

	Secondes SNT	Le web	Durée : 3 H
	Thème web	Fiche prof	
	Contenus	Notions juridiques / Hypertexte / Langages HTML et CSS / URL / Moteur de recherche : principes et usages / paramètre de sécurité d'un navigateur	

KIT STARTER

Activité 1 : Comment crée-t-on une page web ?



Question du professeur :

« Connaissez-vous des solutions pour créer une page web ? »

Le Web Introduction au HTML et au CSS

Nous allons nous intéresser à un acteur fondamental du développement web, le couple **HTML + CSS** (*Hyper Text Markup Language* et *Cascading Style Sheets*).

« Qu'est-ce que le HTML ? »

L'*Hypertext Markup Language*, généralement abrégé HTML, est le format de données conçu pour représenter les pages web. C'est un langage de balisage permettant d'écrire de l'hypertexte, d'où son nom. HTML permet également de structurer sémantiquement et de mettre en forme le contenu des pages, d'inclure des ressources multimédias, dont des images, des formulaires de saisie, et des programmes informatiques. Il permet de créer des documents interopérables avec des équipements très variés de manière conforme aux exigences de l'accessibilité du web. Il est souvent utilisé conjointement avec des langages de programmation (JavaScript) et des formats de présentation (feuilles de style en cascade).



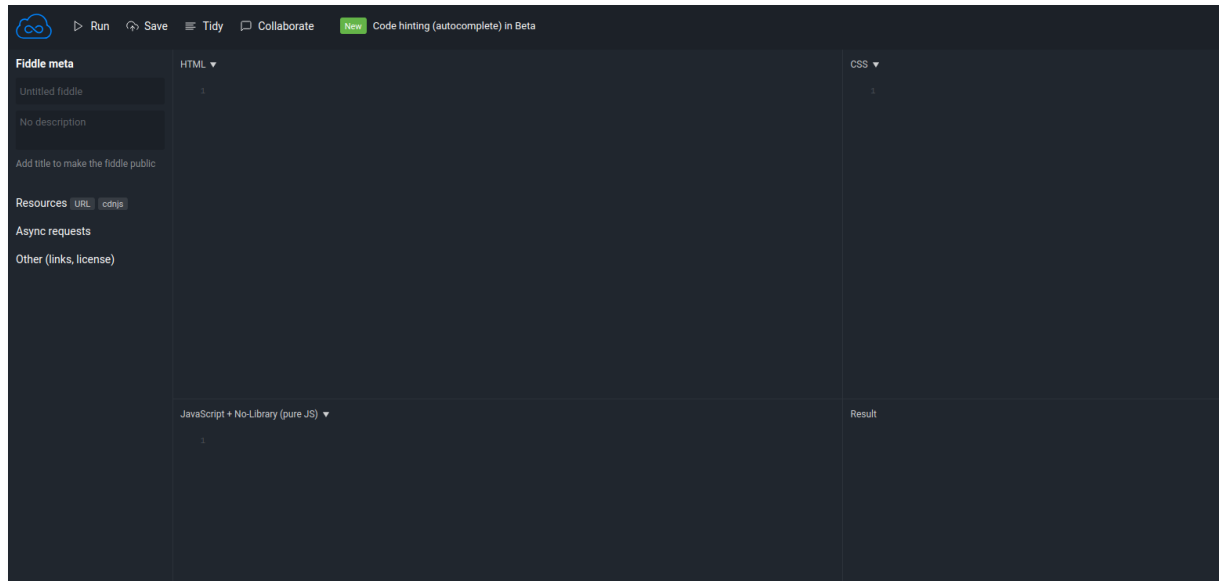
A retenir : Pour l'instant, nous allons retenir deux éléments de cette définition « **conçu pour représenter les pages web** » et « **un langage de balisage** ».

Grâce au HTML vous allez pouvoir, dans votre navigateur (Firefox, Chrome, Opera, ...), afficher du texte, afficher des images, proposer des hyperliens (liens vers d'autres pages web), afficher des formulaires et même maintenant afficher des vidéos (grâce à la dernière version du HTML, l'HTML5).

HTML n'est pas un langage de programmation (comme le Python par exemple), ici, pas question de conditions, de boucles...c'est un **langage de description**.

