



FORMATION

IT - Digital - Management



m2iinformation.fr



HTML5 & CSS3

Les fondamentaux du développement Web



Nehemie “Alfred” BALUKIDI
CTO/Software Engineering & Data|ML/AI



Introduction

Technologies du web :

- **HTML** (HyperText Markup Language) : Structurer le document
- **CSS** (Cascading StyleSheets) : Feuilles de styles
- **JavaScript**: Actions, Interactions avec l'utilisateur ...



Préparer l'environnement

1. Installer VSCode : <https://code.visualstudio.com/>
2. Installer google chrome : <https://www.google.com/intl/fr/chrome/>
3. Télécharger L'extension chrome Web Developer :
<https://bit.ly/3AMflKT>
4. Installer les extensions vs code suivants :
 - a. Emmet
 - b. Prettier
 - c. Live server



1.

Le HTML



Introduction HTML

→ **HTML1:** Naissance du HTML, crée par Tim Berners-Lee en 1991



→ **HTML2:** (1994) => Nouveautés mineures.

→ **HTML3:** (1996) => Tableaux, Scripts, positionnements...

→ **HTML4:** (1998) => Tableaux complexes, formulaires, dissociation des styles (CSS).



→ **HTML5:** (2014) Version actuelle => Simplification, nouveautés ++



Les bases du HTML

- HTML est un langage de balise.
- Une balise permet d'indiquer la nature du texte.
- La syntaxe pour utiliser une balise est la suivante:
 - ◆ `<balise_name> </balise_name>` ou `<balise_name />`
- La syntaxe pour déclarer une page html est divisée en 4 items:
 - ◆ `<!DOCTYPE html>`
 - ◆ `<html> </html>`
 - ◆ `<head> </head>`
 - ◆ `<body> </body>`
- L'ajout de commentaire en HTML se fait de la manière suivante :
 - ◆ `<!-- Mon commentaire -->`



Les balises

- Il faut toujours **structurer** son code à l'aide des **balises**.
- Quelques exemples :

Paragraphe	<code><p> </p></code>
Titres	<code><h1> </h1> , <h2> </h2> ... <h6> </h6></code>
Saut de ligne	<code>
</code>
Listes(ordonnées/non ordonnées)	<code> el1 el2 ... </code> <code> el1 el2 ... </code>
image	<code></code>
Lien(relatif ou absolu)	<code> Text </code>
Texte important	<code>... , ..., <mark>...</mark></code>



Les attributs

- Un attribut est une **information additionnelle/spécification** liée à une balise.
- Les attributs se déclarent dans les **balises ouvrantes** et sont **cumulables**.
- La syntaxe d'un attribut est la suivante :
 - ◆ `<balise_name atr1="value1" atr2="value2" ...> <balise_name/>`
- Quelques exemples :
 - ◆ `<h1 color="red" align="center" ...> </h1>`
 - ◆ ``
 - ◆ ``



Les spécificités du **tableau**

- Principales balises pour afficher un tableau :
 - ◆ Structure : `<table></table>`
 - ◆ En-têtes (header): `<thead></thead>`
 - ◆ Contenu (body): `<tbody></tbody>`
 - ◆ Ajout d'une ligne: `<tr></tr>`
 - ◆ Ajout de colonne : `<th></th>` et `<td></td>`

- Exemple d'attribut :
 - ◆ Bordures : `border="1"` (border=0 par défaut !)
 - ◆ Fusionner (horizontalement) : `colspan="2"`
 - ◆ Fusionner (verticalement) : `rowspan="2"`
 - ◆ Gérer l'alignement: `align="right"`
 - ◆ Fixer la taille de cellule: `width="50%"`



Les spécificités du formulaire

→ Principales balises pour afficher un formulaire :

- ◆ Structure : `<form></form>`
- ◆ Identification unique : `name="form_name"`
- ◆ Attribut de méthode: `method="post"`

→ Balises de formulaires :

- ◆ Textes : `<label>`
- ◆ Champs de saisie: `<input>` (text, number, date, checkbox, password ...)
- ◆ Liste déroulante: `<select>` - `<option></option>` - `</select>`
- ◆ Bouton: `<button></button>`
- ◆ Champs de saisie spécial: `<textarea>`

Les balises sémantiques de structure

header	Représente l'en-tête de la page(Logo, navigation,etc.)
nav	Menu de navigation
main	Contenu principal de la page
section	Sert à regrouper des contenus en fonction de leur thématique
article	La balise <article> sert à englober une portion généralement autonome de la page pouvant être repris dans d'autres sites(blog, magazine)
aside	Informations complémentaires au document
footer	Pied de page situé en bas du document(contact,mentions légales,etc.)



