

# CRÉATION ET GESTION DES SITES WEB: HTML 5 ET CSS 3

**ISTIC**

Enseignant: KWETTE NOUMSI Loïc Bastien

November 28, 2017

# Contenu du cours

<b>1</b>	<b>AVANT-PROPOS</b>	<b>4</b>
<b>2</b>	<b>INTRODUCTION AU WEB</b>	<b>6</b>
<b>3</b>	<b>LES OUTILS NÉCESSAIRES</b>	<b>8</b>
3.1	Les éditeurs . . . . .	8
3.2	Les navigateurs web . . . . .	9
3.3	Autres logiciels . . . . .	10
3.4	TP : installation et utilisation des principaux outils . . . . .	10
<b>4</b>	<b>CRÉER SA PAGE WEB AVEC LE HTML</b>	<b>11</b>
4.1	Présentation . . . . .	11
4.2	Balises et leurs attributs . . . . .	11
4.2.1	Les balises paires . . . . .	11
4.2.2	Les balises orphelines . . . . .	11
4.2.3	Les attributs . . . . .	12
4.3	Les règles d'écritures . . . . .	12
4.4	Structures d'une page web . . . . .	14
4.4.1	La balise <code>&lt;!DOCTYPE html&gt;</code> . . . . .	14
4.4.2	La balise <code>&lt;html&gt;&lt;/html&gt;</code> . . . . .	14
4.4.3	La balise <code>&lt;head&gt;&lt;/head&gt;</code> . . . . .	14
4.4.4	Le corps de la page <code>&lt;body&gt;&lt;/body&gt;</code> . . . . .	15
4.4.5	<code>&lt;!-- LES COMMENTAIRES --&gt;</code> . . . . .	15
<b>5</b>	<b>RÉDIGER LE CONTENU DE SA PAGE WEB</b>	<b>16</b>
5.1	Organiser le texte . . . . .	16
5.1.1	Les paragraphes . . . . .	16
5.1.2	Les titres . . . . .	17
5.1.3	Les listes . . . . .	18
	Les listes non ordonnées . . . . .	18
	Les listes ordonnées . . . . .	19
5.2	Insérer des images . . . . .	19
5.2.1	Les types d'images . . . . .	19
5.2.2	Les erreurs à éviter . . . . .	21
5.2.3	Insertion d'images . . . . .	21
	Ajouter une infobulle . . . . .	21
	Miniature cliquable . . . . .	22

	Création d'une figure . . . . .	23
5.3	Créer des liens . . . . .	23
5.3.1	Un lien vers un autre site . . . . .	24
5.3.2	Lien vers une autre page du site . . . . .	24
5.3.3	Les ancres . . . . .	25
5.3.4	Cas pratique de liens . . . . .	25
5.4	Les tableaux . . . . .	26
5.4.1	Les attributs de la balise <code>&lt;table&gt;</code> . . . . .	27
5.4.2	Fusionner les cellules d'un tableau . . . . .	29
5.4.3	Structurer un tableau . . . . .	31
5.5	Les formulaires . . . . .	32
5.5.1	Créer un formulaire . . . . .	32
5.5.2	Les zones de texte . . . . .	33
	Zone de texte monoligne . . . . .	34
	Quelques attributs supplémentaires . . . . .	34
5.5.3	Zone de mot de passe . . . . .	35
5.5.4	Zone de texte multiligne . . . . .	35
5.5.5	Zone pour l'email . . . . .	36
5.5.6	Zone pour url d'un site web . . . . .	37
5.5.7	Numéro de téléphone . . . . .	37
5.5.8	Zone de recherche . . . . .	37
5.5.9	Nombre . . . . .	38
5.5.10	Date . . . . .	38
5.5.11	Les éléments d'option . . . . .	39
	Les cases à cocher . . . . .	39
	Les zones d'option . . . . .	40
	Les listes déroulantes . . . . .	40
5.5.12	Finaliser et envoyer le formulaire . . . . .	41
	Regrouper les champs . . . . .	42
	Sélection automatique . . . . .	42
	Rendre un champ obligatoire . . . . .	42
	Bouton d'envoi . . . . .	43
5.6	La vidéo et l'audio . . . . .	44
5.6.1	Les formats audio et vidéo . . . . .	44
	Les formats audio . . . . .	44
	Les formats vidéo . . . . .	45
5.6.2	Insertion d'un élément audio . . . . .	45
5.6.3	Insertion d'un élément vidéo . . . . .	46
<b>6</b>	<b>ÉCRIRE LES FEUILLES DE STYLES AVEC LE CSS</b>	<b>48</b>
<b>7</b>	<b>LES ANNEXES</b>	<b>49</b>
7.1	Liste des balises HTML 5 . . . . .	49
7.1.1	Les balises de premier niveau . . . . .	49
7.1.2	Les balises d'en-tête . . . . .	50
7.1.3	Les balises d'architecture . . . . .	51
7.1.4	Les balises de structuration de texte . . . . .	51

7.1.5	Les balises de liste . . . . .	52
7.1.6	Les balises de tableau . . . . .	53
7.1.7	Les balises de formulaire . . . . .	53
7.1.8	Les balises multimédia . . . . .	54
7.1.9	Les autres balises . . . . .	54
7.1.10	Les attributs standard . . . . .	54
7.1.11	Les attributs standard . . . . .	55
7.2	Liste des propriétés CSS . . . . .	56
7.2.1	Les propriétés de mise en forme . . . . .	56
7.2.2	Les propriétés de couleur et de fond . . . . .	57
7.2.3	Les propriétés de boîtes ou blocs . . . . .	58
7.2.4	Les propriétés des tableaux . . . . .	59
7.2.5	Les propriétés de positionnement et d’affichage . . . . .	60
7.2.6	Les propriétés des listes . . . . .	60
7.3	Documentation . . . . .	61

# Chapter 1

## AVANT-PROPOS

Aujourd'hui, les sites web sont partout et permettent d'accéder à toutes sortes de services, que ce soit faire ses achats en ligne, s'informer de l'actualité, communiquer avec le monde entier, lire des tutoriels ou tout simplement rechercher une information.

L'objectif principal de ce cours est consacré à la création de sites web. Pour réussir un site, il faut en effet respecter une certaine démarche, choisir et maîtriser des techniques... Vous trouverez ici les bases de deux d'entre elles, les deux techniques essentielles aujourd'hui : **HTML 5** (HyperText Markup Language) pour rédiger les contenus des sites internet et **CSS 3** (Cascading Style Sheets) pour créer la mise en page et les différentes présentations desdits contenus pour divers terminaux allant des navigateurs internet aux téléphones portables.

Pour y arriver, nous suivrons dans l'ordre des chapitres, le cheminement ci-après :

**Chapitre 1 : Introduction au web** – Pour vous permettre de mieux appréhender le fonctionnement d'un site web.

**Chapitre 2 : Les outils nécessaires** – Vous présenteront la gamme d'outils dont vous aurez besoin pour mettre sur pieds un site.

**Chapitre 3 : Créer sa page web avec le HTML** – Vous présentera la structure de base d'une page web et les concepts de balises.

**Chapitre 4 : Rédiger le contenu de sa page web** – Vous parlera de tout ce qu'on peut introduire dans une page web (texte, images, liens, sons, vidéos,...) et comment le faire.

**Chapitre 5 : Écrire les feuilles de style avec le CSS** – Vous aidera à faire la décoration de vos pages web.

**Chapitre 6 : Ergonomie et structuration web** – Comment disposer les différents composants d'une page web afin qu'elle soit attrayante à visiter par toutes sortes d'internautes.

**Chapitre 7 : Les standards du web** – IL y a des règles et des normes à respecter quand on doit concevoir et publier un site web. Ce sera l'objet de ce chapitre.

**Chapitre 8 : Concevoir un site pas à pas** vous recommandera les étapes à respecter dans la conception d'un site web de phase initiale à la finale.

**Chapitre 9 : Les limites du HTML et introduction aux sites web dynamiques** – IL y a des choses qu'on ne peut encore faire rien qu'avec du HTML et CSS (Exemple : Gérer les interactions des utilisateurs.), heureusement qu'il existe des langages de programmation web dynamiques.

**Chapitre 10 : Quelques conseils** – On en a toujours besoin.

Les chapitres 2 à 8 seront accompagnés de travaux pratiques pour valider les connaissances acquises aux chapitres concernés.

A la fin de ce document vous trouverez la liste complète des balises HTML et des propriétés CSS suivies de quelques références bibliographiques pour perfectionner votre apprentissage.

Bonne formation !  
KWETTE NOUMSI.

# Chapter 2

## INTRODUCTION AU WEB

Le web (en français, toile d'araignée) symbolise le réseau maillé de serveurs d'informations formant une toile d'araignée. Ces serveurs vont des pages personnelles aux interfaces de base de données. Par extension on parle de web pour un serveur de pages HTML.

En effet, pour créer un site web, on doit parler un langage que l'ordinateur comprend. Les sites web sont donc construits à l'aide des langages HTML et CSS. Lorsqu'on crée son site, on écrit du code informatique dans les langages HTML et CSS. Ce code est appelé le **code source**. Pourquoi ? Parce que ce code est la source d'un résultat visuel : votre site web à l'aide d'un navigateur web.

Les langages HTML et CSS sont à la base du fonctionnement de tous les sites web. Quand vous consultez un site avec votre navigateur, il faut savoir que, en coulisses, des rouages s'activent pour permettre au site web de s'afficher. L'ordinateur se base sur ce qu'on lui a expliqué en HTML et CSS pour savoir ce qu'il doit afficher, comme le montre les figures suivantes :



Figure 2.1: LANGAGE HTML ET CSS

Dans ce cours nous allons avoir affaire à deux langages qui se complètent car ils ont des rôles différents :

- **HTML (HyperText Markup Language)** : il a fait son apparition dès 1991 lors du lancement du Web. Son rôle est de gérer et organiser le contenu. C'est donc en HTML que vous écrirez ce qui doit être affiché sur la page : du texte, des liens, des images... Vous direz par exemple : Ceci est mon titre, ceci est mon menu, voici le texte principal de la page, voici une image à afficher, etc. . En théorie, il est possible de faire tout un site rien qu'avec le langage HTML. Toutefois, celui-ci ne serait pas très esthétique...



Figure 2.2: TRADUCTION PAR L'ORDINATEUR ET RESULTAT VISIBLE A L'ECRAN

- **CSS (Cascading Style Sheets, aussi appelées Feuilles de style)** : c'est là qu'intervient le langage CSS. Son rôle est en quelque sorte de « décorer » votre site web, lui donner de l'allure. On utilise le CSS en particulier pour réaliser la mise en page du site, pour définir la police, la taille du texte, la couleur du texte et du fond, etc. Ce langage est venu compléter le HTML en 1996.

En fait, ces deux langages permettent tout simplement de séparer le fond de la forme :

- Le fond est l'affaire du langage HTML ;
- Alors que le langage CSS s'occupera de la forme.

### Remarque très importante !!!

1. De nombreuses personnes confondent (à tort) Internet et le Web. Il faut savoir que le Web fait partie d'Internet. Internet est un grand ensemble qui comprend, entre autres : le Web, les e-mails, la messagerie instantanée, etc.
2. Vous avez peut-être aussi entendu parler du langage XHTML. Il s'agit d'une variante du HTML qui se veut plus rigoureuse et qui est donc un peu plus délicate à manipuler. Pour faire simple, le HTML est apparu le premier en 1991 inventé par Tim Berners-Lee. Début 2000, le W3C (World Wide Web Consortium), système mondial qui définit les règles de codage des pages web, a lancé le XHTML en indiquant que ce serait l'avenir mais le XHTML n'a pas percé comme on l'espérait. Retour aux sources en 2009 : le W3C abandonne le XHTML et décide de revenir au HTML pour le faire évoluer. Il y a beaucoup de confusion autour de ces langages, alors qu'ils se ressemblent beaucoup. Aucun n'est vraiment meilleur que l'autre, il s'agit de deux façons de faire différentes.

Dans ce cours, nous allons travailler sur la dernière version de HTML (HTML 5) et celle du CSS (CSS 3) qui sont aujourd'hui considérés comme les langages d'avenir auxquels tout développeur web est incité à utiliser.

Le chapitre suivant vous fournira les logiciels dont vous avez besoin pour créer un site web.



# Chapter 3

## LES OUTILS NÉCESSAIRES

Il existe effectivement de nombreux logiciels dédiés à la création de sites web. Il s'agit en général des éditeurs, logiciels pour écrire du code HTML et CSS et des logiciels pour voir nos pages web, les navigateurs web.

### 3.1 Les éditeurs

Ce sont des logiciels d'édition de sites web c'est-à-dire qui nous permettent de saisir, modifier et enregistrer notre code source. On peut classer ces logiciels de création de site web en deux catégories :

- Les **WYSIWYG** (What You See Is What You Get - Ce Que Vous Voyez Est Ce Que Vous Obtenez): ce sont des programmes qui se veulent très faciles d'emploi, ils permettent de créer des sites web sans apprendre de langage particulier.

Le tableau ci-dessous vous présente les plus connus :

Logiciel	Système d'exploitation	Prix	Qualité du code	Adresse
Nvu	Windows, Mac et Linux	Gratuit et Open Source	Assez bonne	<a href="http://www.nvu.com">http://www.nvu.com</a>
Adobe GoLive	Windows et Mac	580 €	Moyenne	<a href="http://www.adobe.fr/products/golive/">http://www.adobe.fr/products/golive/</a>
Macromedia Dreamweaver	Windows et Mac	480 €	Moyenne	<a href="http://www.adobe.fr/products/dreamweaver">http://www.adobe.fr/products/dreamweaver</a>
FrontPage	Windows et Mac	250 €	Mauvaise	<a href="http://www.microsoft.com/france/frontpage">http://www.microsoft.com/france/frontpage</a>
Word	Windows et Mac	290 €	Très mauvaise	<a href="http://www.microsoft.com/france/word">http://www.microsoft.com/france/word</a>

Leur principal défaut est la qualité souvent assez mauvaise du code HTML et CSS qui est automatiquement généré par ces outils. Un bon créateur de site web doit tôt ou tard connaître HTML et CSS, c'est pourquoi nous ne vous recommandons pas l'usage de ces outils.

