



FORMATION

IT - Digital - Management

24/02/2022



m2iinformation.fr



COURS HTML CSS



m2information.fr

PLAN

I. HTML


- A. INTRODUCTION
- B. LES BALISES DE STRUCTURE
- C. LES AUTRES BALISES
- D. LES ATTRIBUTS
- E. LISTES
- F. TABLEAUX
- G. FORMULAIRE

II. CSS

- A. INTRODUCTION
- B. SELECTEURS
- C. UNITES
- D. COULEURS
- E. LES PROPRIÉTÉS
- F. BOX MODEL
- G. FLEXBOX
- H. CSS GRID
- I. MOBILE FIRST ET RESPONSIVE DESIGN

INSTALLATION

1. Installez VSCode : <https://code.visualstudio.com/>
2. Installez les extensions VSCode suivantes :
 - Lorem ipsum
 - Prettier – Code formatter
 - HTML Preview
 - HTMLC CSS Support
 - Live server




↓ Windows
Windows 8, 10, 11

User Installer x64 x86 Arm64

System Installer x64 x86 Arm64

.zip x64 x86 Arm64

CLI x64 x86 Arm64



↓ .deb
Debian, Ubuntu

↓ .rpm
Red Hat, Fedora, SUSE


.deb x64 Arm32 Arm64

.rpm x64 Arm32 Arm64

.tar.gz x64 Arm32 Arm64

Snap Snap Store

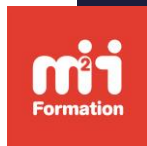
CLI x64 Arm32 Arm64



↓ Mac
macOS 10.11+

.zip Intel chip Apple silicon Universal

CLI Intel chip Apple silicon



PARTIE HTML



m2information.fr

INTRODUCTION



HISTORIQUE

1991

- Naissance du **HTML1**
- Créé par Tim Berners-Lee

1996

- **HTML3**
- Tableaux, scripts, positions, etc.

2014

- **HTML5 version actuelle**
- Simplification
- Nouveautés sémantiques

1994

- **HTML2**
- Nouveautés mineures

1998

- **HTML4**
- Tableaux complexes
- Formulaires
- Dissociation des style (CSS).

- HTML pour *HyperText Markup Language*.
- Langage de *balisage* (utilise des balises) pour **structurer** le contenu d'une *page Web*.
- Les navigateurs interprètent les balises et affiche le rendu correspond sur une page web.

- Une balise permet d'indiquer la nature d'un élément du document.
- Motif (pattern, model) d'une balise :
 - `<balise_name></balise_name>` : ouvrante et fermante ;
 - ou `<balise_name/>` : orpheline ou auto-fermante.

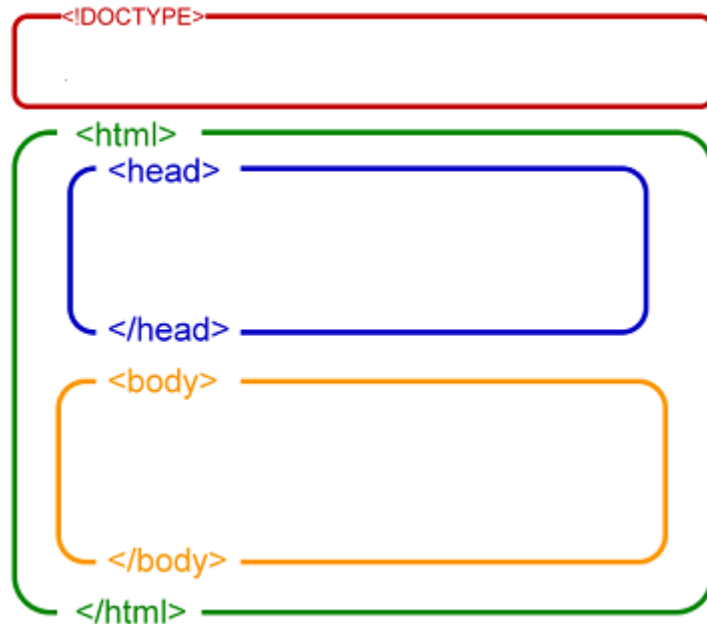
- Les commentaires
 - `<!-- ceci est un commentaire -->`
 - Visible dans le code source mais ce n'est pas interprété par les navigateurs

BALISES DE STRUCTURE



STRUCTURE DE BASE

[Source image Apcpedagogie.com](https://www.apcpedagogie.com)



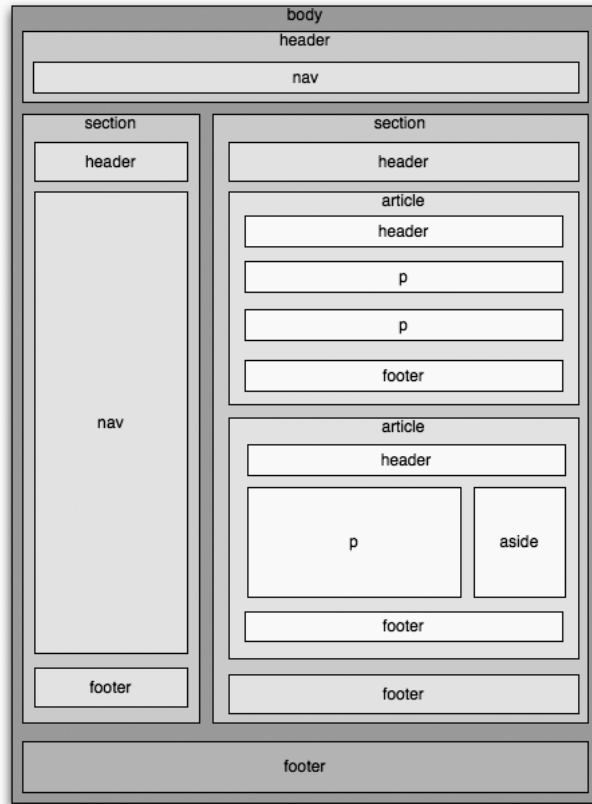
- **DOCTYPE** : indiquer aux navigateurs que le **document** est de type **HTML**
- **HTML** : balise parente contenant toutes les balises *head* et *body*
- **HEAD** : balise d'en tête contenant des informations (meta-données) destinées aux navigateurs
- **BODY** : balise contenant le corps du document *HTML* (tout ce qui sera visible par l'utilisateur sur la page web)