2.

Le CSS



Introduction au CSS

- Naissance du CSS pour accompagner le HTML en 1996
- Le CSS permet de gérer les couleurs, styles, polices, positionnement, ...
- Complémentaire avec le HTML
- Comment l'utiliser ?
 - ◆ Directement dans le code HTML avec la balise `<style> </style>`
 - ◆ Dans un fichier externe `.css`
 - ◆ Lien classique : `<link rel="stylesheet" type="text/css" href="style.css" />`



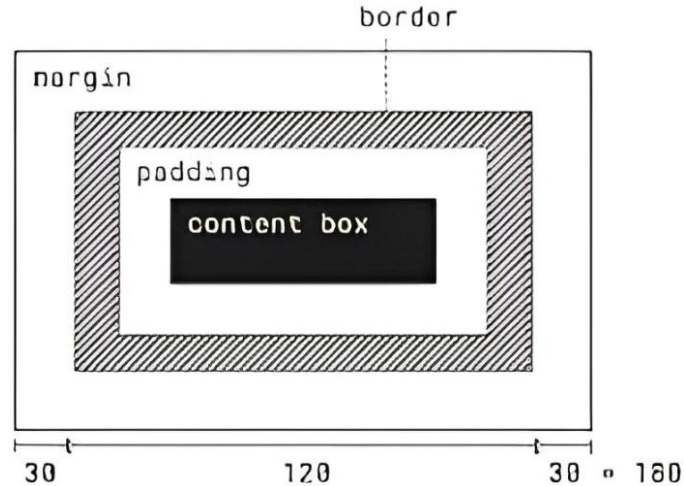
La Syntaxe

```
selecteur{  
    propriété : valeur;  
}
```

Ici, le **sélecteur** c'est l'élément ciblé

Le Modèle de boîte

```
div {  
  box-sizing: border-box;  
  width: 120px;  
  padding: 20px;  
  border: 4px solid;  
  margin: 30px;  
}
```





Les unités de mesures

Unités de mesures absolues

- ▶ px = pixels
- ▶ cm = centimètres
- ▶ mm = millimètres

Unités de mesures relatives

- ▶ % = pourcentage
- ▶ rem = unité relative à la taille des caractères de l'élément parent
- ▶ vw = unité relative à la largeur de la fenêtre du navigateur(en %)
- ▶ vh = unité relative à la hauteur de la fenêtre du navigateur(en %)

Sélecteurs CSS

Sélecteur d'élément ou balise HTML	<pre>p { color: ■ red; }</pre>
Sélecteur de class	<pre>.text-underline { color: ■ red; }</pre>
Sélecteur d'ID	<pre>#fname { color: ■ red; }</pre>
Sélecteurs universels	<pre>* { color: ■ red; }</pre>
Sélecteur d'attribut	<pre>a[href="https://example.org"] { color: ■ green; }</pre>

Combinateurs

Sélection des enfants directs	<pre>div > p{ color: red; }</pre>
Sélecteur des enfants à n'importe quel niveau	<pre>div p{ color: red; }</pre>
Voisin direct	<pre>div + p{ color: red; }</pre>
Autres...	...



Quelques exemples de **propriété**

Propriété	Syntaxe	Exemples
Couleurs	color background-color	color: red; / color: rgb(255, 99, 71) / color:#F10606 background-color: gray; / background-color: #808080
Police	font-family	font-family: Times; / font-family: Impact, Verdana, "Arial Black";
Taille texte	font-size	font-size: 15px; / font-size: small / font-size: 1,5em;
Styles de police	font-style font-weight ...	font-style: italic; font-weight: bold; / text-decoration: underline;
Alignement	text-align	text-align: center; / text-align: right / text-align: justify
Marges	margin padding	margin: 5px 5px 5px 5px; / margin-top: 10px; / margin-bottom: 30px; / padding: 10px; / padding-left: 15px



Positionnement

- Le positionnement permet de **gérer la mise en page** des éléments à l'aide de la propriété CSS **position**.
- Il faut différencier **4 types principaux** de positionnement :
 - ◆ **(static)**: Valeur par défaut - Positionnement statique des éléments
 - ◆ **relative** : Positionnement ajustée par rapport à la position initiale de l'élément
 - ◆ **absolute** : Positionnement ajustée par rapport au parent le plus proche (possibilité des éléments les uns sur les autres)
 - ◆ **fixed** : semblable à absolute, mais en s'appuyant sur la fenêtre visible
- Positionnement avec **top**, **left**, **bottom** et **right**. Les valeurs **négatives** sont possibles.
- La propriété **z-index** permet de prioriser l'élément à afficher (en cas de chevauchement)