- **Les éditeurs de texte:** ce sont des programmes dédiés à l'écriture de code. On peut en général les utiliser pour de multiples langages, pas seulement HTML et CSS. Ils se révèlent être de puissants alliés pour les créateurs de sites web !

En fonction des systèmes d'exploitation, voici une liste d'éditeurs de texte que vous pourrez utiliser:

Logiciel	Système d'exploitation	Prix	Libre ?	Commentaires	Adresse
Bloc-Notes	Windows	Gratuit (livré avec Windows)	Non	Simple, très simple, trop simple ?	Aucune (de série avec Windows)
Notepad++	Windows	Gratuit	Oui	Excellent éditeur, simple de prise en main. Recommandé.	<a href="http://notepad-plus.sourceforge.net">http://notepad-plus.sourceforge.net</a>
PSPad	Windows	Gratuit	Non	Excellent éditeur, très complet.	<a href="http://www.pspad.com/fr">http://www.pspad.com/fr</a>
Smultron	Mac OS X	Gratuit	Oui	Un bon éditeur pour Mac.	<a href="http://smultron.sourceforge.net">http://smultron.sourceforge.net</a>
jEdit	Windows, Mac et Linux	Gratuit	Oui	Fonctionne sur tous les ordinateurs. Puissant.	<a href="http://www.jedit.org/">http://www.jedit.org/</a>
Emacs	Linux, existe aussi pour Windows et pour Mac OS X	Gratuit	Oui	Puissant, mais délicat de prise en main.	<a href="http://www.gnu.org/software/emacs/">http://www.gnu.org/software/emacs/</a>
vim	Linux, existe aussi pour Windows et pour Mac OS X	Gratuit	Oui	Puissant, mais délicat de prise en main.	<a href="http://www.vim.org/">http://www.vim.org/</a>
UltraEdit	Windows	Payant (environ 30 €)	Non	Puissant et réputé.	<a href="https://www.ultraedit.com/">https://www.ultraedit.com/</a>






Si vous êtes sous Windows, dans la suite de ce cours je vous recommande chaudement Notepad++. Contrairement à Bloc-Notes, il permet de colorer intelligemment le code HTML et CSS, ce qui vous permet de vous repérer plus facilement dans votre code source. Ce n'est pas négligeable.

Autres logiciels dont vous aurez besoins sont les navigateurs web.

## 3.2 Les navigateurs web

Un navigateur est un logiciel qui permet de voir les sites web. Il analyse le code HTML et CSS des pages et produit un résultat visuel.

Il existe de nombreux navigateurs différents. Voici les principaux à connaître :

Navigateur	OS	Téléchargement	Commentaires
 Google Chrome	Windows Mac Linux	<a href="#">Téléchargement</a>	Le navigateur de Google, simple d'emploi et très rapide. <i>C'est le navigateur que j'utilise au quotidien.</i>
 Logo Firefox	Windows Mac Linux	<a href="#">Téléchargement</a>	Le navigateur de la fondation Mozilla, célèbre et réputé. Je l'utilise fréquemment pour tester mes sites web.
 IE9	Windows	<a href="#">Téléchargement</a> (Déjà installé sur Windows)	Le navigateur de Microsoft, qui équipe tous les PC Windows. Je l'utilise fréquemment pour tester mes sites web.
 Logo Safari	Windows Mac	<a href="#">Téléchargement</a> (Déjà installé sur Mac OS X)	Le navigateur d'Apple, qui équipe tous les Mac.
 Logo Opera	Windows Mac Linux	<a href="#">Téléchargement</a>	L'éternel <i>outsider</i> . Il est moins utilisé mais propose de nombreuses fonctionnalités.

Le problème avec les navigateurs est qu'ils interprètent parfois le HTML et CSS différemment pour un résultat visuel différent. Cela est dû au fait que les navigateurs ne connaissent pas toujours les dernières fonctionnalités de HTML et CSS. Par exemple, Internet Explorer a longtemps été en retard sur certaines fonctionnalités CSS (et paradoxalement, il a aussi été en avance sur quelques autres). Il vous est donc recommandé de tester continuellement votre site sur au moins sur Google Chrome, Mozilla Firefox et Internet Explorer à la fois pendant sa création, afin d'être sûr que tous vos visiteurs aient un résultat correct.

**NB:** Il existe également des variantes de ces navigateurs pour les Smartphones :

- Safari Mobile pour l'iPhone d'Apple ;
- Chrome Mobile pour les téléphones dotés d'Androïde de Google;
- Internet Explorer Mobile pour les Windows Phone de Microsoft.

Notez que les navigateurs pour mobiles prennent en charge la plupart des dernières fonctionnalités de HTML et CSS. De plus, le système de mise à jour automatisé des mobiles nous garantit que les utilisateurs auront le plus souvent les dernières versions. Sachez néanmoins que des différences existent entre ces différents navigateurs mobiles et qu'il vous est également conseillé de tester votre site sur ces appareils ! En particulier, l'écran étant beaucoup moins large, il faudra vérifier que votre site s'affiche correctement.

En dehors des éditeurs et des navigateurs qui sont incontournables dans la conception de votre site, il existe d'autres logiciels qui vous faciliteront la vie dans les multiples tâches que vous aurez à réaliser.

### 3.3 Autres logiciels

Les plus utiles sont :

- **WAMP SERVER:** Le serveur web. C'est lui qui est chargé de renvoyer des pages web aux internautes;
- **La boîte-à-couleur:** Utiliser dans l'écriture des feuilles de style pour aider à effectuer votre de couleur de choix de plusieurs manières;
- **FileZilla:** Logiciel FTP (File Transfer Protocol) pour transférer vos fichiers chez l'hébergeur de votre site web via Internet.

### 3.4 TP : installation et utilisation des principaux outils

Voir le dossier OUTILS remis en cours par l'enseignant.

Maintenant que vous soyez dotés du nécessaire, entrons dans le vif du sujet et voyons comment créer vos pages web avec HTML 5.

# Chapter 4

## CRÉER SA PAGE WEB AVEC LE HTML

### 4.1 Présentation

Les pages web sont des documents informatiques qui peuvent contenir du texte, des images, des tableaux, des formulaires et d'autres éléments multimédias. Elles sont créées par les webmasters à l'aide des langages HTML et CSS, et sont visualisées par les internautes grâce à un navigateur web.

Le langage HTML, qui définit le contenu de ces pages, n'est d'autres qu'un ensemble d'instructions appelées tags ou balises.

### 4.2 Balises et leurs attributs

Les balises HTML sont des instructions données à l'ordinateur pour lui indiquer la nature du texte qu'elles encadrent. Elles veulent dire par exemple : « Ceci est le titre de la page », « Ceci est une image », « Ceci est un paragraphe de texte », etc. Dans un document HTML vous repèrerez facilement les balises. Elles sont entourées de « chevrons », c'est-à-dire des symboles < et >, comme ceci :

`<nom_balise>Contenu</nom_balise>`

On distingue 2 types de balises.

#### 4.2.1 Les balises paires

Elles s'ouvrent (balise d'ouverture `<nom_balise>`), contiennent du texte puis se referment (balise de fermeture `</nom_balise>`).

Exemple: `<title>ISTIC</title>`. Ce codage signifie que tout ce qui se trouve entre `<title>` et `</title>` est le titre de la page.

#### 4.2.2 Les balises orphelines ou monoatomiques ou auto-fermantes

Ce sont des balises qui servent le plus souvent à insérer un élément à un endroit précis (par exemple une image). Il n'est pas nécessaire de délimiter le début et la fin de l'image, on veut juste dire à l'ordinateur « Insère une image ici ». Une balise orpheline s'écrit comme ceci :

<nom\_balise />. Exemple: <img />

### 4.2.3 Les attributs

Les caractéristiques de chaque balise peuvent être précisées par des informations complémentaires que l'on désigne en tant qu'attributs de la balise. Il peut s'agir par exemple de la définition de la largeur, de la hauteur ou de l'alignement du contenu. Les attributs d'une balise sont toujours définis dans la balise d'ouverture et doivent être séparés les uns des autres par au moins une espace typographique.

Chaque attribut doit avoir une valeur, même s'il ne peut prendre qu'une valeur unique. Aucune valeur n'est donc implicite du moment que l'attribut figure dans la balise d'ouverture. La présence de certains attributs est obligatoire dans quelques balises particulières, ce que nous préciserons systématiquement quand ce sera le cas. La plupart du temps, les attributs d'une balise sont facultatifs et c'est au programmeur de déterminer leur définition par rapport au cas qu'il doit traiter.

Nombre d'attributs ont une valeur par défaut. Cela signifie que même si on ne les définit pas dans la balise, celui-ci se comporte comme si nous avions défini explicitement cette valeur. Il est donc important de connaître ce type d'attribut et de ne pas négliger de les définir avec une autre valeur si ce comportement par défaut n'est pas désiré. La valeur de tous les attributs doit être définie entre des guillemets doubles. La syntaxe conforme d'une balise ayant des attributs est donc la suivante :

```
<nom_balise attribut1="valeur1" attribut2="valeur2">Contenu</nom_balise>
```

Le contenu d'une balise peut être constitué de texte ou d'autres balises qui, eux-mêmes, peuvent en contenir d'autres, et ainsi de suite. Cet ensemble d'inclusion constitue la hiérarchie du document HTML.

Mais avant avant de commencer à coder en HTML nos pages web, il y a des règles à respecter dans son écriture.

## 4.3 Les règles d'écritures

Un document HTML doit respecter certaines règles bien simples :

1. Les balises et les attributs sont sensibles à la casse et doivent être écrits en minuscules.  
Par exemple, <body> et non plus <BODY> comme en HTML.
2. Les balises paires doivent avoir obligatoirement une balise d'ouverture et une balise de fermeture.  
Par exemple, on ne doit plus écrire :

```
<ol>  
  <li>Item1  
  <li>Item2  
.
```

Mais plutôt écrire de cette manière:

```
<ol>
  <li>Item1</li>
  <li>Item2</li>
</ol>
```

3. Les balises orphelines ne comportent qu'une seule balise et doivent se terminer par les caractères `</>` précédés d'une espace pour marquer la fin de la balise.

Par exemple, il ne faut plus écrire :

```
<br>
```

Mais plutôt écrire:

```
<br/>
```

4. Les balises ne doivent pas se chevaucher. Ils doivent donc obéir au principe premier ouvert - dernier fermé . Dans ce cas, la première balise est le parent du second et celui-ci est enfant du premier. Par exemple, le code suivant est incorrect :

```
<aside>Ce bloc contient un texte <h1>Important</aside></h1>
```

Voici le code correct:

```
<aside>Ce bloc contient un texte <h1>Important</h1></aside>
```

5. Tous les attributs doivent avoir une valeur incluse entre des guillemets doubles (`"`). Les différents attributs de la même balise doivent être séparés par au moins une espace. Par exemple, il ne faut plus écrire:

```
<a href=http://google.fr title=Se rendre sur Google>Google</a>
```

Mais plutôt écrire:

```
<a href="http://google.fr" title="Se rendre sur Google">Google</a>
```

6. À tous les attributs utilisés doit être donnée une valeur, y compris ceux dont la valeur est unique. Par exemple, il ne faut plus écrire :

```
<input type="checkbox" checked disabled />
```

Désormais, il faut écrire:

```
<input type="checkbox" checked="checked" disabled="disabled" />
```

7. L'attribut « `name` » qui servaient à identifier certaines balises (`<a>`, `<form>`, par exemple) est supprimé et doit être remplacé par l'attribut « `id` ».

Maintenant que sont fixées les règles d'écriture du code html, nous pouvons commencer à écrire le contenu de nos pages web.

## 4.4 Structures d'une page web

Voici le code source correspondant à la structure de base d'une page web:

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <link rel="stylesheet" href="code.css" />
    <title>titre de la page web</title>
  </head>
  <body>
    Ecrire votre code HTML ici
  </body>
</html>
```

Vous noterez que les balises s'ouvrent et se ferment dans un ordre précis. Par exemple, la balise `<html>` est la première que l'on ouvre et c'est aussi la dernière que l'on ferme (tout à la fin du code, avec `</html>`).

Essayons d'apporter plus d'explication à ce code.

### 4.4.1 La balise `<!DOCTYPE html>`

Il s'agit d'une information sur la version du langage HTML que vous utilisez. Durant cet ouvrage, nous utiliserons le langage HTML en version 5 qui succède au langage XHTML. C'est la seule balise, avec celle de commentaire que nous verrons tout à l'heure, qui ne respecte pas les règles d'écriture des balises que nous venons de découvrir. Ne soyez donc pas surpris en particulier par la présence de majuscules dans cette balise. Considérez que c'est « l'exception qui confirme la règle ».

### 4.4.2 La balise `<html></html>`

C'est la balise principale du code. Tout les autres balises devront se trouver entre `<html>` et `</html>`. Il n'y a aucune balise en dehors hormis la balise spéciale `<!DOCTYPE>` que l'on vient de voir.

### 4.4.3 La balise `<head></head>`

La balise `<head>` contient des informations d'en-tête qui ne sont généralement pas affichées sur la page. On y trouvera en particulier toujours :

- Le titre de la page : `<title>`. Le titre d'une page web est affiché dans la barre de titre du navigateur comme nous le verrons plus loin. Choisissez-le avec soin car il a beaucoup d'importance pour les moteurs de recherche qui référencent les pages web, comme Google. Toute page doit avoir un titre qui décrit ce qu'elle contient. Il est conseillé de garder le titre assez court (moins de 100 caractères en général) ;
- La balise `<link />`: C'est dans cette balise que vous indiquerez le chemin des fichiers à inclure dans votre page web comme les fichier `.css` ou `.js` .

- La balise `<meta />` indique l'encodage utilisé dans votre fichier .html . L'encodage indique la façon dont le fichier est enregistré. C'est lui qui détermine comment les caractères spéciaux vont s'afficher (accents, idéogrammes chinois et japonais, caractères arabes, etc.). Il y a plusieurs techniques d'encodage portant des noms bizarres et utilisées en fonction des langues : **ISO-8859-1**, **OEM 775**, **Windows-1253**... Une seule cependant devrait être utilisée aujourd'hui autant que possible : **UTF-8**. Cette méthode d'encodage permet d'afficher sans aucun problème pratiquement tous les symboles de toutes les langues de notre planète ! C'est pour cela que nous avons indiqué **utf-8** dans cette balise.

Il faut aussi que votre fichier soit bien enregistré en UTF-8. C'est le cas le plus souvent sous Linux par défaut mais, sous Windows, il faut généralement le dire au logiciel. Pour définir l'encodage en UTF-8 dans Notepad++, vous pouvez faire ceci :

- Allez dans le menu Format > Encoder en UTF-8 (sans BOM) pour que votre fichier soit enregistré en UTF-8 dès le début. Cela ne s'applique qu'au fichier actuellement ouvert ;
- Pour ne pas avoir à le faire pour chaque nouveau fichier, je vous conseille d'aller dans le menu Paramétrage > Préférences, onglet Nouveau document/Répertoire, puis sélectionnez UTF-8 sans BOM dans le cadre Encodage.

#### 4.4.4 Le corps de la page `<body></body>`

C'est là que se trouve la partie principale de la page. Tout ce que nous écrirons ici sera affiché à l'écran. C'est à l'intérieur du corps que nous écrirons la majeure partie de notre code.

#### 4.4.5 `<!-- - LES COMMENTAIRES - -->`

Tout ce qui est contenu entre les symboles `<!--` et `-->` est considéré par le navigateur comme des commentaires et n'est pas affiché dans la page, même s'ils se trouvent dans l'élément `<body>`.

Exemple:

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8"/>
    <link rel="stylesheet" href="code.css" />
    <title>titre de la page web</title>
  </head>
  <body>
    <!-- Cette balise sert à mettre un fichier
         en téléchargement sur la page web -->
    <a href="fichier.pdf" title="Télécharger le fichier">Download</a>
  </body>
</html>
```

