

# A LA DECOUVERTE DU LANGAGE HTML



**I. Introduction.** \_\_\_\_\_ **2**

**II. Les pages web.** \_\_\_\_\_ **3**

**III. Le langage HTML.** \_\_\_\_\_ **4**

**IV. Apprendre le HTML.** \_\_\_\_\_ **5**

**V. Les éléments html de base.** \_\_\_\_\_ **10**

**VI. Aller plus loin dans HTML.** \_\_\_\_\_ **19**

- Matériel : Ordinateur, Connexion internet.
- Logiciels : Navigateur.
- Pré-requis pour prendre un bon départ :

	☹	☺	☺☺	☺☺☺
S'inscrire sur un site <b>et retenir ses identifiants</b> !				
Manipulations informatiques basiques : ouvrir, copier, enregistrer un fichier ; aller et trouver sur Internet etc.				

NOM et Prénom : .....

Terminale S spé ISN.

Ce cours n'a pas la prétention de former des spécialistes en Html loin de là.

Il s'agira simplement de découvrir les concepts et notions de base du langage html et de susciter l'envie d'aller plus loin.

## I. INTRODUCTION.

### A. Préambule :

Après avoir manipulé et transformé l'Information grâce à l'Algorithmique-Programmation, on veut maintenant la diffuser de manière conforme, c-à-d être capable de la reconstituer à l'identique ailleurs.

Intéressons-nous à ce problème de la **reconstitution à l'identique** (le problème de la diffusion sera vu dans le cours sur les réseaux).

Imaginons que je veuille transmettre à une personne un document texte avec une belle mise en forme.

- 1<sup>ère</sup> idée : je lui téléphone et je lui dicte seulement le contenu. Quel problème cela pose-t-il ?

.....

Conséquence de cette première idée : on distingue le contenu de l'information de sa mise en forme.

- 2<sup>ème</sup> idée : je lui envoie une photocopie ou un scan : le contenu et la mise en forme sont ainsi préservés.

Le problème est que cette photocopie ou ce scan ne sont que de simples .....

Je suis passé du document texte à une image avec les limitations qui en découlent : impossibilité de dissocier le contenu de la mise en forme afin de ne manipuler que le contenu (copier-coller par exemple ou correction) ou que la mise en forme (modification de couleur par exemple).

Conséquence de cette deuxième idée : le contenu et la mise en forme ne doivent pas devenir immuables.

- 3<sup>ème</sup> idée ?

### B. Problématique :

La problématique est donc la suivante : comment faire en sorte qu'un document comporte non seulement son Information (son texte brut, ses images, ses sons etc.), mais toutes les indications qui permettent de reconstituer sa mise en forme à l'identique ?

On a inventé pour cela **les langages de description** dont le plus célèbre d'entre eux : .....

Voyons cela sur une page web.

## II. LES PAGES WEB.

### A. Que se cache-t-il derrière ?

→ Ouvrez votre navigateur puis aller sur n'importe quelle page web.

→ Faites un clic droit sur la page puis « Code source de la page ».

Que s'affiche-t-il ? .....

Un truc dans ce genre là ? .....

```

1 <!DOCTYPE html>
2 <html lang="fr" dir="ltr" class="no-js" data-ffo-opensans="false" data-ffo-zillaslab="false">
3 <head prefix="og: http://ogp.me/ns#">
4   <meta charset="utf-8">
5
6   <meta http-equiv="X-UA-Compatible" content="IE=Edge">
7   <script>(function(d) { d.className = d.className.replace(/no-js/, ''); })(document.documentElement);</script>
8
9
10
11
12
13 <title>Les bases du HTML - Apprendre le Web | MDN</title>
14
15 <meta name="viewport" content="width=device-width, initial-scale=1">
16 <meta name="robots" content="index, follow">
17 <link rel="home" href="/fr/">
18 <link rel="copyright" href="#copyright">
19

```

On ne voit plus que du texte, en partie coloré, dans un langage abscons !

Quelle est la toute première ligne de ce texte ? .....

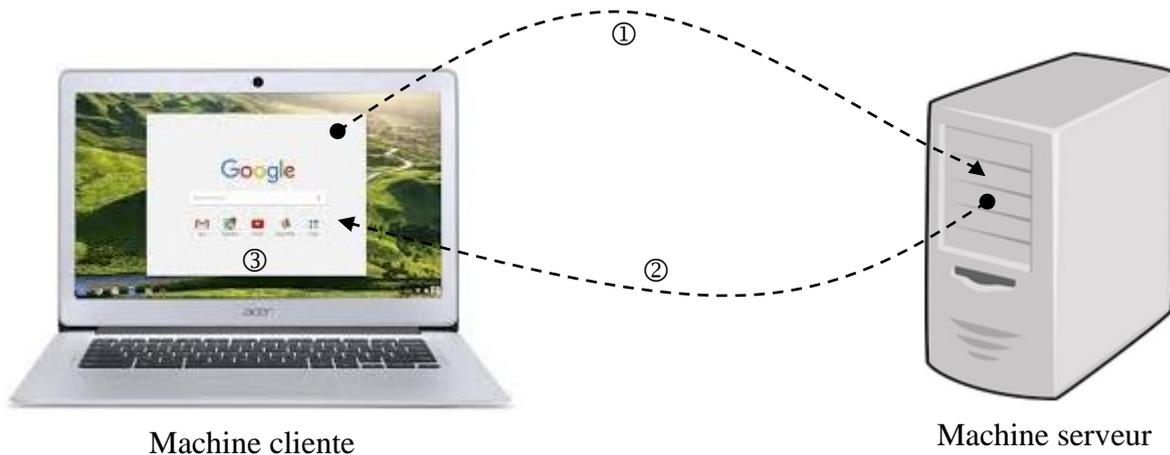
Toutes les pages web sont écrites avec un langage qui s'appelle le langage .....

