion.php" method="post" name="fconnexion">
  <label>Login </label><input name="login" type="text" placeholder="Taper votre login" size="30"/>
  <label>Mot de passe </label><input name="pass" type="password" size="20"/>
</form>
```

Résultat :

Figure 12 : Champs de saisie mono ligne et de mot de passe

NB : pour organiser le formulaire en regroupant les champs ayant trait au même sujet, on utilise les balises **<fieldset></fieldset>** pour encadrer les éléments à regrouper.

3) LES CHAMPS D'ENVOI DE FICHIER

Pour créer un champ d'envoi de fichier on définit l'attribut **type** de la balise **<input>** à "file". Dans ce cas, un bouton **Parcourir** ou **Browse** apparaît à droite de la zone.

Pour utiliser un champ d'envoi de fichier dans un formulaire, la balise **<form>** doit avoir les attributs **method** et **enctype** avec les valeurs respectives **post** et **multipart/form-data**.

Exemple : `<form action="script1.php" method="post" enctype="multipart/form-data">`

Pour restreindre les fichiers à envoyer à des types précis, on utilise l'attribut **accept** dont la valeur est la liste des types MIME acceptés.

Exemple : `<input type="file" name="photo" accept="image/*" />`
Ce champ accepte l'envoi de tous les types d'images.

Pour limiter la taille des fichiers à envoyer, on insère un **champ caché** dont l'attribut **name** sera **MAX_FILE_SIZE** et l'attribut **value** sera la **taille maximale du fichier en octets**.

Exemple complet :

Ecrire le code HTML du formulaire ci-dessous :

Figure 13 : Exemple de champ d'envoi de fichier

```
<form name="formcv" method="post" enctype="multipart/form-data" action="traitement.php">
<fieldset><legend><strong>Transfert de CV</strong></legend>
  <label>Votre CV (au format PDF ou Word - Taille max. 10 Mo): </label>
```

```

<input name="cvfile" type="file" size="20" accept="application/pdf,application/msword"/><br />
<!--Accepte les fichiers au format PDF et Word -->
<input name="MAX_FILE_SIZE" value="10000000" type="hidden"/><!--Taille maxi 10 Mo -->
</fieldset>
</form>

```

4) LES BOUTONS RADIO

Pour créer une liste d'options à choix unique, on utilise la balise **<input />** en définissant l'attribut **type** à la valeur **"radio"**.

L'attribut **name** des options appartenant à un même groupe doit avoir la même valeur pour chaque option.

L'attribut **value** définit la valeur qui sera envoyée au serveur si l'utilisateur clique sur l'option.

Pour définir par défaut le choix d'une option dans un groupe d'options, on donne la valeur **"checked"** à l'attribut **checked** (checked="checked").

Syntaxe :

```

<input type="radio" name="nomgroupe" value="valeur1" />
-----
-----
<input type="radio" name="nomgroupe" value="valeurN" />

```

Exemple:

```

<fieldset><legend>Qualité du produit</legend>
  <label><input type="radio" name="info" value="Excellent" />Excellent</label><br />
  <label><input type="radio" name="info" value="Bon" checked="checked" />Bon</label><br />
  <label><input type="radio" name="info" value="Moyen" />Moyen</label><br />
  <label><input type="radio" name="info" value="Mauvais" />Mauvais</label><br />
  <label><input type="radio" name="info" value="Mauvais" />Indifférent</label>
</fieldset>

```

Dans l'exemple ci-contre l'option Bon est choisie par défaut.

The image shows a web form with a title "Qualité du produit" and a legend. Below the legend, there are five radio button options: "Excellent", "Bon", "Moyen", "Mauvais", and "Indifférent". The "Bon" option is selected, indicated by a filled radio button.

Figure 14 : Exemple de boutons radio

5) LES BOUTONS DE CHOIX MULTIPLES OU CASES A COCHER

Ils permettent de faire plusieurs choix dans un même groupe de cases à cocher. Ils ont les mêmes arguments que les boutons radio.

Mais l'attribut **type** doit prendre la valeur **"checkbox"**. Le nom du champ doit être un tableau lorsqu'il s'agit d'un groupe de case à cocher.

Syntaxe :

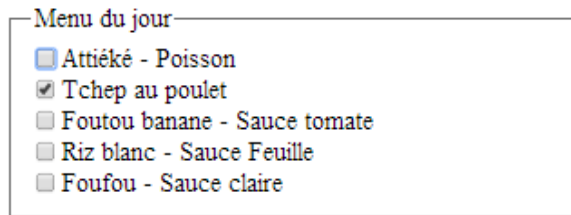
```

<input type="checkbox" name="nom1" value="valeur1" />
-----
-----
<input type="checkbox" name="nomN" value="valeurN" />

```

Exemple :

Soit une société de distribution de repas qui désire concevoir un formulaire dans le but de permettre aux clients de passer leur commande dans le menu du jour. Dans cet exemple le plat Tchep au poulet est choisi par défaut. Le formulaire doit se présenter comme suit :



Menu du jour

- Attiéké - Poisson
- Tchep au poulet
- Foutou banane - Sauce tomate
- Riz blanc - Sauce Feuille
- Foufou - Sauce claire

Figure 15 : Exemple de cases à cocher

Aperçu du code HTML

```
<fieldset><legend>Menu du jour</legend>
  <input type="checkbox" name="repas1" value="Attiekepoisson" /><label>Attiéké - Poisson</label><br />
  <input type="checkbox" name="repas2" value="Tcheppoulet" checked="checked" /><label>Tchep au
  poulet</label><br />
  <input type="checkbox" name="repas3" value="foutoutomate" /><label>Foutou banane - Sauce
  tomate</label><br />
  <input type="checkbox" name="repas4" value="rizfeuille" /><label>Riz blanc - Sauce Feuille</label><br />
  <input type="checkbox" name="repas5" value="foufou" /><label>Foufou - Sauce claire</label>
</fieldset>
```

6) LES BOUTONS DE VALIDATION ET DE REINITIALISATION DE FORMULAIRE

On utilise les boutons de validation pour envoyer les données du formulaire au serveur. L'appui sur ce bouton exécute, généralement, le script inscrit dans la valeur de l'attribut **action** de la balise **<form>**.

Syntaxe :

```
<input type="submit" name="nom bouton" value="texte du bouton" />
```

Le bouton de réinitialisation permet de rafraîchir le formulaire afin de retrouver les valeurs de départ.

Syntaxe :

```
<input type="reset" name="nom bouton" value="texte du bouton" />
```

Exemple

```
<input name="valider" type="submit" width="15" value="Envoyer" />
<input name="valider" type="reset" width="15" value="Annuler" />
```

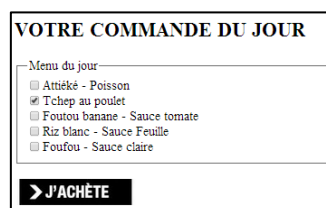
Pour utiliser une image sous forme de bouton de validation, on utilise la syntaxe suivante :

Syntaxe:

```
<input name="nom" type="image" src="url image" width="x" height="y"
alt="texte alternatif" />
```

Exemple :

