

Première NSI  
Chapitre VI - Le Web

# I. Les pages Web

Le HTML permet de mettre du contenu dans la page.

Le style graphique de la page est défini par le langage CSS (Cascade Style Sheets).

L'aspect dynamique de la page Web est gérée par le langage de programmation javascript.

## I.1. Le HTML

### I.1.a. Généralités

#### Définition

Le HTML (HyperText Markup Language) est un langage de balisage qui permet d'écrire le contenu des pages Web.

#### Remarque

Les balises sont composées en général d'une balise entrante et d'une balise fermante.

```
1 <maBalise>
2 ...
3 </maBalise>
```

#### Remarque

Pour des raisons de lisibilité, on indente les balises situées à l'intérieur d'autres balises.

```
1 <balise1>
2   Du texte pour la balise 1.
3   <balise2>
4     Du texte pour la balise 2.
5     <balise3>
6       Le texte de la balise 3.
7     </balise3>
8     Encore du texte pour la balise 2.
9   </balise2>
10  <balise4>
11    Du texte pour la balise 4.
12    <balise5>
13      Le texte de la balise 5.
14    </balise5>
15    Encore du texte pour la balise 4.
16  </balise4>
17  Encore du texte pour la balise 1.
18 </balise1>
```

Chaque balise possède :

- 1 unique parent : la balise qui la contient.
- 0, 1 ou plusieurs enfants : les balises qu'elle contient.

Dans notre exemple :

- `<balise1>...</balise1>` est le parent de `<balise2>...</balise2>` .
- `<balise2>...</balise2>` et `<balise4>...</balise4>` sont les enfants de `<balise1>...</balise1>` .
- `<balise5>...</balise5>` est l'enfant de `<balise4>...</balise4>` .

## Structure d'une page HTML

Une page HTML possède :

- un en-tête qui ne s'affiche pas mais dans lequel on note des données tels que des liens avec d'autres fichiers ou le titre de la page Web
- un corps dans lequel on écrit le contenu de la page Web.

```
1 <html>
2   <head>
3     <meta http-equiv="content-type" content="text/html; charset=utf-8" />
4     <title> Mon titre </title>
5     <link href="CSS/monFichierCSS.css" type="text/css" rel="stylesheet" media="screen">
6     <SCRIPT type="text/javascript" SRC="javascript/monFichierJavascript.js"> </SCRIPT>
7   </head>
8   <body>
9     contenu de la page Web
10  </body>
11 </html>
```

## Remarque

- La ligne 3 indique le type de codage des caractères, ici l'UTF-8.
- La ligne 4 indique le titre de la page, que l'on retrouve dans les onglets des navigateurs.
- La ligne 5 permet de lier la page HTML à un fichier CSS.
- La ligne 6 permet de lier la page HTML à un script Javascript qui se trouve dans un autre fichier.

