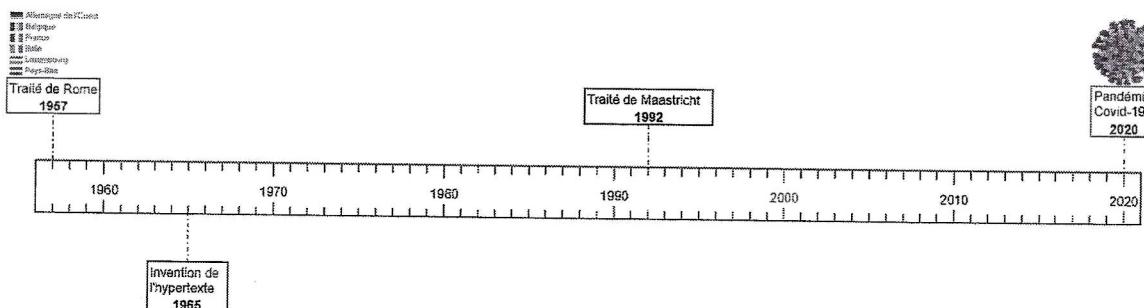




## 1 Introduction au WEB

### Un peu d'histoire :

- 1957 : Traité de Rome/CEE
- 1961 : Mur de Berlin
- 1966 : Révolution culturelle en Chine
- Printemps/mai 1968 : Révoltes étudiantes à Paris, Rome, Prague...
- 1969 : Premiers pas sur la lune
- 1979 : Proclamation de la République islamique en Iran
- 1981 : Premier ordinateur portable
- 1986 : Catastrophe nucléaire de Tchernobyl
- 1989 : chute du mur de Berlin et fin de la Guerre froide
- 1991 : Guerre du Golfe
- 1992 : Traité de Maastricht (citoyenneté européenne)
- 1994 : Fin de l'Apartheid
- 1998 : Apparition de Google
- 1999 : Première mise en circulation de l'euro
- 2001 : Attentat du World Trade Center à New York
- 2001 : Premier iPod
- 2003 : Guerre en Irak
- 2011 : Catastrophe nucléaire à FUKUSHIMA
- 2015 : Attentats meurtriers en France « Charlie Hebdo »



Le **World Wide Web**, plus communément appelé « **Web** » a été développé au **CERN** (Conseil Européen pour la Recherche Nucléaire) par le Britannique **Sir Timothy John Berners-Lee** et le Belge **Robert Cailliau** au début des années 90. À cette époque les principaux centres de recherche mondiaux étaient déjà connectés les uns aux autres, mais pour faciliter les échanges d'information Tim Berners-Lee met au point le système **hypertexte**. Le système hypertexte permet, à partir d'un document, de consulter d'autres documents en cliquant sur des mots clés. Ces mots cliquables sont appelés « **hyperliens** » et sont souvent soulignés et en bleu. Ces hyperliens sont plutôt connus aujourd'hui sous le simple terme de « **liens** ».

The screenshot shows the homepage of the World Wide Web project. The URL in the address bar is [info.cern.ch/hypertext/WWW/TheProject.html](http://info.cern.ch/hypertext/WWW/TheProject.html). The page title is "World Wide Web". The main text reads: "The WorldWideWeb (W3) is a wide-area hypermedia information retrieval initiative aiming to give universal access to a large universe of documents. Everything there is online about W3 is linked directly or indirectly to this document, including an executive summary of the project, Mailing lists, Policy, November's W3 news, Frequently Asked Questions, What's out there? Pointers to the world's online information, subjects, W3 servers, etc. Help on the browser you are using. Software Products A list of W3 project components and their current state. (e.g. Line Mode, X11 Viola, NeXTStep, Servers, Tools, Mail robot, Library) Technical Details of protocols, formats, program internals etc Bibliography Paper documentation on W3 and references. People A list of some people involved in the project. History A summary of the history of the project. How can I help? If you would like to support the web.. Getting code Getting the code by anonymous FTP, etc.

Cette première page web est toujours consultable à l'adresse suivante :  
<http://info.cern.ch/hypertext/WWW/TheProject.html>

Tim Berners-Lee développe le premier navigateur web (logiciel permettant de lire des pages contenant des hypertextes), il l'appelle simplement **WorldWideWeb**. Il faudra attendre 1993 et l'arrivée du navigateur web "NCSA Mosaic" pour que le web commence à devenir populaire en dehors du petit monde de la recherche.

Techniquement le web se base sur trois éléments :

- le protocole **HTTP (HyperText Transfer Protocol)**
- les adresses web **URL (Uniform Resource Locator)**
- le langage de description **HTML (HyperText Markup Language)**

Nous aurons l'occasion de revenir sur ces trois éléments.