```
<input name="valider" type="image" src="images/zendf.jpg" width="103"
height="54" alt="Envoyer" />
```



VOTRE COMMANDE DU JOUR


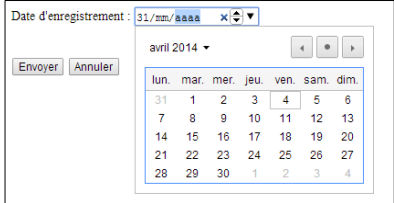

Menu du jour

- Attiéké - Poisson
- Tchep au poulet
- Foutou banane - Sauce tomate
- Riz blanc - Sauce Feuille
- Foufou - Sauce claire

J'ACHÈTE

Figure 16 : Exemple de bouton image

Il existe, avec l'arrivée de HTML 5, de nouveaux types de champs de saisie permettant de rendre plus convivial le contenu d'un formulaire. Pour les obtenir, il suffit de définir les différentes valeurs de l'attribut **type** de l'élément **<input />** comme suite :

Valeur	Description	Aperçu du résultat sur Google Chrome
color	Définit un sélecteur de couleur	
date	Définit un champ date au format (an, mois, jour)	
datetime	Définit un champ de type date et heure au format (an, mois, jour, heure, minute, seconde)	
email	Définit un champ devant contenir une adresse email	
month	Définit un champ de type mois	
number	Définit un champ de saisie numérique	
range	Définit un curseur permettant de choisir un nombre dans un intervalle.	
search	Définit un champ de recherche	
tel	Définit un champ de saisie de numéro de téléphone	
time	Définit un champ de saisie de l'heure	
url	Définit un champ de saisie d'adresse URL	
week	Définit un champ de saisie de semaine et d'année.	

III- LES AUTRES ELEMENTS DE FORMULAIRE

1) LES LISTES

Il s'agit de listes déroulantes ou de listes simples.

Pour créer une liste déroulante on utilise les balises **<select></select>** avec à l'intérieur des balises **<option></option>** pour chaque élément de la liste.

Les attributs de **<option>** sont :

- **value** : définit la valeur envoyée au serveur par le formulaire
- **selected** : pour définir si l'option est choisie par défaut

Syntaxe :

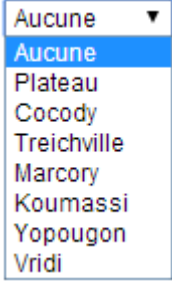
```
<select name="nomliste">
  <option value="valeur1"></option>
  -----
  <option value="valeurN"></option>
</select>
```

Pour définir une liste simple, il suffit d'ajouter l'attribut **size** pour indiquer le nombre d'éléments de la liste.

Pour permettre que plusieurs éléments soient sélectionnés en même temps, on définit l'attribut **multiple** dans la balise `<select>`.


Exemple :

Ecrire le code HTML des listes suivantes :

Lieu de livraison : 

```
<label>Lieu de livraison :</label> <select name="lieu">
  <option value="0">Aucune</option>
  <option value="plateau">Plateau</option>
  <option value="cocody">Cocody</option>
  <option value="treich">Treichville</option>
  <option value="marcory">Marcory</option>
  <option value="koum">Koumassi</option>
  <option value="yop">Yopougon</option>
  <option value="vridi">Vridi</option>
</select>
```

Figure 17 : Exemple de liste déroulante

Lieu de livraison : 

```
<select name="lieu" size="8" multiple="multiple">
  <option value="0">Aucune</option>
  <option value="plateau">Plateau</option>
  <option value="cocody" selected="selected">Cocody</option>
  <option value="treich">Treichville</option>
  <option value="marcory">Marcory</option>
  <option value="koum">Koumassi</option>
  <option value="yop">Yopougon</option>
  <option value="vridi">Vridi</option>
</select>
```

Figure 18 : Exemple de liste

2) CHAMP DE SAISIE MULTI-LIGNE

Ce champ permet de saisir un long texte sur plusieurs lignes (exemples : commentaires, suggestions, descriptions, etc.). Pour insérer ce champ, on utilise la balise `<textarea></textarea>`.

Les attributs possibles de cette balise sont :

1. **Name** : définit le nom de l'élément
2. **Cols** : définit le nombre de caractères visibles sur la largeur de la zone de saisie.
3. **Rows** : définit le nombre de lignes visibles sur la hauteur de la zone de saisie.
4. **Readonly** : indique que la valeur par défaut définie ne peut être modifiée.

Pour définir une valeur par défaut pour la zone de saisie, il suffit de placer le texte entre les balise `<textarea>` et `</textarea>`.

Syntaxe : `<textarea name="nom" cols="N" rows="N" cols="N">Texte par défaut</textarea>`

Exemple :

```
<textarea name="comment" cols="50" rows="5" cols="50">Entrez un commentaire sur notre service ici</textarea>
```

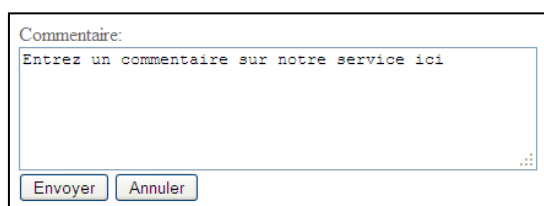


Figure 19 : Exemple de champ de saisie multi-ligne

Exemple complet 1:

La société de livraison de repas à domicile BON APPETIT veut concevoir un formulaire permettant à leurs clients potentiels de s'inscrire en ligne.

Ecrire le code HTML du formulaire ci-dessous :

The screenshot shows a registration form with the following sections:

- Informations de connexion:** Pseudo (text input), Mot de passe (password input), Retaper le mot de passe (password input).
- Informations de connexion (repeated):** Nom et prénoms (text input), Entreprise / Société (text input), Cellulaire / Téléphone (text input), Email (text input), Votre commune (dropdown menu), Adresse exacte (Quartier / Rue / Porte) (text area).
- Tarif souhaité:** Radio buttons for: Moins de 500 F, Entre 500 F et 1000 F, Entre 1000 F et 1500 F, 2000 F et plus, Indifférent.
- Période de livraison souhaitée:** Checkboxes for: Matin (Petit déjeuner), Midi (Déjeuner), Après-midi (Gouter), Soir (Diner).
- Autres précisions:** Text area.
- Buttons:** Valider, Annuler.

Figure 20 : Formulaire complet 1 - Cas du formulaire d'inscription

```
<form name="finscription" action="validinscription.php" method="post">
<h2>Formulaire d'inscription</h2>
  <fieldset><legend>Informations de connexion</legend>
    <label>Pseudo</label><br/> <input name="pseudo" type="text" size="50" maxlength="250" placeholder="Votre pseudo en un seule mot"><br/>
    <label>Mot de passe</label><br/> <input name="pass" type="password" size="50" maxlength="250" placeholder="Votre mot de passe en 8 caracteres
mini"><br/>
    <label>Retaper le mot de passe</label><br/> <input name="repass" type="password" size="50" maxlength="250" placeholder="Retaper le meme mot
de passe">
  </fieldset>
  <fieldset><legend>Identité du client</legend>
    <label>Nom et prénoms</label><br/> <input name="nom" type="text" size="50" maxlength="250" ><br/>
    <label>Entreprise / Société</label><br/> <input name="entreprise" type="text" size="50" ><br/>
    <label>Cellulaire / Téléphone</label><br/> <input name="telephone" type="tel" size="50"><br/>
    <label>Email</label><br/> <input name="email" type="email" size="50" ><br/>
    <label>Votre commune</label><br/>
    <select name="Commune">
      <option value="0" label="Choisir votre commune"></option>
      <option value="plateau" label="Plateau"></option>
      <option value="cocody" label="Cocody"></option>
      <option value="2plateaux" label="Il Plateaux"></option>
      <option value="treichville" label="Treichville"></option>
      <option value="yopougou" label="Yopougou"></option>
    </select><br />
    <label>Adresse exacte (Quartier / Rue / Porte) </label><br/>
    <textarea name="adresse" cols="50" rows="3"></textarea>
  </fieldset>
  <fieldset><legend>Tarif souhaité</legend>
    <label><input name="tarif" type="radio" value="500" >Moins de 500 F</label><br/>
    <label><input name="tarif" type="radio" value="1000">Entre 500 F et 1000 F</label><br/>
    <label><input name="tarif" type="radio" value="1500">Entre 1000 F et 1500 F</label><br/>
    <label><input name="tarif" type="radio" value="2000">2000 F et plus </label><br/>
    <label><input name="tarif" type="radio" value="0">Indifférent</label><br/>
  </fieldset>
  <fieldset><legend>Période de livraison souhaitée</legend>
    <label><input name="periode[]" type="checkbox" value="matin" >Matin (Petit déjeuner)</label><br/>
    <label><input name="periode[]" type="checkbox" value="midi">Midi (Déjeuner)</label><br/>
    <label><input name="periode[]" type="checkbox" value="apresmidi">Après-midi (Gouter)</label><br/>
    <label><input name="periode[]" type="checkbox" value="soir">Soir (Diner) </label><br/>
  </fieldset>
  <label>Autres précisions </label><br/>
  <textarea name="precision" cols="50" rows="3"></textarea><br/>
  <input name="valider" type="submit" value="Valider"><input name="Annuler" type="reset" value="Annuler">
</form>
```

Exemple complet 2:

La société BON APPETIT souhaite également mettre à la disposition de ces partenaires (restaurants, maquis) un formulaire permettant de saisir les informations sur les mets qu'ils proposent. Ecrire le code HTML du formulaire ci-dessous :

