

# WEB

## HTML - CSS - JAVASCRIPT

Yannick CHISTEL

Lycée Dumont d'Urville - CAEN

Janvier 2020

## Introduction

Le mot WEB est une abréviation de l'expression World Wide WEB qui signifie « toile d'araignée de taille mondiale ».

Le principe du WEB consiste à relier des documents entre eux via des liens hypertextes. Un lien hypertexte est un mot, une phrase ou une image qui permet d'accéder au document vers lequel il renvoie.

Pour afficher et lire un document du WEB, on utilise un logiciel appelé **navigateur**. Il en existe de nombreux comme FIREFOX, CHROME, EDGE, OPERA,...

## Langages du WEB

Différents langages sont utilisés pour structurer et afficher les documents du WEB :

- Le langage HTML qui est composé du contenu et de la structure du document à afficher par le navigateur ;
- Le langage CSS qui assure la mise en page et l'apparence graphique du document ;
- Le langage JAVASCRIPT qui permet d'ajouter de l'interactivité au document.

## Définition

HTML est l'acronyme de Hyper Text Markup Language signifiant Langage de Balisage Hyper Texte.

Le Langage HTML est un fichier texte qui est composé du contenu à afficher et de la structure du document selon laquelle le contenu est organisé : titre, paragraphe, emphase, liste, lien, image, etc.

Il existe une version stricte du HTML notée XHTML imposant une syntaxe plus rigoureuse que nous utiliserons.

## Balisage

La structure du document est décrite par des balises, qui sont des mots délimités par les caractères < et > appelés *chevrons*. On a 2 types de balises :

- Les balises **ouvrantes** et **fermantes** : < balise > contenu < /balise >

**Exemple** : < p >le contenu d'un paragraphe < /p >

- Les balises **vides** : < balise/ >

**Exemple** : Pour introduire un retour à la ligne < br/ >

Les balises peuvent contenir des **attributs** qui sont des propriétés supplémentaires à appliquer au contenu à afficher.

## Structure d'un document HTML

Les documents HTML ont la même structure (squelette) générale.

- 1 Déclaration de type nécessaire au navigateur pour savoir que c'est un document HTML.
- 2 Structure du document HTML à afficher elle même décomposée en 2 parties :
  - L'EN-TÊTE (HEAD) qui contient des informations sur le document HTML comme le titre, l'encodage de caractères, des liens vers des ressources à utiliser pour l'affichage, des scripts pour l'interactivité, etc. Ces informations ne sont pas directement visibles dans le contenu affiché.
  - LE CORPS (BODY) qui contient le contenu et la structure du document à afficher.

## Exemple

```
<!DOCTYPE html>                                     <!-- Déclaration de type -->
<html lang="fr">                                       <!-- Structure du document : racine -->
<head>                                                 <!-- Première partie : EN-TÊTE -->
  <!--
  Informations relatives au document html :
  le titre de la page web, encodage de caractères, etc
  -->
</head>
<body>                                                 <!-- Seconde partie : CORPS -->
  <!--
  Contenu et structure du document html :
  le titre du document, des paragraphes de texte, des liens hypertextes, des images, etc
  -->
</body>
</html>
```

## En-tête du document

L'en-tête contient des informations sur le document HTML comme :

- Le titre du document inséré entre les balises `< title >` et `< /title >`. Ce titre est affiché dans l'onglet du navigateur ou la barre d'état de la fenêtre de navigation.
- L'encodage de caractères (ou autres informations sur le document) à utiliser inséré dans une balise vide `< meta / >` et avec l'attribut **charset** ce qui donne : `< meta charset="utf-8" / >`
- Des liens vers des ressources externes peuvent être ajoutés avec la balise `< link >` et des attributs (notamment les feuilles de style CSS).
- Des scripts JAVASCRIPT avec les balises `< script >` et `< /script >`

## Exemple

En-tête d'un document HTML :

```
<head>
  <meta charset="utf-8">
  <title>Introduction au WEB</title>
</head>
```

## Corps du document

Le corps (body) contient le contenu et la structure du document. On peut distinguer différents types de balises :

- ➊ Les balises de structure comme le titre, le paragraphe, une liste, un tableau.
- ➋ Les balises de texte comme le gras, l'italique, l'exposant, l'indice, le lien hypertexte, l'image.
- ➌ Les balises de bloc pour regrouper des éléments ou organiser différents contenus dans une même page comme le contenu principal, un menu de navigation, une barre latérale de ressources, un entête de page ou un pied de page.

## Remarque

L'association `< balise > contenu < /balise >` constitue un **élément HTML**.

- ➊ Un élément peut être simple avec 2 balises et un contenu.
- ➋ Un élément peut imbriquer plusieurs éléments HTML.

