

## 1. Page Web

La **page web** est une ressource du World Wide Web. Les pages web sont conçues pour être consultées avec un navigateur web. Elles sont identifiées par une adresse web.

Techniquement, une page web est généralement constituée d'un document en **Hypertext Markup Language** (HTML) pour le contenu et la structure, de **feuilles de style en cascade** (CSS) pour la mise en page, et de **JavaScript** (JS) pour une interactivité (animation, résultat instantané d'un calcul...).



## 2. Le langage HTML

Le HTML (« Hypertext Mark-Up Language ») est un **langage** dit de « **description** » (de « structuration », de « marquage » ou de « **balisage** ») dont le rôle est de formaliser l'écriture d'un document avec des balises de formatage.

Les **balises** (ou tag en anglais) permettent d'indiquer la façon dont doit être présenté le document et les liens qu'il établit avec d'autres documents.

## 3. Constitution d'une page HTML

Une page Web minimaliste contient le texte suivant :

```
<!DOCTYPE html>
<html>
  <head>
    <title>Ma page HTML</title>
  </head>
  <body>
    <p>Ceci est ma premiere page HTML !</p>
  </body>
</html>
```

La première ligne est le prologue. Il précise que le reste du texte est du HTML (ici HTML 5).

Le texte contient ensuite des balises ouvrantes et fermantes (`<html>` et `</html>`) ou encore (`<title>` et `</title>`).

Chaque balise a une signification. Notre exemple contient les balises suivantes :

- `<html>` : Représente la racine de l'arbre HTML.
- `<head>` : Cette balise contient les informations (méta-données) de la page HTML. Le contenu ne sera pas affiché dans la page HTML.
- `<title>` : Cette balise permet de donner un titre à la page. Ce titre sera affiché par le navigateur dans l'onglet.
- `<body>` : Cette balise représente le contenu de la page. Ce contenu sera affiché par le navigateur.

### 3.1. Écriture des balises

Toute balise ouverte doit être fermée. Elles sont de deux types :

- Balise paire : une balise ouvrante `<nom_de_la_balise>` et une balise fermante `</nom_de_la_balise>`.
  - Par exemple : `<h1>Très gros titre</h1>`
- Balise orpheline : `<nom_de_la_balise />`.
  - Par exemple : `<br />` pour un saut de ligne.

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

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

Chaque balise a une signification qu'il faut bien respecter (on parle de la sémantique des balises).

- `<h1>` pour un très gros titre, `<p>` pour un paragraphe...

L'imbrication des balises doit être respectée :

faux	correct
<code>&lt;balise1&gt;</code>	<code>&lt;balise1&gt;</code>
<code>&lt;balise2&gt;</code>	<code>&lt;balise2&gt;</code>
<code>&lt;/balise1&gt;</code>	<code>&lt;/balise2&gt;</code>
<code>&lt;/balise2&gt;</code>	<code>&lt;/balise1&gt;</code>

## 4. Observation du HTML

Visualiser la page Web <http://www.csszengarden.com/>.



Il est possible de visualiser le code source de la page en cliquant droit dessus.

```

1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4 <meta charset="utf-8">
5 <title>CSS Zen Garden: The Beauty of CSS Design</title>
6 <link rel="stylesheet" media="screen" href="/2012/2012.css?theme=2012">
7 <link rel="alternate" type="application/rss+xml" title="RSS" href="http://www.csszengarden.com/feed.rss">
8 </head>
9 <body>
10 <div id="wrapper" style="width: 100%; height: 100%; text-align: center; background-color: #f0f0f0; padding: 10px 0 0 0;">
11 <div id="content" style="width: 60%; margin: 0 auto; padding: 10px 0 0 0;">
12 <div id="main" style="width: 100%; height: 100%; text-align: center; background-color: #f0f0f0; padding: 10px 0 0 0;">
13 <div id="text" style="width: 60%; margin: 0 auto; padding: 10px 0 0 0;">
14 <div id="text" style="width: 60%; margin: 0 auto; padding: 10px 0 0 0;">
15 </div>
16 </div>
17 </div>
18 </div>
19 </div>
20 </div>
21 </div>
22 </div>
23 </div>
24 </div>
25 </div>
26 </div>
27 </div>
28 </div>
29 </div>
30 </div>
31 </div>
32 </div>
33 </div>
34 </div>
35 </div>
36 </div>
37 </div>
38 </div>
39 </div>
40 </div>
41 </div>
42 </div>
43 </div>
44 </div>
45 </div>
46 </div>
47 </div>
48 </div>
49 </div>
50 </div>
51 </div>
52 </div>
53 </div>
54 </div>
55 </div>
56 </div>
57 </div>
58 </div>
59 </div>
60 </div>
61 </div>
62 </div>
63 </div>
64 </div>
65 </div>
66 </div>
67 </div>
68 </div>
69 </div>
70 </div>
71 </div>
72 </div>
73 </div>
74 </div>
75 </div>
76 </div>
77 </div>
78 </div>
79 </div>
80 </div>
81 </div>
82 </div>
83 </div>
84 </div>
85 </div>
86 </div>
87 </div>
88 </div>
89 </div>
90 </div>
91 </div>
92 </div>
93 </div>
94 </div>
95 </div>
96 </div>
97 </div>
98 </div>
99 </div>
100 </div>

```



Sur la page Web CSS Zen Garden, cliquer droit sur *Inspecter* (Chrome) ou *Examiner l'élément* (FireFox), puis observer le contenu de la page en survolant avec la souris le code HTML.



## 5. Le langage CSS

La mise en page d'une page Web est réalisé grâce au langage CSS (Cascade Style Sheets), **feuilles de style en cascade**.

Observer le pouvoir du langage CSS en sélectionnant différentes feuilles de style dans le site Web CSS Zen Garden (à droite sur la page d'origine).

- Le contenu reste identique (même page HTML) mais c'est la feuille de style qui modifie la mise en page du site Web.

## 6. Observation du CSS

Sur la page Web CSS Zen Garden, Inspecter ou Examiner à nouveau la page (clic droit), puis modifier les paramètres de la feuille de style et observer l'influence sur la mise en forme.



## 7. Le langage Javascript

Le langage Javascript est un langage de programmation qui permet de créer du contenu interactif, de contrôler le contenu multimédia, d'animer des images...

Par exemple, sur la page Web CSS Zen Garden, le Javascript permet de modifier la feuille de style appliquée au site Web.

## 8. Le DOM

Tous les navigateurs ont la même représentation interne en mémoire d'une page HTML : c'est ce qu'on appelle le DOM (Document Object Model).

Cette structure est arborescente. Chaque balise HTML est représentée par un élément (DOM Element).

C'est cette structure qui pourra être manipulée en JavaScript et c'est sur cette structure que les évènements (Event) seront définis.