



Activité III : HTML, CSS, JavaScript

Coté client

Présentation des bases du langage HTML

L'**HyperText Markup Language (HTML)**, est le format de données conçu pour représenter les **pages web**. C'est un langage de **balisage** (markup en anglais) permettant d'écrire de l'**hypertexte**, d'où son nom.

HTML, surtout depuis sa dernière version **HTML5**, 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 est souvent utilisé conjointement avec des langages de programmation (JavaScript, Python, PHP, ...) et des formats de présentation (**CSS**).

les Objectifs :

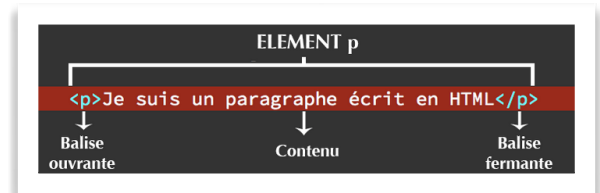
- Rappeler les principes de bases du langage HTML et CSS
- Utiliser les bases du langage Javascript

Le Langage HTML

Les balises (tag)

Le **contenu** d'une page HTML est défini grâce à des **balises** :

- elles **encadrent le contenu** de la page = le texte qui sera affiché
- elles **indiquent la nature** ou ce que signifie le contenu « titre de la page », « image », « paragraphe », « liste », ...



Syntaxe d'une balise : `<nom_de_l'élément>`

Donner des balises que vous avez vu en seconde :

Parmi les balises citées précédemment :

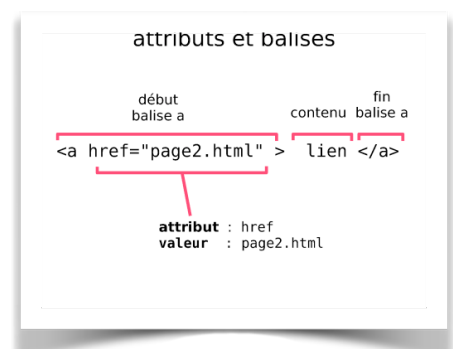
Donner les balises en paire : (Elle permettent d'encadrer du **texte** entre une **balise ouvrante** et une **balise fermante**, de même nature)

Donner les balises orphelines : Ces balises n'encadrent **pas de texte**, elles n'ont donc pas de structure ouverture/fermeture. Elle servent à insérer un élément qu'il n'est pas nécessaire de délimiter, par exemple une image

Les attributs (attribute)

Les **attributs** sont en quelque sorte les *options* des balises. Ils contiennent des informations sur :

- le **contenu** qui n'est pas directement dans le code HTML (un fichier ou un lien) : *image, vidéo, lien, ...*
- la **forme** (ou le **style**) pour son affichage : *dimension, encadrement, alignement, ...*



Légèder cette partie du code html :

```
<img src = "NSI" alt = "_blank" width = "50%" height = "50%">
```

Structure d'une page HTML

Le minimum du minimum pour commencer proprement

```
<!DOCTYPE html>
<html lang="fr">
  <head>
    <meta charset="UTF-8">
    <title>Exemple de page</title>
  </head>
  <body>
    <h1>Bienvenue sur ma page !</h1>
    <p>Ceci est un exemple de page HTML.</p>
  </body>
</html>
```

Donner la signification ou ce qu'elle contient :

lang="fr" :

head :

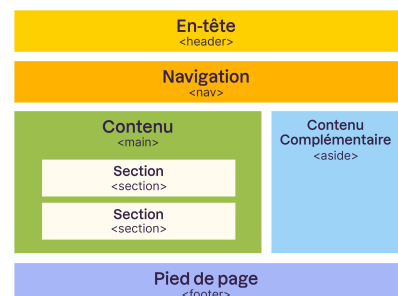
meta charset="UTF-8" :

body :

Organisation des contenus

Depuis [HTML5](#), le contenu d'une page web est doté de **blocs** dont les intitulés sont significatifs : <header> , <nav> , <article> , <section> , <figure> , <audio> , <video> , <embed> , ... des balises aux noms explicites, indiquant ce que signifie leur contenu :

- <header> = entête, permet d'annoncer ce que contient la page,
- <nav> = zone de navigation (menu, ...),
- une liste de liens menant aux contenus du site,
- <article> = article (message de forum, un article de presse, un billet de blog, ...),
- <aside> = contenu contextuel à un autre contenu ...