LES AUTRES BALISES



BALISES SEMANTIQUES

[Source image Studywithdemo.com](#)



- ***header***: dédiée à l'en-tête du body ou d'un sous-bloc body.
- ***footer***: dédiée au pied de page du body ou d'un sous-bloc du body.
- ***nav***: dédié à la navigation (liens, menus).
- ***section***: dédiée au regroupement des contenus de même thématique.
- ***article***: dédiée aux éléments autonomes.
- ***aside***: dédiée aux informations additionnelles.
- ***main***: indiquer le contenu principale de la page.

BALISES DE CONTENU

NOMS	BALISES
Paragraphe	<code><p></p></code>
Titres	<code><h1></h1></code> , <code><h2></h2></code> , <code><h3></h3></code> , <code><h4></h4></code> , <code><h5></h5></code> et <code><h6></h6></code>
Image	<code></code>
Lien (ancree)	<code>Lien cliquable vers Google</code>
bloc	<code><div></div></code>

A VOUS DE JOUER





EXERCICE 1

0-exercices/exercice1.md

LES ATTRIBUTS



LES ATTRIBUTS

- Un **attribut** est une **information complémentaire** ajoutée au sein d'une **balise ouvrante**.
- Les attributs sont **cumulables** (une balise peut avoir 0 à N attributs)
- Les attributs les plus courantes
 - ***id***: identifiant **unique** donnée à une balise dans une page web
 - Un autre élément du document ne peut pas porter le même ID
 - ***class***: propriété donnée à un ensemble d'éléments qui peuvent avoir des caractéristiques ou un style communes
 - ***dataset***: propriété avec un nommage libre, les mots doivent être séparés par le tiret - après le prefix ***data-***
 - Exemples
 - `<header data-content='something here'></header>`
 - `<footer data-info-end='something here'></footer>`
 - ***src***: spécifique aux balises *img*, *script* pour indiquer la source des images ou scripts à charger
 - ***href***: pour indiquer la cible d'un lien *hypertext*

LES LISTES



LES LISTES

[Source image Quora](#)

ul html tag	Result
<pre> Coffee Tea Milk </pre>	<ul style="list-style-type: none"> • Coffee • Tea • Milk
ol html tag	Result
<pre> Coffee Tea Milk </pre>	<ol style="list-style-type: none"> 1. Coffee 2. Tea 3. Milk

- *ul*: balise parente contenant les éléments d'une *liste non ordonnée*
- *ol*: balise parente contenant les éléments d'une *liste ordonnée*
- *li*: balise enfant de *ul* ou *ol* constituant *un élément de la liste*

A VOUS DE JOUER





EXERCICE 2

0-exercices/exercice2.md

LES TABLEAUX



LES TABLEAUX

```
<table border="1">
  <thead>
    <tr>
      <th>Cours</th>
      <th>Nombre d'heures de théorie</th>
      <th>Nombre heures pratique</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>Git</td>
      <td>2</td>
      <td>5</td>
    </tr>
    <tr>
      <td>Fondamentaux Web</td>
      <td>1</td>
      <td>6</td>
    </tr>
    <tr>
      <td>HTML/CSS</td>
      <td>2</td>
      <td>5</td>
    </tr>
  </tbody>
  <tfoot>
    <tr>
      <th>Totaux</th>
      <td>5</td>
      <td>16</td>
    </tr>
  </tfoot>
</table>
```

- Structure
 - <table></table>*
- En-tête (header)
 - <thead></thead>*
- Contenu (body)
 - <tbody></tbody>*
- Pied (footer)
 - <tfoot></tfoot>*
- Ajout d'une ligne
 - <tr></tr>*
- Ajout de colonne d'en-tête
 - <th></th>*
- Ajout colonne du corps du tableau
 - <td></td>*

A VOUS DE JOUER





EXERCICES 3 et 4

0-exercices/exercice3.md

0-exercices/exercice4.md

LES FORMULAIRES



LES FORMULAIRES

[Source image les enovateurs](#)

[Documentation formulaire HTML](#)

```
<!DOCTYPE html>
<html lang="fr">
  <head>
    <meta charset="UTF-8">
    <title>Notre grande fête</title>
  </head>
  <body>
    <h1>Notre grande fête</h1>
    <form>
      <label for="nom">Nom</label>
      <input type="text" name="nom" id="nom" />

      <br /><br />

      <label for="prenom">Prénom</label>
      <input type="text" name="prenom" id="prenom" />

      <br /><br />

      <label for="participation">Participerez-vous à notre grande fête ?</label>
      <select name="participation" id="participation">
        <option value="non">Non</option>
        <option value="peut-etre">Peut-être</option>
        <option value="Oui">Oui</option>
      </select>

      <br /><br />

      <label for="message">Message : </label>
      <textarea name="message" id="message"></textarea>

      <br /><br />

      <button type="submit" name="envoyer" >Envoyer</button>
    </form>
  </body>
</html>
```

- Structure de base
 - `<form></form>`
 - *method="post"* ou *"get"*, attribut méthode d'envoi des données
 - *action="filename_or_controller"*, attribut destination des données lors de la soumission du formulaire
- Balises de formulaires
 - Textes : `<label></label>`
 - Champs de saisie
 - `<input>` (types : *text*, *number*, *date*, *checkbox*, *password*, etc.)
 - `<textarea></textarea>`
 - Liste déroulante
 - `<select> <option>M</option></select>`
 - Boutons
 - `<button></button>`
 - `<input type="button"/>`
 - `<input type="submit"/>`

A VOUS DE JOUER





EXERCICE 5

0-exercices/exercice5.md

HTML5 CHEAT SHEET [source image borntodev](https://sourceimage.borntodev.com/html5-cheat-sheet/)