La notion d'élément HTML est importante pour le CSS et le JAVASCRIPT.

## Balises de structure

- Il existe 6 niveaux de titre balisés par ordre décroissant par **h1**, **h2**, **h3**, **h4**, **h5** et **h6**
- Le paragraphe est balisé par **p**
- Une liste non ordonnée est balisée par **ul** et chaque item de la liste est balisé par **li**
- Une liste ordonnée (numérotée) est balisée par **ol** et chaque item de la liste est balisé par **li**
- Un tableau est balisé par **table**, chaque ligne est balisée par **tr** et chaque colonne est balisée par **td**

## Remarque

- 1 Les balises peuvent contenir des attributs notamment liés aux feuilles de style CSS.
- 2 Toutes ces balises sont ouvrantes et fermantes et entourent les contenus. Ce sont des éléments HTML.

## Exemple :

Voici le corps d'un document HTML contenant un titre, 2 paragraphes et une liste numérotée :

```
<body>
<h1>Un titre de niveau 1</h1>
<p>Ceci est un premier paragraphe</p>
<p>Un second paragraphe suivi d'avec une liste de 2 items:</p>
<ol>
<li>Le premier item;</li>
<li>Le second item numéroté automatiquement</li>
</ol>
</body>
```

Qui affichera :

## Un titre de niveau 1

Ceci est un premier paragraphe

Un second paragraphe suivi d'avec une liste de 2 items:

1. Le premier item;
2. Le second item numéroté automatiquement



## Balises de caractères

- La mise en gras est balisée par **b** et la mise en italique est balisée par *i*
- La diminution de la police de caractère est balisée par **small**
- La mise en indice d'un caractère est balisée par **sub** et la mise en exposant d'un caractère est balisée par **sup**
- La mise en forme d'un code de texte est balisée par **code**

## Balises de lien hypertexte et d'image

- Un mot ou une phrase est transformé en lien hypertexte lorsqu'on l'entoure par la balise **a**. Il faut lui associer un attribut **href** qui contiendra l'adresse du document à relier.
- Une image (photo) peut être intégrée à un document html en utilisant la balise vide **img**. Il faut lui associer un attribut **src** qui contiendra l'adresse de l'image à insérer.

## Remarque

Les balises html structurent et affichent les éléments demandés. L'apparence et la disposition des contenus devra être géré par du CSS.

## Exemple

Voici un code de document HTML complet :

```
<!DOCTYPE html>
<html lang="fr">
<head>
  <meta charset="utf-8">
  <title>Exemple 1</title>
</head>
<body>
<h1>Les balises de texte</h1>
<p>Ce paragraphe contient un 1<sup>er</sup> mot en caractères
  <b>gras</b> et un 2<sup>nd</sup> en <i>italique</i>.</p>
<p>Celui-ci contient un <a
  href="https://fr.wikipedia.org/wiki/Hyperlien">lien hypertexte</a>.</p>
<p>Ce dernier paragraphe contient une image : <br/></p>
</body>
</html>
```

## Les balises de texte

Ce paragraphe contient un 1<sup>er</sup> mot en caractères **gras** et un 2<sup>nd</sup> en *italique*.

Celui-ci contient un [lien hypertexte](https://fr.wikipedia.org/wiki/Hyperlien).

Ce dernier paragraphe contient une image :



## Balises de bloc

- ❶ On peut regrouper les contenus d'un document HTML dans des blocs. Ces blocs sont balisés par **div**.
- ❷ Certain blocs ont une fonction bien précise et sont désignés par des balises spéciales.  
On a par exemple :
  - La balise **nav** qui est utilisée pour les menus.
  - La balise **aside** qui est utilisée pour la navigation à l'intérieur d'un site WEB et souvent positionnée sur le côté de l'écran.
  - La balise **article** qui est utilisée pour le contenu entier d'un document avec un titre, plusieurs paragraphes, des liens et des images.
  - La balise **section** qui est utilisée pour organiser les contenus par section.
  - La balise **header** qui est utilisée pour créer un en-tête.
  - La balise **footer** qui est utilisée pour créer un pied de page.

## Remarques

- ❶ Ces balises de blocs peuvent s'imbriquer. Il est possible d'avoir plusieurs balises **articles** dans une balise **section**. On peut aussi avoir les balises **header** et **footer** dans une balise **article**.
- ❷ Ces balises structurent le contenu et ne gèrent pas leur disposition.

## Exemple

La structure donnée en exemple est imbriquée dans la balise `< body >` du document html.

On peut réaliser la même structure en utilisant uniquement des blocs **div**.

Pour les différencier il sera nécessaire de leur donner un identifiant ou **id**.