Comme pour tout langage de programmation, nous avons tout intérêt à commenter le code HTML afin d'en faciliter la relecture a posteriori. Notez toutefois que les commentaires seront visibles par l'internaute si celui-ci choisit d'afficher le code source de la page dans son navigateur. Évitez donc d'y inclure des informations confidentielles et d'y faire figurer des informations privées.

Vous savez déjà créer une page web, c'est bien. Apprendre à rédiger son contenu, c'est mieux.



# Chapter 5

## RÉDIGER LE CONTENU DE SA PAGE WEB

Après avoir défini la code source minimal d'une page web, voyons ce que nous pourrions mettre comme contenu dans la balise `<body></body>`.

### 5.1 Organiser le texte

Nous verrons successivement dans cette partie :

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

#### 5.1.1 Les paragraphes

Le langage HTML propose la balise `<p></p>` pour délimiter les paragraphes.

Exemple:

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8"/>
    <link rel="stylesheet" href="code.css" />
    <title>titre de la page web</title>
  </head>
  <body>
    <!-- balise paragraphe -->
    <p>Bonjour et bienvenu à l'ISTIC !</p>
  </body>
</html>
```

- `<p>` signifie Début du paragraphe

- `</p>` signifie Fin du paragraphe

A l'intérieur de cette balise vous pouvez insérer d'autres balises qui vous permettront de traiter votre texte à votre guise. On peut citer :

- `<br />` : qui sert à indiquer qu'on voudrait aller à la ligne ;
- `<strong></strong>` : qui sert à mettre en valeur un ou plusieurs mots ;
- `<i></i>` : mettre le mot en italique ;
- `<mark></mark>` : pour marquer un mot ou un texte ; etc.

Exemple:

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8"/>
    <link rel="stylesheet" href="code.css" />
    <title>titre de la page web</title>
  </head>
  <body>
    <!-- balise paragraphe -->
    <p>
      Bonjour et bienvenu à l'ISTIC !<br/>
      Je suis encore étudiant et débutant dans
      la création des sites web alors svp
      <strong>soyez indulgents</strong> j'apprends
      petit à petit comment on crée un site web.
    </p>
    <p>
      Dans quelques semaines, je deviendrai un
      <i><strong>GRAND DEVELOPPEUR</strong></i>.
    </p>
  </body>
</html>
```

### 5.1.2 Les titres

Lorsque le contenu de votre page va s'étoffer avec de nombreux paragraphes, il va devenir difficile pour vos visiteurs de se repérer. C'est là que les titres deviennent utiles. En HTML, on est verni, on a le droit d'utiliser six niveaux de titres différents. Je veux dire par là qu'on peut dire Ceci est un titre très important , Ceci est un titre un peu moins important , Ceci est un titre encore moins important , etc. On a donc six balises de titres différentes :

- `<h1> </h1>`: signifie « titre très important ». En général, on s'en sert pour afficher le titre de la page au début de celle-ci.
- `<h2> </h2>`: signifie « titre important ».
- `<h3> </h3>`: pareil, c'est un titre un peu moins important (on peut dire un « sous-titre » si vous voulez).
- `<h4> </h4>`: titre encore moins important.

- `<h5>` `</h5>`: titre pas important.
- `<h6>` `</h6>`: titre vraiment, mais alors là vraiment pas important du tout.

Voici un exemple concret d'utilisation:

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8"/>
    <link rel="stylesheet" href="code.css" />
    <title>titre de la page web</title>
  </head>
  <body>
    <!-- balise titre -->
    <h1>SITE WEB DE L'ISTIC</h1>
    <p>
      Bonjour et bienvenu à l'ISTIC !<br/>
      Je suis encore étudiant et débutant dans
      la création des sites web alors svp
      <strong>soyez indulgents</strong>.<br/>
      Voici ce que j'apprends:
    </p>
    <h2>HTML 5</h2>
    <p>
      Pour apprendre à rédiger le contenu
      d'un site web.
    </p>
    <h3>CSS 3</h3>
    <p>
      Il va servir à la mise en forme de notre
      site.
    </p>
  </body>
</html>
```

Dans la pratique n'utilisez qu'au plus les balises `<h1>`, `<h2>`, `<h3>`.

### 5.1.3 Les listes

Les listes nous permettent souvent de mieux structurer notre texte et d'ordonner nos informations. Nous allons découvrir ici deux types de listes : les listes non ordonnées ou listes à puces ; les listes ordonnées ou listes numérotées ou encore énumérations.

#### Les listes non ordonnées

- Milt 1
- Milt 2
- Trad 1

C'est un système qui nous permet de créer une liste d'éléments sans notion d'ordre (il n'y a pas de premier ni de dernier ). Créer une liste non ordonnée est très simple. Il suffit d'utiliser la balise `<ul>` que l'on referme un peu plus loin avec `</ul>` en de cette manière:

```

<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8"/>
    <link rel="stylesheet" href="code.css" />
    <title>titre de la page web</title>
  </head>
  <body>
    <!-- balise titre -->
    <h1>SITE WEB DE L'ISTIC</h1>
    <p>
      Bonjour et bienvenu à l'ISTIC !<br/>
    </p>
    <p>
      <h2>Les filières ouvertes à l'ISTIC</h2>
      <ul>
        <li>Milt 1</li>
        <li>Milt 2</li>
        <li>Trad 1</li>
      </ul>
    </p>
  </body>
</html>

```

## Les listes ordonnées

Une liste ordonnée fonctionne de la même façon, seule une balise change : il faut remplacer `<ul></ul>` par `<ol></ol>`. À l'intérieur de la liste, on ne change rien : on utilise toujours des balises `<li></li>` pour délimiter les éléments.

```

<p>
  <h2>Les filières ouvertes à l'ISTIC</h2>
  <ol>
    <li>Milt 1</li>
    <li>Milt 2</li>
    <li>Trad 1</li>
  </ol>
</p>

```

## 5.2 Insérer des images

On n'envisage plus aujourd'hui un site sans images, les sites grand public se doivent donc d'égayer leurs pages avec des illustrations photographiques ou simplement graphiques. L'abus d'images pouvant se révéler aussi nocif que leur absence en termes d'attractivité du site pour le visiteur, il appartient au concepteur d'effectuer un choix judicieux de ses illustrations. De plus, il faut encore tenir compte du poids des images en kilo-octets, Le poids des images est sans commune mesure avec celui du code d'une page HTML et vient rapidement allonger le temps d'affichage complet d'une page si elle en contient beaucoup.

### 5.2.1 Les types d'images

Les navigateurs actuels n'acceptent qu'un nombre restreint de types d'images et il faudra vous limiter aux trois grands types les plus utilisés et enregistrer vos images selon les formats présentés ci-après, qui sont suffisants pour satisfaire tous les besoins d'un concepteur de sites:

- **GIF: Graphics Interface Format**, ayant pour extension « .gif ». Ce format est limité à 256 couleurs et est donc déconseillé pour les photographies ayant une grande plage de teintes différentes. On l'utilisera en priorité pour des icônes, des dessins ou des bandeaux publicitaires car il présente aussi l'avantage de permettre la création de petites animations et l'entrelacement qui va permettre l'affichage progressif de l'image, d'abord en basse résolution puis, au fur et à mesure du chargement, à la résolution maximale.
- **JPEG: Joint Photographic Experts Group**, ayant pour extensions « .jpeg » ou « .jpg ». Ce format permet la création d'images en 24 bits (16 millions de couleurs), et est donc très adapté aux photographies réalistes. En contrepartie toutefois, les images JPEG ont habituellement un poids plus important en Ko, ce qui ralentit leur chargement.
- **PNG: Portable Network Graphics**, ayant pour extension « .png ». Ce format est assez récent et a été conçu comme alternative au format GIF qui faisait l'objet de droits d'auteurs, les inventeurs de ce format pouvant vous réclamer des royalties. Le format PNG créé à l'initiative du W3C est donc libre de droit et permet normalement la création de graphiques et de photographies. Comme il est de création récente, les vieux navigateurs ne le reconnaissent pas, mais comme ils sont désormais très minoritaires, vous pouvez l'employer sans trop de crainte. Le PNG existe en deux versions, en fonction du nombre de couleurs que doit comporter l'image:
  - PNG 8 bits : 256 couleurs;
  - PNG 24 bits : 16 millions de couleurs (autant qu'une image JPEG).

A première vue, une image GIF ou PNG aura un poids plus faible qu'une image JPEG; cependant, c'est la variété des teintes de l'image (la palette des couleurs) qui va influencer le poids final de l'image. Quand vous créez une image, essayez donc systématiquement de l'enregistrer sous les trois formats. Vous serez parfois surpris par le poids réel des images obtenues et vous pourrez choisir la moins lourde, donc celle qui s'affichera le plus vite dans votre page. Ne vous fiez pas non plus au temps d'affichage des images dans votre navigateur lors de vos tests, car le fichier provient directement de votre disque dur dans le navigateur, et ce temps de chargement n'a rien à voir avec celui que les visiteurs du site auront à subir si votre image « pèse » 100 Ko par exemple.

Pour faire bonne mesure, diminuez la définition des photos numériques pouvant atteindre une taille de 2 Mo. De plus, on estime généralement qu'après 5 secondes d'attente, un visiteur moyen commence à s'impatisser. Testez donc vos pages en ligne pour vous mettre dans la situation de vos visiteurs. Si on résume, voici quel format adopter en fonction de l'image que vous avez:

- Une photo : utilisez un JPEG.
- N'importe quel graphique avec peu de couleurs (moins de 256) : utilisez un PNG 8 bits ou éventuellement un GIF.
- N'importe quel graphique avec beaucoup de couleurs : utilisez un PNG 24 bits.
- Une image animée : utilisez un GIF animé.

## 5.2.2 Les erreurs à éviter

- Bannissez les autres formats

Les autres formats non cités ici, comme le format BITMAP (\*.bmp) sont à bannir car bien souvent ils ne sont pas compressés, donc trop gros. Ils ne sont pas du tout adaptés au Web. On peut en mettre sur son site mais le chargement sera vraiment extrêmement long !

- Choisissez bien le nom de votre image

Si vous voulez éviter des problèmes, prenez l'habitude d'enregistrer vos fichiers avec des noms en minuscules, sans espace ni accent, par exemple : mon\_image.png. Vous pouvez remplacer les espaces par le caractère underscore (« \_ ») comme je l'ai fait ici.

## 5.2.3 Insertion d'images

La balise orpheline `<img />` permet d'inclure des images dans une page web. La balise doit être accompagnée de deux attributs obligatoires :

- **src**: il permet d'indiquer où se trouve l'image que l'on veut insérer. Vous pouvez soit mettre un chemin absolu (Ex : <http://www.site.com/fleur.png>), soit mettre le chemin en relatif (ce qu'on fait le plus souvent). Ainsi, si votre image est dans un sous-dossier images, vous devrez taper : `src="images/fleur.png"`
- **alt**: cela signifie « texte alternatif ». On doit toujours indiquer un texte alternatif à l'image, c'est-à-dire un court texte qui décrit ce que contient l'image. Ce texte sera affiché à la place de l'image si celle-ci ne peut pas être téléchargée (cela arrive), ou dans les navigateurs de personnes handicapées (non-voyants) qui ne peuvent malheureusement pas « voir » l'image. Cela aide aussi les robots des moteurs de recherche pour les recherches d'images. Pour la fleur, on mettrait par exemple : `alt="Une fleur"`.

Les images doivent se trouver obligatoirement à l'intérieur d'un paragraphe (`<p></p>`). Voici un exemple d'insertion d'image:

```
<!-- insertion d'image -->
<p>
  Quelques photos de l'ISTIC:<br/>
  <!-- image au chemin absolu -->
  

  <!-- image au chemin relatif à notre page web
  (L'image et notre page web sont dans un même dossier) -->
  

  <!-- image au chemin relatif à notre page web
  (L'image se trouve dans le dossier "fichiers") -->
  