HTML5 CHEAT SHEET

borntoDev

ดาวน์โหลด HTML5 Cheat Sheet ดาวน์โหลดได้ฟรี >



Basic

Tag
ลักษณะการเขียนของ HTML
เปิดด้วยสัญลักษณ์ < และปิดด้วย </>
Element
ส่วนประกอบต่างๆของหน้า HTML แต่ละอัน
ต้องใส่ tag เปิดและ tag ปิด
เช่น <p>borntodev</p> ป็นเป็น 1 element
Attribute
คุณสมบัติของ element เช่น
Comment
เปิดด้วย <!-- และปิดด้วย -->

<!DOCTYPE html>
ระบุว่าเป็น HTML5
<html>
Element หลักของ HTML
<head>
Element สำหรับประกาศเรื่องของหน้า HTML
<title>
Element สำหรับกำหนด "ชื่อเพจ"
(ชื่อที่แสดงบนเว็บเบราว์เซอร์)
<body>
Element สำหรับใส่เนื้อหาทั้งหมดของหน้าเว็บ

Basic Attribute

id
กำหนด id ให้กับ element เป็นตัวเลขเฉพาะ
เช่น: element โดยที่ id id ชื่อ
class
กำหนด class ให้กับ element หลาย element
ใช้ class เดียวกันได้
name
กำหนด name ให้กับ element ใช้บ่อยใน
form element
style
กำหนด CSS ให้กับ element
onclick
กำหนด script ที่: เมื่อทำงานที่คลิก
ที่ element นั้น
title
กำหนด text ที่: แสดงเมื่อเมาส์ชี้ไป
ที่ element นั้น
data
การเก็บข้อมูลเพิ่มเติมนอก HTML
โดย: ไม่แสดงบนหน้า HTML
width
กำหนดความกว้าง
height
กำหนดความสูง

Basic Tag

<div>
สร้าง Block element ที่อยู่ในบรรทัดใหม่
และจะขยายเต็มพื้นที่บนบรรทัดนั้น

สร้าง Inline element จะไม่ขึ้นบรรทัดใหม่
และจะเก็บพื้นที่ในบรรทัดนั้น
<h1>
หัวข้อย่อยของเอกสารตามตัวเลข
มีตั้งแต่ 1 ถึง 10
<p>
Paragraph: ย่อหน้าของ "เป็นย่อหน้า" ใช้
เพื่อเป็นการเขียนย่อหน้า
<a>
Tag เพื่อ "ลิงก์ไปยังเว็บไซต์"
ต้องใส่ attribute "href" เพื่อใส่ลิ้งค์

Tag สำหรับ "ขึ้นบรรทัดใหม่"
<hr>
Tag สำหรับสร้าง "เส้นแนวนอน"

Tag สำหรับ "รูปภาพ"
ต้องใส่ attribute "src" เพื่อใส่ลิ้งค์รูปภาพ
<video>
Tag สำหรับ "วิดีโอ"
<audio>
Tag สำหรับ "เสียง"
<button>
Tag สำหรับสร้าง "ปุ่ม"

Table

<table>
tag สำหรับสร้าง "ตาราง"
<thead>
tag สำหรับใส่ข้อมูล "หัวตาราง"
<tbody>
tag สำหรับใส่ข้อมูล "ตัวตาราง"
<tfoot>
tag สำหรับใส่ข้อมูล "ท้ายตาราง"

Header Element

<title>
สำหรับกำหนด "ชื่อเพจ"
<link>
สำหรับ Import ไฟล์ภายนอกเข้ามา
<script>
สำหรับกำหนด Script ที่ใช้ภายใน
หน้า HTML (ไม่จำเป็นต้องใส่ใน header)
<style>
สำหรับกำหนด CSS ที่ใช้ภายในหน้า
HTML (ไม่จำเป็นต้องใส่ใน header)

List

สร้าง "Unordered List"

สร้าง "Ordered List"

สร้างข้อมูลแต่ละตัวใน "List"
ใช้ได้ทั้งใน และ
<dl>
สร้าง "Description List"
<dt>
สร้างข้อมูลใน <dl>
<dd>
สร้างข้อมูลอธิบาย <dt>

Example

#1: โครงสร้าง HTML5

```
<!DOCTYPE html>
<html>
<head>
<title>"ชื่อเพจ"</title>
</head>
<body>
<div>
</div>
</body>
</html>
```

Example

#2: msId Attribute

```
<button
id="hello"
class="greeting-button"
style="background-color: red;"
onclick="console.log(click)"
title="Hello Greeting Button"
data-mydata="this is my greeting
button">
Hello
</button>
```

Example

#3: การใส่ Table

```
<table>
<tr>
<th>ชื่อ</th>
<th>นามสกุล</th>
<th>อาชีพ</th>
</tr>
<tr>
<td>พต</td>
<td>วิมล</td>
<td>วิ</td>
</tr>
<tr>
<td>พต</td>
<td>วิมล</td>
<td>วิ</td>
</tr>
</table>
```

Example

#5: โครงสร้าง Form

```
<form action="/action_page"
method="post">
<label for="name">Name</label>
<input type="text" name="name">
<label for="address">Address
</label>
<textarea name="address"
id="address" cols="30" rows="4">
</textarea>
<input type="radio" name="gender"
id="male" value="male">
<label for="male">Male</label>
<input type="radio" name="gender"
id="female" value="female">
<label for="female">Female
</label>
<select name="animal" id="animal">
<option value="dog">Dog
</option>
<option value="cat">Cat
</option>
<option value="bird">Bird
</option>
</select>
<input type="submit">
</form>
```

Example

#4: msId List

```
<ul>
<li>Kind of Dog</li>
<li>Kind of Cat</li>
<li>Kind of Bird</li>
<li>Kind of Fish</li>
</ul>
```