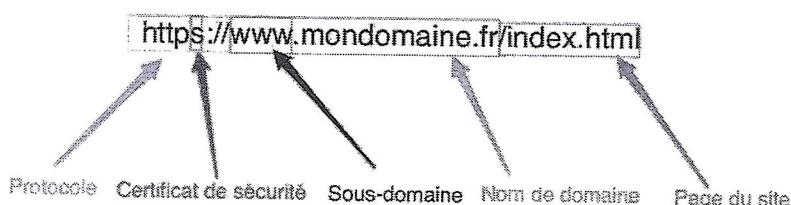
Une chose très importante à bien avoir à l'esprit : Il ne faut pas confondre « **web** » et « **internet** ». Le web s'appuie sur internet qui est un réseau de réseaux utilisant le protocole IP alors que, comme nous venons de le voir, le web est la combinaison de trois technologies : **HTTP**, **URL** et **HTML**. D'ailleurs, il n'y a pas que le web sur internet, on trouve par exemple, les emails avec le protocole **SMTP** (Simple Mail Transfer Protocol) et les transferts de fichiers avec le protocole **FTP** (File Transfer Protocol).

## 2 Les URL

### 2.1 *NOMS DE DOMAINES*

Un nom de domaine permet de situer un serveur sur Internet. Un équipement, appelé **DNS – Domain Name System** – permet d'associer une adresse IP à chaque nom de domaine. Le DNS fait le lien entre le Web et Internet. Les noms de domaine ont été conçus pour simplifier l'utilisation de l'internet : il est plus facile de retenir et taper un nom de domaine qu'une adresse IP. Par exemple, si vous souhaitez accéder à Google, vous pouvez taper <http://172.217.19.227>, mais l'URL de Google est sans doute plus simple à retenir : <http://www.google.fr>

*Exemple d'URL*



### 2.2 *EXTENSIONS DE NOMS DE DOMAINES*

Il existe en grand nombre d'extensions pour les noms de domaines. Certaines ont une portée générale (.com, .biz, .gouv) et ne sont donc pas rattachées à une zone géographique donnée. D'autres ont une portée nationale (.fr, .be, .uk) et sont rattachées à un pays ou une zone géographique donnée. Les extensions à portée générale sont gérées par des sociétés privées (.com, .org), des organisations (.aero, .museum) ou par le gouvernement américain (.gov, .mil).

### 2.3 *URL (UNIFORM RESOURCE LOCATOR)*

Dans la barre d'adresse de votre navigateur vous trouverez, quand vous visitez un site, une adresse du type :

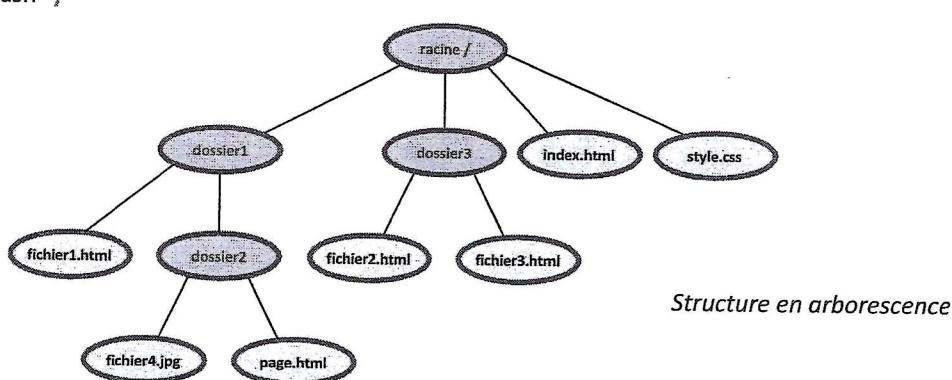
<http://www.site.fr/dossier1/dossier2/page.html>

Cette adresse s'appelle une **URL (Uniform Resource Locator)**.

Elle permet d'identifier une ressource disponible sur un réseau, par exemple un fichier.

Une URL indique « l'endroit » où se trouve la ressource sur un ordinateur. Un fichier peut se trouver dans un dossier qui peut lui-même se trouver dans un autre dossier. On parle d'une structure en arborescence, car elle ressemble à un arbre (à l'envers).