```
<form name="finscription" action="validinscription.php" method="post" enctype="multipart/form-data">
<h2>Saisie des recettes proposées</h2>
<fieldset><legend>Informations sur la recette</legend>
  <label>Nom de la recette</label><br/>
  <input name="nomrecette" type="text" size="50" maxlength="250"><br/>
  <label>Description de la recette</label><br/>
  <textarea name="description" rows="5" cols="50"></textarea><br/>
  <label>Prix du plat pour une personne</label>
  <input name="prix" type="number" min="0" max="10000" step="500" /></label><br/>
  <label>Capacité de production</label>
  <input name="capacite" type="number" min="0" max="100" /></label><br/>
  <label>Période de disponibilité</label><br/>
  <label><input name="periode[]" type="checkbox" value="matin" >Matin (Petit déjeuner)</label><br/>
  <label><input name="periode[]" type="checkbox" value="midi">Midi (Déjeuner)</label><br/>
  <label><input name="periode[]" type="checkbox" value="apresmidi">Après-midi (Gouter)</label><br/>
  <label><input name="periode[]" type="checkbox" value="soir">Soir (Diner) </label><br/>
</fieldset>
<fieldset><legend>Echantillons</legend>
  <label>Télécharger quelques images (Format JPG, PNG, GIF - Taille totale < 5 Mo) <br/>
  <input name="photo" type="file" multiple accept="image/gif,image/jpeg,image/x-png" ></label><br/>
  <input name="MAX_FILE_SIZE" type="hidden" value="5000000" /></label><br/>
</fieldset>
<br/>
  <input name="valider" type="submit" value="Valider"><input name="Annuler" type="reset"
value="Annuler">
</form>
```

Figure 21 : Formulaire complet 2 - Cas de saisie de recette proposée

CHAPITRE VI – LA BALISE D’HYPERLIEN

Un lien hypertexte (hyperlien) est un élément HTML qui renvoie le visiteur à une page ou à un emplacement de la page du même site ou d’un autre site web. Le lien hypertexte est délimité par la balise `<a>`.

Syntaxe générale :

```
<a href="url_page_destination" title="Texte_infobulle" target="cible ">Texte du lien</a>
```

L’attribut **href** permet de définir le chemin de la page de destination du lien.

L’attribut **title** permet de spécifier un texte d’infobulle qui apparaîtra quand le visiteur va pointer le texte du lien.

L’attribut **target** permet de préciser la fenêtre cible dans laquelle la fenêtre de destination va s’afficher.

Les valeurs possibles de l’attribut **target** sont :

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

Il peut avoir plusieurs situations d’utilisation des liens hypertextes en fonction de l’emplacement de la page cible ou de l’usage que l’on veut faire du fichier. Parmi elles, on peut citer :

- Lien vers un autre site web
- Lien vers une autre page de son propre site
- Lien vers une ancre
- Envoyer un email ou télécharger un fichier par le lien

I- LIEN VERS UN AUTRE SITE WEB

Il suffit de préciser dans l’attribut **href** l’**adresse URL absolue** du site web de destination.

Exemple :

Lien vers le site web `www.google.ci`:

```
<a href="http://www.google.ci">Accédez au Google </a>
```

Lien vers la page `inscription.html` du site `www.bonappetit.ci` :

```
<a href="http://www.bonappetit.ci/inscription.html"> Bon appetit </a>
```

Exemple

Lien vers la page `agenda.html` de notre site web :

```
<a href="agenda.html"> Bon appetit </a>
```

II- LIEN VERS UNE AUTRE PAGE DU MEME SITE WEB

Il peut y avoir deux situations dans ce cas :

1) **Les 2 pages sont dans le même dossier**

Dans ce cas, il faut spécifier le nom de la page dans l’attribut **href**.

Exemple : Lien vers la page `agenda.html` de notre site web :

```
<a href="agenda.html"> Bon appetit </a>
```

2) **Les 2 pages sont dans des dossiers différents**

Lorsque la page cible se trouve dans un sous-dossier du dossier courant, la valeur de l’attribut **href** s’écrira : **sous-dossier/pagecible**.

Exemple : soient deux fichiers index.html et inscription.html organisés selon l'arborescence ci-dessous :

```

monsite
|--index.html
|---Contenu
|---inscription.html
|---produit.html

```

Pour créer un lien de la page **index.html** vers la page **inscription.html**, on écrira :

```
<a href="contenu/inscription.html"> Inscription </a>
```

Lorsque la page cible se trouve dans le dossier parent du dossier courant, la valeur de l'attribut **href** s'écrira : **../pagecible**.

Exemple : soient deux fichiers index.html et inscription.html organisés selon l'arborescence précédente :

Pour créer un lien de la page **inscription.html** vers la page **index.html**, on écrira :

```
<a href="../index.html"> Inscription </a>
```

III- LIEN VERS UNE ANCRE

Une ancre est une sorte de point de repère que l'on place sur une page web pour faire référence à une portion de cette page. Elle est très souvent utilisée pour créer un lien permettant d'aller directement au bas ou en haut de la page.

Pour créer une ancre, on ajoute l'attribut **id** dans une balise qui servira de repère sur la page.

Syntaxe de création d'une ancre : `<nom_balise id="nom_ancre">contenu de la balise</nom_balise>`

Exemple : `<h2 id="nom_ancre">Texte du titre</h2>`

Pour faire référence à une ancre, il faut utiliser le signe dièse (#) suivi du nom de l'ancre.

Plusieurs cas peuvent se produire :

- **Cas d'une ancre interne**

Dans ce cas, le lien pointe vers un emplacement (ancre) situé sur la même page.

Syntaxe d'une ancre interne: `Texte du lien vers l'ancre`

Exemple : ` Aller directement à la fin de la page`

- **Cas d'une ancre située sur une page du même site**

Syntaxe : `Texte du lien vers l'ancre`

Exemple : `Cliquer pour aller directement au formulaire d'inscription`

- **Cas d'une ancre située sur une page d'un autre site**

Syntaxe : `Texte du lien vers l'ancre`

Exemple :

```
<a href="http://www.bonappetit.com/client.html#inscription">Inscrivez-vous sur le site bonappetit.com </a>
```

NB: pour créer un lien vers le début d'une page, on écrit : `Début de la page`

CHAPITRE VII – LES BALISES DE LISTE

Elles permettent de créer des listes, d'énumérer des éléments, de créer des sommaires, des menus, etc.

Il existe 04 différents types de listes:

- Les listes non ordonnées (à puces)
- Les listes ordonnées (numérotées)
- Les listes imbriquées
- Les listes de définitions

I- LES LISTES NON ORDONNES OU LISTES A PUCES

Elles sont définies par la balise `` et chaque item (élément) est défini par la balise ``.

Syntaxe :