Le code source d'un site (celui de votre établissement par exemple) est diffusé au vidéoprojecteur, l'enseignant montre qu'il y a des liens nommés `<link>` et des images notées `<img>` qui sont intégrés dans cette page web.

Pour aborder le HTML, nous allons, dans un premier temps utiliser le site **jsfiddle.net**
Après avoir lancé votre navigateur web, tapez <http://jsfiddle.net/> dans la barre d'adresse.
Vous devriez voir apparaître ceci :



Nous allons pour l'instant uniquement utiliser la fenêtre « **HTML** » et la fenêtre « **Result** ».



« À faire vous-même 1 ? » :

Écrivez ou copiez/collez le code HTML suivant dans la fenêtre « HTML » :

```
<h1>Hello World! Ceci est un titre</h1>  
<p>Ceci est un <strong>paragraphe</strong>. Avez-vous bien compris ?</p>
```

Qu'est-ce qui s'affiche dans la fenêtre « Result », après avoir cliqué sur « ▷ Run » (Insérez la capture d'écran rognée du résultat et expliquez les balises) ? ou commenter avec les balises `<!-- et -->`

Hello World! Ceci est un titre

Ceci est un **paragraphe**. Avez-vous bien compris ?

Comme déjà évoqué ci-dessus, en HTML tout est une histoire de balise que l'on ouvre et que l'on ferme.

- Une **balise ouvrante** est de la forme `<nom_de_la_balise>`
- Les **balises fermantes** sont de la forme `</nom_de_la_balise>`

En observant attentivement le code, vous devriez forcément remarquer que toute balise ouverte doit être refermée à un moment ou un autre. La balise ouvrante et la balise fermante peuvent être sur la même ligne ou pas, cela n'a aucune importance, la seule question à se poser ici est : ***ai-je bien refermé toutes les balises que j'ai ouvertes ?***



A retenir : Une structure du type :

```
<balise1>  
<balise2>  
</balise1>  
</balise2>
```

est interdite, la *balise2* a été ouverte après la *balise1*, elle devra donc être refermée avant la *balise1*.

En revanche, l'enchaînement suivant est correct :

```
<balise1>  
<balise2>  
</balise2>  
</balise1>
```

Notez que dans notre exemple nous respectons bien cette règle « d'imbrication » des balises avec la balise `<p>` et la balise `<strong>`.

Il est important de comprendre que chaque balise a une signification qu'il faut bien respecter (on parle de la sémantique des balises).

Exemple de balises

- Par exemple le texte situé entre la balise ouvrante et fermante `<h1>` est obligatoirement un **titre important** (il existe des balises `<h2>`, `<h3>`.....qui sont aussi des titres, mais des titres moins importants (sous-titre)).
- La balise `<p>` permet de définir des **paragraphes**.
- La balise `<strong>` permet de mettre en évidence (**gras**) un élément important.



A retenir : Vous devez aussi savoir qu'il existe des balises qui sont à la fois ouvrantes et fermantes `<balise/>`

- La balise `<br/>` permet de sauter une ligne.

Les attributs

Il est possible d'ajouter des éléments à une balise ouvrante, on parle d'attributs. Une balise peut contenir plusieurs attributs :

```
<ma_balise attribut_1="valeur_1" attribut_2="valeur_2">
```

Il existe beaucoup d'attributs différents, nous allons nous contenter de 2 exemples :

- **attribut id** (id pour identifiant)
- **class**

Nous verrons l'intérêt de ces attributs dans l'activité suivante.



« À faire vous-même 2 ? » : Écrivez ou copiez/collez le code HTML :

```
<h1>Ceci est un titre</h1>
<h2 class="titre_1">Ceci est un sous titre</h2>
<p id="para_1">Ceci est un <strong>paragraphe</strong>. Avez-vous bien compris ?</p>
```

Qu'est-ce qui s'affiche dans la fenêtre « Result » (Insérez la capture d'écran rognée du résultat et expliquez les balises) ?

Ceci est un titre

Ceci est un sous titre

Ceci est un **paragraphe**. Avez-vous bien compris ?



A retenir : Le **HTML n'a pas été conçu pour gérer la mise en page** (c'est possible, mais c'est une mauvaise pratique). Le **HTML** s'occupe uniquement du **contenu et de la sémantique**, pour tout ce qui concerne la **mise en page et l'aspect « décoratif »** (on parle du « style » de la page), on utilisera le **CSS** (Cascading Style Sheets).