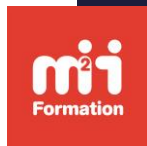
Text

	ตัวหนา	<mark>	เน้นคำ	<sub>	ตัวเล็ก
	ตัวหนา	<small>	ตัวเล็ก	<sup>	ตัวเล็ก
<i>	ตัวเอียง		ขีดฆ่า	<sup>	ตัวเล็ก
	ตัวเอียง	<ins>	ขีดฆ่า	<sup>	ตัวเล็ก

Form

<form>
สร้างฟอร์มสำหรับเก็บข้อมูลจากผู้ใช้
action
attribute ใน <form> ระบุ path หรือ url ที่ต้องการ
ส่งข้อมูล form ไป
method
attribute ใน <form> ระบุ HTML Method มาตรฐาน
เช่น "get" หรือ "post"
target
attribute ใน <form> "self" ส่งข้อมูลไปที่หน้าต่าง
(กำหนดไว้) และ "blank" ส่งข้อมูลไปยังหน้าต่างใหม่
<input>
element สำหรับรับข้อมูลจากผู้ใช้และส่งมาที่เซิร์ฟเวอร์
โดยมี "type"
<input type="text">
หรือ "input type="password">
<input type="radio">
radio button ที่ผู้ใช้เลือก "หนึ่งตัวเลือก"
ใน "หลายตัวเลือก"

COPYRIGHT©2020 BORNTODEV ALL RIGHTS RESERVED



PARTIE CSS



m2information.fr

INTRODUCTION



HISTORIQUE

1996

- Naissance du **CSS1**

2001

- **CSS 2.1**

1998

- **CSS2**

Version
actuelle
CSS3

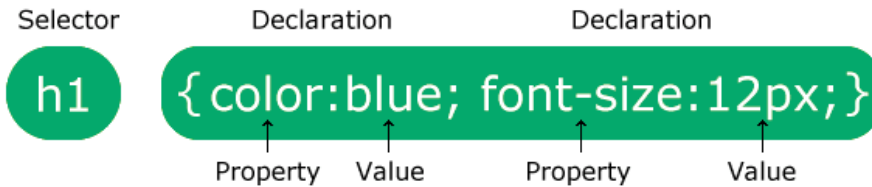
```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link rel="stylesheet" href="main.css">
  <title>Document</title>
  <style>
    p {
      color: orange;
    }
  </style>
</head>
<body>
  <h1 style="color: red; font-weight:bold;">Title</h1>
  <p>Paragraph</p>
</body>
</html>
```

- Le *CSS* permet de gérer
 - Couleurs
 - Polices
 - Mise en page
 - Etc
- Est Complémentaire du *HTML*
- Comment l'utiliser avec le *HTML* ?
 1. Soit dans un **fichier externe** *filename.css* avec une liaison dans le document HTML au niveau du `<head></head>` grâce à la balise `link`.
`<link rel="stylesheet" href="main.css"/>`
 2. Ou directement dans le document HTML avec l'utilisation de la balise `<style></style>`.
 3. Ou directement au sein d'une balise HTML avec l'utilisation de l'attribut *style*.

LES SELECTEURS



[Source image W3Schools](#)



- Sélecteur
 - élément *HTML* ciblé
- Propriété
 - élément de style ciblé
- Valeur
 - valeur à appliquer à l'élément de style ciblé
- /*Commentaires en CSS*/

A VOUS DE JOUER





EXERCICE 6

0-exercices/exercice6.md

SYNTHÈSE SÉLECTEURS

All css selectors

CSS {selectors: cheat-sheet}

Name	CSS	Description	Results
Universal Selector	*	Select all elements	a b c d
Type Selector	div	Select elements of that type Select div elements	a div c div
Class Selector	.c	Select elements with that class Select elements with the c class	a b c d
Id Selector	#i	Select elements with that id Select elements with the id *It is best practice to not use ids in CSS	#a #b #i #d

Name	CSS	Description	Results
Descendant Selector	div a	Select elements that are descendants of the first element Select anchors that are inside a div	div a b a c a d
Direct Child Selector	div > a	Select elements that are direct children of the first element Select anchors that are direct children of a div	div a b a c a d
General Sibling Selector	div ~ a	Select elements that are siblings of the first element and come after the first element Select all anchors that are siblings of a div and come after the div	a div a a
Adjacent Sibling Selector	div + a	Select elements that are siblings of the first element and come directly after the first element Select all anchors that are siblings of a div and come directly after the div	a div a a
Or Selector	div, a	Select elements that match any selector in the list Select all anchors and all divs	a div a b
And Selector	div.c	Select elements that match all the selector combinations Select all divs with the class c	a div c c div

Attribute

Name	CSS	Description	Results
Has Attribute	[a]	Select elements that have that attribute Select elements with the a attribute	a a="1" c d
Exact Attribute	[a="1"]	Select elements that have that attribute with exactly that value Select elements with the a attribute with a value of 1	a a="1" c d
Begins With Attribute	[a^="1"]	Select elements that have that attribute which start with that value Select elements with the a attribute with a value that starts with 1	a="12" a="21"
Ends With Attribute	[a\$="1"]	Select elements that have that attribute which end with that value Select elements with the a attribute with a value that ends with 1	a="12" a="21"
Substring Attribute	[a*="1"]	Select elements that have that attribute which contain that value anywhere Select elements with the a attribute with a value that contains a 1	a="12" a="21"

Pseudo Element

Name	CSS	Description	Results
Before Selector	div::before	Creates an empty element directly before the children of selected element	div before c after
After selector	div::after	Creates an empty element directly after the children of selected element	div before c after

Pseudo Class State

Name	CSS	Description
Hover Selector	button:hover	Select elements that are hovered by the mouse Select buttons that are being hovered
Focus Selector	button:focus	Select elements that are focused. Select buttons that are being focused Focus is set by either clicking on an element or clicking on an element such as a button or anchor tag
Required Selector	input:required	Select inputs that are required Select inputs with the required attribute
Checked Selector	input:checked	Select checkboxes/radio buttons that are checked Select inputs that are checked
Disabled Selector	input:disabled	Select inputs that are disabled Select inputs with the disabled attribute