```
<ul type="type_puce">
<li>Item 1</li>
    <li>Item 2</li>
...
    <li>Item n</li>
</ul>
```

Il existe plusieurs types de puces. Le type est précisé par l'attribut **type**. Les principaux utilisés sont :

- Disc
- Circle
- Square

Exemple 1:

```
<p>Liste des matières :</p>
<ul type="circle">
    <li>Algorithmique</li>
    <li>Langage de programmation</li>
    <li>Merise</li>
    <li>Développement Web</li>
</ul>
```

Exemple 2:

```
<p>Liste des matières :</p>
<ul type="square">
    <li>Algorithmique</li>
    <li>Langage de programmation</li>
    <li>Merise</li>
    <li>Développement Web</li>
</ul>
```

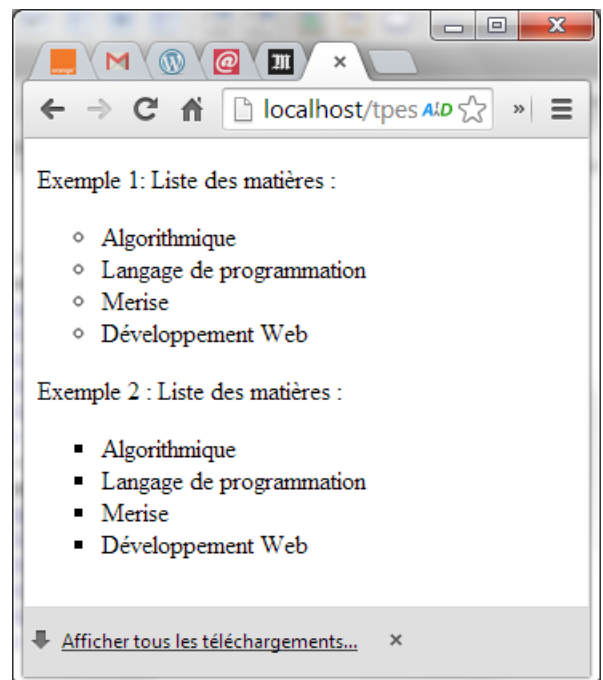


Figure 22 : Aperçu de listes à puces

II- LES LISTES ORDONNEES

Elles sont introduites par la balise `` et chaque élément par la balise ``.

L'attribut **start** permet de définir le début de la numérotation. Pour inverser la numérotation, on ajoute l'attribut **reversed** dans la balise ``.

Syntaxe :

```

<ol>
<li>Item 1</li>
    <li>Item 2</li>
...
    <li>Item n</li>
</ol>

```

Exemples

```

<!--Cas des listes numérotées-->
<p>Exemple 1: Le plan de l'exposé :</p>
<ol>
<li>Introduction</li>
<li>Les causes</li>
<li>Les conséquences</li>
<li>Les perspectives</li>
<li>Conclusion</li>
</ol>
<!--Cas de liste commençant pas à un numéro-->
<p>Exemple 2: Le plan de l'exposé :</p>
<ol start="2">
<li>Introduction</li>
<li>Les causes</li>
<li>Les conséquences</li>
<li>Les perspectives</li>
<li>Conclusion</li>
</ol>
<!--Cas de liste avec numérotation inversée-->
<p>Exemple 3 : Le plan de l'exposé :</p>
<ol reversed>
<li>Introduction</li>
<li>Les causes</li>
<li>Les conséquences</li>
<li>Les perspectives</li>
<li>Conclusion</li>
</ol>

```

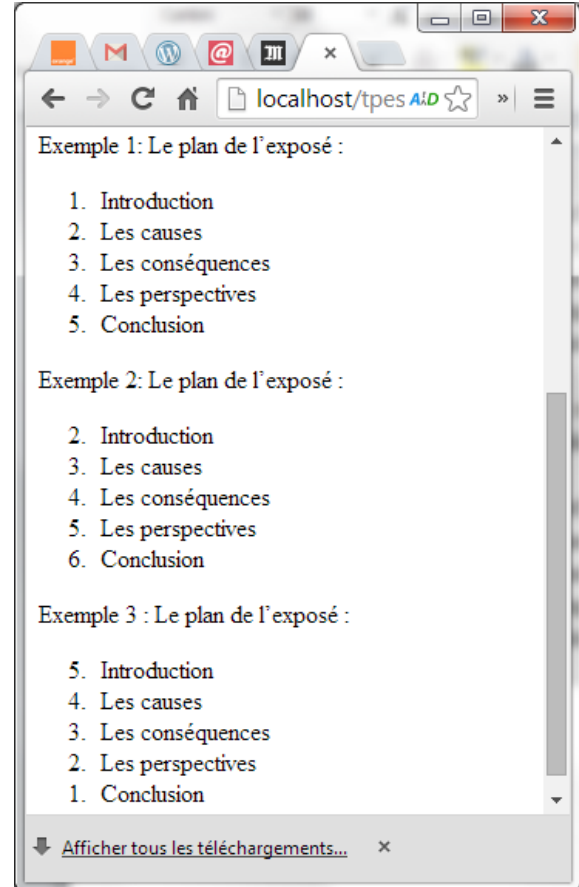


Figure 23 : Aperçu de listes numérotées avec plusieurs variantes

III- LES LISTES IMBRIQUEES

Il est possible de combiner une liste numérotée avec une liste à puces. On obtient une syntaxe comme suite :

```

<ol>
<li> item 1
    <ul>
    <li>Item 1</li>
        <li>Item 2</li>
...
        <li>Item n</li>
    </ul>
    </li>
<li>Item 2</li>
...
    <li>Item n</li>
</ol>

```

Exemple de liste imbriquée :

```

<!--Cas de listes imbriquées-->
<p>Exemple 3 : Le plan de l'exposé :</p>
<ol>
  <li>Introduction</li>
  <li>Les causes
    <ul>
      <li>Les causes lointaines</li>
      <li>Les causes immédiates</li>
      <li>Les causes internes</li>
    </ul>
  </li>
  <li>Les conséquences</li>
  <li>Les perspectives</li>
  <li>Conclusion</li>
</ol>

```

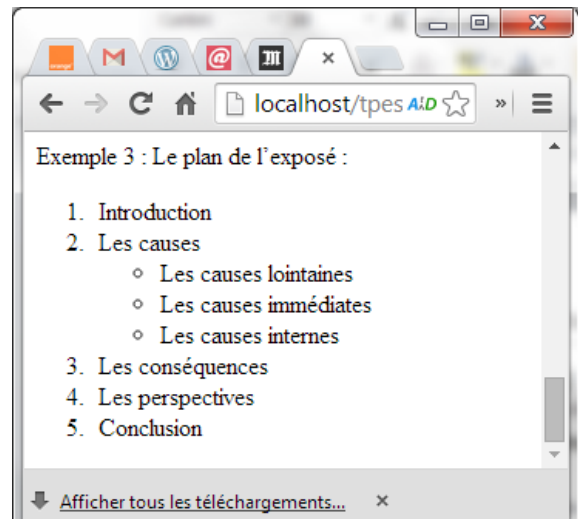


Figure 24: Aperçu de listes imbriquées

IV- LES LISTES DE DEFINITIONS

Ce type de liste permet d'associer un terme avec sa description. Elle est utile pour la création de glossaire ou de dictionnaire par exemple. On la définit en utilisant la balise **<dl>**. Ensuite, on indique chaque terme de la liste avec la balise **<dt>** et sa description avec la balise **<dd>**.

Syntaxe générale :