Dans **JSFIDDLE**, il est possible d'écrire du CSS dans la fenêtre en haut à gauche.



« À faire vous-même 3 ? » :

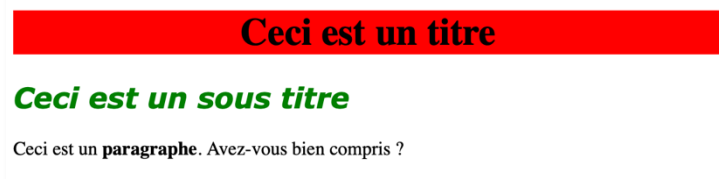
Écrivez ou copiez/collez le code HTML suivant :

```
<h1>Ceci est un titre</h1>
<h2>Ceci est un sous titre</h2>
<p>Ceci est un <strong>paragraphe</strong>. Avez-vous bien compris ?</p>
```

Écrivez ou copiez/collez le code CSS suivant dans la fenêtre « CSS » :

```
h1
{
    text-align: center;
    background-color: red;
}
h2
{
    font-family: Verdana;
    font-style: italic;
    color: green;
}
```

Qu'est-ce qui s'affiche dans la fenêtre « Result » (Insérez la capture d'écran rognée du résultat et expliquez les balises) ?



Dans l'exemple du « À faire vous-même 3 », les propriétés "text-align" et "background-color" seront appliquées au contenu de toutes les balises de type h1 (avec respectivement les valeurs "center" et "red")...



« À faire vous-même 4 ? » :

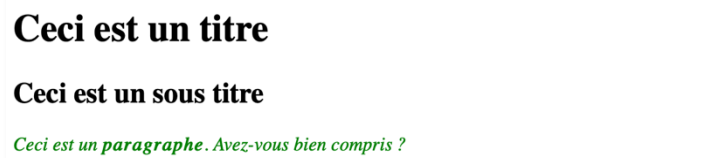
Écrivez ou copiez/collez le code HTML suivant :

```
<h1>Ceci est un titre</h1>
<h2>Ceci est un sous titre</h2>
<p id="para_1">Ceci est un <strong>paragraphe</strong>. Avez-vous bien compris ?</p>
```

Écrivez ou copiez/collez le code CSS suivant :

```
#para_1
{
    font-style: italic;
    color: green;
}
```

Qu'est-ce qui s'affiche dans la fenêtre « Result » (Insérez la capture d'écran rognée du résultat et expliquez les balises) ?
Que remarquez-vous ?



Il est donc possible de cibler un paragraphe et pas un autre en utilisant l'id du paragraphe (en CSS l'id se traduisant par le signe "#").

Il est aussi possible d'utiliser l'attribut class à la place de l'id.

Dans le CSS on utilisera le point "." à la place du "#".

La différence entre "id" et "class" n'est pas très importante.

- L'attribut "class" permet de donner le même nom à plusieurs reprises dans une même page.

Si nous avions eu un 3e paragraphe, nous aurions pu avoir :

```
<p class="para_1">Voici un 3e paragraphe</p>
```

mais nous n'aurions pas pu avoir :

```
<p id="para_1">Voici un 3e paragraphe</p>
```

car le nom "para_1" a déjà été utilisé pour le 1er paragraphe.



A retenir : **JSFIDDLE** est un très bel outil, mais il ne peut pas être utilisé pour la réalisation d'un vrai site internet (ou d'une vraie application web).

Nous allons créer 2 fichiers :

- Un fichier qui contiendra du **HTML** (index.html)
- Un fichier qui contiendra du **CSS** (style.css).



« À faire vous-même 5 ? » :

À l'aide d'un éditeur de texte :

			
https://notepad-plus-plus.org/downloads/v7.9.2/	Capytale sur atrium	Bloc-notes sous Windows	TextEdit sous Mac OS

Créez un nouveau fichier.

Sauvegardez-le en précisant son nom, par exemple "index.html".

