



Nom : SOUALMI

Prénoms : Madjda Maissa

Formation of React , HTML, CSS, JavaScript, Node, MongoDB

Sommaire

Intro to the web	2
What's the Internet ?	2
Intro to the web	2
The request / Response cycle	2
Front end & Back end	2
What do HTML,CSS,JS do ?	3
Setting up our developer environnement	3
Optional VScode theme	4
Html : the essentials	5
Introduction to HTML	5
First HTML page	5
Mozzila developer network (MDN)	6
Paragraph elements	6
Headings	6
Introduction to Chrome Inspector & HTML boilerplate	8
List elements	9
Anchor tags	10
Images	11
Comments	12
HTML : Next steps & semantic	13
What HTML5 actually is ?	13
Block VS inline elements , and <div> elements	13
An odd assortment of elements : HR ,BR , SUP & SUB	14
Entity code	17
Intro to semantic markup	18
Playing with semantic elements	18
Screen reader demonstration	18
Vs code tip : emmet	18
Resources	18

Intro to the web

What's the Internet ?

En résumé l'internet est un tas d'ordinateurs connectés et c'est ce qui nous permet de transporter nos données sur le WWW (World Wide Web) et de faire fonctionner le WEB .

Intro to the web

Un système d'information qui nous permet de partager des ressources sur internet qui sont identifiés par nos URL (Uniform Resource Locator) , et la façon dont les informations sont transférées dont la manière ou le web fonctionne vraiment , c'est qu'il y a le protocole HTTP (Hyper Text TRansfer Protocol) y en a aussi IP TCP.

HTTP est un ensemble normalisé des règles régissant le fonctionnement d'une forme de communication .

The request / Response cycle

🤔 *Quand j'envoie une requête HTTP au serveur , que se passe-t-il ?*

Les serveurs web ne vont pas répondre avec une page web complète comme nous le voyons !!

🔧 Ils répondent avec des instructions , avec un code ou mon navigateur peut comprendre et l'afficher pour moi (Possible de voir le code en faisant un clic droit → code source de la page).

Front end & Back end

Quand j'envoie une requête , elle atteint le serveur Google par exemple , il doit comprendre quels sont les résultats de la recherche ? Annonces ? les droitsetc , puis une page sera d'abord construite , il y a donc beaucoup de logique à comprendre , qui suis-je , qu'est ce que j'ai la permission de voir ? Ce que je veux ? Donc ça c'est le back end (La cuisine du restaurant qui construit la réponse) ,Après Il me renvoie une réponse sous *forme d'accès HTML,JS et CSS* , et c'est ce code s'exécute dans mon navigateur , et c'est tout le code **côté client** ou **frontal**.

What do HTML,CSS,JS do ?

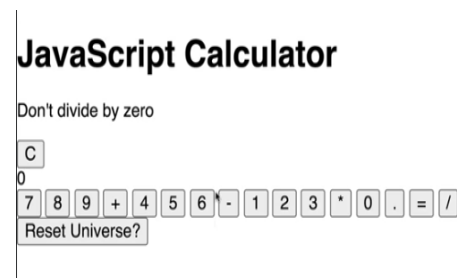
Les trois technologies sont responsables de la création des pages web(sont les seules que le navigateur peut comprendre)

🔧 **HTML** → **noms** (Dinosaures)

CSS → **Adjectifs** (mauves)

JS → **Verbes** (dansent)

→ HTML uniquement (le WHAT de la page)



→ HTML + CSS (une belle forme mais sans actions)

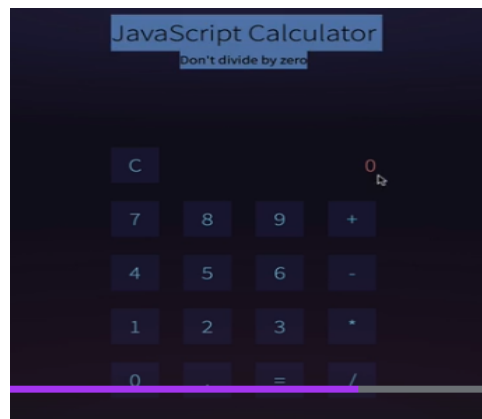


Figure 1 : Interface d'une calculatrice avec JS

→ HTML +CSS + JS (la calculatrice fonctionne bien)

