

## I. Introduction au CSS

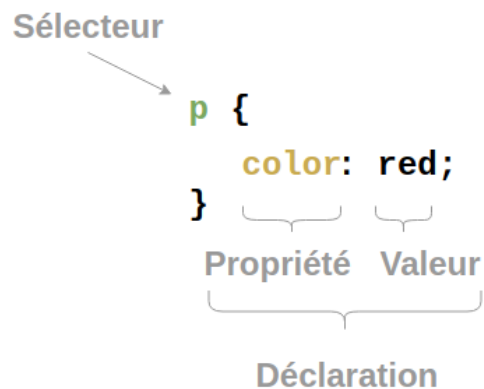
Le **CSS** (Cascading Style Sheets) est un langage utilisé pour décrire la présentation visuelle des pages web. Il permet de contrôler la mise en forme des éléments HTML, comme les couleurs, les polices, les marges, et la disposition. CSS sépare le contenu (HTML) de la présentation, facilitant la maintenance et la modification de l'apparence d'un site web.



### 1. Définition d'une règle CSS

Une **règle CSS** se compose de deux parties principales :

- ✓ **Sélecteur** : Il identifie les éléments HTML à styliser.
- ✓ **Déclaration** : Elle contient les propriétés CSS et leurs valeurs à appliquer aux éléments sélectionnés.



#### Exemple d'utilisation :

html	css
<code>&lt;p&gt; Hello world ! &lt;/p&gt;</code>	<pre>p {   color: blue;   font-size: 16px; }</pre>



Hello world !

## 2. Types de Sélecteurs CSS

### a. Sélecteur de type

Cible un élément HTML par son nom de balise. Par exemple utiliser le sélecteur **p** pour tous les paragraphes.

#### Exemple d'utilisation :

html	CSS
<pre>&lt;p&gt; Hello world ! &lt;/p&gt; &lt;span&gt; Bonjour &lt;/span&gt;</pre>	<pre>span {   color: red; }</pre>



Hello world !

Bonjour

### b. Sélecteur de classe :

Cible un ou plusieurs éléments avec une classe spécifique, précédé d'un point (.).

#### Exemple d'utilisation :

html	CSS
<pre>&lt;p&gt; Hello &lt;/p&gt; &lt;p&gt; Bonjour &lt;/p&gt; &lt;p class="esp"&gt; hola &lt;/p&gt;</pre>	<pre>.esp {   color: red; }</pre>



Hello

Bonjour

hola

### c. Sélecteur d'ID :

Cible un élément unique avec un identifiant spécifique, précédé d'un dièse (#).

#### Exemple d'utilisation :

*Abidi*

LYCÉE MONJI  
SLIM SBIBA

html	CSS
<pre>&lt;p&gt; Hello &lt;/p&gt; &lt;p id="esp"&gt; Bonjour &lt;/p&gt; &lt;p class="esp"&gt; hola &lt;/p&gt;</pre>	<pre>#esp {   color: red; }</pre>



Hello  
 Bonjour  
 hola

#### d. Sélecteur universel :

Cible tous les éléments (\*).

#### Exemple d'utilisation :

html	CSS
<pre>&lt;span&gt; Hello &lt;/span&gt; &lt;p&gt; Bonjour &lt;/p&gt; &lt;div&gt; hola &lt;/div&gt;</pre>	<pre>* {   color: red; }</pre>



Hello  
 Bonjour  
 hola

#### e. Sélecteur d'attribut :

Cible les éléments avec un attribut spécifique. (balise[attribut="valeur"])

#### Exemple d'utilisation :

html	CSS
<pre>Nom : &lt;input type="text"&gt; Tel : &lt;input type="tel"&gt;</pre>	<pre>input[type=text] {   color: red; }</pre>



Nom :  Tel :

### f. Sélecteurs combinés :

Il s'agit de la combinaison d'un sélecteur de balise avec un sélecteur de classe afin de cibler avec plus de précision certains éléments HTML.