</p>
```

## Ajouter une infobulle

Une info bulle vous permet d'avoir une courte information sur un élément. Pour voir l'info bulle apparaître pour une photo, survolez la photo avec la souris. Survolez la photo avec la souris. L'attribut permettant d'afficher une bulle d'aide est le même que pour les liens: il s'agit de « **title** ». Cet attribut est facultatif (contrairement à **alt**). Voici ce que cela peut donner :

```

<!-- insertion d'image -->
<p>
  Une photo prise de l'ISTIC:<br/>
  <!-- image au chemin relatif à notre page web
       (l'image se trouve dans le dossier "fichiers") -->
  
</p>

```

## Miniature cliquable

Si votre image est très grosse, il est conseillé d'en afficher la miniature sur votre site. Ajoutez ensuite un lien sur cette miniature pour que vos visiteurs puissent afficher l'image en taille originale.

Nous allons ainsi disposer de deux versions de ma photo: la miniature et l'image d'origine. Nous les plaçons toutes les deux dans un dossier appelé par exemple « fichiers ». J'affiche la version « istic\_mini.jpg » sur ma page et je fais un lien vers « istic\_ori.jpg » pour que l'image agrandie s'affiche lorsqu'on clique sur la miniature. Voici le code HTML que je vais utiliser pour cela :

```

<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8"/>
    <link rel="stylesheet" href="code.css" />
    <title>ISTIC</title>
  </head>
  <body>
    <!-- insertion d'image -->
    <p>
      Une photo prise de l'ISTIC:<br/>
      <a href="fichiers/istic_ori.jpg">
        
      </a><br/>
      Cliquez sur la photo pour l'agrandir
    </p>
  </body>
</html>

```

**NB:** Si une image illustre le texte (et n'est pas seulement décorative), il est conseillé de la placer au sein d'une balise `<figure>`.

En HTML, les figures sont des éléments qui viennent enrichir le texte pour compléter les informations de la page. Les figures peuvent être de différents types:

- images ;
- codes source ;
- citations ;
- etc.

Bref, tout ce qui vient illustrer le texte est une figure. Nous allons ici nous intéresser aux images mais, contrairement à ce qu'on pourrait croire, les figures ne sont pas forcément des images : un code source aussi illustre le texte.

## Création d'une figure

En HTML 5, voici comment vous pourrez utiliser la balise `<figure>`:

```
<body>
  <!-- Image illustrante -->
  <p>
    <figure>
      
    </figure>
  </p>
</body>
```

Une figure est le plus souvent accompagnée d'une légende. Pour ajouter une légende, utilisez la balise `<figcaption>` à l'intérieur de la balise `<figure>`, comme ceci :

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8"/>
    <link rel="stylesheet" href="code.css" />
    <title>ISTIC</title>
  </head>
  <body>
    <!-- Image illustrante -->
    <p>
      <figure>
        
        <figcaption>Une Classe de l'ISTIC</figcaption>
      </figure>
    </p>
  </body>
</html>
```

## 5.3 Créer des liens

Dans l'acronyme HTML, on peut considérer que le mot le plus important est le mot « Hypertext ». En effet, la création de documents contenant des liens hypertextes est la particularité qui a rendu le Web si populaire.

Les liens hypertextes permettent de passer, d'un simple clic, sur un mot ou sur une image, d'une page à l'autre, qu'elle soit située sur le même serveur ou en n'importe quel point du réseau. Nous parlerons dans ce cas de « lien externe » (sous-entendu au document initial). Il est également possible de passer instantanément d'un point de la page à un autre situé dans la même page et identifié par un nom particulier. Nous parlerons alors de « lien interne » (encore sous-entendu au document initial). Tout contenu essentiel d'une page HTML peut faire l'objet d'un lien hypertexte externe ou interne. Si l'origine d'un lien est très souvent un clic sur une partie d'un texte, nous pourrions tout aussi bien définir celle-ci sur une image, une partie donnée d'une image, un bouton ou encore créer cette redirection.

Il est facile de reconnaître les liens sur une page : ils sont écrits d'une façon différente (par défaut, en bleu et soulignés) et un curseur en forme de main apparaît lorsqu'on pointe dessus.





La balise `<a></a>` permet de créer des liens:

- Vers un autre site;
- Vers une autre page ;
- Relatifs et absolues.

### 5.3.1 Un lien vers un autre site

```
<body>
  <!-- Les liens en HTML -->
  <p>
    Bienvenue à l'ISTIC. Vous pouvez visiter son site web
    <a href="http://www.istic.cm">ICI</a>
  </p>
</body>
```

L'attribut « `href` » (hypertext reference) est obligatoire, puisqu'il indique l'adresse de la page à afficher lors d'un clic sur ce lien. Il est très utile de faire figurer l'attribut « `accesskey` »: les personnes handicapées pourront activer le lien par un appui simultané sur la touche « `Alt` » et la lettre indiquée entre guillemets. L'attribut « `title` » permet d'afficher un texte automatiquement dans une bulle au passage de la souris sur le lien.

```
<body>
  <!-- Les liens en HTML -->
  <p>
    Bienvenue à l'ISTIC. Vous pouvez visiter son site web
    <a href="http://www.istic.cm" accesskey="c" title="Site web de l'ISTIC" >ICI</a>
  </p>
</body>
```

Vous pourrez ajouter d'autres attributs comme:

- « `tabindex` » :il définit un ordre d'accès au lien avec une tabulation, ce qui rend le lien actif si la page en comporte plusieurs. En écrivant le code suivant, il faudra que l'utilisateur tape deux fois sur la touche `Tab` pour rendre le lien actif, la redirection vers la page cible se faisant ensuite avec la touche Entrée.

### 5.3.2 Lien vers une autre page du site

Ce type de lien est généralement utilisé entre les différentes pages qui composent un site web. Pour créer un lien vers sur une autre page, vous devez spécifier son emplacement dans un chemin relatif (Par rapport à la page d'origine) comme dans un chemin absolue (Depuis la racine d'un lecteur). Voici un exemple :

```
<body>
  <!-- Les liens en HTML -->
  <p>
    Bienvenue à l'ISTIC. Vous pouvez visiter son site web
    <!-- lien avec un chemin relatif (les 2 pages sont dans le dossier ) -->
    <a href="m2milt.html" accesskey="c" title="Site web de l'ISTIC" >ICI</a>
  </p>
</body>
```

### 5.3.3 Un lien vers une autre endroit de la même page (les Ancres)

Quand le contenu d'une page est volumineux, l'utilisateur ne peut pas en avoir une vision globale. Une ancre est une sorte de point de repère que vous pouvez mettre dans vos pages HTML lorsqu'elles sont très longues.

En effet, il peut alors être utile de faire un lien amenant plus bas dans la même page pour que le visiteur puisse sauter directement à la partie qui l'intéresse.

Pour créer une ancre, il suffit de rajouter l'attribut `id` à une balise qui va alors servir de repère. Ce peut être n'importe quelle balise, un titre par exemple. Utilisez l'attribut « `id` » pour donner un nom à l'ancre (Pour marquer un endroit de la page). Cela nous servira ensuite pour faire un lien vers cette ancre. Ensuite, il suffit de créer un lien comme d'habitude, mais cette fois l'attribut « `href` » contiendra un dièse (`#`) suivi du nom de l'ancre. Exemple:

```
<!-- Les liens en HTML -->
<p>
    Bienvenue à l'ISTIC. Vous aller apprendre le langage HTML et le CSS
    <!-- lien avec un chemin relatif (les 2 pages sont dans le dossier ) -->
    <a id="#html" >Le HTML 5</a>
    Un document HTML doit toujours avoir une structure de base.<br/>
    Le HTML sert à rédiger le contenu d'une page web.
</p>
<p>
    <a id="#css">Le CSS 3</a>
    Une feuille de style comporte des propriétés CSS.<br/>
    Le CSS sert à effectuer la mise en forme d'une page web.
</p>
<p><a href="#html" accesskey="c" title="Paragraphe sur HTML">Vers le paragraphe sur HTML</a></p>
<p><a href="#css" accesskey="c" title="Paragraphe sur CSS">Vers le paragraphe sur CSS</a></p>
<!-- Si l'ancre se trouve sur une autre page (Exemple istic2.html ) -->
<p><a href="istic2.html#php" accesskey="c" title="Paragraphe sur PHP">Apprendre le PHP</a></p>
```

### 5.3.4 Cas pratique de liens

- Un lien qui affiche une info bulle au survol de la souris:

```
<p>
    <a href="istic.cm" accesskey="c"
        title="Site de l'ISTIC PHP">ISTIC</a>
</p>
```

- Un lien qui ouvre une nouvelle fenêtre ou un nouvel onglet:

```
<p>
    <a href="istic.cm" accesskey="c" target="_blanc"
        title="Site de l'ISTIC PHP">ISTIC</a>
</p>
```

**NB:** Notez cependant qu'il est déconseillé d'abuser de cette technique car elle perturbe la navigation. Le visiteur lui-même peut décider s'il veut ouvrir le lien dans une nouvelle fenêtre. Il fera « **Maj + Clic** » sur le lien pour ouvrir dans une nouvelle fenêtre ou « **Ctrl + Clic** » pour ouvrir dans un nouvel onglet.

- Un lien pour envoyer un e-mail:

```

<p>
  <!-- la valeur "_blanc" de target permettra d'ouvrir
       le lien sur un autre onglet -->
  <a href="mailto:istic@yahoo.fr" accesskey="c" target="_blanc"
     title="Site de l'ISTIC PHP">ISTIC</a>
</p>

```

- Un lien pour télécharger un fichier:

```

<p>
  <!-- lien de téléchargement d'un fichier -->
  <a href="cour_creation_site_web" accesskey="c" target="_blanc"
     title="Site de l'ISTIC PHP">ISTIC</a>
</p>

```

## 5.4 Les tableaux

Dans une page HTML, il est possible de réaliser une présentation d'informations de type textuelle ou graphique sous la forme de tableaux. De la même manière que dans un tableur Excel ou OpenOffice, un tableau créé en HTML est composé de lignes et de colonnes. Cette partie du chapitre sera consacrée à l'insertion des tableaux dans vos pages web.

En HTML, un tableau est délimité par les balises `<table> ... </table>`.

Initialement, les bordures du tableau sont invisibles, donc il faut ajouter dans la balise `<table>` l'attribut « `border` » pour indiquer l'épaisseur de la bordure en pixels.

Également inclus dans la balise `<table>`, les attributs « `cellspacing` » et « `cellpadding` » permettent de préciser en pixels, respectivement l'espacement entre cellules (traits d'encadrement simples si 0, doubles sinon) et les marges intérieures des cellules.

Le tableau se construit ligne par ligne, avec les balises `<tr> ... </tr>` qui signifie table row en anglais, c'est-à-dire rangée du tableau.

Les cellules du tableau sont définies à l'intérieur de chaque ligne, grâce aux balises `<td> ... </td>` qui signifie table data, soit donnée du tableau.

Lorsqu'il s'agit d'une cellule de titre, il est possible d'utiliser à la place les balises `<th> ... </th>` qui signifie table header ou en-tête du tableau : le texte est alors mis en gras et centré.

Les balises `<caption> ... </caption>` sont facultatives : placées sous la balise de début de tableau `<table>`, elles permettent de définir un titre associé au tableau, qui s'écrit par défaut au-dessus de celui-ci.

Pour tout résumer voici comment est structuré un tableau en HTML:

<table>		
<tr>	<th>...</th>	<th>...</th> </tr>
<tr>	<td>...</td>	<td>...</td> </tr>
<tr>	<td>...</td>	<td>...</td> </tr>
</table>		

Voici une illustration de la construction d'un tableau dans une page web:

```

<body>
  <!-- Création d'un tableau -->
  <table border="1" cellspacing="0">
    <caption>Les formations de l'ISTIC</caption>
    <tr>
      <th>Filère</th>
      <th>Prix</th>
      <th>Durée formation</th>
    </tr>
    <tr>
      <td>MILT</td>
      <td>1 000 000</td>
      <td>2 ans</td>
    </tr>
    <tr>
      <td>TRAD</td>
      <td>750 000</td>
      <td>2 ans</td>
    </tr>
    <tr>
      <td>CHINOIS</td>
      <td>750 000</td>
      <td>3 ans</td>
    </tr>
  </table>
</body>

```

### 5.4.1 Les attributs de la balise `<table>`

La balise `<table>` possède l'ensemble des attributs communs, les plus couramment utilisés étant « `title` », qui permet d'expliciter son contenu général, et « `class` » pour lui appliquer des styles, et bien sûr l'attribut `id` pour l'identifier de manière unique. Il possède également les attributs suivants, permettant d'affiner son organisation ou sa présentation par défaut :

- `border="N"` : il définit la largeur des bordures externes et internes qui délimitent le tableau et les cellules. Sa valeur s'exprime exclusivement en nombre de pixels et la valeur 0 est admise pour cacher ses bordures. L'unité px, pour pixel, ne doit pas être indiquée car elle est implicite. Nous pourrions par la suite affiner la taille des bordures du tableau, les lignes et des cellules à l'aide de styles CSS (voir le chapitre 6).
- `width="N px | N%"` : il définit la largeur totale du tableau dans la page. La possibilité de définir une largeur relative en pourcentage est très pratique pour adapter le tableau à l'écran du visiteur.
- `cellpadding="N px | N%"` : il définit la largeur de l'espacement entre le contenu d'une cellule et sa bordure.
- `cellspacing="N px | N%"` : il définit l'espacement entre les bordures de chaque cellule. Pour que cet attribut soit pris en compte, il faut que l'attribut « `border` » ait été par ailleurs défini avec une valeur non nulle. Si vous omettez de définir une valeur pour l'attribut « `cellspacing` », sa valeur est la même que celle de `border`.
- `frame` : il permet de diversifier l'affichage des bordures externes du tableau en n'affichant qu'une partie d'entre elles. Il peut prendre les valeurs suivantes :
  - `void` : supprime toutes les bordures extérieures ;

- **above**: ne laisse subsister que la bordure supérieure ;
  - **below**: ne laisse que la bordure inférieure ;
  - **lhs**: ne laisse que la bordure gauche ;
  - **rhs**: ne laisse que la bordure droite ;
  - **hsides**: ne laisse que les bordures horizontales haute et basse ;
  - **vsides**: ne laisse que les bordures verticales gauche et droite ;
  - **box** ou **border**: ne laisse que les quatre bordures externes.
- **rules**: il définit l’affichage des bordures internes du tableau, situées entre les cellules. Il prend les valeurs suivantes:
    - **none**: supprime toutes les bordures internes ;
    - **rows**: ne laisse que les bordures horizontales ;
    - **cols**: ne laisse que les bordures verticales ;
    - **all**: affiche toutes les bordures. C’est le comportement par défaut ;
    - **groups**: n’affiche que les bordures délimitant les groupes de lignes ou de colonnes définis par les balises `<col/>` et `<colgroup>`, `<tbody>`, `<thead>` et `<tfoot>`, dont nous allons voir les rôles respectifs plus loin.
  - **summary**: il contient un texte décrivant plus complètement le contenu du tableau que l’attribut « **title** ». De plus, ce texte n’est pas utilisé dans les navigateurs visuels habituels, mais uniquement dans les navigateurs oraux destinés aux non-voyants. Vous pourrez pour votre apprentissage tester ces attributs pour mieux cerner leur rôle.