Setting up our developer environnement

- Chrome (Web Browser) : <https://www.google.com/chrome/>
- VS Code (text editor) : <https://code.visualstudio.com/>

Introduction to the web development

Optional VScode theme

- extensions → Material theme → Install

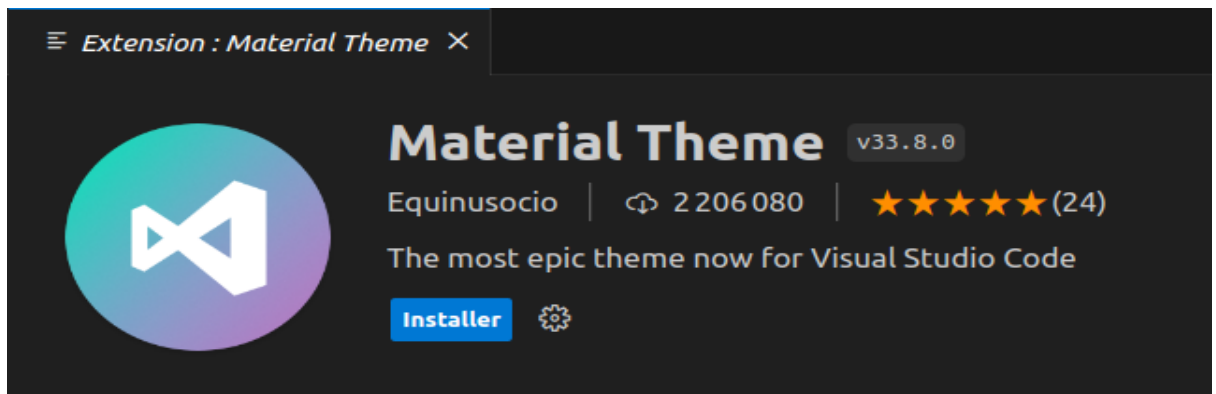


Figure 2 : Material theme

Html : the essentials

Introduction to HTML

Html est un langage de balisage (n'est pas un langage de programmation) , donc il est né pour donner une structure aux textes .

Syntax

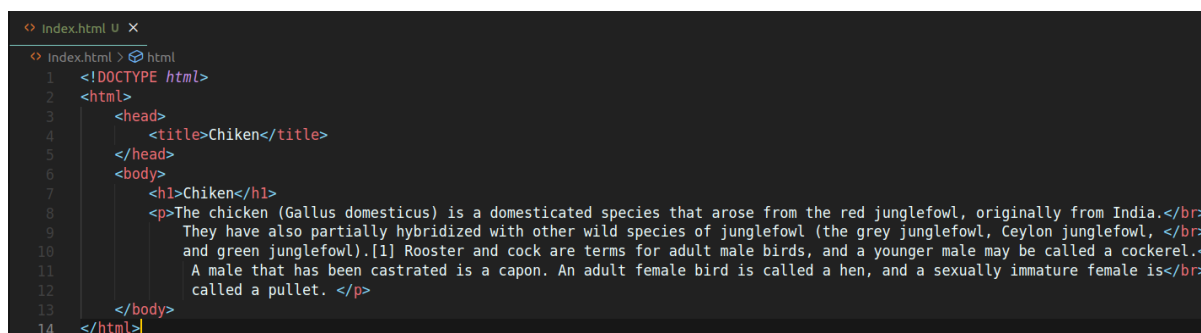
En HTML, la structure du texte repose sur l'utilisation de balises d'ouverture et de fermeture. Voici un exemple de syntaxe générale:

`<balise ouvrante>Contenu du texte</balise fermante>`

Il est important de noter que certaines balises, comme `
` pour un saut de ligne ou `` pour une image, peuvent ne pas nécessiter de balise de fermeture. Dans ces cas-là, vous pouvez utiliser une balise auto-fermante, qui ressemble à ceci :