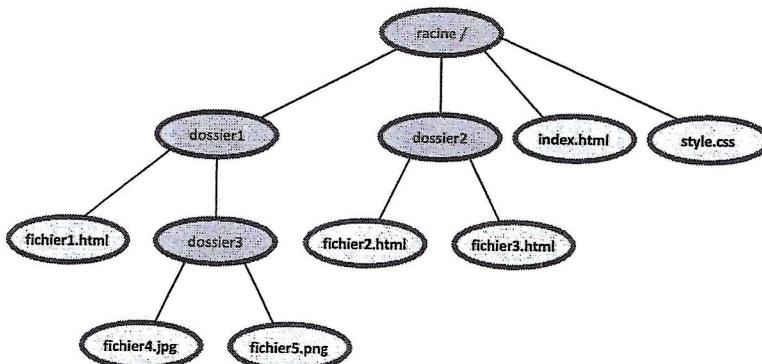
Comme vous pouvez le constater sur l'image ci-dessous, la base de l'arbre s'appelle la **racine** de l'arborescence et elle se représente par un slash /



## 2.4 CHEMIN ABSOLU OU CHEMIN RELATIF ?

Pour indiquer la position d'un fichier, ou celle d'un dossier, dans l'arborescence, il existe deux méthodes :

- indiquer un chemin absolu
- indiquer un chemin relatif



Le **chemin absolu** indique « le chemin » complet depuis la racine.

Par exemple le chemin absolu du fichier **fichier4.jpg** est : /dossier1/dossier3/fichier4.jpg

Remarquez que nous démarrons bien de la racine /. Tout chemin absolu commence par le / de la racine

Attention, les symboles de séparation sont aussi des / .

Imaginons maintenant que le fichier **fichier1.html** fasse appel au fichier **fichier4.jpg**. Il est possible d'indiquer le chemin non pas depuis la racine, mais depuis le **dossier1** qui accueille le **fichier1.html** et qui accueille aussi le **dossier3** qui contient lui-même le **fichier4.jpg**.

Nous parlerons alors de **chemin relatif** qui est dans ce cas : dossier3/fichier4.jpg

Remarquez l'absence du slash / au début du chemin, c'est ce qui nous permet de distinguer un chemin relatif et un chemin absolu.

Imaginons maintenant que le fichier **fichier3.html** fasse appel au fichier **fichier5.png**. Il faut indiquer le chemin relatif du fichier **fichier5.png** depuis l'intérieur du **dossier2**. Comment faire ?

Il faut d'abord « remonter » d'un niveau dans l'arborescence pour se retrouver, dans notre exemple, au niveau de la racine et ainsi pouvoir repartir vers la bonne « branche », celle du **dossier1**. Pour ce faire il faut utiliser deux points : ..

..../dossier1/dossier3/fichier5.png

Il est tout à fait possible de remonter de plusieurs niveaux. Par exemple, ../../ depuis le **dossier3** permet de remonter à la racine.

### Remarque :

L'écriture des chemins, avec des slash / comme séparateurs, est propre aux systèmes dits « UNIX », par exemple GNU/Linux ou encore Mac OS.

Alors que sous Windows, ce n'est pas le slash qui est utilisé, mais l'antislash \

Pour les chemins réseau, et donc le web, pas de problème, c'est toujours le slash qui est utilisé.

## 3 Pages Web : HTML et CSS

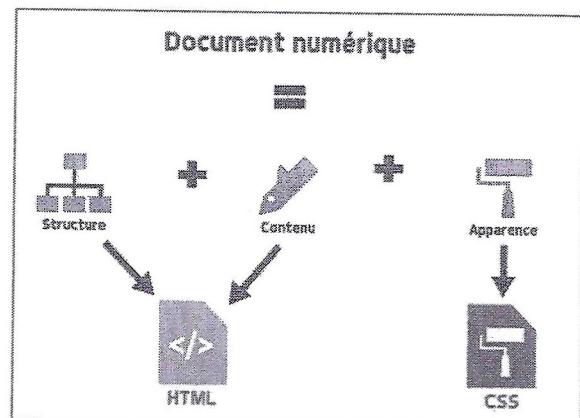
Nous allons nous intéresser à un acteur fondamental du développement web, le couple **HTML + CSS** .

**HTML** Hyper Text Markup Language

**CSS** Cascading Style Sheets

Le code **HTML** se charge de la **structure** et du **contenu** d'une page Web, les titres, les paragraphes, les images, les tableaux, ... etc.

La feuille de style **CSS** se charge de la **présentation** de la page Web, de son style graphique : arrière-plan, type, taille et couleur des polices de caractères utilisées, bordures, ... etc.



## 4 HTML

Dans un premier temps, nous allons exclusivement nous intéresser au HTML. Voici la définition que nous en donne Wikipedia :

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).

Pour l'instant, nous allons retenir deux éléments de cette définition : « conçu pour représenter les pages web »  
« 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) : il n'est pas question de conditions, de boucles... C'est un langage de description.

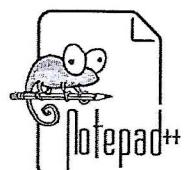
Dans votre navigateur, il est possible de lire le code source de la page WEB sur laquelle vous vous trouvez avec un clic droit sur la page puis « afficher la source » ou « afficher le code source ». Faites l'expérience !