```

<dl>
  <dt>Terme 1</dt>
  <dd>Description terme 1</dd>
  <dt>Terme 2</dt>
  <dd>Description terme 2</dd>
  ...
  <dt>Terme N</dt>
  <dd>Description terme N</dd>
</dl>

```

Exemple :

```

<!--Cas des listes de définitions-->
<p>Quelques définitions de termes en Développement web</p>
<dl>
  <dt>HTML</dt>
  <dd>HyperText Markup Language</dd>
  <dt>URL</dt>
  <dd>Uniform Resource Location</dd>
  <dt>Internet</dt>
  <dd>Réseau mondial d'ordinateurs</dd>
  <dt>HTTP</dt>
  <dd>HyperText Transfert Protocol</dd>
</dl>

```

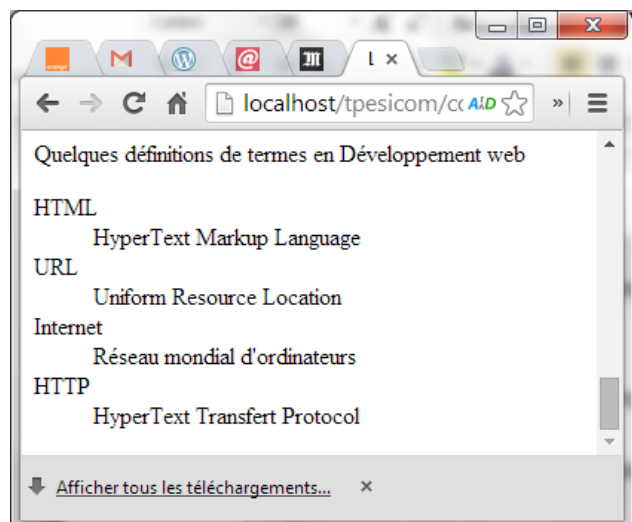


Figure 25: Cas de liste de définitions

CHAPITRE VIII – LES BALISES DE CONTENU EXECUTABLE

Les contenus exécutables ou scripts sont des petits programmes en Javascript ou VBscript exécutés côté client (par le navigateur) pour améliorer l'interactivité avec le visiteur.

Les scripts exécutables sont des fonctions qui s'exécutent généralement lors des événements (clics, survol, appui de touche, etc) liés aux objets tels que les boutons, les images, les liens, etc.

On distingue deux types de scripts :

- Les scripts externes
- Les scripts internes

I- LES SCRIPTS EXTERNES

Ce sont des fichiers externes à la page, avec extension .js pour les fichiers javascript ou .vbs pour les VBscript.

On utilise la balise <script> et l'attribut **src** pour inclure le fichier dans la page.

Syntaxe de la Balise script

```
<script type="text/javascript" src="chemindufichierjavascript.js"></script>
```

II- LES SCRIPTS INTERNES

Les scripts internes sont des bouts de code directement écrits entre les balises <script> et </script>. Ils sont directement inclus dans la page comme suit :

Balise <script>

```
<script type="text/javascript">
function maFonction()
{
  /* du code ici */
}
</script>
```

NB : Contrairement aux scripts externes, les scripts internes ne sont pas réutilisables dans les autres pages.

2^{ème} PARTIE : LES FEUILLES DE STYLE OU CSS

CHAPITRE IX – GENERALITES SUR LES FEUILLES DE STYLE

I. INTRODUCTION

Les feuilles de style en cascade ou Cascade Style Sheet (CSS) ont été adoptées par le W3C pour séparer le contenu des pages HTML et leur apparence. Elles régissent l'apparence des éléments HTML d'une page concernant la disposition, la couleur, les dimensions, etc.

Apparues depuis les années 1996, les versions successives de ce langage ont été :

- CSS 1 ;
- CSS 2.0 ;
- CSS 2.1 ;
- CSS 3, la version actuelle

Si les versions précédentes du HTML permettaient encore de nous passer de ce langage à un certain niveau, avec l'arrivée du HTML 5, le CSS s'impose à tout développeur web.

Elles facilitent ainsi la gestion de la charte graphique d'un site web. En effet, grâce aux feuilles de style, lorsque la charte graphique d'un site composé de plusieurs dizaines de pages web doit être changée, il suffit de modifier la définition des feuilles de style en un seul endroit pour changer l'apparence du site tout entier !

Les feuilles de style permettent entre autres :

- d'avoir une présentation homogène des pages sur tout le site.
- Une maintenance plus rapide du site web.
- une plus grande lisibilité du code HTML.
- le positionnement rigoureux des éléments.
- un chargement plus rapide des pages.

II. PRINCIPE GENERAL

On peut définir une feuille de style dans trois (3) endroits dans le code HTML de la page web :

- Directement à l'intérieur des balises avec l'attribut *style* (méthode pas recommandée) ;
- Dans l'entête **<head>** de la page HTML (peu recommandée) ;
- Dans un fichier externe **.css** (méthode la plus recommandée).

1) Dans l'entête **<head>** de la page HTML

On insère le code CSS dans la balise **<style>** entre les balises d'entête **<head>** de la page HTML.

Syntaxe :

```
<head>
  <meta charset="utf-8" />
  <style>
    Selecteur {
      propriété1: valeur;
      propriété2: valeur;
      ...
      propriétéN: valeur ;
    }
  </style>
</head>
```

Exemple : le code suivant définit une couleur à appliquer aux éléments des listes de la page :



Même si cette méthode fonctionne, elle devient pénible surtout lorsque vous devez définir ou modifier le même style pour plusieurs pages. C'est pourquoi, il est recommandé de recourir à la méthode suivante.

2) Dans un fichier externe .css

Cette méthode consiste à créer un fichier css externe à la page HTML (généralement dans un dossier css) et ensuite de lier ce fichier à la page HTML avec la balise <link>.

Application :

1^{ère} étape : créons un fichier **style.css** (par exemple) dans un dossier **CSS** (par exemple) que l'on aura créée auparavant. Ensuite, écrivons le code CSS de la feuille de style précédente.

2^{ème} étape : A l'intérieur des balises <head> de la page HTML, on insère la balise <link> comme suit :

<link rel="stylesheet" href="url_du_fichier_css">

par exemple <link rel="stylesheet" href="css/styles.css">.

Dans la syntaxe précédente, on a :

- La balise <link> avertit le navigateur qu'il doit chercher un document situé à l'extérieur de la page HTML.
- L'attribut rel="stylesheet" précise que le document en question est une feuille de style externe.
- L'attribut type="text/css" précise le type de feuille de style.
- L'attribut href=" URL " donne l'URL de la feuille de style, c'est-à-dire son emplacement sur Internet.

III. DEFINITION DES SELECTEURS

Revenons sur la syntaxe de définition des feuilles de style précédente.