`<balise />`

First HTML page

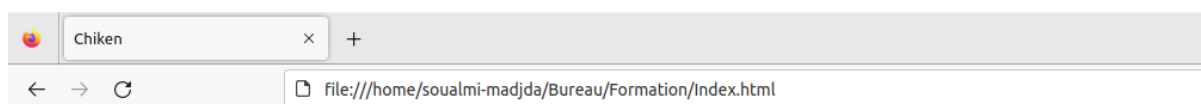


```

1  <!DOCTYPE html>
2  <html>
3      <head>
4          <title>Chicken</title>
5      </head>
6      <body>
7          <h1>Chicken</h1>
8          <p>The chicken (Gallus domesticus) is a domesticated species that arose from the red junglefowl, originally from India.</br>
9              They have also partially hybridized with other wild species of junglefowl (the grey junglefowl, Ceylon junglefowl, </br>
10                 and green junglefowl).[1] Rooster and cock are terms for adult male birds, and a younger male may be called a cockerel.<
11                 A male that has been castrated is a capon. An adult female bird is called a hen, and a sexually immature female is</br>
12                 called a pullet. </p>
13      </body>
14  </html>

```

Figure 3 : Une partie de code HTML



Chicken

The chicken (Gallus domesticus) is a domesticated species that arose from the red junglefowl, originally from India. They have also partially hybridized with other wild species of junglefowl (the grey junglefowl, Ceylon junglefowl, and green junglefowl).[1] Rooster and cock are terms for adult male birds, and a younger male may be called a cockerel. A male that has been castrated is a capon. An adult female bird is called a hen, and a sexually immature female is called a pullet.

Figure 4 : La page HTML correspondante

HTML : the essentials

Mozilla developer network (MDN)

Guide de developpement : <https://developer.mozilla.org/en-US/>

Il s'agit d'une documentation de ressources qui aide à comprendre et familiariser avec les technologies HTML,CSS,JS.

Paragraph elements

La balise `<p>` est utilisée en HTML pour créer un paragraphe de texte. Elle permet de délimiter un bloc de texte qui forme un paragraphe distinct .

Voici un exemple de syntaxe simple de base pour la balise `<p>`

`<p>Contenu du paragraphe</p>`

```
<!--First paragraph-->
<p>For the culinary use of chickens, see Chicken as food. For other uses, see Chicken (disambiguation) and Chooks
(disambiguation)."Rooster" and "Roosters" redirect here. For other uses, see Rooster (disambiguation).
"Cockerel" redirects here. For the Fabergé egg, see Cockerel (Fabergé egg).</p>

<!--Second paragraph-->

<p>The chicken (Gallus domesticus) is a domesticated species that arose from the red junglefowl, originally from India.</br>
They have also partially hybridized with other wild species of junglefowl (the grey junglefowl, Ceylon junglefowl, </br>
and green junglefowl).[1] Rooster and cock are terms for adult male birds, and a younger male may be called a cockerel.<
A male that has been castrated is a capon. An adult female bird is called a hen, and a sexually immature female is</br>
called a pullet. </p>
```

Figure 5 : Utilisation de la balise `<p>`

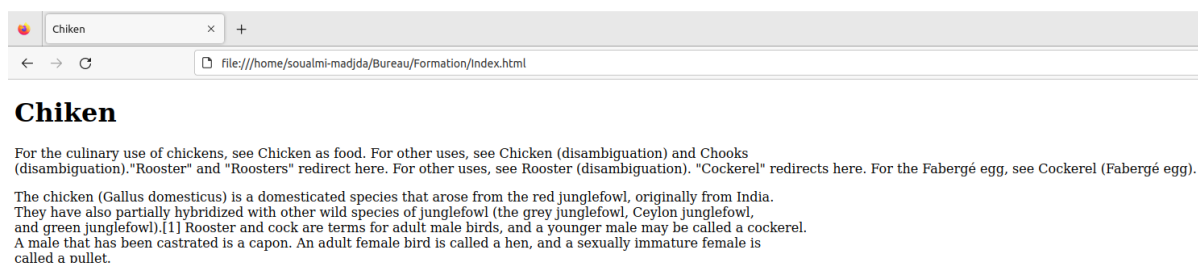


Figure 6 : Structure avec `<p>`

Headings

Les headings sont utilisés pour structurer et organiser le contenu en utilisant une hiérarchie de titres. Ils améliorent la compréhension, la navigation et l'accessibilité du contenu web.