Pour construire un site WEB, il existe des sites et des logiciels dits WYSIWYG (What You See Is What You Get) mais ils produisent du code parfois inutilement complexe et non optimisé et surtout ils ne permettent pas d'apprendre le langage HTML.

Il est possible d'utiliser des sites en lignes comme <http://jsfiddle.net/>.

Mais il est suffisant d'utiliser un simple bloc-notes ou un éditeur de code source qui gère la coloration syntaxique (comme **notepad++** par exemple) : les mots-clés du langage sont ainsi reconnus et mis en évidence ce qui facilite la lecture et la production de code.

Pour disposer de **notepad++** sur votre PC, allez sur le site <https://notepad-plus-plus.org/> pour télécharger la dernière version puis procéder à son installation.



*Voici un premier exemple édité avec notepad++*

...

...

*... et le rendu avec le navigateur Firefox*

F1\_NSI10 - HTML-CSS/Exemple.html - Notepad++  
Édition Recherche Affichage Encodage Langage Paramètres Outils Macro Exécution Modules d'extension Documents 2  
Exemple.html  
1 <!DOCTYPE html>  
2 <html>  
3 <head>  
4 <!-- En-tête de la page -->  
5 <meta charset="utf-8" />  
6 <title>Mon premier site</title>  
7 </head>  
8 <body>  
9 <!-- Corps de la page -->  
10 <h1>Ceci est un titre (h1)</h1>  
11 <h2>Ceci est un sous-titre (h2)</h2>  
12 <p>Premier paragraphe</p>  
13 <p>Second paragraphe</p>  
14 <p>Troisième paragraphe</p>  
15 </body>  
16 </html>  
Hyper Text Mario length: 403 lines: 17 Ln:1 Col:1 Sel:0|0 Windows (CRLF) UTF-8 INS

Mon premier site  
file:///F:/NSI/10 - HTML-CSS/Exemple.html  
Les plus visités LYCEE PAUL-LANGEVIN...  
  
**Ceci est un titre (h1)**  
**Ceci est un sous-titre (h2)**  
Premier paragraphe  
Second paragraphe  
Troisième paragraphe

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

Une balise ouvrante est de la forme **<nom\_de\_balise>**, une balise fermante est de la forme **</nom\_de\_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 espèce d'importance, la seule question à se poser ici est : « Ai-je bien refermé toutes les balises que j'ai ouvertes ? »

Pour terminer avec les généralités sur les balises, il est important de savoir que les balises doivent être correctement imbriquées : Une structure du type : **<balise1><balise2> ... </balise1></balise2>** est interdite, la balise **<balise2>** a été ouverte après la balise **<balise1>**, elle devra donc être refermée avant la balise1.

En revanche, l'enchaînement suivant est correct : **<balise1><balise2> ... </balise2></balise1>**

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

Il est important de respecter les indentations, effectuées avec la touche de tabulation.

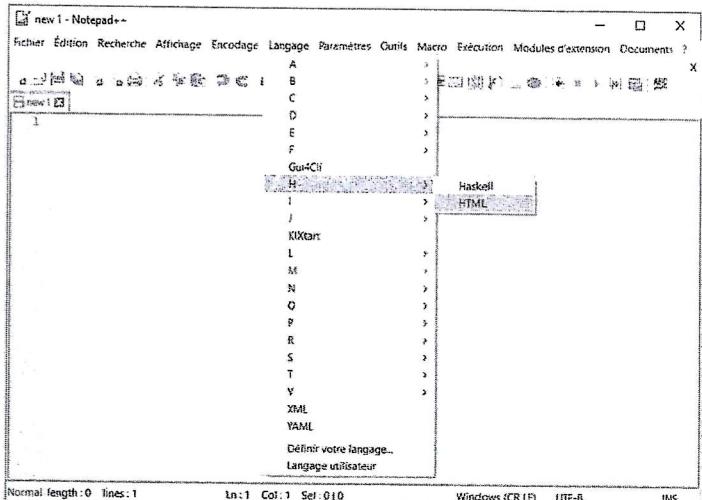
Elles permettent de structurer le code et rend sa lecture beaucoup plus facile.

## Activité 1 :

Utiliser **notepad++** pour éditer le code **HTML** fourni dans l'exemple de la page précédente et valider votre travail en observant le résultat obtenu dans un navigateur.

Lorsque vous lancez **notepad++**, pensez à sélectionner le langage que vous utilisez, afin de bénéficier de la coloration syntaxique, ici le **HTML**, ainsi que de la sauvegarde du fichier avec l'extension appropriée.