#### Exemple d'utilisation :

html	CSS
<pre>&lt;p class="en"&gt; Hello &lt;/p&gt; &lt;p&gt; Bonjour &lt;/p&gt; &lt;p&gt; hola &lt;/p&gt; &lt;div class="en"&gt; reste ... &lt;/p&gt;</pre>	<pre>p.en {   color: red; }</pre>

↓

Hello  
 Bonjour  
 hola  
 reste ...

### g. Sélecteurs multiples :

Les **sélecteurs multiples** permettent d'appliquer les mêmes styles à plusieurs éléments différents en les séparant par une **virgule (,)** dans la feuille de style CSS. Cela évite de répéter plusieurs règles identiques.

#### Exemple d'utilisation :

html	CSS
<pre>&lt;h1&gt;Bienvenue &lt;/h1&gt; &lt;p&gt;paragraphe&lt;/p&gt; &lt;div&gt; une division.&lt;/div&gt;</pre>	<pre>p,h1 {   color: red; }</pre>

↓

Bienvenue  
 paragraphe  
 une division.

fbidi />  
 YCÉE MONJI  
 SLIM SBIBA

## h. pseudo-classe

Une **pseudo-classe** est un mot-clé ajouté à un sélecteur, précédé d'un deux-points : qui permet de définir un **état particulier** d'un élément HTML **sans ajouter de classe ou d'attribut**.

**Liste de pseudo-classes :**

Pseudo-classe	Élément concerné	Moment où elle s'applique
:link	Lien <a>	Lorsqu'un lien <b>n'a pas encore été visité</b>
:visited	Lien <a>	Lorsqu'un lien <b>a déjà été visité</b>
:hover	Tout élément interactif	Lorsqu'on <b>passse la souris dessus</b>
:active	Bouton ou lien	Lors d'un <b>clic actif</b> (appui sans relâche)
:focus	Champs de formulaire	Lorsqu'un élément reçoit le <b>focus</b>

**Exemple d'utilisation :**

html	CSS
<pre>&lt;p&gt;Bienvenue &lt;/p&gt; &lt;p&gt;paragraphe&lt;/p&gt;</pre>	<pre>p:hover {   color: red; }</pre>



Avant le passage de souris : Bienvenue paragraphe
 après le passage : Bienvenue paragraphe

## II. Propriétés de mise en forme du texte en CSS

Le CSS offre un large éventail de propriétés permettant de styliser le texte dans les pages web. Voici quelques-unes des plus courantes, avec des exemples :

### 1. font-family :

Cette propriété permet de définir la police de caractères utilisée pour le texte.

< Abidi />

LYCÉE MONJI  
SLIM SBIBA

**Exemple d'utilisation :**

html	CSS
<code>&lt;p&gt;Bienvenue &lt;/p&gt;</code>	<code>p { font-family :arial,sans-serif ; }</code>

↓

Bienvenue

Ici, le texte des paragraphes utilisera la police **Arial** avec une alternative **sans-serif** si Arial n'est pas disponible. Voici les 5 famille de polices avec des exemples des fonts :

Generic Font Family	Examples of Font Names
Serif	Times New Roman Georgia Garamond
Sans-serif	Arial Verdana Helvetica
Monospace	Courier New Lucida Console Monaco
Cursive	<i>Brush Script MT</i> <i>Lucida Handwriting</i>
Fantasy	<b>Copperplate</b> Papyrus

**2. font-weight :**

Elle permet de définir l'épaisseur du texte. Les valeurs possibles incluent normal, bold, ou des valeurs numériques comme 100 à 900.

**Exemple d'utilisation :**

html	CSS
<code>&lt;p&gt;Bienvenue &lt;/p&gt; &lt;div&gt; division &lt;/div&gt;</code>	<code>p { font-weight :bold ; }</code>

↓

**Bienvenue**  
division

### 3. font-style :

Utilisée pour mettre le texte en italique. Les valeurs possibles sont **normal**, **italic**, ou **oblique**.