Nous avons 6 niveaux de titres allant de h1 à h6 (Nous devons avoir au plus un H1 dans une page et cela devrait toujours être l'en-tête principal du niveau supérieur).

```

<body>
  <h1>Chicken</h1>

  <!--First paragraph-->
  <p>For the culinary use of chickens, see Chicken as food. For other uses, see Chicken (disambiguation) and Chooks </br>
    (disambiguation)."Rooster" and "Roosters" redirect here. For other uses, see Rooster (disambiguation).
    "Cockerel" redirects here. For the Fabergé egg, see Cockerel (Fabergé egg).</p>

</body>
<h2>Terminology</h2>
<p>Chicken may also mean a chick (see for example Hen and Chicken Islands).[20] In fact, chicken was originally <br>
  a term only for an immature, or at least young, bird.[when?] In older sources, and still often in trade and scientific ,<br>
  contexts chicken as a species are typically referred to as common fowl or domestic fowl.[21] </p>

<h2>Etymology</h2>
<p>According to Merriam-Webster, the term rooster (i.e. a roosting bird) originated in the mid- or late 18th century as a <br>
  euphemism to avoid the sexual connotation of the original English cock,[22][23][24] and is widely used throughout North <br>
  America. Roosting is the action of perching aloft to sleep at night.[25</p>

<h3>Behavior</h3>

<h4>Social behaviour</h4>
<p>See also: Empathy in chickens</p>

```

Figure 7 : Utilisation des balises H

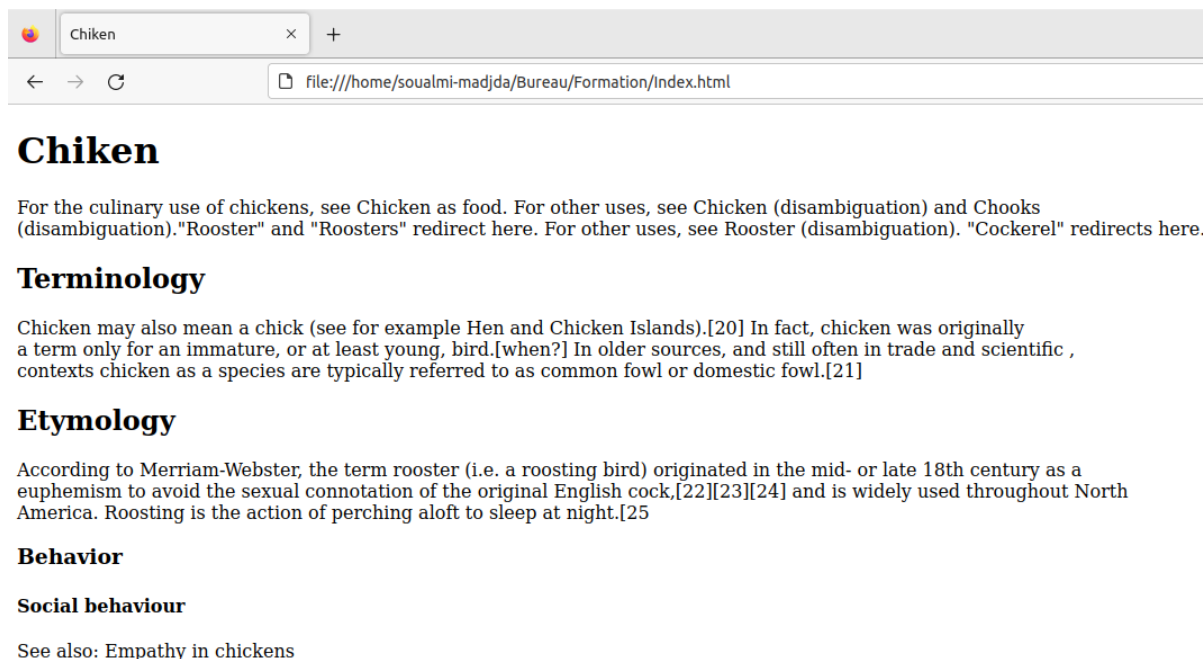


Figure 8 : Structure avec <h>

Important : Nous devrions jamais avoir un H5 par exemple sans H4 dans un H3.

Introduction to Chrome Inspector & HTML boilerplate

🙄 *Les pages HTML Créées, sont-elles valides ? Les balises correctement utilisées?*

🔧 NON , les documents sont imparfaits .