L'indentation utilisée sur les différentes lignes est à respecter scrupuleusement, elle permet aussi de visualiser facilement la structure de la page Web, l'en-tête (head) et le corps (body).



Quelques exemples de balises :

balise	utilisation
<!doctype html>	Cette balise orpheline indique au navigateur que nous utiliserons la dernière version du HTML, c'est-à-dire HTML 5.
<html> ...</html>	Ces balises sont obligatoires, elles délimitent l'ensemble du code HTML.
<head> ... </head>	Ces balises délimitent ce que l'on appelle l'en-tête ; entre ces balises, ce sont des informations sur la page et non pas des éléments qui seront affichés sur la page WEB.
<meta charset = "utf-8">	Cette balise orpheline se place dans l'en-tête et qui indique au navigateur le type d'encodage de la page WEB, ici utf-8 : l'encodage le plus courant actuellement.
<title> ... </title>	Titre du site tel qu'il apparaîtra sur l'onglet du navigateur
<body> ... </body>	Ces balises délimitent le « corps » de la page, ce qui sera effectivement sur la page.
<h1> ... </h1>	Un titre de niveau 1 (on peut utiliser des sous-titres avec h2, ... , h6)
<p> ... </p>	Un paragraphe
 	Cette balise orpheline permet un retour à la ligne sans sortir du paragraphe
<strong>...</strong>	Ces balises délimitent une zone de texte importante dans le thème du site (souvent présenté en gras mais surtout utile pour les moteurs de recherche)
<b> ... </b>	Ces balises délimitent une zone de texte à écrire en gras (« b » pour « bold »)
<i> ... </i>	Ces balises délimitent une zone de texte à écrire en italique.
<u> ... </u>	Ces balises délimitent une zone de texte à souligner.
<sub> ... </sub>	Ces balises délimitent une zone de texte à écrire en indice .
<sup> ... </sup>	Ces balises délimitent une zone de texte à écrire en exposant .
<b>Des balises particulières :</b>	
<div> ... </div>	Il s'agit d'un conteneur générique qui ne possède aucune sémantique. Il est utilisé uniquement pour organiser du contenu. Il est de type « block » (retour à la ligne)
<span> ... </span>	Il s'agit d'un conteneur générique qui ne possède aucune sémantique. Il est utilisé pour organiser du contenu. Il est de type « inline » (pas de retour à la ligne)

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

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

Par exemple :

```
<html lang="fr" > ... </html>
```

permet d'indiquer la langue utilisée sur le site.

Il existe beaucoup d'attributs différents, nous pouvons déjà signaler deux exemples avec les attributs **id** (id pour « identifiant ») et **class**. Nous verrons l'intérêt de ces attributs notamment quand nous aborderons les fichiers **CSS**.

Quelques balises supplémentaires :

- Pour créer un lien vers une autre page WEB, une ancre dans la même page ou envoyer un message électronique (mail) : la balise `<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 apparaissent 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.

`<a href="page1.html">Cliquez ici pour vous rendre sur mon autre page</a>`

Par exemple : `<a href="http://langevin.lyc.ac-amiens.fr/">Lien vers le site du lycée</a>`

Entre les balises ouvrante et fermante, il y a le texte du lien qui s'affichera à l'écran (celui qui sera souligné et de couleur bleue). La balise peut sans problème se trouver en plein milieu d'un paragraphe.

- On peut utiliser la balise `<a>` pour faire un **lien vers un endroit particulier de la page Web**, ce point est appelé une ancre et doit être défini par : `<a name="ancre">bla bla bla bla</a>`

On crée un lien vers cette ancre en utilisant un « # », de la manière suivante :

`<a href="#ancre">lien vers bla bla bla bla</a>`

- On peut utiliser la balise `<a>` pour créer un lien pour envoyer un message électronique à une adresse prédéfinie grâce à l'attribut `mailto` : `<a href="mailto:mon-adresse@domaine.fr">Écrivez-moi !</a>`

Sur la page web, l'action de cliquer sur « Écrivez-moi ! » ouvrira le logiciel de messagerie.

- Pour insérer une image : la balise `<img>`

La balise `<img>` sert à insérer une image : ``

La balise `<img>` est à la fois ouvrante et fermante comme la balise `<br>`. Elle possède deux attributs :

❖ `src` qui doit être égal au nom du fichier image, avec son extension, et au besoin avec le chemin si le fichier se trouve dans un autre dossier, comme indiqué dans le paragraphe sur les « URL ».

❖ `alt` qui doit être égal à un texte qui décrit votre image. Cet attribut est utilisé notamment par les systèmes de description des pages web utilisées par les non-voyants et permet de définir un texte à afficher si le lien vers l'image est faux : il faut donc prendre l'habitude de renseigner systématiquement cet attribut.