### B. Comment une page web s'affiche-t-elle sur l'écran de son ordinateur ?

Toute page web est initialement stockée sur une machine appelée serveur. **Cette page web est un fichier texte écrit en langage html.**

Lorsqu'une personne appelé client veut consulter une page web, il utilise un navigateur (par exemple ..... ) qui agit comme suit :

- ① Une requête (une demande) est envoyée par l'intermédiaire du navigateur à la machine serveur : « envoie-moi tel fichier html. »
- ② Le serveur renvoie normalement le fichier html en question à la machine cliente.
- ③ Le navigateur sur la machine cliente lit le fichier html et peut l'afficher correctement grâce à la description conforme assurée par le langage HTML.



### III. LE LANGAGE HTML.

#### A. Historique du langage HTML :

➤ En 1967, William Tunncliffe montre la nécessité de séparer le contenu de l'Information de son format. En 1969 est alors inventé le GML (Generalized Markup Language), un langage à balises. Son évolution, le SGML, va donner naissance au HTML.

Mais ce n'est qu'en 1993, avec le navigateur web Mosaic, que le HTML va débiter sa vertigineuse expansion.



➤ Les avantages du HTML sont nombreux : ouvert, libre, peu coûteux (un simple éditeur de texte suffit à écrire ses premiers documents HTML), relativement facile à aborder, il représente en outre un bon moyen de dépasser les problèmes de compatibilité entre des systèmes et des formats informatiques différents.

#### B. HTML est un langage de description :

➤ Html n'est pas un langage de programmation contrairement à Python ou .....

Il n'est pas conçu pour exécuter un algorithme : pas de boucles, de tests conditionnels etc. en Html !

Il sert à décrire la structure et le sens du contenu (c-à-d le fond) d'une page web à l'aide de balises.

➤ Il existe d'autres langages de description. Par exemple :

- le langage CSS qui s'occupe de l'habillage (c-à-d la forme) des éléments d'une page web.
- le langage Latex qui permet de structurer et mettre en forme des textes scientifiques.
- le langage MathML qui permet l'affichage de symboles mathématiques sur les pages web.

#### C. HTML : Hyper Text Markup Language.

##### 1. Hyper Texte :

➤ Une page web ne se limite quasiment jamais à du texte mais contient aussi des .....

Une page web peut aussi contenir un formulaire, un jeu, un forum etc.

De plus, la navigation au sein d'une page web n'est pas linéaire comme dans un journal papier mais multidirectionnelle : on peut aller d'une page à une autre au gré de ses envies grâce à des .....

➤ Ce texte enrichi aux multiples ramifications s'appelle de l'.....

##### 2. Markup Language :

➤ **Markup signifie Balise.**

Le langage HTML permet de structurer le contenu d'une page web. Il permet par exemple de définir quel contenu formera un menu, quels seront les paragraphes, les titres, d'insérer des liens, des tableaux etc.

Cette structuration se fait à l'aide de .....

➤ Les balises s'écrivent entre chevrons `< >`. Ces balises sont colorées dans le code source de la page pour mieux les distinguer : c'est ce qu'on appelle la **coloration syntaxique**.

Par exemple dans le code source p.3, on voit le couple de balises `<script> </script>`.

Donner un autre exemple de couple de balises : .....

➤ **La bonne maîtrise du langage HTML revient donc à connaître ces balises, leur syntaxe et la façon dont on les utilise.**

## IV. APPRENDRE LE HTML.

### A. Les éléments html :

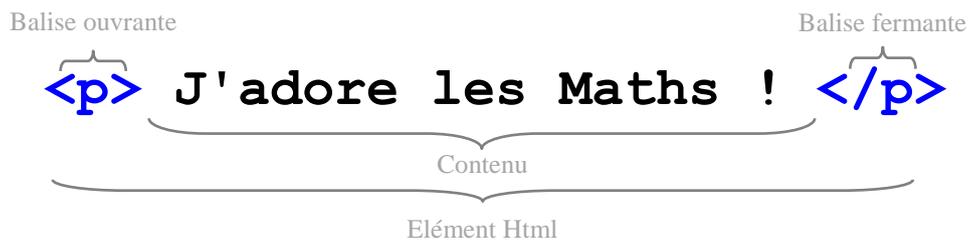
La structure de base du langage html s'appelle l'élément html.

Par exemple, soit le contenu textuel suivant : **J'adore les Maths !**

Si vous souhaitez indiquer que ce contenu fait partie d'un paragraphe, on le place dans un élément de paragraphe `<p>`, ce qui donne l'élément html : `<p> J'adore les Maths ! </p>`

#### 1. Anatomie d'un élément html :

Les composants de base d'un élément html sont :



1. **la balise ouvrante** `< >` : Elle indique le début de l'élément html et de quel type d'élément html il s'agit (ici `<p>` comme paragraphe).  
Pour notre exemple, cela indique le début du paragraphe.
2. **la balise fermante** `</ >` : Elle indique la fin de l'élément html.  
Pour notre exemple, elle indique la fin du paragraphe.
3. **le contenu** « **J'adore les Maths !** » : C'est le contenu de l'élément, ici c'est simplement du texte, mais ce peut être une image, un tableau etc..
4. **l'élément html** : Il est composé de la balise ouvrante, du contenu et de la balise fermante.

#### 2. Attribut (ou paramètre ou complément) d'un élément html :

Les éléments peuvent aussi avoir un ou plusieurs « attributs ». La syntaxe est la suivante :



➤ **Les attributs-paramètres-compléments offrent des informations supplémentaires ou des options supplémentaires pour les éléments html auxquels ils s'appliquent.**

Dans notre exemple, l'attribut `class` permet de particulariser ce paragraphe en le nommant par `super`.

Cette dénomination `super` pourra par exemple être utilisée plus tard pour faire agir une règle de mise en forme CSS sur tous les paragraphes de classe `super`.