Écrivez ou copiez/collez le code suivant dans votre éditeur de texte (sans oublier de sauvegarder quand vous avez terminé) :

```
<!doctype html>
<html lang="fr">
  <head>
    <meta charset="utf-8">
    <title>Voici mon site</title>
  </head>
  <body>
    <h1>Hello World! Ceci est un titre</h1>
    <p>Ceci est un <strong>paragraphe</strong>. Avez-vous bien compris ?</p>
  </body>
</html>
```

Testez votre code à l'aide d'un navigateur web (Firefox ou Chrome) en “double-cliquant” sur le fichier index.html

Qu'est-ce que vous obtenez (Insérez la capture d'écran rognée du résultat et expliquez les balises) ?

Hello World! Ceci est un titre

Ceci est un **paragraphe**. Avez-vous bien compris ?



A retenir : Dans l'exemple du "À faire vous-même 5", vous reconnaissez le code se trouvant entre les balises **<body>** :

<body>

.....

</body>

Tout votre code HTML devra se trouver entre ces 2 balises.

Le reste des balises devraient vous être inconnues. Passons-les en revue :

- La première ligne **<!doctype html>** permet d'indiquer au navigateur que nous utiliserons la dernière version du HTML, le fameux HTML5.
- La balise **<html>** est obligatoire,
- L'attribut **lang="fr"** permet d'indiquer au navigateur que nous utiliserons le français pour écrire notre page.
- Les balises **<head>...</head>** délimitent ce que l'on appelle l'en-tête.

L'en-tête contient, dans notre exemple, deux balises :

- la balise **<meta charset="utf-8">** qui permet de définir l'encodage des caractères (utf-8)
- La balise **<title>** qui définit le titre de la page (attention ce titre ne s'affiche pas dans le navigateur, ne pas confondre avec la balise **<h1>**).



« À faire vous-même 6 ? » :

Toujours à l'aide de l'éditeur de texte, vous allez créer un fichier qui va contenir le CSS de notre page (par exemple `style.css`).

Complétez ce fichier à l'aide du code suivant :

```
h1
{
    text-align: center;
    background-color: red;
}
p
{
    font-family: Verdana;
    font-style: italic;
    color: green;
}
```

Pour l'instant notre CSS ne sera pas appliqué à notre page, pour ce faire, il faut modifier notre code HTML en ajoutant une ligne qui va permettre d'associer notre code CSS à notre page.

Modifiez le code HTML avec la ligne suivante :

```
<link rel="stylesheet" href="style.css">
```

```
<!doctype html>
<html lang="fr">
  <head>
    <meta charset="utf-8">
    <title>Voici mon site</title>
    <link rel="stylesheet" href="style.css">
  </head>
  <body>
    <h1>Hello World! Ceci est un titre</h1>
    <p>Ceci est un <strong>paragraphe</strong>. Avez-vous bien compris ?</p>
  </body>
</html>
```

Testez votre code à l'aide d'un navigateur web en “double-cliquant” sur le fichier `index.html` :

Qu'est-ce que vous obtenez (Insérer la capture d'écran rognée du résultat et expliquez les balises) ?

Hello World! Ceci est un titre

Ceci est un **paragraphe**. Avez-vous bien compris ?

Dans l'exemple que nous venons de voir, les fichiers "*index.html*" et "*style.css*" se trouvent dans le même dossier. Il est souvent utile de placer les fichiers CSS dans un dossier "CSS". Il faudra alors modifier le code HTML en conséquence :

```
<link rel="stylesheet" href="CSS/style.css">
```



A retenir : Pour terminer, voici quelques balises très utilisées.

- La balise "**a**"

```
<a href="mon_autre_page.html">Cliquez ici pour vous rendre sur mon autre page</a>
```

La balise "**a**" permet de créer des liens hypertextes, ce sont ces liens hypertextes qui vous permettent de "voyager" entre les pages d'un site ou entre les sites. Les liens hypertextes sont par défaut soulignés et de couleur bleue (modifiable grâce au CSS). La balise "**a**" possède un attribut "*href*" qui a pour valeur le chemin du fichier que l'on cherche à atteindre ou l'adresse du site cible (exemple : `<a href="http://www.google.fr">Cliquez ici pour vous rendre sur google.fr</a>`). Entre la balise ouvrante et fermante, on trouve le texte qui s'affichera à l'écran (c'est ce texte qui est souligné et de couleur bleue). La balise peut sans problème se trouver en plein milieu d'un paragraphe.