- Pour insérer une liste : les balises `<ul> ... </ul>` et `<ol> ... </ol>`

La balise `<ul>` sert à insérer une **liste non numérotée** (« unordered list »), chaque élément de la liste sera délimité par les balises `<li> ... </li>` :

```
<ul>
    <li>un premier élément</li>
    <li>un deuxième élément</li>
</ul>
```

La balise `<ol>` s'utilise comme la balise `<ul>` et sert à insérer une **liste numérotée** (« ordered list »).

- Pour insérer un formulaire : la balise `<form> ... </form>`

Les formulaires sont des éléments importants des sites internet, ils permettent à l'utilisateur de transmettre des informations.

Un formulaire devra être délimité (idéalement) par une balise `form` :

`<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.

- Pour insérer un bouton : la balise `<button>`

La balise `<button>` est utilisée afin de créer un contrôle interactif qui pourra être utilisé dans un formulaire.

`<button type="button"> Cliquer ici ! </button>`  
`<button type="submit"> Soumettre </button>`

## Activité 2 :

Ajouter dans le **body** de votre fichier **Exemple.html** quelques balises proposées ci-dessus et les tester.  
Par exemple, vous pouvez insérer une image, un lien vers un site externe et une liste numérotée.

*Envie d'en savoir plus sur le HTML ?*

Faites un tour ici : <https://www.w3schools.com/html/default.asp>

The screenshot shows a web browser window displaying the w3schools.com website. The title bar reads "Tutoriel HTML". The address bar shows the URL "https://www.w3schools.com/html/default.asp". The page header includes the w3schools logo and the text "LE PLUS GRAND SITE DE DÉVELOPPEURS WEB AU MONDE". A "SIDENFIER" button is visible in the top right corner. The main content area features a large heading "Tutoriel HTML". Below it are two buttons: "< Accueil" on the left and "Prochain >" on the right. To the left of the main content is a sidebar titled "ACCEUIL HTML" containing a list of HTML topics. In the center, there is a section about HTML version 5, featuring a "HTML 5" logo and text about its standardization and ease of learning. At the bottom of this section is a call-to-action button "Commencez à apprendre le HTML maintenant >". The right side of the page has a sidebar with the text "Apprentissage facile avec HTML "Essayez-le vous-même"".

*Notes :*

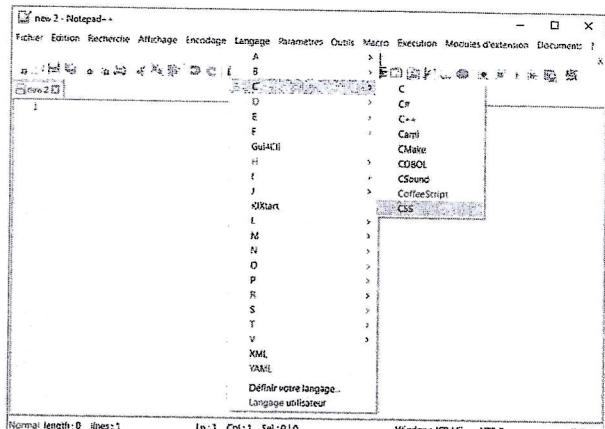
*REMARQUE*

## 5 CSS

Le HTML n'a pas été conçu pour gérer la mise en page. Le HTML s'occupe 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).

On crée un nouveau fichier appelé par exemple, « **style.css** » qu'on va placer dans le même dossier que le fichier HTML « **index.html** » qui va l'utiliser et y faire référence.

Lorsque vous créerez ce nouveau fichier avec **notepad++**, pensez à sélectionner le langage que vous utilisez, afin de bénéficier de la coloration syntaxique, ici le **CSS**, ainsi que de la sauvegarde du fichier avec l'extension appropriée.



Voici un exemple de fichier CSS :