➤ Les attributs doivent toujours avoir :

1. un type (le nom de l'attribut), suivi d'un signe égal « = ».
2. des guillemets (ou quotes) pour encadrer la valeur de l'attribut.

### 3. Imbriquer des éléments html :

➤ On peut placer des éléments html au sein d'autres éléments html, c'est ce qu'on appelle l'**imbrication**.

Par exemple, pour montrer que vous adorez **vraiment** les Maths, vous pouvez placer le mot « adore » dans un élément `<strong>` :

```
<p> J' <strong> adore </strong> les Maths ! </p>
```

➤ Attention, l'imbrication des balises fonctionne exactement comme l'emboîtement des parenthèses selon le principe des poupées ..... : il ne doit pas y avoir de chevauchement, pas comme dans le code incorrect suivant :

```
<p> J' <strong> adore </p> les Maths ! </strong>
```

Les éléments doivent être emboîtés correctement. S'ils se chevauchent, le navigateur essaiera de choisir la meilleure option, qui ne sera peut-être pas le résultat attendu.

➤ Application : Compléter le code source suivant :

```
<.....> <italic> Viva <.....> les Maths ! </h1>
```

### 4. Des éléments html sans contenu :

Certains éléments html n'ont pas de contenu. Ces éléments sans contenu sont appelés **éléments vides**.

➤ Ainsi, l'élément `<img />` qui sert à afficher une image fait partie de ces éléments vides.

Par exemple, pour afficher l'image Maths.jpeg dans un paragraphe, on pourra écrire le code source suivant :

```
<p>  </p>
```

Cet élément `<img />` a-t-il une balise fermante ? ..... Cet élément `<img />` a-t-il un contenu ? .....

Combien d'attributs possède ici cet élément `<img />`? ..... Lesquels ? .....

➤ L'élément `<img />` n'agit pas sur du contenu. Son but est d'insérer une image dans la page Html (là où l'élément `<img />` est placé dans le code). C'est donc pourquoi il n'y a pas aussi de balise fermante.

Nous reviendrons sur cet important élément en p.14.

➤ Il existe évidemment d'autres éléments html vides comme par exemple `<br />` (break) qui insère un retour à la ligne ; ou `<hr />` qui insère une ligne horizontale sur toute la largeur de la page.

➤ La syntaxe classique (balises ouvrante et fermante) `<img> .. <img />` marche aussi parfaitement !

## B. Des outils pour apprendre le langage html :

Créer un document html comprend 2 phases : création et visualisation.

### 1. Solution offline sans nécessité d'une connexion internet :

Nous avons besoin dans ce cas de **2 outils installés localement sur sa machine** pour coder en html :

1. **Un éditeur de texte pour créer et modifier notre code html.** Par exemple :

- Bloc-Notes (Windows) / TextEdit (Mac) pour un usage basique.
- Notepad++ (Windows) / TextWrangler (Mac) pour un usage avancé.

2. **Un navigateur pour tester le code html et visualiser le rendu :** Firefox, Safari etc.

Il faut hélas constamment passer de l'éditeur au navigateur pour créer-modifier le code puis voir le résultat. Ce qui n'est pas forcément commode pour débiter.

## 2. Solution online avec nécessité d'une connexion internet :

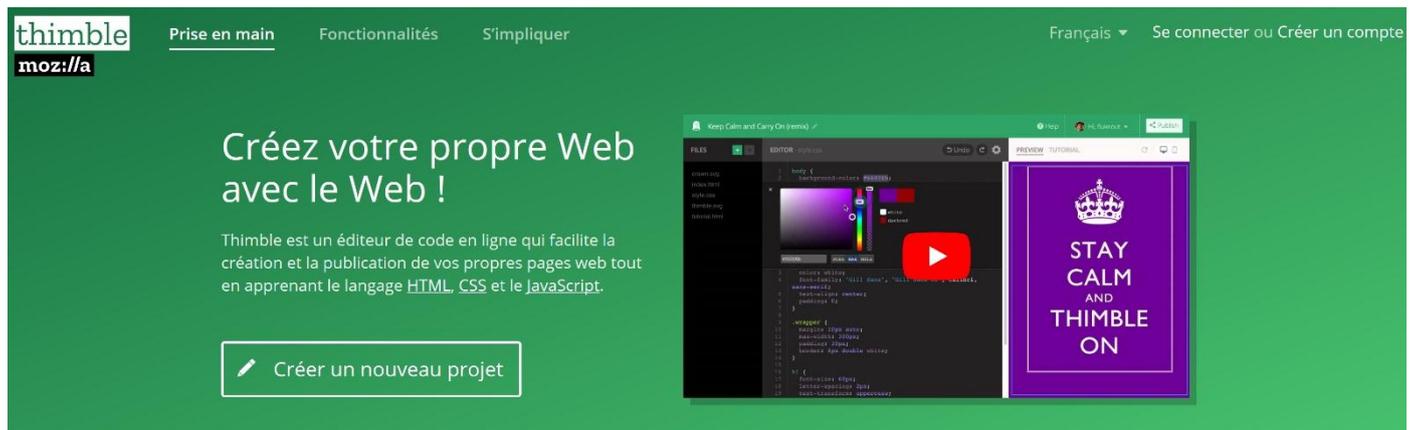
Il existe de nombreux **environnements de développement en ligne sur Internet**.

**Ils font office à la fois d'éditeur et de navigateur** : on modifie le code et on voit en simultanée le rendu de la page web ! Et ils ont encore bien d'autres avantages : prise en charge des langages CSS et Javascript, publication simplifiée etc.

## 3. Thimble :

Nous allons donc pour la suite utiliser l'un de ces environnements de développement en ligne : Thimble de la fondation Mozilla. Thimble (dé à coudre en français) est très simple à utiliser.