PSEUDO-CLASSES

[Source image dev.to/aredev](https://dev.to/aredev)

Pseudo Class	Position/Other	
Name	CSS	Description
First Child Selector	<code>a:first-child</code>	Select elements that are the first child inside a container Select anchors that are the first child
Last Child Selector	<code>a:last-child</code>	Select elements that are the last child inside a container Select anchors that are the last child
Nth Child Selector	<code>a:nth-child(2n)</code>	Select elements that are the nth child inside a container based on the formula Select anchors that are even-numbered children
Nth Last Child Selector	<code>a:nth-last-child(3)</code>	Select elements that are the nth child inside a container based on the formula counting from the end Select anchors that are the third to last child
Only Child Selector	<code>a:only-child</code>	Select elements that are the only child inside a container Select anchors that are the only child
First Of Type Selector	<code>a:first-of-type</code>	Select elements that are the first of a type inside a container Select the first anchor in a container
Last Of Type Selector	<code>a:last-of-type</code>	Select elements that are the last of a type inside a container Select the last anchor in a container
Nth Of Type Selector	<code>a:nth-of-type(2n)</code>	Select elements that are the nth of a type inside a container based on the formula Select every second anchor
Nth Last Of Type Selector	<code>a:nth-last-of-type(2)</code>	Select elements that are the nth of a type inside a container based on the formula counting from the end Select the second to last anchor
Only Of Type Selector	<code>a:only-of-type</code>	Select elements that are the only of a type inside a container Select anchors that are the only anchor in a container
Not Selector	<code>a:not(.c)</code>	Select all elements that do not match the selector inside the container Select all anchor tags that do not have the class

- Ajouter un style spécifique en fonction de l'état dans lequel se trouve l'élément ciblé ou en tenant compte des autres éléments présents dans le document HTML
- [Documentation MDN pseudo-classes](#)
- [Documentation W3Schools sélecteurs](#)
- [Test sélecteurs W3Schools](#)

PSEUDO-ELEMENTS

Pseudo-element	
::first-line	Appliquer des propriétés uniquement à la première lettre de l'élément ciblé.
::before	Ajouter du contenu avant l'élément cible à l'aide la propriété content et de sa valeur (première enfant de la cible)
::after	Ajouter du contenu après l'élément cible à l'aide la propriété content et de sa valeur (dernière enfant de la cible)

Propriété	Valeur	
content	«Contenu entre guillemets »	Associé aux pseudo-elements ::before et ::after dont la valeur est généré par le CSS et affichée par le navigateur

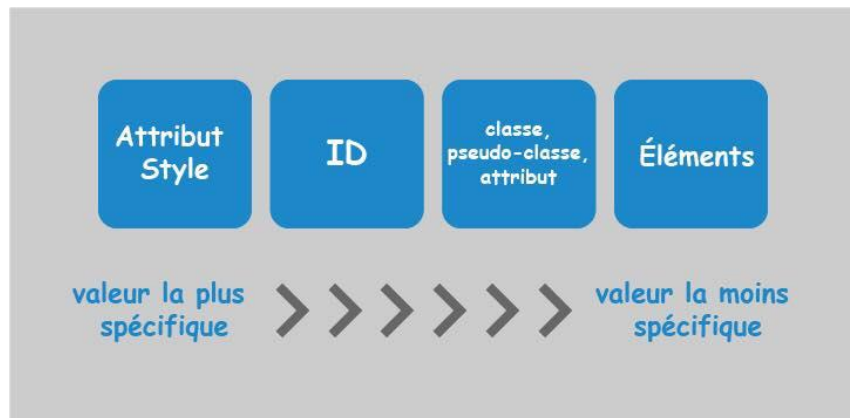
Appliquer les propriétés CSS à une sous-partie de l'élément sélectionné

[Documentation MDN pseudo-éléments](#)

SPÉCIFICITÉS

[Source image gauche w3docs](#)

[Source image droite groupe Sii](#)



- Lorsque plusieurs règles (propriétés et valeurs) sont appliquées à un élément avec des sélecteurs différents qui ciblent ce même élément, le navigateur applique en premier les règles les plus spécifiques.

<code>div.class1 > div.class2.class3.class4 .class5:hover</code>			
0 Inline styles	0 IDs	6 Classes, attributes and pseudo-classes	2 Elements and pseudo-elements
<code>#id1:hover</code>			
0 Inline styles	1 IDs	1 Classes, attributes and pseudo-classes	0 Elements and pseudo-elements

LES UNITES



NUMÉRIQUES ET ABSOLUES

Unité num. et absolue	
<i>cm</i>	Réservé aux feuilles de style dédiée à l'impression
<i>mm</i>	Réservé aux feuilles de style dédiée à l'impression
<i>in</i>	Réservé aux feuilles de style dédiée à l'impression
<i>px</i>	Unité de référence pour les écrans, un écran est défini par sa résolution en pixels

NUMÉRIQUES ET RELATIVES

Unités num. et relatives	
%	Dimension est proportionnelle à la dimension du conteneur (parent) à utiliser principalement pour définir une largeur (<i>width</i>)
<i>em</i>	Relative par rapport à la taille de la police de l'élément ou de l'élément parent
<i>rem</i>	Relative par rapport à la taille de la police de l'élément racine html ou body
<i>vh</i>	Vh pour viewport height (hauteur visible) 1 vh = 1 % de la hauteur visible
<i>vw</i>	Vw pour viewport width (largeur visible) 15 vw = 15 % de la largeur visible

LES COULEURS



Notation	Unités pour les couleurs	Signification
Nom	Nom de la couleur en anglais	Valeur en langage naturel, rapidement limité par rapport à toutes les possibilités (~16 millions)
RGB	rgb(0,255,10)	Triplet des nombres compris entre 0 et 255 (1 octet) qui représente les couleurs rouge, vert et bleu
Hexadécimal encodage 6 octets (notation longue)	#74dd8a	Notation en base 16 permettant d'économiser le nombre d'octets utilisés pour coder une couleur
Hexadécimal encodage 3 octets (notation courte)	#dab	Notation raccourcie de l'hexadécimal en 3 octets lorsque les lettres sont identiques #dab = #ddaabb
RGB avec opacité	rgba(15, 20, 35, .9)	Identique au rgb sauf qu'on tient compte de la transparence dont la valeur est comprise entre 0 et 1