A screenshot of Notepad++ displaying a CSS file named 'style.css'. The code defines styles for the 'body', 'h1', 'h2', and 'special' classes. The 'body' style sets a lightblue background, sans-serif font, 18px size, black color, and margins of 10px, 50px, 20px, 50px. The 'h1' style sets a serif font, 32px size, white color, blue background, and center text-align. The 'h2' style sets a Verdana font, 24px size, green color, italic font-style, and italic font-style. The 'special' class has italic font-style and #606060 color. The '.important' class has red color.

```
1 body { /* Indique que le style défini ci-dessous correspond au texte courant */
2     background-color: lightblue; /* Couleur de l'arrière plan : bleu clair (code #ADD8E6) */
3     font-family: sans-serif; /* Nom de police de caractères */
4     font-size: 18px; /* Taille de la police de caractères : 18 pixels */
5     color: black; /* Couleur de la police de caractères : noir (code #000000) */
6     margin: 10px 50px 20px 50px; /* Marge autour du texte Haut/Droite/Bas/Gauche : ici H=10 D=50 B=20 G=50 */
7 }
8
9 h1 { /* Indique que le style ci-dessous correspond au titre de niveau 1 (h1) */
10    font-family: serif; /* Nom de la police de caractères */
11    font-size: 32 px; /* Taille de la police de caractères : 32 pixels */
12    color: white; /* Couleur de la police de caractères : blanc (code #FFFFFF) */
13    background-color: blue; /* Couleur de l'arrière plan : bleu (code #0000FF) */
14    text-align: center; /* Alignement du texte : centré */
15 }
16
17 h2 { /* Indique que le style ci-dessous correspond aux titres de niveau 2 (h2) */
18    font-family: Verdana; /* Nom de la police de caractères */
19    font-size: 24px; /* Taille de la police de caractères : 24 pixels */
20    color: green; /* Couleur de la police de caractères : vert (code #008000) */
21    font-style: italic; /* Style de la police de caractères */
22 }
23
24 #special { /* Indique le style particulier associé à l'identifiant "special" */
25    font-style: italic; /* Style de la police de caractères */
26    color: #606060; /* Couleur de la police de caractères : gris (code #606060) */
27 }
28
29 .important { /* Indique le style associé à la classe "important" */
30    color: red; /* Couleur de la police de caractères : rouge (code #FF0000) */
31 }
```

Cascade Style Sheets File

length: 1739 lines: 31

Ln:1 Col:1 Sel:0|0

Windows (CR LF) UTF-8 INS

Dans cet exemple, les propriétés définies dans le paragraphe « **body** » seront appliquées au texte courant du corps de la page HTML. Le paragraphe « **h1** » indique toutes les propriétés qui seront appliquées au titre de niveau 1 ... et on fait de même pour les sous-titres « **h2** ». L'avantage de cette méthode est qu'il est très facile de modifier l'apparence complète du site en n'agissant que sur quelques lignes du fichier CSS.

Il est possible de définir un style particulier pour **un seul** des titres, des sous-titres ou des paragraphes en ayant ajouté, dans le fichier HTML, un **identifiant** en attribut dans la balise de cet élément.

L'identifiant ne peut être utilisé qu'une fois dans le fichier HTML !

```
<div id="special"> ... </div>
```

Dans le fichier CSS, on définit toutes les propriétés de cet identifiant "special", précédé d'un « **#** », comme montré dans le fichier donné en exemple ci-dessus.

On peut aussi utiliser un même style pour **plusieurs éléments** du site en indiquant une **classe** en attribut des balises correspondantes : une même « **classe** » peut apparaître à plusieurs endroits du fichier HTML :

```
<div class="important"> Paragraphe important </div>
```

...

```
<div class="important"> Autre paragraphe important </div>
```

Dans le fichier CSS, on définit cette classe, précédée d'un « **.** », comme montré dans le fichier donné en exemple ci-dessus.

## TRÈS important :

Pour que le fichier CSS soit pris en compte dans l'affichage de la page HTML, on doit ajouter une balise orpheline `<link>` entre les balises `<head> ... </head>` du fichier HTML pour indiquer que le style est délégué à ce fichier « `style.css` » :

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

Voici un fichier HTML qui utilise le fichier CSS précédent :

Repérez, en les surlignant, les attributs qui font référence au fichier CSS

```

<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8">
    <title>Spécialité NSI</title>
    <link type="text/css" rel="stylesheet" href="style.css">
</head>
<body>
    <h1>Présentation de la spécialité NSI</h1>
    <p>La spécialité <strong>"Numérique et Sciences Informatiques"</strong> a ouvert en septembre 2019 au lycée Paul Langevin de Beauvais.</p>
    <h2>Le programme de NSI</h2>
    <ul>
        <li id="special">Histoire de l'informatique (partie transversale qui se décline dans chaque partie)</li>
        <li class="important">Représentation des données : type et valeurs de base</li>
        <li class="important">Représentation des données : types construits</li>
        <li class="important">Traitement des données en tables</li>
        <li class="important">Interactions entre l'homme et la machine sur le Web</li>
        <li class="important">Architectures matérielles et systèmes d'exploitation</li>
        <li class="important">Langages et programmation</li>
        <li class="important">Algorithmique</li>
    </ul>
    Le <a href="https://cache.media.education.gouv.fr/file/SP1-MEN-22-1-2019/26/8/spe633_annexe_1062268.pdf">programme</a> officiel de la spécialité NSI<br>
    <br><h2>Liens utiles</h2>
    En savoir plus sur le <a href="https://www.w3schools.com/html/">langage HTML</a><br>
    En savoir plus sur le <a href="https://www.w3schools.com/css/">langage CSS</a><br>
    En savoir plus sur le <a href="https://www.w3schools.com/python/">langage Python</a><br>
    Pour obtenir le nom ou le code RVB des <a href="http://www.proftnj.com/RVB3.htm">couleurs</a><br>
    <br>
    Lien vers le site du <a href="http://langevin.lyc.ac-amiens.fr/">lycée Paul Langevin</a>
    <br><br>
    <span>Site écrit en </span>
    
</body>
</html>

```

Et la page WEB affichée dans un navigateur :

Spécialité NSI

file:///E/\_1NSI/07 - HTML-CSS/index.html

Les plus visités LYCEE PAUL-LANGEVIN...