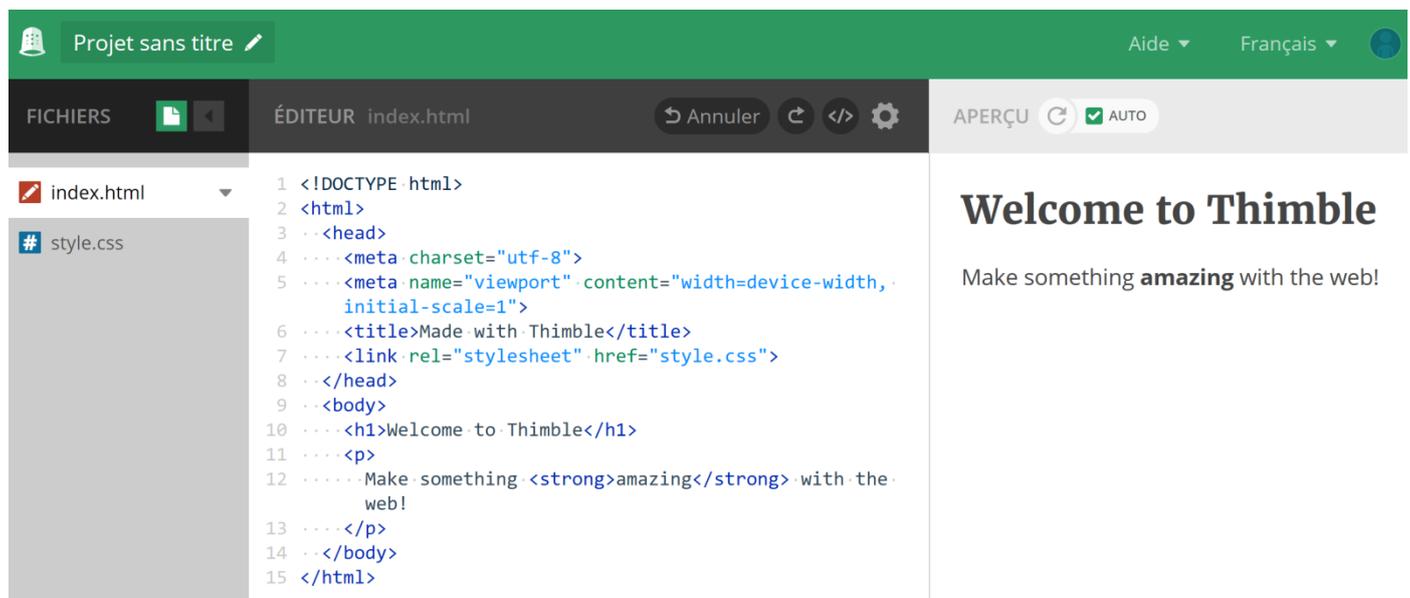
→ *Aller sur le site de Thimble Mozilla.*



*Créer un compte.*

**RETENEZ BIEN LES IDENTIFIANTS QUE VOUS AVEZ CHOISIS !!**

Puis on obtient comme ci-dessous la page web du projet par défaut créé par Thimble.



On voit bien dans l'interface les 2 fonctionnalités importantes :

- l'**Editeur** pour créer modifier le code (colonne du milieu).
- l'**Aperçu** pour tester le code html et visualiser en temps réel le rendu (colonne de droite).

(Pour avoir l'interface en blanc au lieu du noir : aller dans Paramètres (roue dentée) / Couleur du thème.)

Etudions plus en détail le code « minimal » qu'on voit dans la partie Editeur.

## C. Anatomie d'un document HTML :

Voici le code html très simple du premier projet créé par défaut par Thimble :

```

1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8">
5     <meta name="viewport" content="width=device-width,
      initial-scale=1">
6     <title>Made with Thimble</title>
7     <link rel="stylesheet" href="style.css">
8   </head>
9   <body>
10    <h1>Welcome to Thimble</h1>
11    <p>
12      Make something <strong>amazing</strong> with the web!
13    </p>
14  </body>
15 </html>

```

Ce code html contient entre autres :

1. `<!DOCTYPE html>` : le *doctype*.

La balise `<!DOCTYPE html>` est le premier élément censé apparaître dans le code d'une page html.

Ce n'est pas une balise html à proprement parler mais un indicateur qui permet au navigateur de savoir quelles règles appliquer pour la mise en page du document.

Elle définit :

- La version HTML utilisée par ce document.
- Le DTD (pour Document Type Definition) utilisé par le document :

`<!DOCTYPE html>` indique que c'est la version 5 de HTML (la dernière en date) qui est utilisée.

2. `<html>..</html>` : l'élément structurel `<html>`.

**Cet élément obligatoire encadre tout le contenu de la page.**

Cet élément est parfois appelé l'élément racine.

3. `<head>..</head>` : l'élément structurel `<head>`.

**Cet élément obligatoire est utilisé comme un container pour toutes les choses qui font partie de la page Html mais qui ne sont pas visibles.**

C'est dans cet élément qu'on pourra mettre les déclarations de codage des caractères (*character sets*) à utiliser, des mots-clés, l'auteur de la page, une description de la page qui apparaîtra sur les moteurs de recherche, les liens vers les fichiers css (mise en forme) et javascript (comportement de la page).

4. `<meta charset="utf-8">` :

Cet élément indique quel encodage des caractères (character set) est utilisé dans la page : ici utf-8.

Généralement le jeu de caractères utf-8 permet de gérer n'importe quel texte car utf-8 regroupe l'ensemble des caractères utilisés dans les différents langages humains.

5. `<meta name="viewport" content="width=device-width, initial-scale=1">` :

Cet élément permet de contrôler l’affichage sur les navigateurs mobiles : ici largeur = toute la largeur de l’écran de l’appareil et zoom initial à 1 soit la taille normale.

6. `<title>..</title>` : l’élément `<title>`.