#### Exemple d'utilisation :

html	CSS
<pre>&lt;p&gt;Bienvenue dans notre &lt;span&gt; plateforme &lt;/span&gt; de jeux en ligne &lt;/p&gt;</pre>	<pre>span {   font-style :italic ; }</pre>



Bienvenue dans notre *plateforme* de jeux en ligne

### 4. font-size :

↑ Text

Elle permet de définir la taille du texte. Les valeurs peuvent être en pixels (px).

#### Exemple d'utilisation :

html	CSS
<pre>&lt;p&gt;Bienvenue &lt;/p&gt; &lt;p class= "A"&gt; division &lt;/p&gt;</pre>	<pre>p {   font-size :16px ; } .A {   font-size : 32px ; }</pre>



Bienvenue

division

< Abidi />

### 5. font :

Cette propriété regroupe plusieurs propriétés liées aux polices, comme font-style, font-weight, font-size, et font-family dans une seule déclaration.

**Exemple d'utilisation :**

html	css
<code>&lt;p&gt;Bienvenue dans notre site &lt;/p&gt;</code>	<code>p { font: italic bold 16px Arial ; }</code>

↓

***Bienvenue dans notre site***

**6. text-align :**

permet le contrôle l'alignement horizontal du texte. Les valeurs incluent **left**, **right**, **center**, et **justify**.

Alice opened the door and found that it led into a small passage, not much larger than a rat hole: she knelt down and looked along the passage into the loveliest garden you ever saw.	Alice opened the door and found that it led into a small passage, not much larger than a rat hole: she knelt down and looked along the passage into the loveliest garden you ever saw.	Alice opened the door and found that it led into a small passage, not much larger than a rat hole: she knelt down and looked along the passage into the loveliest garden you ever saw.	Alice opened the door and found that it led into a small passage, not much larger than a rat hole: she knelt down and looked along the passage into the loveliest garden you ever saw.
left	center	right	justify

**7. text-shadow :**

Cette propriété permet d'ajouter une ombre au texte. Elle prend 4 valeurs : l'offset horizontal et vertical, le flou, et la couleur.

**Exemple d'utilisation :**

html	css
<code>&lt;h1&gt; un Titre H1 &lt;/h1&gt;</code>	<code>h1 { text-shadow: 2px 2px 5px red; }</code>

↓

**un Titre H1**

**8. text-transform :**

Elle permet de modifier la casse du texte. Les valeurs incluent **uppercase**, **lowercase**, et **capitalize**.



**Exemple d'utilisation :**

html	css
<pre>&lt;h1&gt;Titre principal &lt;/h1&gt; &lt;p&gt;ceci est un texte.&lt;/p&gt;</pre>	<pre>h1 {     text-transform: uppercase; } p {     text-transform: capitalize; }</pre>



# TITRE PRINCIPAL

Ceci Est Un Texte.

**9. color :**

Cette propriété définit la couleur du texte. Les couleurs peuvent être spécifiées en nom (comme red), en hexadécimal .

<b>Maroon</b> #800000	<b>Red</b> #FF0000	<b>Orange</b> #FFA500	<b>Yellow</b> #FFFF00	<b>Olive</b> #808000
<b>Purple</b> #800080	<b>Fuchsia</b> #FF00FF	<b>White</b> #FFFFFF	<b>Lime</b> #00FF00	<b>Green</b> #008000
<b>Navy</b> #000080	<b>Blue</b> #0000FF	<b>Aqua</b> #00FFFF	<b>Teal</b> #008080	
<b>Black</b> #000000	<b>Silver</b> #C0C0C0	<b>Gray</b> #808080		