Par exemple, pour les deux articles, la structure sera :

```
<div id="article1">
  <div>
    <!-- Contenu de l'article 1 -->
  </div>
</div>

<div id="article2">
  <div>
    <!-- Contenu de l'article 1 -->
  </div>
</div>
```

**Remarque :** L'image ci-contre représente la structure. La disposition des blocs sera obtenue uniquement avec du CSS.

```
<header>
...
</header>
```

```
<nav>...</nav>
```

```
<article>
  <div>
    ...
  </div>
</article>
```

```
<aside>
...
...
</aside>
```

```
<article>
  <div>
    ...
  </div>
</article>
```

```
<footer>
...
</footer>
```

## Introduction

Le langage CSS de l'anglais Cascading Style Sheets signifiant feuilles de style en cascade est un langage permettant de définir les propriétés graphiques des éléments HTML constituant une page Web. Cela permet par exemple :

- De modifier la police de caractère, sa taille, sa fonte, son alignement, sa couleur etc.
- De modifier les couleurs des différents éléments HTML de la page comme les fonds.
- De gérer les espacements entre les blocs et leurs dispositions.
- D'ajouter des images de fond et des logos.

## La syntaxe CSS

L'application d'un style CSS se fait par le choix d'un **sélecteur** contenant, entre deux accolades, des couples **propriétés :valeurs** séparés par des points virgules.

Le nombre de propriétés n'est pas limité tant qu'elles existent.

```
sélecteur{  
  propriété : valeur;  
  propriété : valeur;  
  propriété : valeur;  
}
```

## Exemple

Voici le CORPS (body) d'un document en HTML :

```
<body>
  <h1>Un titre en noir</h1>
  <p>Mon premier paragraphe est court mais montre l'étendue
    des règles CSS.</p>
</body>
```

Voici des règles CSS associées aux sélecteurs **h1** et **p** :

```
h1{
  color:red;
  padding:10px;
}
p{
  text-align:right;
  border:solid 1px blue;
}
```

Page HTML **avant** l'application des règles CSS :

## Un titre en noir

Mon premier paragraphe est court mais montre l'étendue des règles CSS.

Page HTML **après** l'application des règles CSS :

## Un titre en noir

Mon premier paragraphe est court mais montre l'étendue des règles CSS.

## Les sélecteurs CSS

Il existe plusieurs manières de choisir un **sélecteur**.

- ❶ En utilisant directement la balise html du document (comme l'exemple précédent).  
Cette méthode est simple mais rencontre une difficulté. Les règles CSS seront appliquées à tous les sélecteurs de même type.  
Dans l'exemple précédent, s'il y avait plusieurs paragraphes, ils seraient tous alignés à droite et encadrés en bleu !
- ❷ En ajoutant un attribut **class** à l'élément HTML. Cette méthode offre 2 avantages :
  - Tous les éléments qui ont la même classe ont les mêmes règles CSS appliquées.
  - Les éléments de même type peuvent avoir des règles CSS différentes en leur attribuant des classes différentes.
- ❸ En donnant un **identifiant** aux éléments HTML du document. Chaque identifiant est unique donc les règles CSS ne s'appliqueront qu'à cet élément.
- ❹ On peut attribuer des règles CSS en mélangeant les trois premières méthodes.  
Un élément paragraphe `< /p >` peut avoir un identifiant et une class ce qui donne par exemple : `< p id="premier" class="rouge" >`

## Exemple

Voici un extrait de document en HTML :

```
<body>
  <h1 id="uno" class="bleu">Un titre en noir</h1>
  <p class="droite">Mon premier paragraphe est court mais
    montre l'étendue des règles CSS.</p>
</body>
```

Voici des règles CSS associés aux sélecteurs :

```
#uno{
  padding:10px;
}
.bleu{
  color:blue;
}
p{
  border:solid 1px blue;
}
.droite{
  text-align:right;
}
```

Page HTML **avant** l'application des règles CSS :

## Un titre en noir

Mon premier paragraphe est court mais montre l'étendue des règles CSS.

Page HTML **après** l'application des règles CSS :

## Un titre en noir

Mon premier paragraphe est court mais montre l'étendue des règles CSS.



## Déclaration de style

La première méthode se fait directement au niveau de la balise et les deux autres méthodes se font dans l'en-tête, entre les balises `< head >` et `< /head >` du fichier html.

**Méthode 1 :** dans la balise html, on insère l'attribut **style** et on lui donne en valeur les règles CSS.

```
<p style="text-align:right;">Le paragraphe avec des règles CSS en attribut</p>
```

**Méthode 2 :** entre les balises `< head >` et `< /head >` du fichier html, on ajoute les balises `< style >` et `< /style >` contenant les propriétés CSS ;

```
<style>
  h1{color:red;padding:10px;}
  p{text-align:right;border:solid 1px blue;}
</style>
```

**Méthode 3 :** entre les balises `< head >` et `< /head >` du fichier html, on ajoute la balise vide **link** et on crée un fichier **style.css** contenant toutes les règles CSS.