```

Selecteur {
    propriété1: valeur;
    propriété2: valeur;
    ...
    propriétéN: valeur ;
}

```


Dans cette syntaxe, le sélecteur représente l'élément ou les éléments auxquels on veut appliquer les différentes propriétés.

IV. QUELQUES TYPES DE SELECTEURS SIMPLES

Il existe plusieurs types de sélecteurs dont quelques-uns.

Sélecteurs	Syntaxe	Exemples du code CSS	Élément auxquels s'applique le CSS
Sélecteur d'élément	Element{ } Element1, element2{ }	p{ } h3{ } li, a{ }	<p> <h3> et <a>
Sélecteur d'attribut	[attribut]{ } Element[attribut=valeur]{ }	[alt] { } input[type=submit] { }	Tout élément possédant un attribut alt Tout élément <input type="submit"/>
Sélecteur universel	*		Tous les éléments de la page
Sélecteur de classe	.nomclasse{ } element.classe{ }	.remarque { } div.remarque { }	Tout élément l'attribut class="remarque" Tout élément <div class="remarque">
Sélecteur d'id	Element#id{ }	img#photo{ }	Tout élément

V. LES PSEUDO-CLASSES ET PSEUDO-ELEMENTS

Les pseudo-classes permettent d'affiner le style appliqué à un certain nombre de balises en définissant une réaction à un événement ou bien à la position relative de l'élément au sein des autres balises.

Contrairement aux classes, le nom des pseudo-classes est prédéfini, il n'est donc pas possible de créer ses propres pseudo-classes.

Tableau 1: Quelques pseudo-classes et pseudo-éléments

Pseudo-classe ou pseudo-élément	Rôle
:hover	Élément lien survolé
:visited	Élément lien visité
:link	Lien
:active	Etat activé d'un élément
:focus	Etat réception de focus
:checked	Etat coché d'un élément checkbox
::first-letter	Pseudo élément désignant la Première lettre

Exemples :

- **a:hover {font-decoration: underline;}** permet de souligner le texte du lien lors du survol de celui-ci
- **textarea:focus {color: #FF0000;}** permet de changer la couleur du texte de la zone de texte lorsque celle-ci reçoit le focus.
- **a:active {color: #FF0000;}** permet de changer la couleur du texte du lien lorsque celui-ci est actif.
- **a:link{color: #FF0500;}** permet de définir les propriétés des liens que visiteur n'a pas encore été sélectionné.
- **a:visited {color: #FF000A;}** permet de définir la couleur des liens hypertextes que le client a déjà visités
- **p ::first-letter** : pseudo-classe d'élément s'appliquant sur la première lettre de chaque paragraphe :

```
p::first-letter{
    font-size: 3em;
    font-weight: bold;
    color:rgb(0,0,120);
    text-decoration:none;
}
```

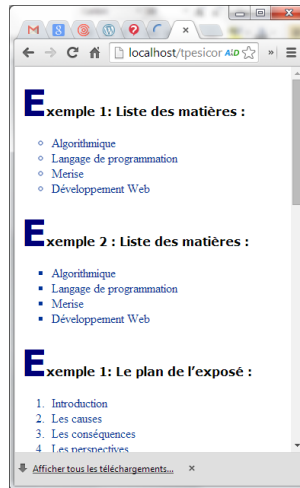


Figure 26: Résultat du code CSS ci-contre

VI. LES PROPRIETES

Elles représentent les différents effets souhaités sur l'élément. La ligne de propriété est composée d'un nom de propriété et d'une valeur séparée par deux points terminée par un point-virgule.

Exemple : font-weight: bold;

Il existe un nombre important de propriétés CSS même si toutes ne sont pas encore prises en compte par tous les navigateurs.

Nous énumérerons quelques propriétés selon les groupes d'éléments auxquelles elles peuvent s'appliquer.

Tableau 2 : Quelques propriétés sur le texte

Propriétés	Rôles	Quelques valeurs possibles
Color	Couleur du texte	Blue, red, green, yellow, rgb(0,0,255), rgba(0,0,0,0.5), #003399
Font	Déclaration groupée de propriétés sur un texte (taille, couleur, police, etc.)	font-style font-weight font-size/line-height font-family bold 16px "Trebuchet MS", Arial, Helvetica, sans-serif ;
Font-weight	Epaisseur du texte	normal, bold, light, bolder, lighter valeur numérique
Font-size	Taille de police	xx-small, x-small, small, medium, large, x-large, xx-large, taille en points (pt), cm, %
Font-style	Style de police du texte	normal, italic, oblic
Font-family	Famille de police	Arial ou Georgia, "Times New Roman", Times, serif
Font-stretch	Etirement du texte	Condensed, expanded
Text-decoration	Décoration du texte	blink, underline, line-through, overline ou none (clignotant, souligné, barré, surligné ou aucune)
Text-indent	Indentation de texte	en pouces (in), en cm, ou en %
Text-align	Alignement du texte	left, center, right, justify
Text-transform	Définit la casse de texte	uppercase, lowercase, capitalize (majuscule, minuscule, première lettre en majuscule)
Word-spacing	Espace entre les mots	En points (pt), pouces (in), en cm, en pixels (px), ou en %
Line-height	Interligne du texte	En points (pt), pouces (in), en cm, en pixels (px), ou en %

Exemple :

```
p{
    font-family: Georgia, "Times New Roman", Times, serif; /*Famille de police*/
    font-size:16px; /* taille 16*/
    font-weight:bold; /*texte en gras*/
    text-align:justify;
}
```

Tableau 3 : Propriétés de fond

Propriétés	Rôles	Quelques valeurs possibles
background-color	Couleur de fond	"#RRGGBB", "#F03300A", blue, green
background-image	Image de fond	url(..images/bonappetitfig.jpg)
background-repeat	Répétition de l'arrière-plan	repeat, repeat-x, repeat-y, no-repeat
background-attachment	Comportement de l'image l'arrière-plan lors du défilement	scroll, fixed
background-position	Position de l'image d'arrière-plan par rapport au coin supérieur gauche	top, center, bottom, left, center ou right En points (pt), pouces (in), en cm, en pixels (px), ou en %
background-size	Taille de l'image de fond par rapport aux dimensions de l'élément	80% 80%
background	Groupe de propriétés d'arrière-plan	url(test.jpg) fixed repeat

Exemple :

```
body{
    background-image:url(..images/bonappetitfig.jpg);
    background-repeat:no-repeat;
    background-attachment:fixed;
    background-position:left top;
    background-size:10% 15%;
}
```



Tableau 4 : Propriétés d'ombres et transparence

Propriétés	Rôles	Quelques valeurs possibles
Text-shadow	Ombrage du texte	
Text-outline	Contour du texte	
box-shadow	Ombrage de l'élément boîte	
opacity	Opacité de l'élément	

Exemple:

```
.logo{
    width:15%;
    height:15%;
    box-shadow: 10px 10px 5px #666666;
}
```

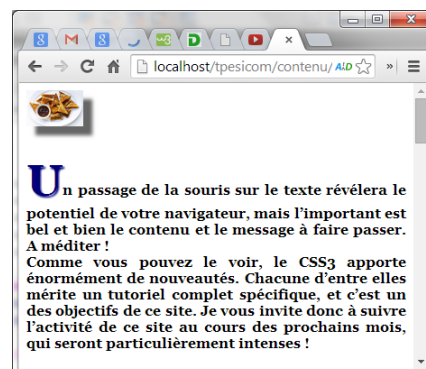


Tableau 5 : Quelques propriétés de bordure

Propriétés	Rôles	Quelques valeurs possibles
border	Groupe de propriétés de bordure	border: 1px solid #CCC;
Border-width	Largeur de bordure	2px
Border-color	Couleur de bordure	Gray, #CCC
Border-spacing	Espacement de bord	0px
Border-radius	Rayon du bord arrondi	2px 2px 3px 3px
Outline	Contour	
Outline-style	Style du contour	Solid, dotted
Outline-color	Couleur du contour	blue
Outline-width	Largeur de contour	1px

Exemple:

```
table, tr, td{
    border-width:1px;
    border-color:#CCCCCC;
    border-style:solid;
    border-spacing:0px;
    width:60%;
}
```

LISTE DES PROPRIETES

Type	Propriété	Valeur	Description
Marges	margin-top	En points (pt), pouces (in), en cm, en pixels (px), ou en %	Valeur de la marge supérieure
	margin-right	En points (pt), pouces (in), en cm, en pixels (px), ou en %	Valeur de la marge de droite
	margin-bottom	En points (pt), pouces (in), en cm, en pixels (px), ou en %	Valeur de la marge inférieure
	margin-left	En points (pt), pouces (in), en cm, en pixels (px), ou en %	Valeur de la marge de gauche
	margin		Raccourci pour les propriétés de marge
	padding-top	En points (pt), pouces (in), en cm, en pixels (px), ou en %	Ecartement entre l'élément et le bord supérieur
	padding-right	En points (pt), pouces (in), en cm, en pixels (px), ou en %	Ecartement entre l'élément et le bord droit
	padding-bottom	En points (pt), pouces (in), en cm, en pixels (px), ou en %	Ecartement entre l'élément et le bord inférieur
	padding-left	En points (pt), pouces (in), en cm, en pixels (px), ou en %	Ecartement entre l'élément et le bord gauche
	padding		Raccourci vers les propriétés d'écartement
Listes	list-style-type	disc, circle ou square	Type de puces et de numérotation
	list-style-image	URL	Permet de remplacer les puces par une image
	list-style-position	inside ou outside	Spécifie si les puces sont à l'intérieur ou à l'extérieur du texte
	list-style		Raccourci vers les propriétés de liste

Exemple complet :

```

1 <html>
2 <head>
3   <title>Exemples de CSS</title>
4   <STYLE type="text/css">
5     <!--
6       /*Styel des titres avec la balise H1*/
7       h1{
8         font-family: Verdana; font-size: 18px; font-style: bold; color: red; text-indent:0.5in
9       }
10      /*Definition de 2 classes pour la balise <a>*/
11      a.vert{
12        font-family:verdana; font-size:14px; color: green
13      }
14      a.rouge{
15        font-family:verdana; font-size:14px; color: red
16      }
17      /*Definition de classe universelle*/
18      .citation{
19        font-family:calibri; font-size:16px; font-style:italic; color:blue
20      }
21 |
22      -->
23   </STYLE>
24 </head>
25 <body>
26   <h1>Bonjour</h1>
27   <div >
28     <p>Paragraphe 1 :<span class="citation">Ne faites pas aux autres ce que vous ne souhaitez pas que l'on vous
29     fasse</span></p>
30     <p>Paragraphe 2 <a class="vert" href="">Cliquez ici</a></p>
31     <p>Paragraphe 3 <a class="rouge" href="">Cliquez ici</a></p>
32   </div>
33 </body>
34 </html>

```

Figure 27 - Exemple complet classes universelles

Résultat attendu :

Bonjour

Paragraphe 1 : *Ne faites pas aux autres ce que vous ne souhaitez pas que l'on vous fasse*

Paragraphe 2 [Cliquez ici](#)

Paragraphe 3 [Cliquez ici](#)

Figure 28- Aperçu de la page obtenue

MINI-PROJET:**Projet 1 : Réalisation d'un site web de la classe BTS IDA 1^{ère} Année****APERÇU DE LA PAGE D'ACCUEIL DU SITE**


Ecole Supérieure d'Informatique et de Commerce

Bienvenue sur le site des étudiants de BTS IDA 1^{ère} Année
La programmation et le développement web n'ont plus de secret pour nous.

[Accueil](#) [Formation](#) [Récompense](#) [Bibliothèque](#) [Infos](#) [Contactez-nous](#)

Présentation de la filière

Le mot « algorithme » est un terme d'origine arabe, venant plus précisément du nom du célèbre mathématicien de Bagdad (780-850), AL-KHUIWARIZMI qui écrivit le premier ouvrage de résolution des équations linéaires et quadratiques.

L'algorithmique est l'ensemble des règles et des techniques qui sont impliquées dans la définition et la conception d'algorithmes.

Un algorithme est donc une suite ordonnée et finie d'instructions élémentaires permettant de résoudre un problème.

Le dictionnaire Robert définit l'algorithme comme un ensemble de règles opératoires propres à un calcul.

News TIC

[Droit à l'oubli : Google s'incline devant la Justice européenne](#)

Google a mis en ligne un formulaire permettant à toute personne de demander la suppression de son moteur de recherche des résultats de recherche inclu...

[Droit à l'oubli : Google s'incline devant la Justice européenne](#)

Google a mis en ligne un formulaire permettant à toute personne de demander la suppression de son moteur de recherche des résultats de recherche inclu...

[Droit à l'oubli : Google s'incline devant la Justice européenne](#)

Google a mis en ligne un formulaire permettant à toute personne de demander la suppression de son moteur de recherche des résultats de recherche inclu...

Album Photos

Quelques photos d'étudiants de la Classe



Rubriques

- Accueil
- Formation
- Récompense
- Bibliothèque
- Infos
- Contactez-nous

Copyright © 2013-2014 - BTS IDA 1 ESICOM Plateau- Tous droits réservés
 Plateau, Près de l'Hôtel IBIS
 Contactez-nous ici

CODE HTML 5

```

<!DOCTYPE HTML>
<html>
<head>
<meta charset="utf-8">
<title>Bienvenue sur le site de la classe BTS IDA 1ère Année</title>
<link rel="stylesheet" href="style.css" />
</head>

<body>
  <div id="corps">
    <header>
      <div id="titre_ecole">
        
        <h2>Ecole Supérieure d'Informatique et de Commerce</h2>
      </div>
      <div id="banniere">
        <h1> Bienvenue sur le site des etudiants de BTS IDA 1ère Année</h1>
        <h3> La programmation et le developpement web n'ont plus de secret pour nous. </h3>
      </div><!-->
    </header>
    <nav>
      <ul id="menu_h">
        <li> <a href="#"> Accueil </a> </li>
        <li> <a href="#"> Formation </a> </li>
        <li> <a href="#"> Récompense </a> </li>
        <li> <a href="#"> Bibliothèque </a> </li>
        <li> <a href="#"> Infos </a> </li>
        <li> <a href="#"> Contactez-nous </a> </li>
      </ul>
    </nav>
    <section>
      <article>
        <h2>Presentation de la filière</h2>
        <p> Le mot « algorithmme » est un terme d'origine arabe, venant plus précisément du nom du
<strong>célèbre mathématicien de Bagdad (780-850), AL-KHUWARIZMI</strong> qui écrit le premier ouvrage de
résolution des équations linéaires et quadratiques.</p>
        <p>L'algorithmique est l'ensemble des règles et des techniques qui sont impliquées dans la définition et la
conception d'algorithmes.
</p>
        <p>Un algorithme est donc une suite ordonnée et finie d'instructions élémentaires permettant de résoudre un
problème.</p>
        <p>Le dictionnaire Robert définit l'algorithme comme un ensemble de règles opératoires propres à un
calcul.</p>
      </article>

      <article >
        <h2>Album Photos</h2>
        <figure>
          <figcaption>Quelques photos d'étudiants de la Classe</figcaption>
          <a href="images/btsida1_ens.png" target="_blank"></a>
          <a href="images/btsida1_garcons.png" target="_blank"></a>
          <a href="images/btsida1_filles.png" target="_blank"></a>
        </figure>
      </article>
  </div>

```



```

        <a href="images/btsida1_ens.png" target="_blank"></a>
        <a href="images/btsida1_ens.png" target="_blank"></a>
    </figure>
</article>