LES PROPRIÉTÉS



LES PROPRIÉTÉS

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

LES PROPRIÉTÉS

	Propriété	Exemple(s)
Bordure	<i>border</i>	<i>border : 1px solid black ;</i>
Bordures arrondies	<i>border-radius</i>	<i>border-radius : 10px ;</i>
Soulignements et autres décorations	<i>text-decoration</i>	<i>text-decoration : underline line-through overline none ;</i>
Ombres block	<i>box-shadow</i>	<i>box-shadow : 6px 6px 0px black ;</i>

[Source image christian Lisangola](#)

BOX MODEL

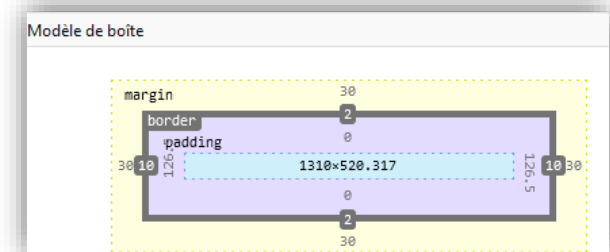


BOX MODEL

- Block
 - Occupe toute la largeur disponible
 - Élément suivant se positionne en bas (retour à la ligne)
 - Propriétés *width* & *height* modifiables

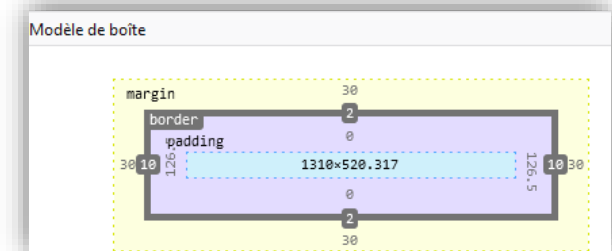
- Inline
 - Occupe la place nécessaire qu'il lui faut
 - Élément suivant se positionne à la suite
 - Propriétés *width* et *height* non modifiables

- Inline-block
 - Se comporte comme un *inline*
 - Propriétés *width* & *height* modifiables



BOX SIZING

- Définit la façon dont la hauteur et la largeur totale d'un élément sont calculées à l'aide de la propriété *box-sizing*
- Valeur par défaut *content-box*
 - A la largeur (*width*) et à la hauteur (*height*) de base de l'élément, viennent s'ajouter les marges intérieures (*padding*), les marges extérieures (*margin*) et la bordure (*border*)
- Valeur *border-box*
 - Prend en compte le *padding*, *margin* et *border* dans le calcul du *width* et *height*
 - Valeurs du *width* et *height* incluent le *margin*, *padding* et *border*



A VOUS DE JOUER





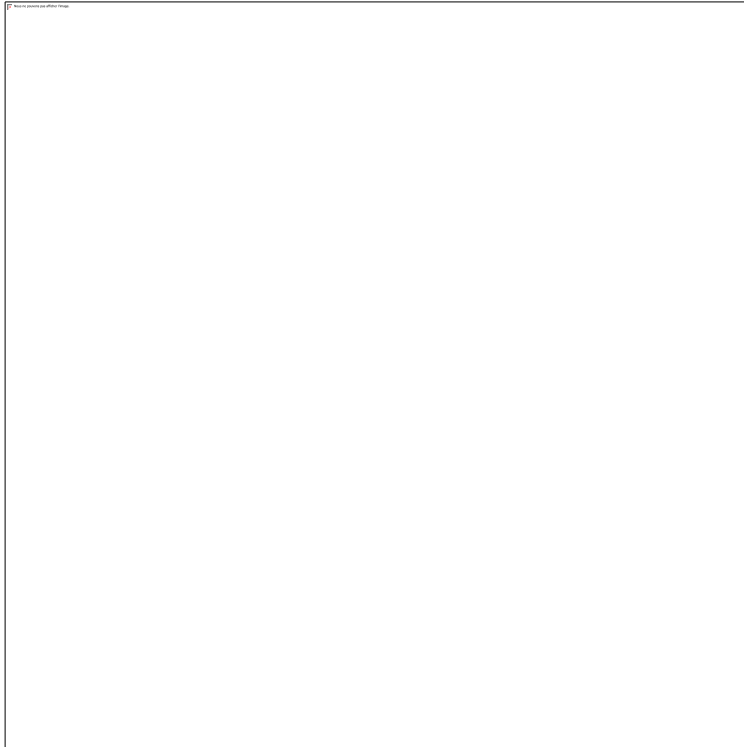
EXERCICE 7

0-exercices/exercice7.md



SYNTHÈSE FLEXBOX

[Source image Samanthaming](#)



- Boîtes flexibles
- Un *conteneur* (l'élément parent)
 - Rendu flexible grâce à la déclaration *display : flex*
 - Les éléments vont se positionner de manière flexible à l'intérieur selon une direction
- 2 axes
 - *Axe principale* ou main axis
 - *Axe secondaire* ou cross axis ou axe transversal

FLEXBOX CHEAT SHEET [source image Mariana Simon](#)

Flexbox Cheat Sheet

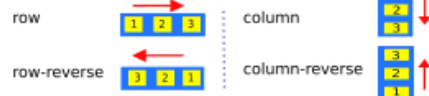
@simonpaix

Parent properties

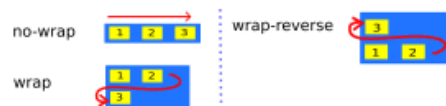
display: enables flex context for all direct children.

```
.container{
  display: flex; // or inline-flex
}
```

flex-direction: sets the main-axis.