- La balise "**img**"

Comme vous devez déjà vous en douter, la balise image sert à insérer des.....images :

```

```

la balise **img** est à la fois ouvrante et fermante comme la balise "**br**".

Elle possède 2 attributs :

- "**src**" qui doit être égal au nom du fichier image (ou au chemin si le fichier image se trouve dans un autre dossier).
- "**alt**" qui doit être égal à une description de votre image (cet attribut est utilisé notamment par les systèmes de description des pages web utilisées par les non-voyants), il faut donc systématiquement renseigner cet attribut.

- Les balises "**form**", "**input**" et "**button**"

Les formulaires sont des éléments importants des sites internet, ils permettent à l'utilisateur de transmettre des informations. Un formulaire devra être délimité par une balise "**form**" (même si ce n'est pas une obligation) :

```
<form>
....
</form>
```

Il existe différentes balises permettant de construire un formulaire, notamment la balise "**input**". Cette balise possède un attribut "**type**" qui lui permet de jouer des rôles très différents.

La balise "**button**" nous sera aussi d'une grande utilité.



« À faire vous-même 7 ? » :

Créez un fichier html contenant le code suivant :

```
<!doctype html>
<html lang="fr">
  <head>
    <meta charset="utf-8">
    <title>Voici mon site</title>
  </head>
  <body>
    <form>
      <p>voici un champ de texte : <input type="text"/></p>
      <p>voici une checkbox <input type="checkbox"/></p>
      <button>Cliquez ici !</button>
    </form>
  </body>
</html>
```

Testez votre code à l'aide d'un navigateur web en “double-cliquant” sur le fichier html que vous venez de créer.

Qu'est-ce que vous obtenez (Insérer la capture d'écran rognée du résultat et expliquez les balises) ?

voici un champ de texte :

voici une checkbox ☐

- Les balises "**div**" et "**span**"

Ces deux balises sont très utilisées (surtout la balise "**div**"). Pourtant, il faudrait, autant que possible, les éviter, pourquoi ?

Parce qu'elles n'ont aucune signification particulière, ce sont des balises dites "génériques".

À quoi servent-elles alors ?

À organiser la page, à regrouper plusieurs balises dans une même entité.

Pourquoi 2 balises génériques ?

Parce que "**div**" est une balise de type "**block**" et que "**span**" est une balise de type "**inline**" !

Sans vouloir trop entrer dans les détails, il faut bien comprendre que l'ordre des balises dans le code HTML a une grande importance. Les balises sont affichées les unes après les autres, on parle du flux normal de la page.

C'est ici qu'entrent en jeu les balises de type "**block**" et les balises de type "**inline**".

- Les contenus des balises de type "**block**" (*p, div, h1, ...*) se placent sur la page web les uns en dessous des autres.
- Les contenus des balises de type "**inline**" (*strong, img, span, a*) se placent sur la page web les uns à côté des autres.



A retenir : Cela revient à dire qu'une balise de type "**block**" prend toute la largeur de la page alors qu'une balise de type "**inline**" prend juste la largeur qui lui est nécessaire.



« À faire vous-même 8 ? » :

Créez un fichier html contenant le code suivant :

```
<!doctype html>
<html lang="fr">
  <head>
    <meta charset="utf-8">
    <title>Voici mon site</title>
  </head>
  <body>
    <div>div est une balise de type "block"</div>
    <p>la balise p est une autre balise de type block</p>
    <span>En revanche, span est une balise de type "inline"</span>
    <a href="www.google.fr">Et voici une autre balise de type "inline"</a>
    <h1>h1 est bien une balise de type "block"</h1>
    <span>la malheureuse balise span est "obligée" de se placer en dessous</
span>
  </body>
</html>
```

Testez votre code.

Qu'est-ce que vous obtenez (Insérer la capture d'écran rognée du résultat et expliquez les balises) ?

```
div est une balise de type "block"
la balise p est une autre balise de type block
En revanche, span est une balise de type "inline" Et voici une autre balise de type "inline"
h1 est bien une balise de type "block"
la malheureuse balise span est "obligée" de se placer en dessous
```



« À faire vous-même 9 ? » : Projet SNT-EMC (Évaluation)

OBJECTIFS

FORME : Création d'une page WEB.