HTML skelton

- **<!DOCTYPE html>** : *Indicateur de navigateur qui indique que nous utiliserons HTML5 (la dernière version).*
- **L'en-tête < head>** : Aussi appelé élément de métadonnées contient essentiellement des éléments qui n'apparaissent pas sur la page.
- **Le corps du document <body>** : Représente tout le contenu de notre document.

Voici un exemple pour un document HTML valide

```
Index.html U X
Index.html > ...
1  <!DOCTYPE html> <!-- Indicateur de navigateur qui indique que nous utiliserons HTML5-->
2  <html lang="fr"> <!--root element -->
3      <head>
4          <title> Web page </title><!--Titre de notre document , qui s'affiche sur l'onglet du navigateur-->
5          <meta charset="UTF-8">
6          <meta name="Web page" content="Formation - udemy-->
7      </head>
8      <body>
9          <!--Contenu de la page web-->
10     </body>
11 </html>
```

Figure 9 : Code d'une page HTML valide

Important : Ils doivent apparaître une et une seule fois.

List elements

Type de liste	Code correspondant	Affichage
Liste à puces (<i>unordered list</i>)	<pre><!--Unordred list--> Item 1 Item 2 Item 3 </pre>	<ul style="list-style-type: none"> • Item 1 • Item 2 • Item 3
Liste numérotée (<i>ordered list</i>)	<pre><!--Ordred list--> Item 1 Item 2 Item 3 </pre>	<ol style="list-style-type: none"> 1. Item 1 2. Item 2 3. Item 3
Liste de définition (<i>definition list</i>)	<pre><dl><!-- definition list--> <dt>Item 1 </dt> <dd>Définition de l item 1</dd> <dt>Item 2 </dt> <dd>Définition de l item 2</dd> <dt>Item 3</dt> <dd>Définition de l item 3</dd> </dl></pre>	<ul style="list-style-type: none"> Item 1 Définition de l item 1 Item 2 Définition de l item 2 Item 3 Définition de l item 3

Important : Il peut y avoir des listes imbriquées

Note : Nous pouvons utiliser des attributs pour modifier le type de numérotation des listes ordonnées, ainsi que le style des puces elles-mêmes. Voici quelques exemples:

```
<ul type = "square"><!--Unordred list-->
  <li>Item 1</li>
  <li>Item 2</li>
  <li>Item 3</li>
</ul>

<ol type="i"><!--Ordred list-->
  <li>Item 1</li>
  <li>Item 2</li>
  <li>Item 3</li>
</ol>
```

Figure 10 : Changement des types de numérotation et des puces

- Item 1
 - Item 2
 - Item 3
-
- i. Item 1
 - ii. Item 2
 - iii. Item 3

Figure 11 : Styles de numérotation et des puces différent

Anchor tags

Le principe d'utilisation des liens sur HTML est de permettre aux utilisateurs de naviguer entre différentes pages web en cliquant sur des éléments de lien. Les liens sont créés à l'aide de l'élément `<a>` (anchor) et l'attribut `href` est utilisé pour spécifier la cible du lien, c'est-à-dire l'URL de la page vers laquelle vous souhaitez diriger les utilisateurs.

Voici un exemple simple de code HTML pour créer un lien :

```
<p>Pour bien comprendre le principe des liens sur html  
  <a href="https://www.udemy.com/course/the-web-developer-bootcamp/learn/lecture/21915884#overview">Cliquez ici</a>  
  <!-- balise pour l'insertion d'un lien -->  
</p>  
<p>Pour accéder à une page du même dossier <a href="Reference.html">autre page web</a></p>
```

Figure 12 : Utilisation des liens

Pour bien comprendre le principe des liens sur html [Cliquez ici](https://www.udemy.com/course/the-web-developer-bootcamp/learn/lecture/21915884#overview)

Pour accéder à une page du même dossier [autre page web](#)

Figure 13 : la page web contenant des liens

Images

Sur HTML, les images sont utilisées pour afficher des éléments visuels sur une page web. Le principe d'utilisation des images est assez simple :

1. Balise `` : Pour insérer une image, vous devez utiliser la balise `` avec certains attributs obligatoires.
2. Attribut "src" : Cet attribut spécifie l'URL ou le chemin d'accès de l'image. Par exemple : ``.
3. Attribut "alt" : Cet attribut définit un texte alternatif qui sera affiché si l'image ne peut pas être chargée ou si le lecteur d'écran est utilisé. Il est recommandé d'utiliser cet attribut pour des raisons d'accessibilité. Par exemple : ``.
4. Attributs optionnels : Il existe d'autres attributs optionnels que vous pouvez utiliser, tels que "width" (largeur de l'image), "height" (hauteur de l'image), "title" (texte affiché au survol de l'image), etc.