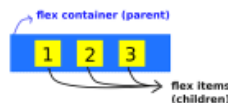
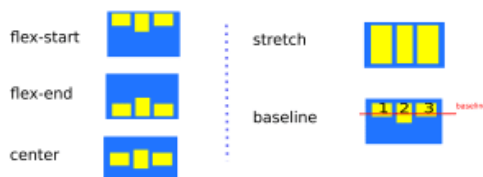
flex-wrap: allows the items to wrap as needed.



justify-content: defines alignment along the main axis.

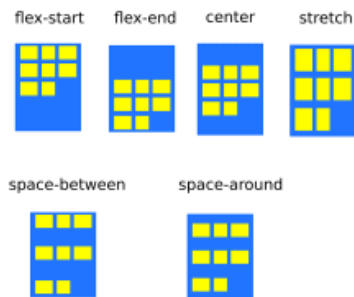


align-items: defines alignment along the cross axis.



LearnPine

align-content: aligns multiple lines, like justify-content does with individual items.



Children properties

order: changes the order of flex items.

```
.item {
  order: 3 // the default is 0
}
```



flex-grow: allows item to grow using remaining space.

```
.item-1 { flex-grow: 0; } //default
.item-2 { flex-grow: 1; }
```



Tip: If all items have flex-grow: 1, the remaining space is distributed equally.

flex-shrink: defines the ability for a flex item to shrink.

```
.one { flex-shrink: 1; }
.two { flex-shrink: 2; }
.three { flex-shrink: 3; }
.four { flex-shrink: 4; }
```

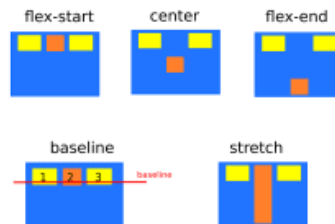


Tip: Defaults to 1. The highest the value the more it shrinks compared to siblings.

flex-basis: sets the default size of a flex item. It accepts:

- specific values : pixels, rm, %
- auto : defaults to width or height property
- content : automatic sizing, based on its content
- global values : inherit, initial, unset

align-self: overrides default alignment (or the one specified by align-items) for a specific item.



A VOUS DE JOUER





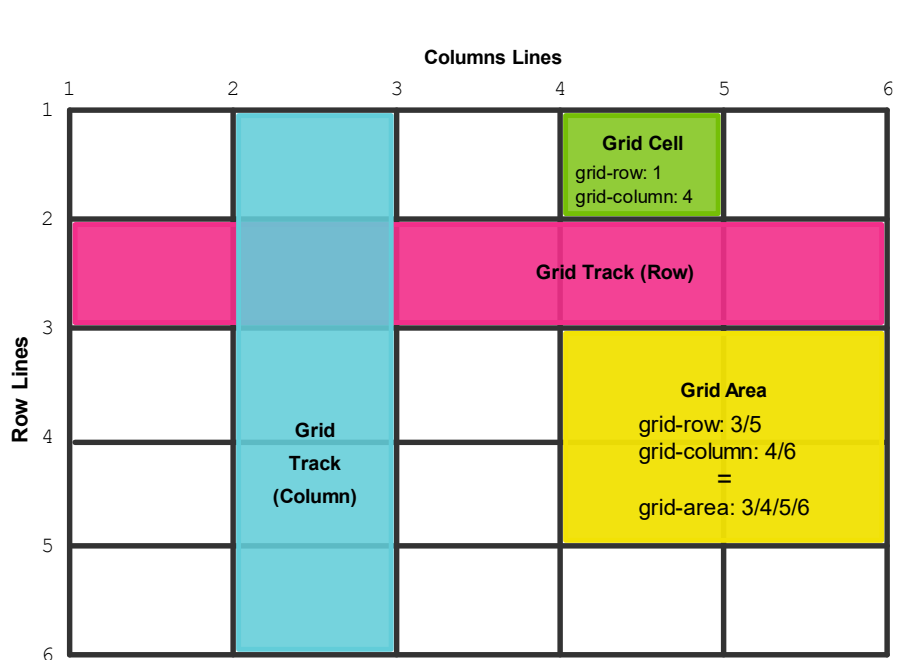
EXERCICE 8

0-exercices/exercice8.md

CSS GRID

SYNTHÈSE CSS GRID

[Source image Kustavo](#)



- Grille
- Un *conteneur* (l'élément parent)
 - Transformer en grille grâce à la déclaration *display: grid*
 - Les éléments vont se positionner selon 2 dimensions (colonne et ligne).
- Unité *fr unit* (fraction unit) correspond à une fraction de l'espace disponible dans le conteneur de la grille
 ([Source Developer Mozilla](#))

CSS GRID CHEAT SHEET [source image Simon Paix](#)

Grid Cheat Sheet



Parent properties

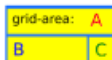
display: enables grid context for all direct children.
`.container{ display: grid; } // or inline-grid`

grid-template: defines the rows and columns of the grid. Set track size values and line-names(optional).

`grid-template-columns: 10px 30px auto 20%;`
`grid-template-rows: repeat(3, 20px);`

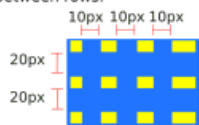
grid-template-areas:

`"A A A A"`
`B B B C"`



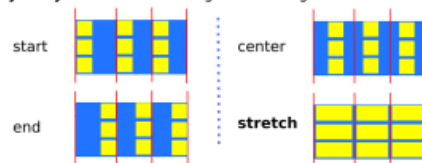
grid-gap: sets the size of the grid lines, the gutters between columns and between rows.

`column-gap: 10px;`
`row-gap: 20px;`

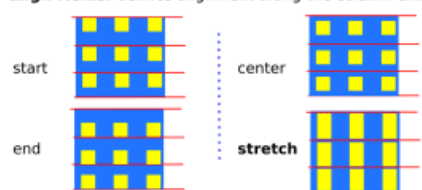


tip: default values are in bold

justify-items: defines alignment along the row axis.



align-items: defines alignment along the column axis.



grid-auto-flow: defines how to automatically place grid items that aren't explicitly placed.



grid container (parent)



justify-content: justifies all grid content on row axis if total grid size is smaller than container.



space-between **space-around** **space-evenly**



align-content: justifies all grid content on column axis if total grid size is smaller than container.