**Cet élément obligatoire dans la version 5 de HTML définit le titre de la page.**

C'est ce titre qui apparaîtra sur l'onglet et qui est utilisé pour décrire la page lorsqu'elle est placée dans le marque-pages.

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

Cet élément permet de lier cette page html à la feuille de style « style.css » écrite en langage CSS pour la mise en forme.

9. `<body>..</body>` : l’élément structurel `<body>`.

**Cet élément obligatoire est celui qui contient tout le contenu qui sera *visible* : cela peut être du texte, des images, des vidéos, des jeux, etc.**

Quels sont les 4 éléments obligatoires de n’importe quelle page web et à quoi correspondent-ils ?

- 1.
- 2.
- 3.
- 4.

Indiquer dans le tableau si les éléments suivants doivent être insérer à l’intérieur de `<head>` ou de `<body>` :

	dans <code>&lt; head &gt;...&lt; / head &gt;</code>	dans <code>&lt; body &gt; ... &lt; / body &gt;</code>
une image		
le titre de la page affichée sur l’onglet du navigateur		
le titre d’un article du document		
les liens vers des fichiers externes de sytle CSS ou de comportement Javascript		
l’auteur de la page		
des articles		
le menu de la page		

## V. LES ELEMENTS HTML DE BASE.

### A. Baliser le texte :

#### 1. Les titres : éléments <h1> .. <h6>.

Les balises utilisées sont <h..> comme *header*.

Eléments	Commentaires
<h1>..</h1>	1) Titre principal.
<h2>..</h2>	2) Titre de chapitre, de section.
<h3>..</h3>	3) Sous titre.
<h4>..</h4>	4) Sous sous titre.
<h5>..</h5>	5) Etc.
<h6>..</h6>	6) Etc.

Attributs possibles	Commentaires	Exemples de valeurs possibles de l'attribut
• <b>title</b> ="..."	Ajoute une infobulle au contenu.	
Appliquer une règle de style css	Exemples de règles de style css	Exemples de valeurs possibles de la règle de style
• <b>style</b> ="..."	<b>font-family</b> : police de caractère. <b>font-size</b> : taille du texte en pixels. <b>font-weight</b> : épaisseur de la police. <b>text-align</b> : alignement du texte. <b>color</b> : couleur du texte.	<b>Arial, Verdana, etc.</b> <b>30 px, etc.</b> <b>bold, lighter, normal etc.</b> <b>justify, center, left, etc.</b> <b>red, yellow, etc.</b>

➤ Exemple : Si on veut mettre le texte « J'adore les Maths ! » en titre de niveau 2, centré, avec en infobulle « chapitre 1 », on peut écrire en html (bien observer la syntaxe) :

```
<h2 style="text-align:center" title="chapitre 1" >J'adore les Maths !</h2>
```

→ A vous : Sur *Thimble*, changer le contenu du titre h1 par un texte de votre choix, rajouter une infobulle et appliquer des règles de style de votre choix (couleur, taille de la police, alignement du texte etc.).

➤ Remarque :

Ne pas confondre les éléments <h1> <h6> avec l'élément <title> !

L'élément <title> a un nom trompeur ! Il n'indique pas un titrage comme <h1> <h6> mais le nom de la page web qui s'affichera sur la barre d'onglet. Donc <title> est placé dans l'élément d'en-tête <head> alors que <h1> <h6> sont visibles donc placés dans le corps de la page c-à-d dans l'élément <body>.

## 2. Les paragraphes : élément <p>.

La balise utilisée est <p> comme *paragraph*.

Elément	Commentaires
<p>..</p>	1) Met en forme le texte contenu dans un paragraphe. 2) Texte justifié par défaut (pas de décalage par rapport aux bords). 3) Apparition d'un saut de ligne avant son contenu. 4) Apparition d'un saut de ligne après son contenu.

Attributs possibles	Commentaires	Exemples de valeurs possibles de l'attribut
• <b>title</b> =".."	Ajoute une infobulle au contenu.	
Appliquer une règle de style css	Exemples de règles de style css	Exemples de valeurs possibles de la règle de style
• <b>style</b> =".."	<b>font-family</b> : police de caractère. <b>font-size</b> : taille du texte en pixels. <b>font-weight</b> : épaisseur de la police. <b>text-align</b> : alignement du texte. <b>text-indent</b> : décalage de la première ligne du texte en pixels. <b>color</b> : couleur du texte.  <b>display</b> : joue un rôle important dans la disposition des éléments. Indique le mode d'affichage des éléments et leur comportement par rapport aux éléments adjacents.	<b>Arial, Verdana, etc.</b> <b>30 px, etc.</b> <b>bold, lighter, normal etc.</b> <b>justify, center, left, etc.</b> <b>25 px, etc.</b>  <b>red, yellow, etc.</b>  <b>none, block, inline, inline-block, inline-table, list-item, table, etc.</b>

➤ Exemple : Si on veut mettre le texte « J'adore les Maths ! » dans un paragraphe, en justifié, en taille 25 pixels, avec une indentation de 25 pixels, on peut écrire en html (bien observer la syntaxe) :

```
<p style="text-indent:25px ; font-size:25px ; text-align:justify"> J'adore les Maths !</p>
```

→ A vous : Sur *Thimble*, dans le corps de la page, rajouter 2 paragraphes contenant du texte de votre choix et appliquer leur des règles de style de votre choix.

➤ Remarque :

La balise <span> est une alternative pour encapsuler du texte.

A la différence des paragraphes <p>, l'élément <span> n'est pas de type block : en particulier il ne prend pas toute la largeur de la page. On peut utiliser <span> si on veut juste encadrer une partie du texte.

### 3. Les listes : éléments `<ul>` et `<ol>`.

Les types de listes utilisés fréquemment sont les listes non-ordonnées et les listes ordonnées.

#### **Listes non-ordonnées <ul> ou Listes à puces :**

Ce sont des listes sans ordre (par exemple une liste de courses).