Exemple:

```
<!-- Création d'un tableau -->
<table border="8" cellspacing="2px" cellpadding="10">
  <caption><strong>Les formations de l'ISTIC</strong></caption>
  <tr>
    <th>Filère</th>
    <th>Prix</th>
    <th>Durée formation</th>
  </tr>
  <tr>
    <td>MILT</td>
    <td>1 000 000</td>
    <td>2 ans</td>
  </tr>
  <tr>
    <td>TRAD</td>
    <td>750 000</td>
    <td>2 ans</td>
  </tr>
  <tr>
    <td>CHINOIS</td>
    <td>750 000</td>
    <td>3 ans</td>
  </tr>
</table>
```

### 5.4.2 Fusionner les cellules d'un tableau

Dans certains tableaux complexes, vous aurez besoin de « fusionner » des cellules entre elles. Un exemple de fusion ? Regardez le tableau à la figure suivante, qui dresse une liste de jeux vidéo pour filles et garçons :

Nom du jeu	Pour les filles ?	Pour les garçons ?
Resident Evil	Trop terrifiant	Oui
Les Sims	Oui	Trop ringard
Super Mario Kart	Pour tout le monde !	



Cellules fusionnées

Pour le dernier jeu, vous voyez que les cellules ont été fusionnées : elles ne font plus qu'une. C'est exactement l'effet qu'on cherche à obtenir.

Pour effectuer une fusion, on rajoute un attribut à la balise `<td>`. Il faut savoir qu'il existe deux types de fusion :

- **La fusion de colonnes** : c'est ce que nous venons de voir dans cet exemple. La fusion s'effectue horizontalement. On utilisera l'attribut « **colspan** ».
- **La fusion de lignes** : là, deux lignes seront groupées entre elles. La fusion s'effectuera verticalement. On utilisera l'attribut « **rowspan** ».

Comme vous le savez, vous devez donner une valeur à l'attribut (que ce soit **colspan** ou **rowspan**). Il faut indiquer le nombre de cellules à fusionner entre elles. Dans notre exemple, nous avons fusionné deux (02) cellules : celle de la colonne « Pour les garçons ? » et celle de « Pour les filles ? ».

On devra donc écrire :

```

<!-- Création d'un tableau -->
<table border="1" cellspacing="0">
  <caption><strong>Jeux vidéos pour le filles et les garçons</strong></caption>
  <tr>
    <th>Nom du jeu</th>
    <th>Pour les filles ?</th>
    <th>Pour les garçons ?</th>
  </tr>
  <tr>
    <td>Résident Evil</td>
    <td>Trop térifiant</td>
    <td>Oui</td>
  </tr>
  <tr>
    <td>Les Sims</td>
    <td>Oui</td>
    <td>Trop ringard</td>
  </tr>
  <tr>
    <td>Super Mario Kart</td>
    <!-- fusion de 2 cellules -->
    <td colspan="2">Pour tout le monde</td>
  </tr>
</table>

```

Si vous voulez, vous pouvez inverser l'ordre de votre tableau, au lieu de mettre les titres en haut, vous pouvez les placer à gauche. C'est une autre façon de voir le tableau: au lieu de le construire en hauteur, on peut le construire en longueur. Dans ce cas, le **colspan** n'est plus adapté, c'est un **rowspan** qu'il faut utiliser:

```

<!-- Création d'un tableau -->
<table border="1" cellspacing="0">
  <caption><strong>Jeux vidéos pour le filles et les garçons</strong></caption>
  <tr>
    <th>Nom du jeu</th>
    <td>Résident Evil</td>
    <td>Les Sims</td>
    <td>Super Mario Kart</td>
  </tr>
  <tr>
    <th>Pour les filles ?</th>
    <td>Trop térifiant</td>
    <td>Oui</td>
    <!-- fusion de 2 cellules -->
    <td rowspan="2">Pour tout le monde</td>
  </tr>
  <tr>
    <th>Pour les garçons ?</th>
    <td>Oui</td>
    <td>Trop ringard</td>
  </tr>
</table>

```

Résultat : les cellules sont fusionnées verticalement (figure suivante) !

Jeux vidéos pour le filles et les garçons			
Nom du jeu	Résident Evil	Les Sims	Super Mario Kart
Pour les filles ?	Trop térifiant	Oui	Pour tout le monde
Pour les garçons ?	Oui	Trop ringard	

### 5.4.3 Structurer un tableau

Si votre tableau est particulièrement gros, il est recommandé de le diviser en plusieurs parties. Cela rend la structure du tableau plus claire pour le navigateur, et votre code sera plus lisible par la même occasion. On peut créer jusqu'à trois parties dans un tableau:

- **Un en-tête de tableau:** on le place à l'intérieur d'une balise `<thead></thead>`.
- **Un corps de tableau:** c'est la partie centrale, que l'on place à l'intérieur de `<tbody></tbody>`.
- **Un pied de tableau:** situé tout en bas du tableau, le pied se place à l'intérieur de `<tfoot></tfoot>`.

Le schéma suivant en est une parfaite illustration de la structuration d'un tableau:

Passagers du vol 377				
En-tête du tableau	Nom	Age	Pays	<thead>
	Carmen	33 ans	Espagne	
	Michelle	26 ans	Etats-Unis	
Corps du tableau	François	43 ans	France	<tbody>
	Martine	34 ans	France	
	Jonathan	13 ans	Australie	
	Xu	19 ans	Chine	
Pied du tableau	Nom	Age	Pays	<tfoot>

Un tableau découpé en plusieurs parties

Exemple:



```

<table border="1" cellspacing="0">
  <thead> <!-- En-tête du tableau -->
    <tr>
      <th>Pays</th>
      <th>Capitale</th>
    </tr>
  </thead>
  <tfoot> <!-- Pied du tableau -->
    <tr>
      <th>Pays</th>
      <th>Capitale</th>
    </tr>
  </tfoot>
  <tbody> <!-- Corps du tableau -->
    <tr>
      <td>France</td>
      <td>Paris</td>
    </tr>
    <tr>
      <td>Espagne</td>
      <td>Madrid</td>
    </tr>
    <tr>
      <td>Royaume-Uni</td>
      <td>Londres</td>
    </tr>
  </tbody>
</table>

```

## 5.5 Les formulaires

S'inscrire dans une mailing-list, laisser son avis dans un livre d'or, saisir un mot-clé dans un moteur de recherche ou passer une commande en ligne, toutes ces actions aujourd'hui devenues très courantes pour nombre d'entre nous mais ne sont possibles que grâce à l'existence des formulaires insérés dans une page web.

Tout échange de données entre un visiteur (le poste client) et l'ordinateur hébergeant le site (le serveur), opéré via le protocole HTTP, se fait donc via les saisies effectuées dans un formulaire.

Dans cette partie du cours, vous apprendrez à créer des formulaires tels que vous les voyez sur de nombreux sites web sur Internet.

### 5.5.1 Créer un formulaire

La balise principale pour créer un formulaire est la balise `<form>` `</form>`. Elle permet d'en indiquer le début et la fin.

```
<!-- Creation d'un formulaire -->
<form>
|   <p>Contenu du formulaire</p>
</form>
```

Une fois qu'un visiteur aura rempli le formulaire, il cliquera sur le bouton « Envoyer ». Mais où ces informations vont-elles être envoyées et par quel moyen ? Pour indiquer cela, il faut ajouter impérativement deux attributs à la balise `<form>`:

- **method**: cet attribut permet d'indiquer la méthode d'envoi des données. C'est un concept un peu abstrait quand on ne connaît pas des langages de script comme le **PHP** ou le **Java EE**. Quoi qu'il en soit, voici les deux méthodes possibles:
  - **method="get"**: les informations seront transmises dans l'adresse de la page. Le visiteur verra ainsi les informations qu'il a rentrées, transiter dans sa barre d'adresse comme ceci: `http://www.site.com/page.php?nom=dupontprenom=jean` . Cette méthode d'envoi est toutefois limitée à 255 caractères, ce qui fait qu'on l'utilise assez rarement avec les formulaires.
  - **method="post"**: c'est la méthode la plus adaptée aux formulaires. Cette fois, le visiteur ne verra pas transiter les informations qu'il a rentrées dans sa barre d'adresse (ce qui peut paraître plus propre ). De plus, le principal avantage de cette méthode est qu'elle permet d'envoyer beaucoup plus d'informations à la fois.
- **action**: indique l'adresse de la page vers laquelle doit être redirigé le visiteur une fois qu'il a cliqué sur le bouton « Envoyer ». Cette page doit être une page de script et donc comporter par exemple l'extension `.php` ou `.jsp` , mais pas l'extension `.html` .

Cela nous donne donc un code qui ressemble à ceci:

```
<!-- Creation d'un formulaire -->
<form method="post" action="/traitementFormulaire" >
|   <p>Contenu du formulaire</p>
</form>
```

Tour à tour, nous découvrirons en HTML, comment insérer des zones de texte (Monolignes, multilignes, mot de passe), les zones de saisies enrichies (E-mail, téléphone, url, date, recherches), les éléments d'option (Cases à cocher, zones d'option, liste déroulante) et les boutons.

## 5.5.2 Les zones de texte

Il faut savoir qu'il y a deux zones de texte différentes:

- **La zone de texte monoligne**: comme son nom l'indique, on ne peut y écrire qu'une seule ligne. Elle sert à saisir des textes courts, par exemple un pseudo;
- **La zone de texte multiligne**: cette zone de texte permet d'écrire une quantité importante de texte sur plusieurs lignes, par exemple une dissertation sur l'utilité du HTML dans le développement des pays d'Asie du Sud-Est (ce n'est qu'une suggestion hein...).

## Zone de texte monoligne

Votre pseudo :  Une zone de texte monoligne

Pour créer une zone de texte à une ligne, on doit écrire:

```
<!-- Code HTML -->
<form method="post" action="traitement">
  <p>
    <label for="pseudo">Votre pseudo</label> :
    <input type="text" name="pseudo" id="pseudo" />
  </p>
</form>
```

Explication:

- `<label></label>`: pour le libeller de la zone de texte;
- `<input></input>`: pour insérer le type zone et ses attributs;
- L'attribut `name`: pour donner un nom au type de zone (Notre zone de texte s'appellera `pseudo`);
- Les attributs `id` et `for`: pour créer un lien entre le type de zone et le label. Ces deux attributs doivent toujours avoir la même valeur.

## Quelques attributs supplémentaires

On peut ajouter un certain nombre d'autres attributs à la balise `<input />` pour personnaliser son fonctionnement:

- On peut agrandir le champ avec `size`;
- On peut limiter le nombre de caractères que l'on peut saisir avec `maxlength`;
- On peut pré-remplir le champ avec une valeur par défaut à l'aide de `value`;
- On peut donner une indication sur le contenu du champ avec `placeholder`. Cette indication disparaîtra dès que le visiteur aura cliqué à l'intérieur du champ.

Dans l'exemple suivant, la zone de texte contient une indication permettant de comprendre ce qu'il faut saisir; le champ fait 30 caractères de long mais on ne peut écrire que 10 caractères maximum à l'intérieur:

```

<!-- Code HTML -->
<form method="post" action="traitement">
  <p>
    <label for="pseudo">Votre pseudo :</label>
    <input type="text" name="pseudo" id="pseudo"
      placeholder="Ex: KWETTE" size="30" maxlength="10" />
  </p>
</form>

```

Résultat :

Votre pseudo :

### 5.5.3 Zone de mot de passe

Vous pouvez facilement faire en sorte que la zone de texte se comporte comme une « zone de mot de passe », c'est-à-dire une zone où on ne voit pas à l'écran les caractères saisis. Pour créer ce type de zone de saisie, utilisez l'attribut `type="password"`.