Les noms de ces blocs définissent le **type de contenu**, mais ne spécifie pas leur forme (position, dimension, couleurs, police de caractère, comportement, ...). Pour cela, on utilisera des feuilles de style CSS.

Les feuilles de style (CSS)

Les feuilles de style en cascade, généralement appelées CSS (Cascading Style Sheets), forment un langage informatique qui décrit la présentation des documents HTML et XML. Les standards définissant CSS sont publiés par le [World Wide Web Consortium](http://www.w3.org/) (W3C).

les avantages des feuilles de style :

- La structure du document et la présentation peuvent être gérées dans des fichiers séparés ;
- Les pages HTML d'une site web peuvent partager les mêmes feuilles de styles.
- Un même document peut donner le choix entre plusieurs feuilles de style, (exemple une pour l'impression et une pour la lecture à l'écran)
- Le code HTML est considérablement réduit en taille et en complexité, puisqu'il ne contient plus de balises ni d'attributs de présentation.

HTML
(pas de CSS)



HTML + CSS



On peut écrire du code en langage CSS à trois endroits différents :

1. dans un fichier **.css** (méthode la plus recommandée) ;
2. dans l'en-tête `<head>` du fichier HTML ;
3. directement dans les balises du fichier HTML via un attribut `style` (méthode la moins recommandée).

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

Application des styles à un élément

Pour pouvoir appliquer un style à un contenu, il va déjà falloir le cibler, c'est-à-dire trouver un moyen d'indiquer qu'on souhaite appliquer tel style à un contenu en particulier.

Pour cela, nous allons utiliser des **sélecteurs**.

De manière très schématique et très simplifiée, nous allons utiliser nos sélecteurs en CSS pour cibler des contenus HTML et leur appliquer des styles.

```
/* Changer la couleur de fond de la page */
body {
  background-color: lightblue;
  font-family: Arial, sans-serif; /* Police de caractère */
}

/* Styliser le titre (h1) */
h1 {
  color: darkblue; /* Couleur du titre */
  text-align: center; /* Centrer le titre */
  font-size: 3em; /* Taille de la police */
}

/* Styliser le paragraphe */
p {
  color: darkgreen; /* Couleur du texte */
  font-size: 1.2em; /* Taille de la police */
  margin: 20px; /* Ajouter de l'espace autour du paragraphe */
}
```



A partir de notre image, donner les sélecteurs, Les propriétés et les valeurs de chaque propriété.

De quelle couleur seront les titres, les paragraphes ?

De quelle couleur sera le fond de la page HTML ?

Où vont se trouver les titres de la page

Lorsqu'on applique un style à une balise, tout ce qui est encadré par ce type de balise prendra ce style. On peut vouloir faire des exceptions.

Il existe deux possibilités pour distinguer des balises ; leur attribuer :

- une **classe** : attribut class= »nom_de_la_classe »
- un **identifiant** : attribut id= »nom_d_identifiant » (en principe unique)

à une classe

Pour attribuer à une balise une **classe**, on lui ajoute l'attribut class :

```
<div class="adresse">
Lycée Blaise Pascal
<br>
36 av Carnot
<br>
63037 CLERMONT FERRAND
</div>
```

Remarque : plusieurs balises peuvent appartenir à la même classe

on applique un style à une classe dans le code CSS :

```
.adresse
{
border-width:10px;
border-style:ridge;
}
```

Ainsi, seules les balises appartenant à la classe « adresse » auront ce style.

à un identifiant

Le principe est le même que pour les classes, hormis le fait qu'un identifiant doit être unique sur une page web.

```
<p id="age">
<em>25 ans</em>
</p>
```

Le sélecteur CSS d'identifiant utilise la syntaxe suivante :

```
#age
{
box-shadow: 10px 10px 5px #888888;
}
```

Ainsi, Une seule balise contenant un id unique aura ce style.

Comment le sélecteur fait la différence entre une "class" et un "id" ?