## Structure du corps d'une page HTML

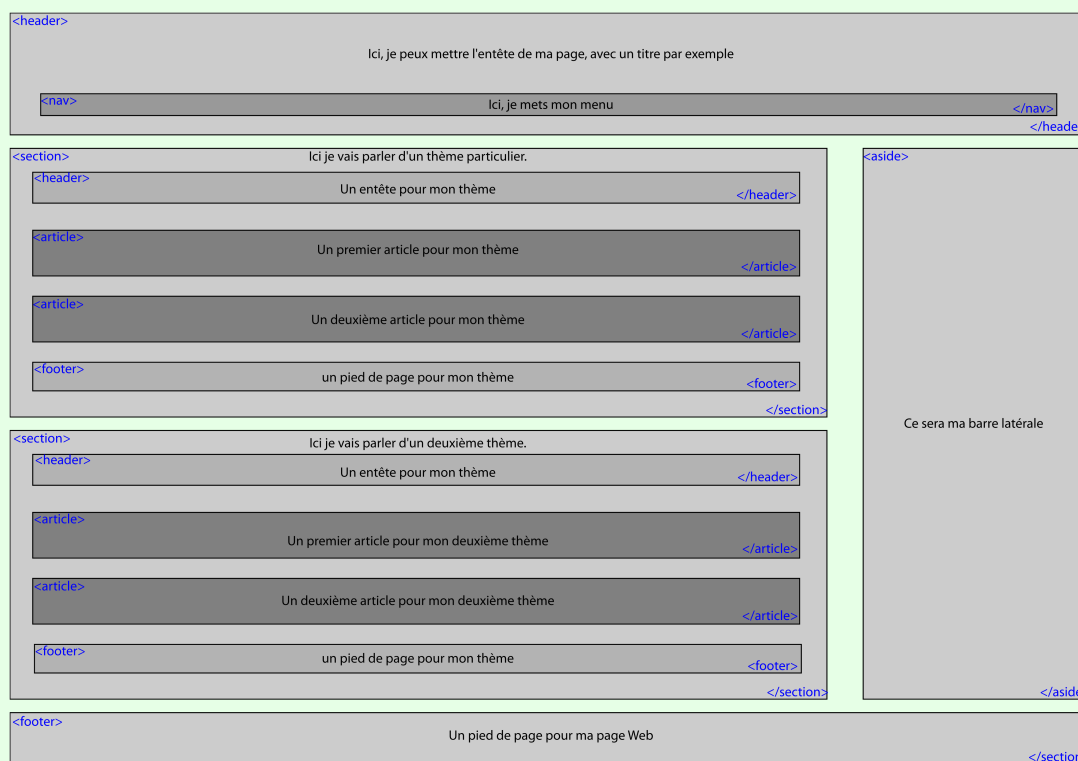
Afin de se repérer facilement dans le contenu du corps, il faut le structurer. Ce dernier doit donc ressembler à :

```
1 <header>
2   Ici , je peux mettre l'entête de ma page, avec un titre par exemple
3   <nav>
4     Ici , je mets mon menu
5   </nav>
6 </header>
7 <section>
8   Ici je vais parler d'un thème particulier.
9   <header>
10    Un en-tête pour mon thème
11  </header>
12  <article>
13    Un premier article pour mon thème
14  </article>
15  <article>
16    Un deuxième article pour mon thème
17  </article>
18  <footer>
19    un pied de page pour mon article
20  </footer>
21 </section>
```

## Structure du corps d'une page HTML - suite

```
22 <section>
23     Ici je vais parler d'un deuxième thème.
24     <header>
25         Un entête pour mon thème
26     </header>
27     <article>
28         Un premier article pour mon thème
29     </article>
30     <article>
31         Un deuxième article pour mon thème
32     </article>
33 </section>
34     un pied de page pour mon article
35 </footer>
36 </section>
37 <aside>
38     Ce sera ma barre latérale
39 </aside>
40 <footer>
41     Un pied de page pour ma page Web
42 </footer>
```

Ce qui donne une structure du type ci-dessous :



### I.1.b. Les balises

En plus de celles que nous avons vu précédemment, voici une liste non exhaustive des balises HTML.

- `<div>...</div>` : crée une zone de largeur la largeur de son parent.
- `<span>...</span>` : crée une zone de largeur la largeur de son contenant.
- `<br>` : retour à la ligne. Remarque : il n'y a pas de balise fermante.
- `` : insère une image.

#### Attention

Lorsqu'on indique une source (ici, celle de l'image), on l'indique toujours en chemin relatif, c'est-à-dire par rapport à l'emplacement du fichier dans lequel on indique la source (ici, par rapport à l'emplacement de `ma_premiere_page_Web.html`).

#### taille de l'image

On peut choisir la taille de l'image en ajoutant les attributs `height` (hauteur) ou `width` (largeur) dans la balise.

```
1 
2 
3 
```

La ligne 3 déformera l'image si vous ne gardez pas le rapport hauteur/largeur de l'image. Les deux autres lignes ne déformeront pas l'image.