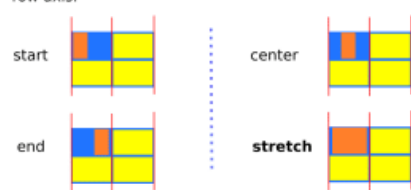


space-between **space-around** **space-evenly**

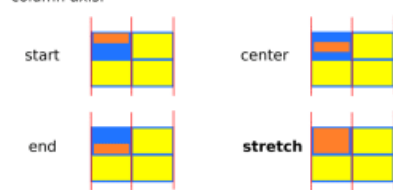


Children properties

justify-self: aligns an item inside a single cell along the row axis.



align-self: aligns an item inside a single cell along the column axis.



grid-column: determines the item's location based on a start and an end column lines (or a span).

`grid-column-start: 2;`
`grid-column-end: 4;`
`// or grid-column: 2 / 4`



grid-row: same but for the row location.

`grid-row-start: 1;`
`grid-row-end: 3;`
`// or grid-row: 1 / span 2`



[Source image velog Danubi21](#)



- Le positionnement permet de gérer la mise en page des éléments à l'aide de la propriété CSS *position*.
- Elle peut prendre 5 valeurs :
 1. *static* : valeur par défaut.
 2. *relative* : Positionnement ajusté par rapport à la position initiale de l'élément.
 3. *absolute* : Positionnement ajusté par rapport au parent le plus proche qui n'est pas en position *static*.
 4. *fixed* : Semblable à *absolute*, cependant l'élément est toujours visible sur la fenêtre en scrollant (navigation sur la page).
 5. *sticky* : Positionnement *relative* puis se transforme en *fixed* en scrollant.

[Source image MDN](#)

CSS Demo: z-index

The image shows a CSS demo for the z-index property. On the left, there is a list of five CSS rules: `z-index: auto;`, `z-index: 1;`, `z-index: 3;`, `z-index: 5;`, and `z-index: 7;`. The first rule is highlighted with a blue border. On the right, a visual representation shows three overlapping rectangular boxes. The top box is blue and labeled `z-index: 6`. The middle box is light blue and labeled `z-index: 4`. The bottom box is light blue and labeled `z-index: 2`. Below these, there is a yellow box labeled `Change my z-index`. At the bottom, there are three overlapping pink boxes, all labeled `z-index: auto`.

- La propriété position est complétée par les propriétés *top*, *left*, *bottom* et *right* pour positionner l'élément à l'endroit souhaité.

Les valeurs sont numériques avec une unité absolue ou relative. Ils peuvent être négatives .

- La propriété **z-index** permet de gérer l'affichage des éléments lorsqu'ils se chevauchent.

A VOUS DE JOUER



EXERCICES 9 À 11

`0-exercices/exercice9.md`

`0-exercices/exercice10.md`

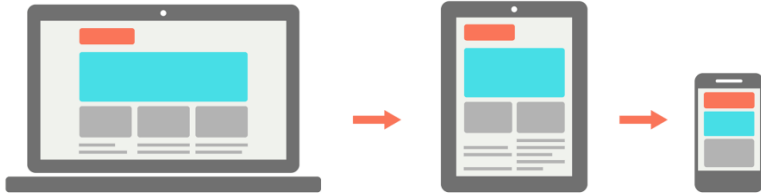
`0-exercices/exercice11.md`

MOBILE FIRST ET RESPONSIVE DESIGN



MOBILE FIRST ET RESPONSIVE DESIGN

[Source image Frederic Gonzalo](#)



Responsive Web Design

Mobile First Web Design



- Le **Mobile First** est méthodologie de conception inventée par *Luke Wroblewski* en 2011.

Concevoir son application d'abord sur **mobile** puis adapter le contenu pour les écrans plus grands.

- Le **Responsive Web Design** est une méthode qui permet de créer une page Web qui **s'adapte automatiquement** par rapport à la **taille de l'écran** d'affichage.

- Ces méthodes sont possible grâce à la mise en place des *medias queries* dans le style *CSS*.

1 seul code *HTML* et des règles *CSS* différentes selon les tailles d'écran.

MISE EN PLACE DES MEDIA QUERIES

[Source image Christian Lisangola](#)

```
@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 */
}
```

[Source image Megawebdesign](#)



Types de média et opérateurs logiques cf. documentation	
<i>all</i>	Caractéristique pour désigner tout type de media (<i>screen</i> , <i>print</i> , <i>speech</i>)
<i>only</i>	Caractéristique pour indiquer les règles à appliquer uniquement pour un type de media spécifique
<i>min-width/width/max-witdh</i>	Largeur minimal (au moins) ou fixe ou maximal (jusqu'à)
<i>orientation</i>	Orientation du périphérique, <i>portrait</i> ou <i>landscape</i> (paysage)
<i>and</i>	Opérateur logique permettant de combiner les caractéristiques
<i>not</i>	Opérateur logique permettant de spécifier ce qu'on ne veut pas (l'opposé)

TAILLES DES ÉCRANS

Tailles	
Maximum 600px de large	Smartphone, montre
Minimum 600px de large	Orientation portrait tablette et phablette (entre smartphone et tablette)
Minimum 768px de large	Orientation paysage tablette
Minimum 992px de large	Ordinateurs portable et de bureau
Minimum 1200px de large	Écrans de télévision , ordinateurs portables, ordinateurs de bureau très large

- Les tailles des écrans peuvent varier en fonction des librairies CSS utilisées plus tard. Ici, les tailles sont celles définies par le site [W3schools](https://www.w3schools.com/css/css3_media_queries.asp)

A VOUS DE JOUER

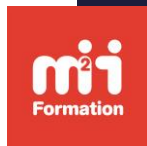


EXERCICES 12 À 14

0-exercices/exercice12.md

0-exercices/exercice13.md

0-exercices/exercice14.md



FIN



m2information.fr