Complétons notre formulaire. Demandons maintenant au visiteur son pseudo et son mot de passe:

```

<!-- Code HTML -->
<form method="post" action="traitement">
  <p>
    <!-- Zone de texte -->
    <label for="pseudo">Votre pseudo :</label>
    <input type="text" name="pseudo" id="pseudo"
      placeholder="Ex: KWETTE" size="30" maxlength="20" /><br />
    <!-- Zone de mot de passe -->
    <label for="pass">Votre mot de passe :</label>
    <input type="password" name="pass" id="pass" />
  </p>
</form>

```

Résultat :

Votre pseudo :

Votre mot de passe :

### 5.5.4 Zone de texte multiligne

Pour créer une zone de texte multiligne, on change de balise : nous allons utiliser `<textarea>` `</textarea>`. Comme pour tout autre élément du formulaire, il faut lui donner un nom avec « `name` » et utiliser un label qui explique de quoi il s'agit.

```

<!-- Code html -->
<form method="post" action="traitement.php">
  <p>
    <label for="avis">Votre avis sur l'université</label><br/>
    <textarea id="avis" name="avis" size=""></textarea>
  </p>
</form>

```

On peut modifier la taille du `<textarea>` de deux façons différentes:

- **En CSS:** il suffit d'appliquer les propriétés CSS `width` et `height` au `<textarea>`.
- **Avec des attributs:** on peut ajouter les attributs `rows` et `cols` à la balise `<textarea>`. Le premier indique le nombre de lignes de texte qui peuvent être affichées simultanément, et le second le nombre de colonnes.

```
<!-- Code html -->
<form method="post" action="traitement.php">
  <p>
    <label for="avis">Votre avis sur l'université</label><br/>
    <textarea id="avis" name="avis" rows="10" cols="40">
      Je pense que c'est la meilleur Université d'Afrique !
    </textarea>
  </p>
</form>
```

Résultat:

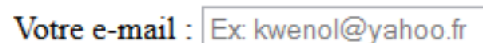


### 5.5.5 Zone pour l'email

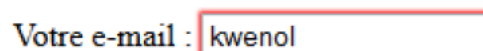
Vous pouvez demander à saisir une adresse e-mail:

```
<!-- Code HTML -->
<form method="post" action="traitement">
  <p>
    <!-- Zone pour e-mail -->
    <label for="mail">Votre e-mail :</label>
    <input type="email" name="mail" id="mail"
      placeholder="Ex: kwenol@yahoo.fr" /><br />
  </p>
</form>
```

Résultat:



Attention certains browsers peuvent afficher une indication si l'adresse n'est pas un e-mail, c'est ce que fait Firefox par exemple (figure suivante):



### 5.5.6 Zone pour url d'un site web

Avec le type url, on peut demander à saisir une adresse absolue (commençant généralement par http://):

```
<!-- Code HTML -->
<form method="post" action="traitement">
  <p>
    <!-- Zone pour site web -->
    <label for="site">Avez-vous un site web :</label>
    <input type="url" name="site" id="site"
      placeholder="Ex: http://:www.cfpa.cm" /><br />
  </p>
</form>
```

Résultat:

Avez-vous un site web :

### 5.5.7 Numéro de téléphone

Ce champ est dédié à la saisie de numéros de téléphone:

```
<!-- Code HTML -->
<form method="post" action="traitement">
  <p>
    <!-- Numero de telephone -->
    <label for="numTel">Téléphone :</label>
    <input type="tel" name="numTel" id="numTel"
      placeholder="( +237 ) 6 99 30 68 30" /><br />
  </p>
</form>
```

Résultat:

Téléphone :

### 5.5.8 Zone de recherche

On peut créer un champ de recherche comme ceci:

```

<!-- Code HTML -->
<form method="post" action="traitement">
  <p>
    <!-- Zone de recherche -->
    <label for="find">Recherche :</label>
    <input type="search" name="find" id="find" /><br />
  </p>
</form>

```

### 5.5.9 Nombre

Ce champ permet de saisir un nombre entier:

```

<!-- Code html -->
<input type="number"/>

```

Le champ s'affichera en général avec des petites flèches pour changer la valeur (figure suivante).

5

▲

▼

Champ de saisie de nombre

Vous pouvez personnaliser le fonctionnement du champ avec les attributs suivants:

- **min**: valeur minimale autorisée.
- **max**: valeur maximale autorisée.
- **step**: c'est le « pas » de déplacement. Si vous indiquez un pas de 2, le champ n'acceptera que des valeurs de 2 en 2 (par exemple 0, 2, 4, 6...).

### 5.5.10 Date

Différents types de champs de sélection de date existent:

- **date**: pour la date (05/08/1985 par exemple) ;
- **time**: pour l'heure (13:37 par exemple) ;
- **week**: pour la semaine ;
- **month**: pour le mois ;
- **datetime**: pour la date et l'heure (avec gestion du décalage horaire) ;
- **datetime-local**: pour la date et l'heure (sans gestion du décalage horaire).

Vous pourrez adapter votre choix en fonction de vos besoin mais sachez qu'à part Opéra, peu de navigateurs gèrent ce type de champ.

### 5.5.11 Les éléments d'option

HTML 5 vous offre une ribambelle d'éléments d'options à utiliser dans votre formulaire. Ce sont des éléments qui demandent au visiteur de faire un choix parmi une liste de possibilités. Nous allons passer en revue:

- les cases à cocher;
- les zones d'options;
- les listes déroulantes.

#### Les cases à cocher

Pour créer des cases à cocher, Nous allons réutiliser la balise `<input />`, en spécifiant cette fois le `type="checkbox"`:

```
<!-- Code HTML -->
<form method="post" action="traitement">
  <p>
    Cochez les cours que vous faites :<br />
    <input type="checkbox" name="infog" id="infog" />
    <label for="infog">INFOGRAPHIE</label><br />
    <input type="checkbox" name="jour" id="jour" />
    <label for="jour">JOURNALISME</label><br />
    <input type="checkbox" name="real" id="real" />
    <label for="real">REALISATEUR</label><br />
    <input type="checkbox" name="3d" id="3d" /> <label
    for="3d">MONTAGE 3D</label>
  </p>
</form>
```

Résultat:

Cochez les cours que vous faites :

- ☒ INFOGRAPHIE
- ☐ JOURNALISME
- ☐ REALISATEUR
- ☒ MONTAGE 3D

N'oubliez pas de donner un nom différent à chaque case à cocher, cela vous permettra d'identifier plus tard lesquelles ont été cochées par le visiteur. Enfin, sachez que vous pouvez faire en sorte qu'une case soit cochée par défaut avec l'attribut `checked="checked"`:

```
Cochez les cours que vous faites :<br />
<input type="checkbox" name="infog" id="infog" checked />
```



## Les zones d'option

Les zones d'options vous permettent de faire un choix (et un seul) parmi une liste de possibilités. Elles ressemblent un peu aux cases à cocher mais il y a une petite difficulté supplémentaire : elles doivent être organisées en groupes. Les options d'un même groupe possèdent le même nom (name), mais chaque option doit avoir une valeur (value) différente. La balise à utiliser est toujours un `<input />`, avec cette fois la valeur radio pour l'attribut type. Les choses seront plus claires sur l'exemple ci-dessous:

```
<!-- Code HTML -->
<form method="post" action="traitement">
  <p>
    Veuillez indiquer la tranche d'âge dans laquelle vous vous
    situez :<br />
    <input type="radio" name="age" value="moins15" id="moins15"
    /> <label for="moins15">Moins de 15 ans</label><br />
    <input type="radio" name="age" value="medium15-25"
    id="medium15-25" /> <label for="medium15-25">15-25 ans</label><br />
    <input type="radio" name="age" value="medium25-40"
    id="medium25-40" /> <label for="medium25-40">25-40 ans</label><br />
    <input type="radio" name="age" value="plus40" id="plus40" checked/>
    <label for="plus40">Encore plus vieux que ça ?!</label>
  </p>
</form>
```

Résultat:

Veuillez indiquer la tranche d'âge dans laquelle vous vous situez :

- ☐ Moins de 15 ans
- ☐ 15-25 ans
- ☐ 25-40 ans
- ☒ Encore plus vieux que ça ?!

## Les listes déroulantes

Les listes déroulantes sont un autre moyen élégant de faire un choix parmi plusieurs possibilités. Le fonctionnement est un peu différent. On va utiliser la balise `<select> </select>` qui indique le début et la fin de la liste déroulante. On ajoute l'attribut `name` à la balise pour donner un nom à la liste. Puis, à l'intérieur du `<select> </select>`, nous allons placer plusieurs balises `<option> </option>` (une par choix possible). On ajoute à chacune d'elles un attribut `value` pour pouvoir identifier ce que le visiteur a choisi.

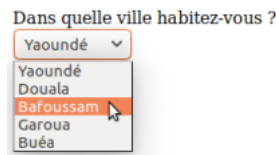
Voici un exemple d'utilisation:

```

<!-- Code html -->
<form method="post" action="traitement.php">
  <p>
    <label for="ville">Dans quelle ville habitez-vous ?</label><br/>
    <select id="ville" name="ville">
      <option value="yaounde">Yaoundé</option>
      <option value="douala">Douala</option>
      <option value="bafoussam">Bafoussam</option>
      <option value="garoua">Garoua</option>
      <option value="buea">Buéa</option>
    </select>
  </p>
</form>

```

Résultat:



Si vous voulez qu'une option soit sélectionnée par défaut, utilisez cette fois l'attribut **selected**:

```

<option value="bafoussam" selected="selected">Bafoussam</option>

```

Vous pouvez aussi grouper vos options avec la balise `<optgroup>` `</optgroup>`. Dans notre exemple, pourquoi ne pas séparer les villes en fonction de leur région ?

```

<!-- Code html -->
<form method="post" action="traitement.php">
  <p>
    <label for="ville">Dans quelle ville habitez-vous ?</label><br/>
    <select id="ville" name="ville">
      <optgroup label="Centre">
        <option value="yaounde">Yaoundé</option>
        <option value="mbyo">Mbal Mayo</option>
      </optgroup>
      <optgroup label="Ouest">
        <option value="bafoussam" selected="selected">Bafoussam</option>
        <option value="bandjoun">Bandjoun</option>
      </optgroup>
      <optgroup label="Extrême-Nord">
        <option value="touboro">Touboro</option>
        <option value="kaele">Kaélé</option>
      </optgroup>
    </select>
  </p>
</form>

```

### 5.5.12 Finaliser et envoyer le formulaire

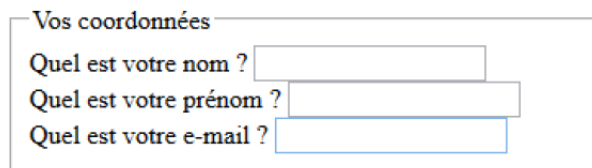
Nous y sommes presque. Il ne nous reste plus qu'à agrémenter notre formulaire de quelques dernières fonctionnalités (comme la validation), puis nous pourrions ajouter le bouton d'envoi du formulaire.

## Regrouper les champs

Si votre formulaire grossit et comporte beaucoup de champs, il peut être utile de les regrouper au sein de plusieurs balises `<fieldset>`. Chaque `<fieldset>` peut contenir une légende avec la balise `<legend>`. Regardez cet exemple:

```
<!-- Code HTML -->
<form method="post" action="traitement">
  <p>
    <fieldset>
      <legend>Vos coordonnées</legend> <!-- Titre du fieldset -->
      <label for="nom">Quel est votre nom ?</label>
      <input type="text" name="nom" id="nom" /><br />
      <label for="prenom">Quel est votre prénom ?</label>
      <input type="text" name="prenom" id="prenom" /><br />
      <label for="email">Quel est votre e-mail ?</label>
      <input type="email" name="email" id="email" /><br />
    </fieldset>
  </p>
</form>
```

Résultat:



Vos coordonnées

Quel est votre nom ?

Quel est votre prénom ?

Quel est votre e-mail ?

## Sélectionner automatiquement un champ

Vous pouvez placer automatiquement le curseur dans l'un des champs de votre formulaire avec l'attribut `autofocus`. Dès que le visiteur chargera la page, le curseur se placera dans ce champ. Par exemple, pour que le curseur soit par défaut dans le champ prénom:

```
<label for="prenom" autofocus="autofocus">Quel est votre prénom ?</label>
```

## Rendre un champ obligatoire

Vous pouvez faire en sorte qu'un champ soit obligatoire en lui donnant l'attribut `required`.

```
<label for="nom" required="required">Quel est votre nom ?</label>
```

## Bouton d'envoi

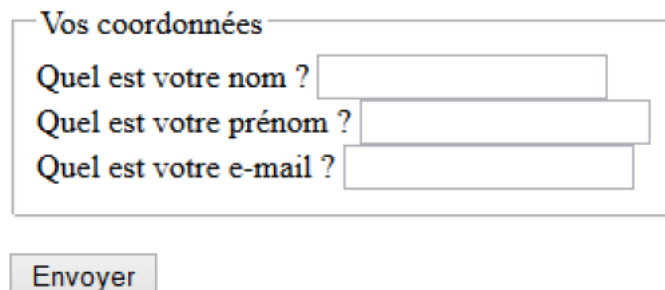
Il ne nous reste plus qu'à créer le bouton d'envoi. Là encore, la balise `<input />` vient à notre secours. Elle existe en quatre versions:

- `type="submit"`: le principal bouton d'envoi de formulaire. C'est celui que vous utiliserez le plus souvent. Le visiteur sera conduit à la page indiquée dans l'attribut `action` du formulaire;
- `type="reset"`: remise à zéro du formulaire ;
- `type="image"`: équivalent du bouton `submit`, présenté cette fois sous forme d'image. Rajoutez l'attribut `src` pour indiquer l'URL de l'image.
- `type="button"`: bouton générique, qui n'aura (par défaut) aucun effet. En général, ce bouton est géré en *JavaScript* pour exécuter des actions sur la page. Nous ne l'utiliserons pas ici.

Pour créer un bouton d'envoi on écrira donc par exemple:

```
<form method="post" action="traitement">
  <p>
    <fieldset>
      <legend>Vos coordonnées</legend> <!-- Titre du fieldset -->
      <label for="nom" required >Quel est votre nom ?</label>
      <input type="text" name="nom" id="nom" /><br />
      <label for="prenom" autofocus >Quel est votre prénom ?</label>
      <input type="text" name="prenom" id="prenom" /><br />
      <label for="email">Quel est votre e-mail ?</label>
      <input type="email" name="email" id="email" /><br />
    </fieldset>
  </p>
  <input type="submit" value="Envoyer" />
</form>
```

Résultat:



Lorsque vous cliquez sur le bouton « Envoyer », le formulaire vous amène alors à la page indiquée dans l'attribut `action`. Souvenez-vous, nous avons imaginé une page fictive : « *traitement.php* ». Le problème, c'est que vous ne pouvez pas créer cette page seulement en HTML. Il est nécessaire d'apprendre un nouveau langage, comme le « PHP », pour pouvoir récupérer les informations saisies et décider quoi en faire.

## 5.6 La vidéo et l'audio

Depuis l'arrivée de *Youtube* et *Dailymotion*, il est devenu courant aujourd'hui de regarder des vidéos sur des sites web. Il faut dire que l'arrivée du haut débit a aidé à démocratiser les vidéos sur le Web.