- `<a>...</a>` : insère une ancre, ou lien hypertexte.

```
1 <a href="adresse_du_lien"> affichage </a>
```

`adresse_du_lien` est :

- soit une adresse distante (sur Internet) du type : `https://www.w3schools.com/tags/tag_a.asp`
- soit une adresse locale :
  - \* relative du type : `deuxieme_page.html`
  - \* absolue (non recommandée) du type : `C:/leDossierDeMonSite/deuxieme_page.html`

#### Remarque

Le fichier au bout du lien peut être une page Internet, une image, un fichier média (son ou vidéo), etc.

- `<h1>...</h1>` à `<h6>...</h6>` : titres et sous-titres. Plus le numéro de la balise est petit, plus la police de caractère sera grande.
- `<b>...</b>` : en gras.
- `<i>...</i>` : en italique.
- `<u>...</u>` : souligné.
- `<button>...</button>` : un bouton. Le contenu de la balise correspond à ce qui s'affiche sur le bouton.

La liste des balises est référencée sur le site [W3schools](http://W3schools)

Certaines balises ont besoin d'attributs, d'autres non. Mais on peut affecter certains attributs, dits **universels**, à toutes les balises, notamment les attributs `id` et `classe`.

### Remarque

Lorsqu'on affecte une valeur à un attribut pour un élément de la page Web, c'est toujours dans la balise ouvrante de ce dernier.

## I.1.c. Les classes

### Définition

Une classe est un groupe. Lorsqu'on affecte à une balise une classe, via l'attribut `class`, elle fera automatiquement partie de cette classe.

### Propriété

Lorsqu'on donne une propriété à une classe alors tous les éléments de cette classe sont affectés.

### Exemple

Soit le code HTML suivant :

```
1 <span class = "machin truc">Bonjour</span>
2 <span class = "chose machin">le</span>
3 <span class = "chose truc">monde</span>
```

Si la classe `machin` correspond à la propriété « couleur rouge », la classe `truc` correspond à la propriété « bordure noire double » et la classe `chose` correspond à la propriété « grandeur du texte plus grande que la normale », alors on obtiendra :

Bonjour

le

monde

## I.1.d. Les identifiants

### Définition

Un identifiant, comme son nom l'indique identifie une balise. Il doit être unique. Plusieurs balises ne peuvent avoir le même identifiant. Pour donner un identifiant à une balise, on utilise l'attribut `id`.

### Exemple

Soit le code HTML suivant :

```
1 <span id = "bidule">Bonjour</span>
2 <span id = "hop">à tous.</span>
```

Si on donne à l'identifiant `bidule` la propriété « couleur verte » et à l'identifiant `hop` la propriété « bordure noire », on obtient :

Bonjour

à tous.

## I.2. Le CSS

### Définition

Le CSS (Cascade Style Sheet) permet de mettre en forme la page Web.

### Remarque

Si le style de la page web est dans un autre fichier, il est nécessaire de lier le fichier CSS au fichier HTML.

Pour appliquer des propriétés à des balises, il y a trois façons de faire :

- en utilisant le nom de la balise, auquel cas la propriété affectera toutes les balises.

### Exemple

```
1 article {
2     mes propriétés
3 }
```

- en utilisant une classe, auquel cas la propriété affectera toutes les balises appartenant à la classe.

### Exemple

```
1 .laClasse {
2     mes propriétés
3 }
```

- en utilisant un identifiant, auquel cas la propriété affectera seulement la balise d'identifiant donné.

### Exemple

```
1 #monIdentifiant {
2     mes propriétés
3 }
```

## I.3. Le Javascript

### I.3.a. Généralités

### Définition

Javascript est un langage de programmation qui permet de dynamiser les pages Web.