Voici un exemple complet d'utilisation d'une image dans HTML :

```
<!--Insertion d'une image localement existante-->
<!--<caption>"La dernière version de HTML"</caption>--> <!--Légende d'une image-->

<!--Insertion d'une image qui n'existe pas en local-->
```

Figure 14 : Utilisation des images dans HTML



Figure 15 : Images en HTML

Comments

Les commentaires en HTML sont utilisés pour ajouter des annotations ou des explications dans le code HTML. Voici comment utiliser les commentaires en HTML :

```
<!-- C'est un commentaire sur une seule ligne -->
```

HTML : Next steps & semantic

What HTML5 actually is ?

La dernière version stable du [HTML](#), HTML5, transforme le HTML qui était un simple balisage pour structurer des documents en une plate-forme complète de développement d'applications. Parmi ses autres caractéristiques, HTML5 comporte de nouveaux éléments et des [API JavaScript](#) pour améliorer le stockage, le multimédia et l'accès au matériel.

Block VS inline elements , `` and `<div>` elements

`<div>` : l'élément de division du contenu

L'élément HTML **`<div>`** (ou division) est le conteneur générique du contenu du flux. Il n'a aucun effet sur le contenu ou la mise en page tant qu'il n'est pas mis en forme d'une manière quelconque à l'aide de CSS.

Exemple

```
<div> <!--bloc pour regrouper les éléments lien-->
  <p>Pour accéder à google
    <a href="google.com">Cliquez ici</a>
    <!--balise pour l'insertion d'un lien-->
  </p>
  <p>Pour accéder à une page du même dossier <a href="Reference.html">autre page web</a></p>
</div>
```

Figure 16 : Code html correspondant à l'élément `<div>`

``

L'élément HTML **``** est un conteneur générique en ligne (*inline*) pour les contenus phrasés. Il ne représente rien de particulier. Il peut être utilisé pour grouper des éléments afin de les mettre en forme (grâce aux attributs `class` ou `id` et aux

règles [CSS](#)) ou parce qu'ils partagent certaines valeurs d'attribut comme [lang](#). Il doit uniquement être utilisé lorsqu'aucun autre élément sémantique n'est approprié.

`` est très proche de l'élément `<div>`, mais l'élément `<div>` est [un élément de bloc](#), alors que `` est [un élément en ligne](#).

Exemple

```
<div> <!--bloc pour regrouper les éléments lien-->
  <p>Pour accéder à <span>google</span> <!--Pour traiter cette partie de texte différemment -->
  <!--balise pour l'insertion d'un lien-->
</p>
<p>Pour accéder à une page du même dossier <a href="Reference.html">autre page web</a></p>
</div>
```

Figure 17 : Code HTML correspondant à l'élément ``

An odd assortment of elements : HR ,BR , SUP & SUB

`<hr>` : l'élément de rupture thématique (règle horizontale)

L'élément HTML `<hr>` représente un changement thématique entre des éléments de paragraphe (par exemple, un changement de décor dans un récit, un changement de sujet au sein d'une section).