Cependant, aucune balise HTML ne permettait jusqu'ici de gérer la vidéo. Il fallait à la place utiliser un plugin, comme Flash. Encore aujourd'hui, Flash reste de loin le moyen le plus utilisé pour regarder des vidéos sur *Youtube*, *Dailymotion*, *Vimeo* et ailleurs. Mais utiliser un plugin a de nombreux défauts : on dépend de ceux qui gèrent le plugin (en l'occurrence, l'entreprise *Adobe*, qui possède *Flash*), on ne peut pas toujours contrôler son fonctionnement, il y a parfois des failles de sécurité ... Au final, c'est assez lourd.

C'est pour cela que deux nouvelles balises standard ont été créées en HTML 5 : `<video>` et `<audio>`!

### 5.6.1 Les formats audio et vidéo

Lorsque nous avons présenté les images et la balise `<img />`, nous avons commencé par un petit tour d'horizon des différents formats d'images (JPEG, PNG, GIF, etc.). Pour la vidéo et l'audio, nous allons faire pareil.

#### Les formats audio

Pour diffuser de la musique ou n'importe quel son, il existe de nombreux formats. La plupart d'entre eux sont compressés (comme le sont les images JPEG, PNG et GIF) ce qui permet de réduire leur poids:

- **MP3**: vous ne pouvez pas ne pas en avoir entendu parler ! C'est l'un des plus vieux, mais aussi l'un des plus compatibles (tous les appareils savent lire des MP3), ce qui fait qu'il est toujours très utilisé aujourd'hui.
- **AAC**: utilisé majoritairement par Apple sur iTunes, c'est un format de bonne qualité. Les iPod, iPhone et autres iPad savent les lire sans problème.
- **OGG**: le format Ogg Vorbis est très répandu dans le monde du logiciel libre, notamment sous Linux. Ce format a l'avantage d'être libre, c'est-à-dire qu'il n'est protégé par aucun brevet.
- **WAV (format non compressé)**: évitez autant que possible de l'utiliser car le fichier est très volumineux avec ce format. C'est un peu l'équivalent du Bitmap (BMP) pour l'audio.

Aucun navigateur ne gère tous ces formats à la fois. Retenez surtout la compatibilité pour les MP3 et OGG:

Navigateur	MP3	OGG
Google Chrome	Oui	Oui
Internet Explorer	Oui	-
Mozilla Firefox	-	Oui
Safari	Oui	-
Opera	-	Oui

Pour résoudre ce problème, il faudra proposer différents formats aux navigateurs qui sélectionneront alors celui qu'ils savent lire.

## Les formats vidéo

Le stockage de la vidéo est autrement plus complexe. On a besoin de trois éléments:

- **Un format conteneur:** c'est un peu comme une boîte qui va servir à contenir les deux éléments ci-dessous. On reconnaît en général le type de conteneur à l'extension du fichier: AVI, MP4, MKV...
- **Un codec audio:** c'est le format du son de la vidéo, généralement compressé. Nous venons de les voir, on utilise les mêmes: MP3, AAC, OGG...
- **Un codec vidéo:** c'est le format qui va compresser les images. C'est là que les choses se corsent, car ces formats sont complexes et on ne peut pas toujours les utiliser gratuitement. Les principaux à connaître pour le Web sont:
  - **H.264:** l'un des plus puissants et des plus utilisés aujourd'hui... mais il n'est pas 100% gratuit. En fait, on peut l'utiliser gratuitement dans certains cas (comme la diffusion de vidéos sur un site web personnel), mais il y a un flou juridique qui fait qu'il est risqué de l'utiliser à tout va.
  - **Ogg Theora:** un codec gratuit et libre de droits, mais moins puissant que H.264. Il est bien reconnu sous Linux mais, sous Windows, il faut installer des programmes pour pouvoir le lire.
  - **WebM:** un autre codec gratuit et libre de droits, plus récent. Proposé par Google, c'est le concurrent le plus sérieux de H.264 à l'heure actuelle.

Compatibilité des codecs vidéo sur les différents navigateurs:

Navigateur	H.264	Ogg Theora	WebM
Google Chrome	-	Oui	Oui
Internet Explorer	Oui	-	Oui
Mozilla Firefox	-	Oui	Oui
Safari	Oui	-	-
Opera	Oui	Oui	Oui

### 5.6.2 Insertion d'un élément audio

La balise `<audio>` que nous allons découvrir est reconnue par tous les navigateurs récents, y compris Internet Explorer à partir de la version 9 (IE9).

En théorie, il suffit d'une simple balise pour jouer un son sur notre page:

```
<!-- Code html -->  
<audio src="fichier.mp3"></audio>
```

En pratique, c'est un peu plus compliqué que cela. Si vous testez ce code... vous ne verrez rien ! En effet, le navigateur va seulement télécharger les informations générales sur le fichier (on parle de métadonnées) mais il ne se passera rien de particulier.

Vous pouvez compléter la balise des attributs suivants:

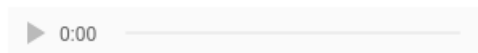
- **controls:** pour ajouter les boutons « Lecture », « Pause » et la barre de défilement. Cela peut sembler indispensable, et vous vous demandez peut-être pourquoi cela n'y figure pas par défaut, mais certains sites web préfèrent créer eux-mêmes leurs propres boutons et commander la lecture avec du JavaScript.

- **width**: pour modifier la largeur de l'outil de lecture audio.
- **loop**: la musique sera jouée en boucle.
- **autoplay**: la musique sera jouée dès le chargement de la page. Évitez d'en abuser, c'est en général irritant d'arriver sur un site qui joue de la musique tout seul !
- **preload**: indique si la musique peut être préchargée dès le chargement de la page ou non. Cet attribut peut prendre les valeurs:
  - **auto** (par défaut) : le navigateur décide s'il doit précharger toute la musique, uniquement les métadonnées ou rien du tout.
  - **metadata**: charge uniquement les métadonnées (durée, etc.).
  - **none**: pas de préchargement. Utile si vous ne voulez pas gaspiller de bande passante sur votre site.

Il faudra donc toujours ajouter les contrôles comme dans le code ci-dessous:

```
<!-- Code html -->
<audio controls>
  <source src="fichiers/Inspiration.mp3"></source>
  <source src="fichier.ogg"></source>
</audio>
```

Résultat:



### 5.6.3 Insertion d'un élément vidéo

Il suffit d'une simple balise `<video>` pour insérer une vidéo dans la page:

```
<!-- Code html pour ajouter une vidéo-->
<video src="fichiers/configuration-router-cisco.mp4" width="600"
  poster="fichiers/videos.png" controls>
  Si la vidéo n'est pas affichée mettez à jour le navigateur
</video>
```

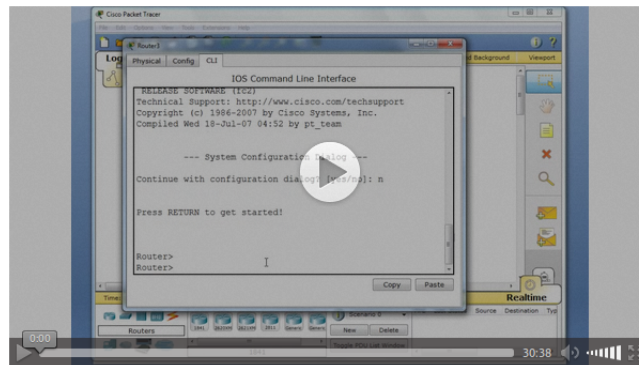
Voici les attributs que peut prendre la balise `<video>`:

- **poster**: image à afficher à la place de la vidéo tant que celle-ci n'est pas lancée. Par défaut, le navigateur prend la première image de la vidéo mais, comme il s'agit souvent d'une image noire ou d'une image peu représentative de la vidéo, je vous conseille d'en créer une ! Vous pouvez tout simplement faire une capture d'écran d'un moment de la vidéo.
- **controls**: pour ajouter les boutons « Lecture », « Pause » et la barre de défilement. Cela peut sembler indispensable, mais certains sites web préfèrent créer eux-mêmes leurs propres boutons et commander la lecture avec du JavaScript. En ce qui nous concerne, ce sera largement suffisant !
- **width**: pour modifier la largeur de la vidéo.



- **height**: pour modifier la hauteur de la vidéo.
- **loop**: la vidéo sera jouée en boucle.
- **autoplay**: la vidéo sera jouée dès le chargement de la page. Là encore, évitez d'en abuser, c'est en général irritant d'arriver sur un site qui lance quelque chose tout seul !
- **preload**: indique si la vidéo peut être préchargée dès le chargement de la page ou non. Cet attribut peut prendre les valeurs:
  - **auto** (par défaut): le navigateur décide s'il doit précharger toute la vidéo, uniquement les métadonnées ou rien du tout.
  - **metadata**: charge uniquement les métadonnées (durée, dimensions, etc.).
  - **none**: pas de préchargement. Utile si vous souhaitez éviter le gaspillage de bande passante sur votre site.

Résultat:



Vous utiliserez la balise `<source>` à l'intérieur de la balise `<video>` pour proposer différents formats. Le navigateur prendra celui qu'il reconnaît:

```
<!-- Code html pour ajouter une vidéo-->
<video width="600" poster="fichiers/videos.png" controls>
  <source src="fichiers/configuration-router-cisco.mp4">
  <source src="fichiers/configuration-router-cisco.webm">
  <source src="fichiers/configuration-router-cisco.ogv">
  Si la vidéo n'est pas affichée mettez à jour le navigateur
</video>
```

Les *iPhone*, *iPad* et *iPod* ne reconnaissent à l'heure actuelle que le format H.264 (fichier .mp4)... et uniquement si celui-ci apparaît en premier dans la liste ! Je vous recommande donc d'indiquer le format H.264 en premier pour assurer une compatibilité maximale.

Après avoir passé toute une première partie du cours à ne travailler que sur le HTML, nous allons maintenant découvrir le CSS qui vient le compléter pour vous aider à mettre en forme votre page web.



## Chapter 6

# ÉCRIRE LES FEUILLES DE STYLES AVEC LE CSS

A suivre ...

# Chapter 7

## LES ANNEXES

### 7.1 Liste des balises HTML 5

#### 7.1.1 Les balises de premier niveau

Les balises de premier niveau sont les principales balises pour réaliser le « code minimal » d'une page html.

BALISE		ATTRIBUT / DESCRIPTION	
<!-- Commentaire -->		Commentaire	
<!DOCTYPE html>		Déclaration du Doctype	
BALISE DE PREMIER NIVEAU – Code minimal d'une page web			
<html>		Indiquez au navigateur que le document est en HTML	
<head>		En-tête de la page	<body> Corps de la page

## 7.1.2 Les balises d'en-tête

Ces balises sont toutes situées dans l'en-tête de la page web, c'est-à-dire entre `<head>` et `</head>`:

BALISE D'EN-TÊTE – Entre les balises <code>&lt;head&gt;</code>			
<code>&lt;title&gt;</code>	Titre de la page, tres important pour le référencement		
<code>&lt;script&gt;</code>	Inserer script	<code>&lt;style&gt;</code>	Inserer CSS
<code>&lt;noscript&gt;</code>	Message à afficher si le script n'est pas toleré		
<code>&lt;base /&gt;</code>	URL par défaut <code>&lt;base href="http://41mag.fr/" target="_blank" /&gt;</code>		
<code>&lt;link /&gt;</code>	CSS	<code>&lt;link rel="stylesheet" type="text/css" href="#" /&gt;</code>	
	PAGE	<code>&lt;link rel="start" href="index.html" /&gt;</code>	
	RSS	<code>&lt;link rel="alternate" type="application/rss+xml" href="#" /&gt;</code>	
	FAVICON	<code>&lt;link rel="shortcut icon" type="image/x-icon" href="#" /&gt;</code>	
<code>&lt;meta /&gt;</code>	TITLE		Titre de la page
	DESCRIPTION		Description de la page
	KEYWORDS		Mots-clés de la page
	ROBOTS		Restrictions pour les robots
	SYNDICATION-SOURCE		Indique l'URL d'origine
	ORIGINAL-SOURCE		Indique que c'est l'original
	NOTRANSLATE		Ne sera pas traduit
	HTTP-EQUIV CHARSET		Jeux de caractères
	HTTP-EQUIV REFRESH		Raffraichir la page
	HTTP-EQUIV PRAGMA		Définit le cache du navig.

### 7.1.3 Les balises d'architecture

BALISE D'ARCHITECTURE	
<code>&lt;header&gt;</code>	Définit l'en-tête d'une section ou d'une page
<code>&lt;footer&gt;</code>	Définit le bas d'une section ou d'une page
<code>&lt;hgroup&gt;</code>	Définit les informations d'en-tête d'une section ou d'une page
<code>&lt;details&gt;</code>	Définit les détails d'un élément
<code>&lt;summary&gt;</code>	Définit l'en-tête des détails d'un élément
<code>&lt;menu&gt;</code>	Définit un menu en forme de liste
<code>&lt;section&gt;</code>	Définit une section
<code>&lt;article&gt;</code>	Définit un article
<code>&lt;aside&gt;</code>	Définit un élément latéral
<code>&lt;nav&gt;</code>	Définit un groupe de liens de navigations
<code>&lt;iframe&gt;</code>	Introduit une page html dans une frame
<code>&lt;div&gt;</code>	Calque ou section
<code>&lt;span&gt;</code>	Section de type inline

### 7.1.4 Les balises de structuration de texte

Ces balises permettent de construire le squelette de notre site web.