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

## Positionnement des éléments HTML

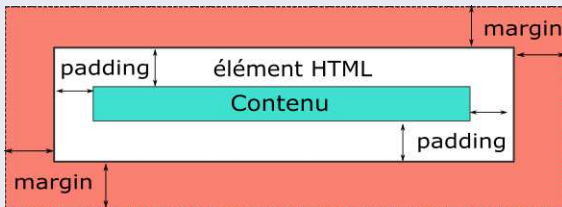
Les éléments HTML sont comme des boîtes avec du contenu. Les boîtes sont de 2 types :

- inline : les liens, le gras, les images, etc
- block : les titres, les paragraphes, les div et les éléments nav, article, aside, etc.

Selon le type, l'affichage et le positionnement sont différents.

- Pour un élément HTML inline, il se comporte comme du texte.
- Pour un élément HTML de type block, il occupe toute la largeur de la page par défaut. Il faut attribuer des règles CSS pour avoir un positionnement particulier.

Des règles CSS permettent de gérer les espaces autour et dans les éléments HTML :



## Présentation

Le langage JAVASCRIPT a été créé par Brendan Eich en 1985 pour permettre aux navigateurs (FIREFOX, CHROME, ...) d'exécuter des programmes pour les documents WEB.

- JAVASCRIPT est un langage de programmation : des variables (numériques, chaînes de caractères, tableaux,...), des conditions (if ... else), des boucles et des fonctions ;
- JAVASCRIPT permet de modifier le document HTML et les règles CSS du document ;
- JAVASCRIPT est à l'écoute d'événements qui sont liés aux actions du lecteur du document HTML : click de souris, déplacement de la souris, pression sur touche du clavier, chargement d'un document, ...
- JAVASCRIPT peut faire des requêtes HTTP (formulaires).

## Déclaration

Un script JAVASCRIPT se déclare avec la balise `< script >` dans le HEAD du document HTML. Il existe deux méthodes :

- **Méthode 1** : le script JAVASCRIPT est inséré entre les balises `< script >` et `< /script >`.
- **Méthode 2** : le script JAVASCRIPT est écrit dans un fichier externe d'extension JS et appelé par la balise `< script src="js/monscript.js" type="text/javascript" >< /script >`.

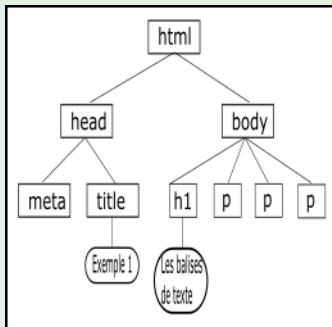
## DOM

Un document HTML est constitué de différents éléments html imbriqués les uns dans les autres. Cette structure peut se représenter par un arbre dont les noeuds sont les éléments html, (balises) reliés entre eux par des branches et les feuilles sont les contenus de texte. Cette structure s'appelle le DOM pour Document Object Model.

## Exemple

Voici un code de document HTML et son DOM :

```
<!DOCTYPE html>
<html lang="fr">
<head>
  <meta charset="utf-8">
  <title>Exemple 1</title>
</head>
<body>
<h1>Les balises de texte</h1>
<p>Ce paragraphe contient un 1<sup>er</sup> mot en caractères
  <b>gras</b> et un 2<sup>nd</sup> en <i>italique</i>.</p>
<p>Celui-ci contient un <a
  href="https://fr.wikipedia.org/wiki/Hyperlien">lien hypertexte</a>.</p>
<p>Ce dernier paragraphe contient une image : <br/></p>
</body>
</html>
```



## Parcourir le DOM

Le script JAVASCRIPT ne peut s'exécuter qu'à la fin du chargement de la page WEB. Ainsi, il est possible de parcourir le DOM du document pour y effectuer des modifications. La sélection des éléments se fait sur :

- Les sélecteurs : h1, p, a, div, etc.
- Les identifiants des éléments html : id="identifiant"
- Les class CSS appliquées aux sélecteurs : class="nom de classe"
- Les éléments parents, enfants, frères, etc grâce à des méthodes spécifiques.

## Exemple

Pour accéder à l'élément paragraphe **p**, on peut utiliser la méthode JAVASCRIPT **querySelectorAll('sélecteur');**

Cete méthode va parcourir tout le DOM et sélectionner tous les éléments **p**. On obtient donc en résultat un tableau contenant chaque élément.

```
let mesParagraphes = document.querySelectorAll('p');  
for (i=0; i < mesParagraphes.length; i++) {  
    mesParagraphes[i].style.color = "blue";  
}
```