**HEXADECIMAL COLOR CODES**

Red Color Code    Green Color Code    Blue Color Code

# **RR GG BB**

Signifies Hex Color Code

Each of the two-digit color codes are in the range 00...FF hexadecimal

**EXAMPLES**

**#FFCC99    #3C00A6**

**Exemple d'utilisation :**

html	css
<pre>&lt;h1&gt;Titre principal &lt;/h1&gt;</pre>	<pre>h1 {     color: lime; }</pre>



LYCÉE MONJI M SBIBA  
**Titre principal**

## XII. Propriétés de couleur et de fond :

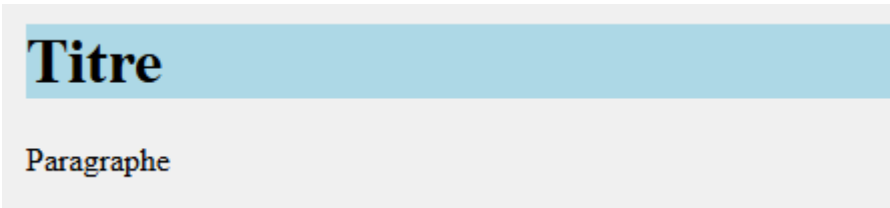
### 1. background-color

La propriété **background-color** définit la couleur de fond d'un élément. Elle peut être spécifiée à l'aide de noms de couleurs, de valeurs hexadécimales, ....

#### Exemple d'utilisation :

html	css
<pre>&lt;h1&gt;Titre &lt;/h1&gt; &lt;p&gt;Paragraphe&lt;/p&gt;</pre>	<pre>body {background-color: #f0f0f0;} h1 {background-color: lightblue;}</pre>

↓




### 2. background-image

La propriété **background-image** permet d'appliquer une image en tant qu'arrière-plan d'un élément. L'image peut être spécifiée par une URL.

#### Exemple d'utilisation :

html	css
<pre>&lt;h1&gt;Titre &lt;/h1&gt; &lt;p&gt;Paragraphe&lt;/p&gt;</pre>	<pre>body { background-image: url("backg.png"); color:white; }</pre>

↓

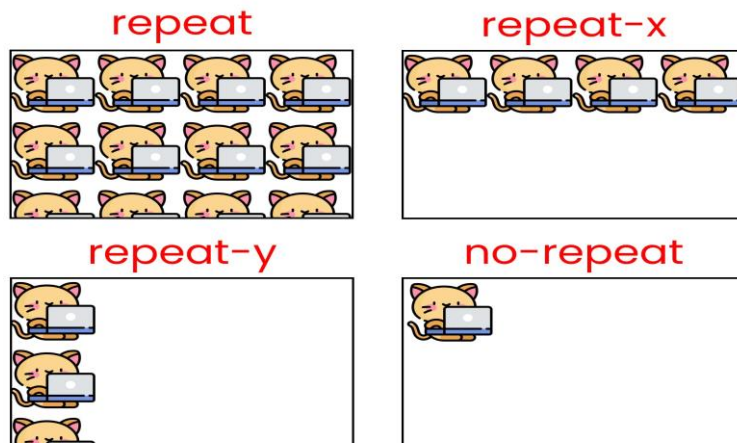


### 3. background-repeat :

La propriété **background-repeat** contrôle la répétition de l'image de fond. Par défaut, une image de fond se répète pour couvrir l'élément. Vous pouvez choisir de désactiver la répétition ou de répéter l'image horizontalement ou verticalement.

#### Valeurs :

- ❖ **repeat** (par défaut) : L'image de fond se répète dans les deux directions.
- ❖ **repeat-x** : L'image se répète uniquement horizontalement.
- ❖ **repeat-y** : L'image se répète uniquement verticalement.
- ❖ **no-repeat** : L'image ne se répète pas.