### Remarque

Javascript peut utiliser la programmation événementielle, c'est-à-dire une programmation qui exécute des instructions quand on fait une action sur des éléments de la page Web associée.

### Remarque

Si les instructions Javascript sont dans un autre fichier, il est nécessaire de lier le fichier Javascript au fichier HTML.

### I.3.b. Variables et fonctions

Comme en Python, on peut utiliser des variables et des fonctions en Javascript.

Pour les variables locales, on écrit :

```
1 var maVariable = valeur
```

#### Remarque

Si on oublie le mot-clé `var`, la variable sera globale.

Pour les variables globales, on écrit :

```
1 maVariable = valeur
```

Pour les fonctions, Il y a deux façons de faire :

```
1 var maFonction = function(argument1, argument2, ...){
2     mesInstructions
3 }
```

ou

```
1 function maFonction(argument1, argument2, ...){
2     mesInstructions
3 }
```

### I.3.c. Les événements

Il existe beaucoup d'événements en Javascript que l'on peut retrouver sur le site [W3schools](https://www.w3schools.com).

Nous nous intéresserons uniquement à l'événement `onclick`, qui exécute une séquence d'instructions lorsqu'on clique sur un élément de la page Web.

Pour programmer en programmation événementielle, il y a trois façons de faire :

- 1<sup>ère</sup> façon : en HTML

```
1 <element onclick="monScript">
```

`monScript` est bien un script Javascript dans une chaîne de caractères, c'est-à-dire une série d'instructions séparées par des `;`.

Cette façon n'est recommandée uniquement s'il y a peu d'instructions (une ou deux) à exécuter.

L'événement utilisé `onclick` est un attribut de l'élément `element`.

- 2<sup>ème</sup> façon : en Javascript

```
1 monObjet = document.getElementById("monIdentifiant");
2 monObjet.onclick = function(){
3     monScript
4 }
```

Il faut donc connaître l'identifiant de l'élément sur lequel on doit cliquer pour exécuter les instructions.



- 3<sup>ème</sup> façon : en Javascript avec la méthode `addEventListener()`

```
1 monObjet = document.getElementById("monIdentifiant");
2 monObjet.addEventListener("click", monScript)
```

ou

```
1 var maFonction = function{
2     monScript
3 }
4 monObjet = document.getElementById("monIdentifiant");
5 monObjet.addEventListener("click", maFonction)
```

Il faut donc connaître l'identifiant de l'élément sur lequel on doit cliquer pour exécuter les instructions.

### I.3.d. Diversité et unité des langages de programmation

D'un langage à l'autre, on programme différemment, cependant certaines particularités peuvent rester identiques : création de fonctions, affectations, etc.

En Python :

```
1 def delta(a, b, c):
2     return b * b - 4 * a * c
3 print("delta = " + str(delta(1, 2, 3)))
```

En Javascript :

```
1 function delta(a, b, c) {
2     return b * b - 4 * a * c
3 }
4 document.write("delta = " + delta(1, 2, 3))
```

ou

```
1 var delta = function(a, b, c) {
2     return b * b - 4 * a * c
3 }
4 document.write("delta = " + delta(1, 2, 3))
```

En PHP :

```
1 function delta(a, b, c) {
2     return b * b - 4 * a * c
3 }
4 echo "delta = ".delta(1, 2, 3)
```

En C++ :

```
1 float delta(float a, float b, float c)
2 {
3     return b * b - 4 * a * c;
4 }
5 cout << "delta = " << delta(1, 2, 3);
```

Les quatre exemples ci-dessus font exactement la même chose, à savoir affiche la chaîne de caractères :

```
1 delta = -8
```

## **II. Exercices**

### **II.1. Exercice 1**

Analyser le site *adresseIP*.

### **II.2. Exercice 2**

Créer une page sur thème informatique. Vous devrez créer au moins 3 fichiers le HTML, le CSS et le Javascript. Vous devrez utiliser la programmation événementielle pour dynamiser votre page.