Exemple

```
<body>

  <!--Contenu de la page web-->
  <h1>HTML5</h1>
  <hr>
```

Figure 18 : Code HTML correspondant à l'élément `<hr>`

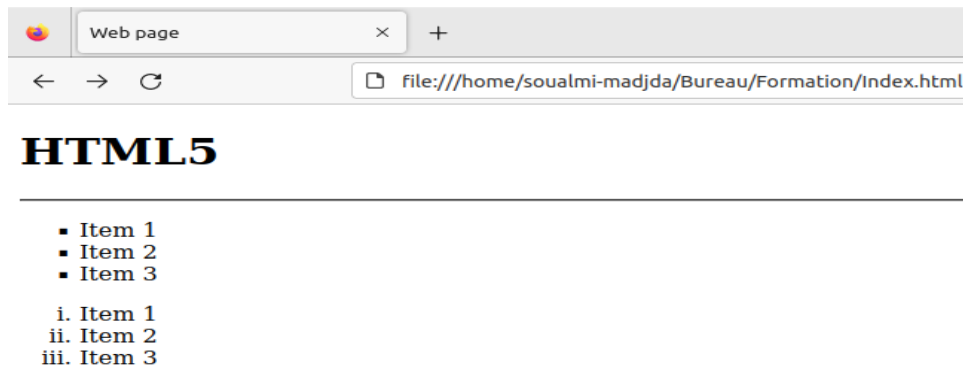


Figure 19 : Résultat d'utilisation de l'élément <hr>

 : l'élément de saut de ligne

L'élément HTML **
** crée un saut de ligne (un retour chariot) dans le texte. Il s'avère utile lorsque les sauts de ligne ont une importance (par exemple lorsqu'on écrit une adresse ou un poème).

Exemple

```
<!--Exemple d'un poeme pour utiliser <br> -->
<h2>Poème</h2>
<p>
  Tu as un peu de temps,<br>
  La lecture peut se faire ici.<br>
  Sans être dérangé.<br>
  Des mots pour s'évader.<br>
  <br>
  Tu as un peu de temps,<br>
  Laisse les rimes te bercer.<br>
  Mots changeant une réalité,<br>
  Une minute,ou plus longtemps.<br>
</p>
```

Figure 20 : Utilisation de l'élément

Poème

Tu as un peu de temps,
La lecture peut se faire ici.
Sans être dérangé.
Des mots pour s'évader.

Tu as un peu de temps,
Laisse les rimes te bercer.
Mots changeant une réalité,
Une minute, ou plus longtemps.

Figure 21 : Résultat d'utilisation de l'élément

<sup> : l'élément de mise en exposant

L'élément HTML **<sup>** est utilisé, pour des raisons typographiques, afin d'afficher du texte en exposant (plus haut et généralement plus petit) par rapport au bloc de texte environnant.

<sub> : l'élément de mise en indice

L'élément HTML **<sub>** est utilisé, pour des raisons typographiques, afin d'afficher du texte en indice (sous la ligne de base et généralement plus petit) par rapport au bloc de texte environnant.

Exemple

```
<!--Utilisation des élément sup & sub -->  
<p>Almost every developer's favorite molecule is C<sub>8</sub>H<sub>10</sub>N<sub>4</sub>  
  
<p>The <em>Pythagorean theorem</em> is often expressed as the following equation:</p>  
<p><var>a<sup>2</sup></var> + <var>b<sup>2</sup></var> = <var>c<sup>2</sup></var></p>
```

Figure 22 : Code correspondant aux éléments <sup> & <sub>

SUB & SUP

Almost every developer's favorite molecule is C₈H₁₀N₄

The *Pythagorean theorem* is often expressed as the following equation:

$$a^2 + b^2 = c^2$$

Figure 23 : Résultat d'utilisation des éléments <sup> & <sub>

Entity code

Les entités HTML sont des codes spéciaux utilisés en HTML pour représenter des caractères spéciaux et éviter les conflits de syntaxe, assurant ainsi un affichage correct dans les navigateurs. Elles incluent des entités pour les caractères spéciaux, réservés, ainsi que des symboles mathématiques et monétaires. Leur utilisation garantit un contenu bien formaté et conforme aux normes du langage HTML.

Voici un lien pour voir tous les codes

<https://html.spec.whatwg.org/multipage/named-characters.html>

Exemple

```
<!--Entities-->
<p> 1 &lt; 2 </p> <!--C'est le symbole < -->
```

Figure 24 : code correspondant au symbole "<"

Entities

1 < 2

Figure 25 : Résultat d'utilisation d'entité

HTML : Next steps & semantic

Intro to semantic markup

Playing with semantic elements

Screen reader demonstration

Vs code tip : emmet

Resources



<https://www.udemy.com/course/the-web-developer-bootcamp/learn/lecture/22587518#overview>



<https://github.com/SWLMadjda/WebDevelopment-Udemy->



<https://trello.com/b/0PVRE1XQ/web-developer-bootcamp>