#### Exemple d'utilisation :

html	css
<pre>&lt;h1&gt;Titre &lt;/h1&gt; &lt;p&gt;Paragraphe&lt;/p&gt;</pre>	<pre>body {   background-color:silver;   background-image: url("backg2.png");   background-repeat: repeat-y; }</pre>



#### 4. background-size :

La propriété **background-size** définit la taille de l'image d'arrière-plan. Vous pouvez définir des dimensions précises, utiliser des pourcentages, ou employer des mots-clés comme **cover** ou **contain**.

##### Valeurs :

- ❖ **auto** (par défaut) : La taille de l'image de fond reste à sa taille d'origine.
- ❖ **cover** : L'image couvre entièrement l'élément tout en gardant ses proportions.
- ❖ **contain** : L'image est redimensionnée pour être complètement visible à l'intérieur de l'élément.
- ❖ **Valeurs libre** : comme 100px 50px, pour des largeurs et hauteurs spécifiques.



##### Exemple d'utilisation :

html	CSS
<pre>&lt;h1&gt;Titre &lt;/h1&gt; &lt;p&gt;Paragraphe&lt;/p&gt;</pre>	<pre>body {   background-color:silver;   background-image: url("backg2.png");   background-repeat: no-repeat;   background-size :cover ; }</pre>



**Titre**

Paragraphe



## 5. background

La propriété **background** est une propriété **raccourcie** qui permet de définir plusieurs propriétés de fond en une seule déclaration, comme background-color, background-image, background-repeat, backgroundposition, et background-size.

### Syntaxe :

**background:** [background-color] [background-image] [background-repeat] [position / taille];

### Exemple d'utilisation :

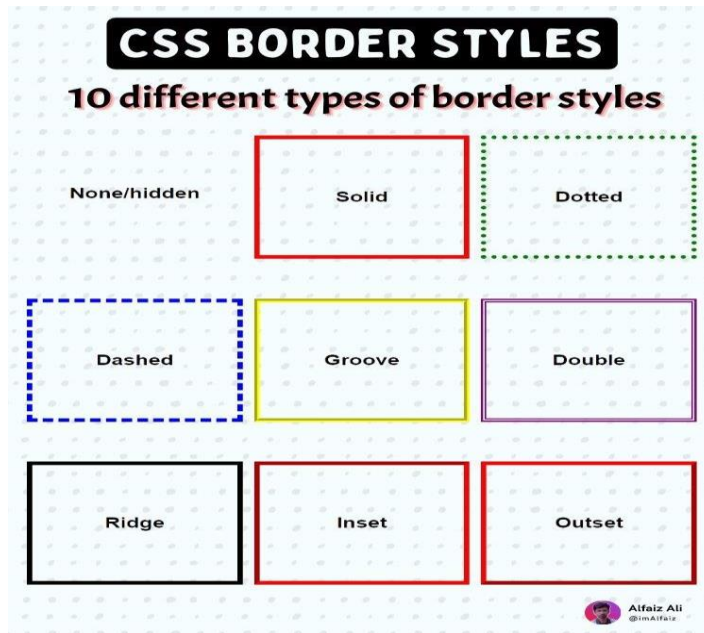
html	css
<pre>&lt;h1&gt;Titre &lt;/h1&gt; &lt;p&gt;Paragraphe&lt;/p&gt;</pre>	<pre>body {   background: lightblue url('background.jpg')   no-repeat center / cover; }</pre>



## propriétés des bordures :

### 5. 1. border-style

La propriété **border-style** définit le style de la bordure d'un élément. Elle peut prendre différentes valeurs pour spécifier l'apparence de la bordure (solide, en pointillés, en tirets, etc.).

**Exemple d'utilisation :**

html	css
<pre>&lt;h1&gt;Titre principal &lt;/h1&gt; &lt;p&gt; paragraphe &lt;/p&gt;</pre>	<pre>h1 { border-style: solid; }</pre>



**Titre principal**

paragraphe

**6. border-color**

La propriété **border-color** définit la couleur de la bordure. Elle peut être spécifiée en utilisant un nom de couleur, une valeur hexadécimale, RGB, ....