BALISE DE STRUCTURATION DE DE TEXTE			
<code>&lt;h1&gt;</code> à <code>&lt;h6&gt;</code>	Créer un titre		
<code>&lt;p&gt;</code>	Paragraphe	<code>&lt;a&gt;</code>	Lien
<code>&lt;strong&gt;</code>	Mise en exergue	<code>&lt;b&gt;</code>	Texte en gras
<code>&lt;em&gt;</code>	Italique en exergue	<code>&lt;i&gt;</code>	italique
<code>&lt;mark&gt;</code>	Marqueur de texte	<code>&lt;small&gt;</code>	Rétrécis le texte
<code>&lt;sub&gt;</code>	Mise en indice	<code>&lt;sup&gt;</code>	Mise en Exposant
<code>&lt;adress&gt;</code>	Définit une adresse	<code>&lt;cite&gt;</code>	Citation
<code>&lt;abbr&gt;</code>	Abréviation	<code>&lt;dfn&gt;</code>	Définition
<code>&lt;del&gt;</code>	Texte supprimé	<code>&lt;ins&gt;</code>	Texte ajouté
<code>&lt;time&gt;</code>	Date / Horaire	<code>&lt;meter&gt;</code>	Mesure
<code>&lt;code&gt;</code>	Portion de code	<code>&lt;pre&gt;</code>	Texte préformaté
<code>&lt;br /&gt;</code>	Saut de ligne	<code>&lt;wbr&gt;</code>	Empl pr saut 2 ligne
<code>&lt;blockquote&gt;</code>	Longue citation	<code>&lt;q&gt;</code>	Courte citation
<code>&lt;samp&gt;</code>	Echantillon	<code>&lt;var&gt;</code>	Variable
<code>&lt;kbd&gt;</code>	Raccourci clavier	<code>&lt;bdo&gt;</code>	Sens du texte

### 7.1.5 Les balises de liste

Ce sont des balises HTML permettant de créer des listes (listes à puces, listes numérotées, listes de définitions ...)

BALISE DE LISTE			
<code>&lt;ul&gt;</code>	Liste non-ordonné	<code>&lt;ol&gt;</code>	Liste ordonné
<code>&lt;li&gt;</code>	Élément de liste	<code>&lt;dl&gt;</code>	Liste de définition
<code>&lt;dt&gt;</code>	Terme à définir	<code>&lt;dd&gt;</code>	Définition du terme

### 7.1.6 Les balises de tableau

BALISE DE TABLEAU			
<code>&lt;table&gt;</code>	Tableau	<code>&lt;caption&gt;</code>	Titre du tableau
<code>&lt;thead&gt;</code>	En-tête du tableau	<code>&lt;tbody&gt;</code>	Corps du tableau
<code>&lt;tr&gt;</code>	Ligne du tableau	<code>&lt;th&gt;</code>	Cellule d'en-tête
<code>&lt;td&gt;</code>	Cellule du tableau	<code>&lt;tfoot&gt;</code>	Bas du tableau
<code>&lt;col&gt;</code>	Colonne du tableau	<code>&lt;colgroup&gt;</code>	Groupe de colonne

### 7.1.7 Les balises de formulaire

BALISE DE FORMULAIRE				
<form>	Formulaire		<fieldset>	Regroupe plusieurs éléments du formulaire
<legend>	Titre d'un groupe		<label>	Titre d'un élément
<datalist>	Liste déroulante		<select>	Liste sélectionnable
<option>	Élément d'une liste		<optgroup>	Grp d'éléments d'une list
<textarea>	Zone de texte		<keygen>	Génération d'une clé
<button>	Bouton cliquable		<command>	Bouton de commande
<output>	Définit un type de sortie			
<input />	button	file	radio	text
	checkbox	hidden	range	time
	color	image	reset	url
	date	month	search	
	datetime	number	submit	
	email	password	tel	

### 7.1.8 Les balises multimédia

BALISE MULTIMEDIA			
<code>&lt;area&gt;</code>	Zone cliquable à l'interieur d'une image		
<code>&lt;audio&gt;</code>	Contenu audio	<code>&lt;canvas&gt;</code>	Graphique
<code>&lt;img /&gt;</code>	Image ou photo	<code>&lt;progress&gt;</code>	Progression
<code>&lt;figure&gt;</code>	Groupe d'element multimedia	<code>&lt;figcaption&gt;</code>	Légende du groupe d'élément
<code>&lt;video&gt;</code>	Vidéo	<code>&lt;source&gt;</code>	Source du media
<code>&lt;map&gt;</code>	Carte / image	<code>&lt;param&gt;</code>	Parametre d'objet
<code>&lt;embed /&gt;</code>	Contenu exterieur	<code>&lt;object&gt;</code>	Objet du cont. ext

### 7.1.9 Les autres balises

AUTRE			
<code>&lt;rp&gt;</code>	Annotation pr le nav	<code>&lt;rt&gt;</code>	Explication ruby
<code>&lt;ruby&gt;</code>	Annotation ruby	<code>&lt;hr /&gt;</code>	Barre horizontale

### 7.1.10 Les attributs standard

ATTRIBUT STANDAR			
<b>Acceskey</b>	Raccourci clavier	<b>itemprop</b>	Utilisé pr un gp d'élémnt
<b>class</b>	Attribut une classe	<b>context-menu</b>	Menu contextuel d'elemnt
<b>lang</b>	Langage d'un élément	<b>spellcheck</b>	Correction automatique
<b>data-</b>	Définit un attribut	<b>style</b>	Applique un style
<b>dir</b>	Direction du texte	<b>subject</b>	Définit l'elmnt correspondt
<b>draggable</b>	Element deplacable	<b>tabindex</b>	Définit l'ordre d'un tabl0
<b>hidden</b>	Element caché	<b>title</b>	Titre de l'élément
<b>id</b>	Nomme un élément	<b>conteneditable</b>	Element editable
<b>item</b>	Utilisé pour un groupe d'élément		

### 7.1.11 Les attributs standard

EVENEMENT			
Pour la balise <body>	Pour formulaire	Pour la souris	Pour les medias
Onafterprint, onbeforeprint, onafterload, onblur, onerror, onfocus, onhaschange, onload, onmessage, onoffline, ononline, onpagehide, onpageshow, onpopstate, onredo, onresize, onstorage, onundo, onunload	Onblur, onchange, oncontextmenu, onfocus, onformchange, onforminput, oninput, oninvalid, onsubmit, onselect	Onclick, ondblclick, ondrag, ondragend, ondragenter, ondragleave, ondragover, ondragstart, ondrop, onmousedown, onmouseup, onmousemove, onmouseout, onmousewheel, onscroll	Onabort, oncanplay, oncanplaytought, ondurationchange, onemptied, onended, onerror, onloaddata, onloadstart, onpause, onplay, onplaying, onprogress, onseeked, onsuspend, onwaiting, onvolumechange



## 7.2 Liste des propriétés CSS

### 7.2.1 Les propriétés de mise en forme

Une mise en forme de texte représente tout ce qui touche à la présentation du texte proprement dit : le gras, l'italique, le souligné, la police, l'alignement, etc.

Propriété	Valeurs (exemples)	Description
<code>font-family</code>	<i>police1, police2, police3, serif, sans-serif, monospace</i>	Nom de police
<code>@font-face</code>	<i>Nom et source de la police</i>	Police personnalisée
<code>font-size</code>	1.3em, 16px, 120%...	Taille du texte
<code>font-weight</code>	bold, normal	Gras
<code>font-style</code>	italic, oblique, normal	Italique
<code>text-decoration</code>	underline, overline, line-through, blink, none	Soulignement, ligne au-dessus, barré ou clignotant
<code>font-variant</code>	small-caps, normal	Petites capitales
<code>text-transform</code>	capitalize, lowercase, uppercase	Capitales
<code>font</code>	-	Super propriété de police. Combine : <code>font-weight</code> , <code>font-style</code> , <code>font-size</code> , <code>font-variant</code> , <code>font-family</code> .
<code>text-align</code>	left, center, right, justify	Alignement horizontal
<code>vertical-align</code>	baseline, middle, sub, super, top, bottom	Alignement vertical (cellules de tableau ou éléments <code>inline-block</code> uniquement)
<code>line-height</code>	18px, 120%, normal...	Hauteur de ligne
<code>text-indent</code>	25px	Alinéa
<code>white-space</code>	pre, nowrap, normal	Césure
<code>word-wrap</code>	break-word, normal	Césure forcée
<code>text-shadow</code>	5px 5px 2px blue (horizontale, verticale, fondu, couleur)	Ombre de texte

## 7.2.2 Les propriétés de couleur et de fond

Propriété	Valeurs (exemples)	Description
<code>color</code>	<i>nom</i> , <code>rgb(rouge,vert,bleu)</code> , <code>rgba(rouge,vert,bleu,transparence)</code> , <code>#CF1A20...</code>	Couleur du texte
<code>background-color</code>	<i>Identique à color</i>	Couleur de fond
<code>background-image</code>	<code>url('image.png')</code>	Image de fond
<code>background-attachment</code>	<code>fixed</code> , <code>scroll</code>	Fond fixe
<code>background-repeat</code>	<code>repeat-x</code> , <code>repeat-y</code> , <code>no-repeat</code> , <code>repeat</code>	Répétition du fond
<code>background-position</code>	<i>(x y)</i> , <code>top</code> , <code>center</code> , <code>bottom</code> , <code>left</code> , <code>right</code>	Position du fond
<code>background</code>	-	Super propriété du fond. Combine : <code>background-image</code> , <code>background-repeat</code> , <code>background-attachment</code> , <code>background-position</code>
<code>opacity</code>	0.5	Transparence

### 7.2.3 Les propriétés de boîtes ou blocs

Propriété	Valeurs (exemples)	Description
<code>width</code>	150px, 80%...	Largeur
<code>height</code>	150px, 80%...	Hauteur
<code>min-width</code>	150px, 80%...	Largeur minimale
<code>max-width</code>	150px, 80%...	Largeur maximale
<code>min-height</code>	150px, 80%...	Hauteur minimale
<code>max-height</code>	150px, 80%...	Hauteur maximale
<code>margin-top</code>	23px	Marge en haut
<code>margin-left</code>	23px	Marge à gauche
<code>margin-right</code>	23px	Marge à droite
<code>margin-bottom</code>	23px	Marge en bas
<code>margin</code>	23px 5px 23px 5px ( <i>haut, droite, bas, gauche</i> )	Super-propriété de marge. Combine : <code>margin-top</code> , <code>margin-right</code> , <code>margin-bottom</code> , <code>margin-left</code> .

<code>padding-left</code>	23px	Marge intérieure à gauche
<code>padding-right</code>	23px	Marge intérieure à droite
<code>padding-bottom</code>	23px	Marge intérieure en bas
<code>padding-top</code>	23px	Marge intérieure en haut
<code>padding</code>	23px 5px 23px 5px (haut, droite, bas, gauche)	Super-propriété de marge intérieure. Combine : <code>padding-top</code> , <code>padding-right</code> , <code>padding-bottom</code> , <code>padding-left</code> .
<code>border-width</code>	3px	Épaisseur de bordure
<code>border-color</code>	nom, rgb(rouge,vert,bleu), rgba(rouge,vert,bleu,transparence), #CF1A20...	Couleur de bordure
<code>border-style</code>	solid, dotted, dashed, double, groove, ridge, inset, outset	Type de bordure
<code>border</code>	3px solid black	Super-propriété de bordure. Combine <code>border-width</code> , <code>border-color</code> , <code>border-style</code> . Existe aussi en version <code>border-top</code> , <code>border-right</code> , <code>border-bottom</code> , <code>border-left</code> .
<code>border-radius</code>	5px	Bordure arrondie
<code>box-shadow</code>	6px 6px 0px black (horizontale, verticale, fondu, couleur)	Ombre de boîte

## 7.2.4 Les propriétés des tableaux

Propriété	Valeurs (exemples)	Description
<code>border-collapse</code>	collapse, separate	Fusion des bordures
<code>empty-cells</code>	hide, show	Affichage des cellules vides
<code>caption-side</code>	bottom, top	Position du titre du tableau

## 7.2.5 Les propriétés de positionnement et d’affichage

Propriété	Valeurs (exemples)	Description
<code>display</code>	block, inline, inline-block, table, table-cell, none...	Type d'élément ( <code>block</code> , <code>inline</code> , <code>inline-block</code> , <code>none</code> ...)
<code>visibility</code>	visible, hidden	Visibilité
<code>clip</code>	rect (0px, 60px, 30px, 0px) <i>rect (haut, droite, bas, gauche)</i>	Affichage d'une partie de l'élément
<code>overflow</code>	auto, scroll, visible, hidden	Comportement en cas de dépassement
<code>float</code>	left, right, none	Flottant
<code>clear</code>	left, right, both, none	Arrêt d'un flottant
<code>position</code>	relative, absolute, static	Positionnement
<code>top</code>	20px	Position par rapport au haut
<code>bottom</code>	20px	Position par rapport au bas
<code>left</code>	20px	Position par rapport à la gauche
<code>right</code>	20px	Position par rapport à la droite
<code>z-index</code>	10	Ordre d'affichage en cas de superposition. La plus grande valeur est affichée par-dessus les autres.

## 7.2.6 Les propriétés des listes

Propriété	Valeurs (exemples)	Description
<code>list-style-type</code>	disc, circle, square, decimal, lower-roman, upper-roman, lower-alpha, upper-alpha, none	Type de liste
<code>list-style-position</code>	inside, outside	Position en retrait
<code>list-style-image</code>	url('puce.png')	Puce personnalisée
<code>list-style</code>	-	Super-propriété de liste. Combine <code>list-style-type</code> , <code>list-style-position</code> , <code>list-style-image</code> .

## 7.3 Documentation

Pour découvrir de nouvelles pistes pour poursuivre votre apprentissage de HTML et de CSS consulter les références: [1], [2], [3], [4], [5], [6].

# Bibliography

- [1] Matthieu NEBRA. *Apprenez à créer votre site web avec html5 et css3*. OpenClassroom, 2013.
- [2] Premiers pas en CSS et XHTML. *Francis DRAILLARD*. Eyrolles, 2008.
- [3] Bien développer pour le web 2.0. *Christophe PORTENEUVE*. Eyrolles, 2007.
- [4] 4<sup>e</sup> édition *Réussir un projet de site web*. *Christophe PORTENEUVE*. Eyrolles, 2006.
- [5] Ergonomie web. Pour des sites web efficaces. *Amélie BOUCHER*. Eyrolles, 2009.
- [6] HTML5 et CSS3 Cours et exercices corrigés. *Jean Engels*. Eyrolles, 2012.