FOND : Vous devez élaborer une page web (Html + Css) en binôme, en utilisant toutes les balises* étudiées en classe. Votre page web devra porter sur l'une des 4 thématiques suivantes :

• La communauté LGBT+ dans la chanson française : ○ <i>Amour censure</i> de Hoshi ○ <i>Ta reine</i> de Angèle ○ <i>Petit gars</i> de Suzanne ○ <i>Normal</i> ou <i>Grave</i> d'Eddy de Pretto	• La communauté LGBT+ dans l'art : ○ <i>Le voguing</i> aux USA ○ La communauté LGBT+ dans l'art ○ La communauté LGBT+ au cinéma
• La communauté LGBT+ dans le sport ○ Homosexualité et rugby ○ Homosexualité et sports de combat	• Le traitement des LGBT+ dans différents pays ○ Les LGBT+ aux USA ○ Les thérapies de guérisons en France ○ Donner son sang en France quand on est homosexuel (historiquement et actuellement) ○ LGBTQ+ et religions dans le monde

THEMATIQUE :

- EMC : Le constat des discriminations en raison de l'homosexualité et les moyens pour lutter contre ces discriminations.
- SNT : Le Web

MOTS CLÉS :

- EMC : Stéréotype, préjugé, orientation sexuelle, homosexualité, homophobie, égalité, liberté, lutte contre toute forme de discrimination.
- SNT : données, informations, algorithmes, langages, programme, machines, internet.

RÉFÉRENCE AU PROGRAMME EMC : Égalité et discrimination.

CONSIGNES : Pour votre sujet vous DEVEZ :

- Effectuer des recherches sur Internet et croiser vos sources.
- Votre page web doit comprendre une sitographie (minimum quatre sites différents) que vous devez commenter : ce site est-il fiable et pourquoi ? Ce site est-il objectif ou subjectif et pourquoi ?
- D'abord vous présenterez votre sujet et ses enjeux (par exemple : si vous prenez une chanson, vous devez parler de l'auteur, du contexte de sortie de la chanson).
- Ensuite vous expliquerez en quoi votre sujet est intéressant aux vues des avancées des droits obtenus par la communauté LGBT+ ou, au contraire, en quoi ils montrent les difficultés rencontrées par les membres de cette communauté.
- Et pour finir, après avoir décrit votre page web, vous commenterez à l'oral vos lignes de codes Html et Css en expliquant vos balises.
- Vous pouvez ajouter d'autres balises à condition de savoir les expliquer à l'oral.

* Balises :

html : <h1> <h2> <p> <p id="para_1"> <h2 class="titre_1"> <!doctype html> <html lang="fr"> <head>	<meta charset="utf-8"> <title> <body> <link rel="stylesheet" href="style.css"> <form> ☐	<button> <div> CSS: text-align: background-color: font-family: font-style: color: #para_1
--	--	---

EXEMPLES DE TRAVAUX ELEVES :



ANNEXE : La liste complète des balises HTML5

Balise	Description
<u><!-- ... --></u>	Définit un commentaire
<u><!DOCTYPE></u>	Définit le type du document
<u><a></u>	Définit un lien
<u><abbr></u>	Définit une abréviation
<u><address></u>	Définit une adresse
<u><area></u>	Définit une zone à l'intérieur d'une image
<u><article></u>	Définit un article
<u><aside></u>	Définit une partie latérale au contenu
<u><audio></u>	Définit un fichier audio
<u></u>	Texte en gras
<u><base></u>	Définit une URL de base pour tous les liens de la page
<u><bdo></u>	Définit la direction du texte
<u><blockquote></u>	Définit une longue citation
<u><body></u>	Définit le corps de la page
<u>
</u>	Saut de ligne
<u><button></u>	Définit un bouton cliquable
<u><canvas></u>	Définit un graphique
<u><caption></u>	Légende du tableau
<u><cite></u>	Définit une citation
<u><code></u>	Mise en forme d'un texte en code informatique
<u><col></u>	Définit une colonne d'un tableau
<u><colgroup></u>	Définit un groupe de colonne pour un tableau
<u><command></u>	Définit un bouton de commande
<u><datalist></u>	Définit une liste d'options
<u><dd></u>	Définition d'un terme
<u></u>	Définit un texte supprimé
<u><details></u>	Définit les détails d'un élément
<u><dfn></u>	Définition
<u><div></u>	Définit un calque ou une section
<u><dl></u>	Liste de définition
<u><dt></u>	Définition d'un terme
<u></u>	Mise en exergue d'un texte – italique