**Exemple d'utilisation :**

html	css
<pre>&lt;h1&gt;Titre principal &lt;/h1&gt;</pre>	<pre>h1 { border-style: solid; border-color :red ; }</pre>



**Titre principal**

## 7. border-width

La propriété **border-width** contrôle l'épaisseur de la bordure. Vous pouvez définir une épaisseur spécifique en pixels ou utiliser des valeurs prédéfinies telles que **thin**, **medium**, ou **thick**.

### Exemple d'utilisation :

html	css
<pre>&lt;h1&gt;Titre principal &lt;/h1&gt; &lt;p&gt; un paragraphe&lt;/p&gt;</pre>	<pre>h1,p { border-style: solid; border-color :red ; } h1{border-width :10px ;}</pre>



**Titre principal**

un paragraphe

## 4. border-radius

La propriété **border-radius** permet de créer des coins arrondis pour les bordures. Vous pouvez définir un rayon spécifique pour les coins en pixels ou en pourcentage.

### Exemple d'utilisation :

html	css
<pre>&lt;h1&gt;Titre principal &lt;/h1&gt;</pre>	<pre>h1 { border-style: solid; text-align:center; border-color :red ; border-radius :50% ; }</pre>



**Titre principal**

## 8. 5. border

La propriété **border** est une **propriété raccourcie** qui permet de définir en une seule déclaration le style, la couleur et l'épaisseur de la bordure.

### Syntaxe :

```
border: [épaisseur] [style] [couleur];
```

### Exemple d'utilisation :

html	css
<code>&lt;h1&gt;Titre principal &lt;/h1&gt;</code>	<code>h1 { border : 2px dashed navy ; }</code>

Titre principal

## XIII. Propriétés des listes

### 1. list-style-type

La propriété **list-style-type** définit le type de puce ou de numérotation à utiliser pour les éléments d'une liste. Elle peut être appliquée à des listes ordonnées (`<ol>`) et non ordonnées (`<ul>`).

Valeurs possibles :

armenian	circle	decimal	georgian	decimal-leading-zero	lower-alpha	lower-greek	lower-roman	square	upper-alpha	upper-roman
U.	°	1.	Ⴀ.	01.	a.	α.	i.	■	A.	I.
F.	°	2.	Ⴁ.	02.	b.	β.	ii.	■	B.	II.
Q.	°	3.	Ⴂ.	03.	c.	γ.	iii.	■	C.	III.
Ⴀ.	°	4.	Ⴃ.	04.	d.	δ.	iv.	■	D.	IV.
Ლ.	°	5.	Ⴃ.	05.	e.	ε.	v.	■	E.	V.

### Exemple d'utilisation :

html	css
<code>&lt;span&gt;programme &lt;/span&gt; &lt;ol&gt; &lt;li&gt; workshop 1 &lt;/li&gt; &lt;li&gt; pause &lt;/li&gt; &lt;li&gt; workshop 2 &lt;/li&gt; &lt;/ol&gt;</code>	<code>ol { list-style-type : georgian ; }</code>

programme

- Ⴀ. workshop 1
- Ⴁ. pause
- Ⴂ. workshop 2

LYCÉE MONJI  
SLIM SBIBA

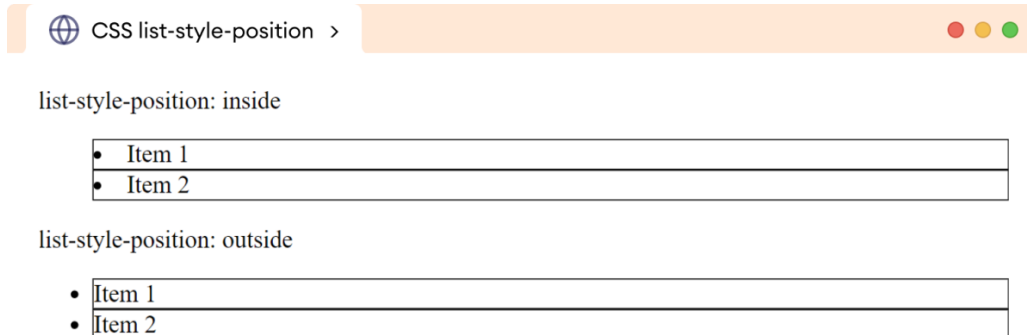


## 2. 2. list-style-position

La propriété **list-style-position** détermine la position de la puce ou du numéro par rapport au contenu de l'élément de la liste.