Le CSS dynamique

- Le CSS permet de modifier les style **dynamiquement** grâce au pseudo-classes

```
selecteur:pseudo-class{  
    propriété : valeur;  
}
```

Action	Syntaxe
Survol	hover
Clic	active
cliqué	focus
Sélectionné	checked
Activé / désactivé	enabled / disabled

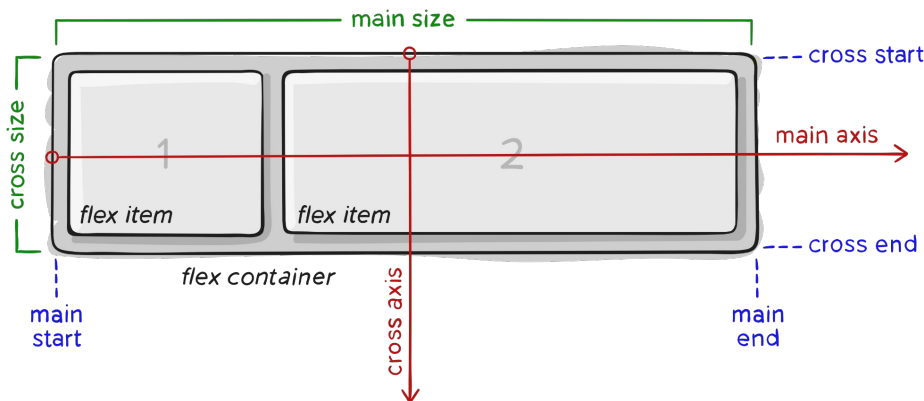


Display, div et span

- La propriété display définit le type d’affichage utilisé (**inline** vs **block**)
- Les balises **div** et **span** sont des **conteneurs génériques**
- Ils agissent sur la propriété **display**

Introduction au **Flexbox**

Un module CSS visant à fournir un moyen plus efficace d'agencer, d'aligner et de répartir l'espace entre les éléments d'un conteneur, même lorsque leur taille est inconnue et/ou dynamique





Responsive web design

La responsive web design est une approche qui permet de créer des pages web qui sont toujours bien rendu quelque soit la taille du périphérique d'affichage.

En css nous allons mettre en place des sites responsives grâce au **medias queries**.



Media queries

C'est une technique qui permet d'appliquer les propriétés css en fonction des **caractéristiques du périphérique** utilisé pour la consultation de la page HTML.



Break points

Les valeurs des **points de rupture** (breakpoint) des CSS3 Media Queries sont les tailles pour lesquelles l'affichage change sur le site.



Notion des break-points

Breakpoint

min-width: 480px

min-width: 768px

min-width: 992px

min-width: 1200px

Cible

Mobile

Tablette-portrait

Tablette paysage

Ecran large(desktop)



Le mobile en premier(Mobile First)

Il est recommandé de toujours concevoir pour le mobile en premier. Ceci signifie concevoir pour mobile avant de concevoir pour ordinateur de bureau ou tout autre appareil (cela rendra l'affichage de la page plus rapide sur les appareils plus petits)



Implications du Mobile First

- Rester sur l'essentiel
- Ne rien ajouter de superflue
- Pouvoir accéder le plus rapidement et facilement possible aux infos
- Cibler les performances

Mise en place des media queries

```
@media screen and (min-width: 480px) {  
    /* Votre code CSS */  
}  
  
@media screen and (min-width: 768px) {  
    /* Votre code CSS */  
}  
  
@media screen and (min-width: 992px) {  
    /* Votre code CSS */  
}  
  
@media screen and (min-width: 1200px) {  
    /* Votre code CSS */  
}
```



3.

Accessibilité



Introduction à l'ARIA

L'ARIA(Applications Internet Riches et Accessibles en Français) est un ensemble de rôles et d'attributs qui définissent comment rendre le contenu et les applications web accessibles.

Ces rôles et attributs doivent être utilisés sur des éléments HTML qui ne fournissent pas **sémantiquement** beaucoup de support pour les fonctionnalités d'**accessibilité**.



Utilisation de l'ARIA

HTML ▾

```
<div  
  id="percent-loaded"  
  role="progressbar"  
  aria-valuenow="75"  
  aria-valuemin="0"  
  aria-valuemax="100"  
>
```



Atelier Performance



Fin.