La balise utilisée est `<ul>` comme *unordered list*.

Élément	Commentaires
<code>&lt;ul&gt;..&lt;/ul&gt;</code>	<ol style="list-style-type: none"> <li>1) Met en forme le texte contenu sous forme de liste à puces sans ordre.</li> <li>2) Sans attribut, la puce par défaut est •.</li> <li>3) Doit contenir au moins 2 items de liste.</li> <li>4) Chaque item d'une liste est balisé avec un élément <code>&lt;li&gt;</code>.</li> <li>5) Une sous liste peut être imbriquée dans une liste.</li> </ol>

Appliquer une règle de style css	Exemples de règles de style css	Exemples de valeurs possibles de la règle de style
• <code>style=".."</code>	<code>list-style-type :</code> Type de puce	<ul style="list-style-type: none"> <li>• <code>disc</code> (par défaut)</li> <li>○ <code>circle</code></li> <li>• <code>square</code></li> </ul>

#### ➤ Exemple :

```

1 <ul style="list-style-type:circle">
2   <li>Catégorie</li>
3   <li>Catégorie
4     <ul style="list-style-type:square">
5       <li>Sous catégorie</li>
6       <li>Sous catégorie</li>
7       <li>Sous catégorie</li>
8     </ul>
9   </li>
10  <li>Catégorie</li>
11 </ul>

```

⇒

- Catégorie
- Catégorie
  - Sous catégorie
  - Sous catégorie
  - Sous catégorie
- Catégorie

→ A vous : Sur *Thimble*, dans le corps de la page, ajouter une liste non ordonnée de 2 items de votre choix (puce carrée) avec pour le dernier item, une sous liste de 2 éléments de votre choix (puce pointée).

**Listes ordonnées <ol> :**

Ce sont des listes avec un certain ordre (par exemple une recette de cuisine).

La balise utilisée est `<ol>` comme *ordered list*.

Elément	Commentaires
<code>&lt;ol&gt;..&lt;/ol&gt;</code>	1) Met en forme le texte contenu sous forme de liste ordonnée. 2) Sans attribut, la numérotation par défaut est entière et commence à 1. 3) Doit contenir au moins 2 items de liste. 4) Chaque item d'une liste est balisé avec un élément <code>&lt;li&gt;</code> . 5) Une sous-liste peut être imbriquée dans une liste.

Appliquer une règle de style CSS.	Règle de style CSS	Exemples de valeurs possibles de la règle de style
• <code>style="..."</code>	<b>list-style-type :</b> Type de numérotation	<ul style="list-style-type: none"> <li>• <b>decimal</b> : 1. 2. 3. ... (par défaut)</li> <li>• <b>decimal-leading-zero</b> : 01. 02. 03. ...</li> <li>• <b>lower-alpha</b> : a. b. c. ...</li> <li>• <b>upper-alpha</b> : A. B. C. ...</li> <li>• <b>upper-roman</b> : I. II. III. ...</li> </ul>

➤ **Exemple :**

```

9   <body>
10  <ol style="list-style-type:upper-alpha">
11    <li>Catégorie</li>
12    <li>Catégorie
13      <ol style="list-style-type:decimal">
14        <li>Sous catégorie</li>
15        <li>Sous catégorie</li>
16        <li>Sous catégorie</li>
17      </ol>
18    </li>
19    <li>Catégorie</li>
20  </ol>
21  </body>

```

⇒

- A. Catégorie
- B. Catégorie
  - 1. Sous catégorie
  - 2. Sous catégorie
  - 3. Sous catégorie
- C. Catégorie

→ A vous : Sur Thimble, dans le corps de la page, ajouter une liste ordonnée de 3 items de votre choix (numérotation lettres minuscules) avec pour le dernier item, une sous liste de 2 éléments de votre choix (numérotation chiffres romains).

## B. Images : élément <img />.

La balise utilisée est `<img />` comme *image*.

Elément	Commentaires
<code>&lt;img /&gt;</code>	<ol style="list-style-type: none"> <li>1) Affiche une image dans une page web.</li> <li>2) Cette image peut être stockée à l'intérieur ou à l'extérieur du site web.</li> <li>3) Attribut « obligatoire » <code>src=""</code> qui indique l'adresse du fichier image.</li> <li>4) Attribut « obligatoire » <code>alt=""</code> qui indique le texte « de secours ».</li> <li>5) Pas de balise fermante donc pas de contenu. Que des attributs !</li> </ol>

Attributs possibles	Commentaires	Exemples de valeurs possibles de l'attribut
• <code>src=""</code>	<ul style="list-style-type: none"> <li>• <code>src</code> comme <i>source</i>.</li> <li>• Attribut « obligatoire ».</li> <li>• Indique l'adresse interne ou externe du fichier image.</li> </ul>	<ul style="list-style-type: none"> <li>• Exemple d'adresse interne : <code>/dossier-images/img1.jpg</code></li> <li>• Exemple d'adresse externe : <code>https://www.yalamaths...</code></li> </ul>
• <code>alt=""</code>	<ul style="list-style-type: none"> <li>• <code>alt</code> comme <i>alternative</i>.</li> <li>• Attribut « obligatoire ».</li> <li>• Affiche un texte de secours si l'image ne s'affiche pas.</li> <li>• Affiche en infobulle ce texte de secours si l'image s'affiche.</li> </ul>	<code>alt="Texte de secours"</code>
Appliquer une règle de style CSS.	Exemples de règles de style CSS	Exemples de valeurs possibles de la règle de style
• <code>style=""</code>	<p><code>width</code>: largeur en pixels, en % etc.</p> <p><code>height</code>: hauteur en pixels, en % etc.</p> <p><code>border</code>: épaisseur de la bordure en pixels</p>	<p>200 px, 100%, etc.</p> <p>300 px, 50%, etc.</p> <p>10 px, etc.</p>

### ➤ Exemples :

• ``

Affichera une image prenant toute la largeur de la page et dont le fichier test.png est situé dans le dossier images du site web.