<u><embed></u>	Définit un contenu extérieur (audio, vidéo ...)
<u><fieldset></u>	Regroupe plusieurs éléments d'un formulaire
<u><figcaption></u>	Légende d'un groupe d'élément multimédia
<u><figure></u>	Définit un groupe d'élément multimédia
<u><footer></u>	Définit le bas d'un section ou d'une page
<u><form></u>	Définit un formulaire
<u><h1> to <h6></u>	Définit un titre par degré importance de 1 à 6
<u><head></u>	Définit l'en-tête d'un document
<u><header></u>	Définit le haut d'une section ou d'une page
<u><hgroup></u>	Regroupe les informations du haut d'une page ou section
<u><hr></u>	Barre horizontale
<u><html></u>	Définit un document html
<u><i></u>	Texte en italique
<u><iframe></u>	Introduit un page html dans une frame
<u></u>	Définit une image
<u><input></u>	Définit un champ
<u><ins></u>	Définit un texte insérer
<u><keygen></u>	Générateur de clé pour un formulaire
<u><kbd></u>	Raccourcis ou touche du clavier
<u><label></u>	Légende d'un groupe d'élément de formulaire
<u><legend></u>	Titre d'un groupe d'élément d'un formulaire
<u></u>	Élément d'une liste
<u><link></u>	Définit les relations entre les documents
<u><map></u>	Définit une carte
<u><mark></u>	Mise en valeur d'un mot ou d'un texte – Texte marqué
<u><math></u>	Définit une formule mathématique
<u><menu></u>	Définit un menu en liste
<u><meta></u>	Définit des informations relatives au document
<u><meter></u>	Définit une unité de mesure
<u><nav></u>	Définit un groupe de liens de navigation
<u><noscript></u>	Définit une alternative au script non supporté
<u><object></u>	Définit un objet du contenu extérieur multimédia
<u></u>	Définit une liste ordonné
<u><optgroup></u>	Regroupe d'une liste d »option dans un formulaire

<u><option></u>	Option d'une liste dans un formulaire
<u><output></u>	Définit un type de sortie
<u><p></u>	Définit un paragraphe
<u><param></u>	Définit les paramètres d'un objet
<u><pre></u>	Texte pré-formaté
<u><progress></u>	Définit une progression
<u><q></u>	Définit une courte citation
<u><rp></u>	Annotation ruby si le script n'est pas supporté
<u><rt></u>	Annotation ruby d'explication
<u><ruby></u>	Définit une annotation en ruby
<u><samp></u>	Définit un échantillon de code
<u><script></u>	Définit un script
<u><section></u>	Définit une section
<u><select></u>	Définit une liste sélectionnable
<u><small></u>	Minimise l'importance d'un texte
<u><source></u>	Définit la source d'un contenu multimédia
<u></u>	Définit une section de type inline
<u></u>	Mise en exergue d'un texte – Texte en Gras
<u><style></u>	Définit un style CSS
<u><sub></u>	Mise en indice d'un texte
<u><summary></u>	Définit l'en-tête des détails d'un document ou section
<u><sup></u>	Mise en exposant d'un texte
<u><svg></u>	Définit une image vectorielle
<u><table></u>	Définit un tableau
<u><tbody></u>	Définit le corps d'un tableau
<u><td></u>	Définit une cellule d'un tableau
<u><textarea></u>	Définit une zone de texte
<u><tfoot></u>	Définit le bas d'un tableau
<u><th></u>	Définit une cellule d'en-tête d'un tableau
<u><thead></u>	Définit le haut d'un tableau
<u><time></u>	Définit une unité de temps
<u><title></u>	Définit le titre d'un document
<u><tr></u>	Définit une ligne de tableau
<u><track></u>	Définit une unité de temps pour les éléments <audio> et <video>.

<u></u>	Définit une liste non-ordonné
<u><var></u>	Définit une variable
<u><video></u>	Définit une vidéo
<u><wbr></u>	Définit un possible retour à la ligne