</section>

<aside>
<h2>News TIC</h2>
<p class="titreinfo"><a href="http://www.lemondeinformatique.fr/" target="_blank">Droit à l'oubli :
Google s'incline devant la Justice européenne</a></p>
<p>Google a mis en ligne un formulaire permettant à toute personne de demander la suppression de son
moteur de recherche des résultats de recherche inclu...</p>
<p class="titreinfo"><a href="http://www.lemondeinformatique.fr/" target="_blank">Droit à l'oubli :
Google s'incline devant la Justice européenne</a></p>
<p>Google a mis en ligne un formulaire permettant à toute personne de demander la suppression de son
moteur de recherche des résultats de recherche inclu...</p>
<p class="titreinfo"><a href="http://www.lemondeinformatique.fr/" target="_blank">Droit à l'oubli :
Google s'incline devant la Justice européenne</a></p>
<p>Google a mis en ligne un formulaire permettant à toute personne de demander la suppression de son
moteur de recherche des résultats de recherche inclu...</p>
</aside>
<footer>
    <h2>Rubriques</h2>
    <ul>
    <li><a href="#">Accueil</a></li>
    <li><a href="#">Formation</a></li>
    <li><a href="#"> Récompense </a> </li>
    <li><a href="#"> Bibliothèque </a> </li>
    <li><a href="#"> Infos </a> </li>
    <li><a href="#"> Contactez-nous </a> </li>
    </ul>
    <p>Copyright &copy; 2013-2014 - BTS IDA 1 ESICOM Plateau- Tous droits réservés<br/>
Plateau, Près de l'Hôtel IBIS<br />
Contactez-nous <a href="mailto:esicom.ida2014@gmail.com">ici</a></p>
</footer>
</div>
</body>
</html>

```

CODE CSS

```

@charset "utf-8";
/* CSS Document */

body{
    background-color:rgba(191,191,191,0.5);/* Gris
avec transparence de 0.5 */
}
/*bloc de corps de la page contenant toutes les données*/
#corps{
    width:80%;
    height:100%;
    background-color:#FFFFFF; /*Gris */
    margin:auto;
    border-radius:10px 10px; /*Bordures arrondies*/
    border:2px solid rgb(214,16,5);
}
/*Taille du logo*/
#logo{
    width:30%;
    height:30%;
    vertical-align:middle; /*Aligner verticalement */
    margin-left:20px; /*Marge exterieure de gauche*/
}
/*Cadre logo et nom de l'ecole*/
#titre_ecole{
    background-color:#fde5d9; /*Couleur de fond rose
du logo*/
    border:1px solid #fde5d9;
    border-radius:10px 10px 0px 0px; /*Bordures
arrondies*/
}
/*banniere*/
#banniere{
    background-color:rgba(20,35,126,1); /*#14237e;*/
    border:1px solid rgba(20,35,126,1);
}
/*Nom de l'ecole*/
header div h2{
    display:inline-block;
    font-family:"Trebuchet MS", Arial, Helvetica, sans-
serif;
    color:#14237e; /*Le bleu du logo*/
    font-weight:lighter; /*Rendre l'écriture fine*/
    margin-left:50px; /*Marge exterieure de gauche*/
}
/*Titre de bienvenue*/
header h1{
    font-family:"Trebuchet MS", Arial, Helvetica, sans-
serif;
    color:rgba(255,255,255,1); /*#14237e; Le bleu du
logo*/
    font-weight:bold; /*Rendre l'écriture fine*/
    font-size:26px;
    text-align:center;
    text-shadow:0px 1px 2px #FFFFFF;
}
/*Slogan*/
header h3{
    font-family:"Trebuchet MS", Arial, Helvetica, sans-
serif;
    color:rgba(214,16,5,0.9); /*#14237e; Le bleu du
logo*/

```

```

font-style:italic; /*Rendre l'écriture italique*/
font-size:16px;
text-align:center;
}
/*Menu */
nav{
    width:100%;
    text-align:center;
    font-family:Arial, Helvetica, sans-serif;
}
/*Element texte du menu*/
nav ul li{
    display:inline-block;
    margin-left:15px;
    padding:10px;
    border: 2px dotted #999999; /*
}
/*Element conteneur ul du menu*/
nav ul{
    list-style-type:none;
    background-color:#000; /*#666; Gris foncé*/
    margin:0;
}
nav a{
    font-size: 1.1em;
    color:#CCC; /*#181818; Gris*/
    text-decoration:none;
}
nav a:hover{
    font-size: 1.1em;
    color:#6CF; /*#CCC; Gris clair*/
    text-decoration:none;
}
/*Presentation*/
section{
    width:70%;
}
/*News TIC*/
aside{
    width:25%;
}
article, section, aside{
    display:inline-block;
    vertical-align:top;
    border:1px solid #fde5d9;
    border-radius:10px 10px 0px 0px; /*Bordures
arrondies*/
    box-shadow: 0px 0px 20px #aaa;
    -moz-box-shadow: 0px 0px 20px #aaa;
    -webkit-box-shadow: 0px 0px 20px #aaa;
    margin-top:5px;
    margin-left:5px;
    margin-bottom:5px;
    margin-right:5px;
}
article p, aside p{
    font-family:Arial, Helvetica, sans-serif;
    font-size:14px;
    color:#000;
    padding-left:15px;
}

```

```
article h2{
    font-family:Arial, Helvetica, sans-serif;
    font-size:20px;
    color:#000;
    border-bottom-style:dotted;
    border-bottom-color:#FF0000;
    border-bottom-width:thin;
    background-color:#CCCCCC;
    padding-left:15px;
    margin-top:0;
    border-radius:10px 10px 0px 0px; /*Bordures
arrondies*/
}
aside h2{
    font-family:Arial, Helvetica, sans-serif;
    font-size:16px;
    color:#000;
    border-bottom-style:ridge;
    border-bottom-color:#FF0000;
    border-bottom-width:thin;
    padding-left:10px;
    border-radius:10px 10px 0px 0px; /*Bordures
arrondies*/
}
.titreinfo{
    font-family:Arial, Helvetica, sans-serif;
    font-size:12px;
    font-weight:bold;
    color:#999999;
    padding-left:10px;
}
/*Album photo*/
figure img{
    display:inline-block;
    border:1px solid rgb(214,16,5);
    margin:2px;
    width:25%;
    height:25%;
}
figcaption{
    color:#333;
    font-family:Arial, Helvetica, sans-serif;
    font-size:12px;
}

/*pied de page*/
footer, footer h2{
    font-family:Arial, Helvetica, sans-serif;
    font-size:14px;
    background-color:#000000;
    color:#CCCCCC;
    padding-left:5px;
    padding-top:5px;
}
footer a{
    text-decoration:none;
    color:#CCCCCC;
}
footer ul{
    display:inline-block;
    vertical-align:top;
    margin-left:50px;
}
}
footer p{
    display:inline-block;
    margin-left:100px;
    text-align:center;
}
}
```

Aperçu de la page d'accueil du site

The screenshot shows the homepage of the ESICOM website. At the top, the ESICOM logo is displayed in red and blue, with the text "Ecole Supérieure d'Informatique et de Commerce" to its right. Below the logo, a blue banner reads "Bienvenue sur le site des étudiants de BTS IDA 1ère Année" and "La programmation et le développement web n'ont plus de secret pour nous." A navigation menu includes "Accueil", "Formation", "Récompense", "Bibliothèque", "Infos", and "Contactez-nous".

The main content area is divided into two columns. The left column features a section titled "Présentation de la filière" with text explaining the term "algorithme" and its historical origin with Al-Khwarizmi. Below this is an "Album Photos" section with the caption "Quelques photos d'étudiants de la Classe" and five small photographs of students in blue uniforms. The right column is titled "News TIC" and contains two identical news items, each with a link to a Google search and a message about a search engine issue.

At the bottom, a "Rubriques" section lists navigation links: Accueil, Formation, Récompense, Bibliothèque, Infos, and Contactez-nous. A copyright notice reads: "Copyright © 2013-2014 - BTS IDA 1 ESICOM Plateau- Tous droits réservés. Plateau, Près de l'Hôtel IBIIS. Contactez-nous ici".

BIBLIOGRAPHIE

Commentcamarche. (2006, Juin). *Feuille de style - CSS*. Retrieved Octobre 2010, from Portail encyclopedique commentcamarche.net: <http://www.commentcamarche.net>

Goetter, R. (2005,2007). *CSS 2 - Pratique du design web*. Groupe EYROLLES.