• Pour positionner une image (la centrer par exemple), on peut utiliser la technique suivante : on place l'image dans un conteneur à tout faire `<div>` et on modifie l'alignement avec la règle de style `text-align` :

```
<div style="text-align: center">
```

```
  
```

```
</div>
```

Remarque : On aurait pu utiliser n'importe quelle autre balise conteneur autre que `<div>` : `<p>` par exemple.

→ Sur Thimble, créer un dossier Images (colonne de gauche), puis y transférer une image de votre choix.

Puis dans le corps de votre page web, faire afficher cette image centrée, avec un texte de secours, une hauteur de moitié de la page et une largeur de 300 pixels.

## C. Les liens : élément <a>.

Les liens sont très importants, ce sont eux qui font que le web est une **toile faite d’hypertexte**.

La balise utilisée est **<a>** comme *anchor* (*ancree*).

Elément	Commentaires
<code>&lt;a&gt;..&lt;/a&gt;</code>	<ol style="list-style-type: none"> <li>1) Ajoute un lien cliquable sur tout ce qui est contenu entre ces 2 balises.</li> <li>2) Ce peut être un texte, une image etc.</li> <li>3) Attribut « obligatoire » <code>href=""</code> qui indique l’adresse du document cible vers lequel le lien doit pointer.</li> <li>4) Attribut conseillé <code>target=""</code> qui indique où s’ouvre ce document cible. Sans target, le document cible s’ouvre par défaut dans le même onglet du navigateur.</li> </ol>

Attributs possibles	Commentaires	Exemples de valeurs possibles de l’attribut
• <code>href=""</code>	<ul style="list-style-type: none"> <li>• href comme <i>hypertext reference</i>.</li> <li>• Attribut « obligatoire ».</li> <li>• Indique l’adresse interne ou externe du document cible vers lequel le lien doit pointer.</li> </ul>	<ul style="list-style-type: none"> <li>• Exemple d’adresse interne : <code>/accueil/index.html</code></li> <li>• Exemple d’adresse externe : <code>http://www.yalamaths.free.fr</code></li> </ul>
• <code>target=""</code>	<ul style="list-style-type: none"> <li>• target comme <i>cible</i>.</li> <li>• Attribut conseillé.</li> <li>• Indique où s’ouvrira le document cible.</li> </ul>	<code>_blank</code> La cible s’ouvre dans un nouvel onglet du navigateur. Ne pas oublier le tiret !
• <code>title=""</code>	Ajoute une infobulle au contenu.	
Appliquer une règle de style css	Exemples de règles de style css	Exemples de valeurs possibles de la règle de style
• <code>style=""</code>	On peut évidemment appliquer certaines règles de style vues précédemment.	

➤ Exemples :

• `<a href="accueil.html" style="font-style:italic">Retour à l’accueil </a>`

Le texte « Retour à l’accueil » sera mis en italique et comportera un lien cliquable vers la page accueil.html du même site. Cette page accueil.html s’ouvrira par défaut dans le même onglet du navigateur.

• Une image test.png se trouve dans le dossier images. On veut la rendre cliquable pour qu’elle ouvre Google dans un nouvel onglet, avec une infobulle, de hauteur 50% de la page et avec un texte de secours.

```
<a href="https://www.google.fr" target="_blank" title="Vers Google">
  
</a>
```

Remarque : On pourra rendre cliquable certaines zones d’une image grâce à l’élément `<map>`.

→ A vous : Sur *Thimble*, rajouter sur un titre un lien en italique avec une infobulle vers le site de votre choix. Ce lien doit s’ouvrir dans une nouvelle page. Puis faites de même avec votre image.

## D. Les tableaux : balises <table>.

Les tableaux sont très importants en html. Ils ont 2 fonctions essentielles :

1. Présenter l'information de manière structurée sous forme de ... tableaux !
2. Faire de la mise en page élémentaire à l'aide de tableaux invisibles (bordures à 0).

### 1. Informations présentées sous forme de tableau :

Les balises minimales pour créer un tableau sont <table>, <tr> et <td>.

Élément	Commentaires
<table>..</table>	Déclaration du tableau.
<caption>..</caption>	Donne un titre au tableau qui sera écrit juste au-dessus du tableau.
<thead>..</thead>	En tête du tableau.
<tr>..</tr>	Ligne du tableau (table row).
<td>..</td>	Toujours encapsulé dans <tr> ou <thead>, crée une cellule dans la ligne.
<tfoot>..</tfoot>	Ligne de pied du tableau.

➤ Un exemple vaut mieux qu'un long discours :

```

10 <table cellspacing="2px" cellpadding="2px"
    rules="all" style="border:dotted 1px black">
11   <caption>Titre du tableau</caption>
12   <thead>
13     <tr>
14       <th width="200">Titre Colonne 1</th>
15       <th width="80">Titre Colonne 2</th>
16     </tr>
17   </thead>
18   <tr>
19     <td>Valeur 1</td>
20     <td>Valeur 2</td>
21   </tr>
22   <tr>
23     <td>Valeur 1</td>
24     <td>Valeur 2</td>
25   </tr>
26 </table>

```

⇒

Titre du tableau	
Titre Colonne 1	Titre Colonne 2
Valeur 1	Valeur 2
Valeur 1	Valeur 2

On voit dans les balises <table> et <th> quelques exemples d'attributs de mise en forme pour les tableaux :

Attributs possibles	Commentaires	Valeurs possibles de l'attribut
• <b>border</b> ="..."	<ul style="list-style-type: none"> <li>• Indique l'épaisseur en pixels des bordures extérieures du tableau.</li> <li>• Mettre à 0px pour avoir un tableau invisible (utile pour de la mise en page).</li> </ul>	0px, 1px, 8px ...
• <b>cellpadding</b> ="..."	Espacement en pixels entre le bord des cases et leur contenu.	0px, 5px, 8px ...
• <b>cellspacing</b> ="..."	Espacement en pixels entre les cases.	0px, 7px, 8px ...