Autres marque-pages

# Présentation de la spécialité NSI

La spécialité "Numérique et Sciences Informatiques" a ouvert en septembre 2019 au lycée Paul Langevin de Beauvais.

## Le programme de NSI

- Histoire de l'informatique (partie transversale qui se décline dans chaque partie)
- Représentation des données : type et valeurs de base
- Représentation des données : types construits
- Traitement des données en tables
- Interactions entre l'homme et la machine sur le Web
- Architectures matérielles et systèmes d'exploitation
- Langages et programmation
- Algorithmique

Le programme officiel de la spécialité NSI

## Liens utiles

En savoir plus sur le langage HTML  
 En savoir plus sur le langage CSS  
 En savoir plus sur le langage Python  
 Pour obtenir le nom ou le code RVB des couleurs

Lien vers le site du lycée Paul Langevin

HTML CSS

Site écrit en

Pour bien observer le rôle du fichier CSS, voici la même page WEB affichée dans un navigateur sans l'usage de ce fichier CSS :

The screenshot shows a Microsoft Edge browser window. The title bar says "Spécialité NSI". The address bar shows "file:///E/\_1NSI/07 - HTML-CSS/index.html". Below the address bar, there's a toolbar with icons for back, forward, search, and refresh. A status bar at the bottom right says "Autres marque-pages". The main content area contains the following text:

## Présentation de la spécialité NSI

La spécialité "Numérique et Sciences Informatiques" a ouvert en septembre 2019 au lycée Paul Langevin de Beauvais.

### Le programme de NSI

- Histoire de l'informatique (partie transversale qui se décline dans chaque partie)
- Représentation des données : type et valeurs de base
- Représentation des données : types construits
- Traitement des données en tables
- Interactions entre l'homme et la machine sur le Web
- Architectures matérielles et systèmes d'exploitation
- Langages et programmation
- Algorithmique

Le [programme officiel de la spécialité NSI](#)

### Liens utiles

En savoir plus sur le [langage HTML](#)  
En savoir plus sur le [langage CSS](#)  
En savoir plus sur le [langage Python](#)  
Pour obtenir le nom ou le code.RVB des [couleurs](#)

Lien vers le site du [lycée Paul Langevin](#)

Site écrit en 

### Activité 3 :

Utiliser **notepad++** pour éditer le code **HTML** fourni dans l'exemple de la page précédente ([index.html](#)), ainsi que le code **CSS** fourni ([style.css](#)) et valider votre travail en observant le résultat obtenu dans un navigateur.  
Vous placerez ces deux fichiers dans un dossier qui contiendra aussi le fichier **HTML-CSS.jpg** fourni sur le serveur, et auquel votre fichier **HTML** fait appel.

## 6 Travail à réaliser

Vous allez utiliser l'éditeur **notepad++**, pour écrire plusieurs pages web qui seront accessibles à partir d'hyperliens placés sur celle réalisée dans le cours lors de l'activité 3 ([index.html](#)). Chacune de ces nouvelles pages web contiendra les titres des chapitres du cours correspondant et le lien vers vos documents PDF et/ou des images (captures, photos, ...) issues de vos cours.  
Vous devrez, bien évidemment, adapter le fichier CSS ([style.css](#)) à vos goûts personnels (polices, couleurs, ...).  
Les ressources proposées sur les liens de [w3schools.com](#) vous permettront de n'avoir aucune limite à votre créativité.  
Tous les fichiers seront rangés dans des dossiers organisés avec une arborescence digne de ce nom (voir chapitre 2.4).

~~~~~

En plus du **HTML** et du **CSS**, envie de faire un jeu en **Javascript** ?



Un casse-brique 2D, expliqué étape par étape, dans un tutoriel en ligne ... et en français !

... à ajouter éventuellement sur une page web de votre site.

[https://developer.mozilla.org/fr/docs/Games/Workflows/2D\\_Breakout\\_game\\_pure\\_JavaScript](https://developer.mozilla.org/fr/docs/Games/Workflows/2D_Breakout_game_pure_JavaScript)