Attributs possibles	Commentaires	Valeurs possibles de l'attribut
• <b>rules</b> ="..."	Indique quelles bordures intérieures sont visibles.	<b>all, none, cols, rows, ...</b>
• <b>frame</b> ="..."	Indique quelles bordures extérieures sont visibles.	<b>above, below, border, hside, vside, box, void</b>
• <b>width</b> ="..."	Indique la largeur totale du tableau en pixels ou en pourcentage.	<b>500px, 100% ...</b>
• <b>colspan</b> ="..."	Fusion horizontale du nombre indiqué de cases.	<b>un nombre entier</b>
• <b>rowspan</b> ="..."	Fusion verticale du nombre indiqué de cases.	<b>un nombre entier</b>
Appliquer une règle de style css	Exemples de règles de style css	Exemples de valeurs possibles de la règle de style
• <b>style</b> ="..."	On peut évidemment appliquer certaines règles de style vues précédemment.	

➤ Exemple :

```

10 <table cellpadding="3px" cellspacing="0px"
    rules="all" style="border:dashed 1px black; text-align:center;">
11   <tr>
12     <th style="width:140px;">Titre1</th>
13     <th style="width:100px;">Titre2</th>
14   </tr>
15   <tr>
16     <td>Valeur1</td>
17     <td>Valeur2</td>
18   </tr>
19   <tr>
20     <td colspan="2" style="background-color:lightsteelblue;">Cellule avec
    colspan</td>
21   </tr>
22   <tr>
23     <td rowspan="2" style="background-color :
    lightgrey;">Cellule avec rowspan</td>
24     <td>Valeur3</td>
25   </tr>
26   <tr>
27     <td>Valeur4</td>
28   </tr>
29 </table>

```

⇒

Titre1	Titre2
Valeur1	Valeur2
Cellule avec colspan	
Cellule avec rowspan	Valeur3
	Valeur4

➤ Pour plus d'exemples, voir : [http://hypermedia.univ-paris8.fr/jean/internet/ex\\_table.html](http://hypermedia.univ-paris8.fr/jean/internet/ex_table.html)

→ A vous : Sur Thimble, créer le tableau suivant avec en plus un titre de tableau.

Titre1 (largeur 100px)	Titre2 (largeur200px)
Valeur1	
Valeur2	Valeur3
Valeur4	

## 2. Utiliser les tableaux pour faire de la mise en page basique :

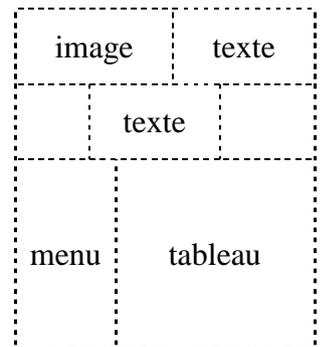
➤ Contrairement à un traitement de texte comme Word, lorsqu'on code en html, on ne voit pas tout de suite ce que cela donne (avec Thimble si, ouf !).

Très difficile ainsi de placer un élément lorsque sa position est autre que left, center, right, top ou bottom.

On ne peut pas par exemple positionner à la souris finement des images comme avec Word.

Pour contourner ce problème, on peut utiliser un tableau (dont on aura rendu les bordures invisibles !) : les images seront alors rangées dans des cases plus ou moins fusionnées du tableau et on n'aura plus qu'à jouer sur la largeur de ses cases pour bien positionner les images.

➤ Ainsi, lorsqu'on veut concevoir une page internet, il faut avoir en tête ce découpage de la page en cases de tableau, ce qui permettra une mise en page facile des éléments en jouant sur les marges des cases du tableau.



## VI. ALLER PLUS LOIN DANS HTML.

Ces quelques pages hélas ne feront pas de vous des virtuoses du HTML. ☹

Se perfectionner en HTML revient donc à connaître toutes les balises, leurs attributs et valeurs.

Cela revient aussi à maîtriser les règles de style qui peuvent s'appliquer aux éléments html : c'est l'objet d'un prochain cours sur le langage CSS.

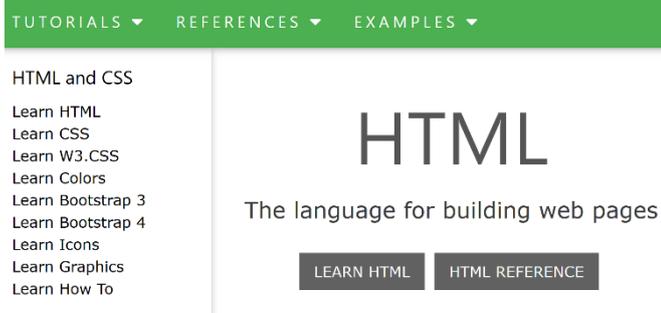
### A. Des aides précieuses :

- Pour tout apprendre sur Html et le reste :

**www.w3schools.com**

LE site d'apprentissage (édité par le W3C, l'organisme chargé de la standardisation des technologies du Web). Super site.

**w3schools.com**



- Pour avoir la liste des balises et attributs :

**www.startyourdev.com**

Bien foutu.

A avoir constamment ouvert dans un onglet.

Bienvenue sur StartYourDev.com

**Html5 - Css3 - Xml/Xsl - Javascript**



- Pour tuer le temps : **appli SoloLearn**

A installer d'urgence sur son smartphone !



### B. Un mini CV en ligne :

Même si on n'est pas des pros, on peut déjà s'amuser un peu ☺.

→ A vous : Sur *Thimble*, créer un mini CV :

- *3 pages reliées les unes aux autres avec des titres, des images.*
- *avec une liste ordonnée et une liste non ordonnée.*
- *avec au moins 2 tableaux.*