### Valeurs possibles :

- **inside** : la puce ou le numéro est à l'intérieur du bloc, donc aligné avec le texte de l'élément.
- **outside** (par défaut) : la puce ou le numéro est à l'extérieur du bloc, décalé à gauche du texte.



### Exemple d'utilisation :

html	css
<pre>&lt;span&gt;programme &lt;/span&gt; &lt;ol&gt; &lt;li&gt; workshop 1 &lt;/li&gt; &lt;li&gt; pause &lt;/li&gt; &lt;li&gt; workshop 2 &lt;/li&gt; &lt;/ol&gt;</pre>	<pre>ol { list-style-position: outside; } li { background-color: pink; }</pre>

↓

programme

1. workshop 1
2. pause
3. workshop 2

## 3. 3. list-style-image


La propriété **list-style-image** permet d'utiliser une image en tant que puce dans une liste.

**Exemple d'utilisation :**


html	CSS
<pre>&lt;span&gt;programme &lt;/span&gt; &lt;ol&gt; &lt;li&gt; workshop 1 &lt;/li&gt; &lt;li&gt; pause &lt;/li&gt; &lt;li&gt; workshop 2 &lt;/li&gt; &lt;/ol&gt;</pre>	<pre>ol { list-style-image: url("motif.jpg"); }</pre>

↓


programme



workshop 1



pause



workshop 2

**4. 4. list-style**

La propriété **list-style** est une propriété **raccourcie** qui permet de définir simultanément **list-style-type**, **list-style-position**, et **list-style-image** en une seule déclaration.

**Syntaxe :**

**list-style:** [type] [position] [image];

**Exemple d'utilisation :**

html	CSS
<pre>&lt;span&gt;programme &lt;/span&gt; &lt;ol&gt; &lt;li&gt; workshop 1 &lt;/li&gt; &lt;li&gt; pause &lt;/li&gt; &lt;li&gt; workshop 2 &lt;/li&gt; &lt;/ol&gt;</pre>	<pre>ol { list-style: decimal inside url("motif.jpg"); }</pre>

Dans cet exemple, la liste utilise une image **motif.png**, avec un style de puce décimale comme secours, et les puces sont alignées à l'intérieur du texte.

LYCÉE MONJI  
SLIM SBIBA

## XIV. Propriétés des tableaux

### 5. 1. table-layout

La propriété **table-layout** détermine la manière dont la largeur des colonnes d'un tableau est calculée. Elle influence la rapidité du rendu du tableau et la manière dont les colonnes sont redimensionnées.

#### Valeurs possibles :

##### table-layout: auto

ID	Description	Prix
1	Petit texte	10€
2	Texte beaucoup plus long qui élargit la colonne	20€

##### table-layout: fixed

ID	Description	Prix
1	Petit texte	10€
2	Texte beaucoup plus long qui va être coupé ou passer à la ligne	20€


#### Exemple d'utilisation :

##### html

```
<table>
  <tr>
    <th>Colonne courte</th>
    <th>Colonne avec un texte
    beaucoup plus long qui ne devrait
    pas élargir</th>
  </tr>
  <tr>
    <td>123</td>
    <td> texte pour voir
    l'effet.</td>
  </tr>
</table>
```

##### css

```
table {
  table-layout: fixed;
}
td, th {
  border: 1px solid black;
}
```



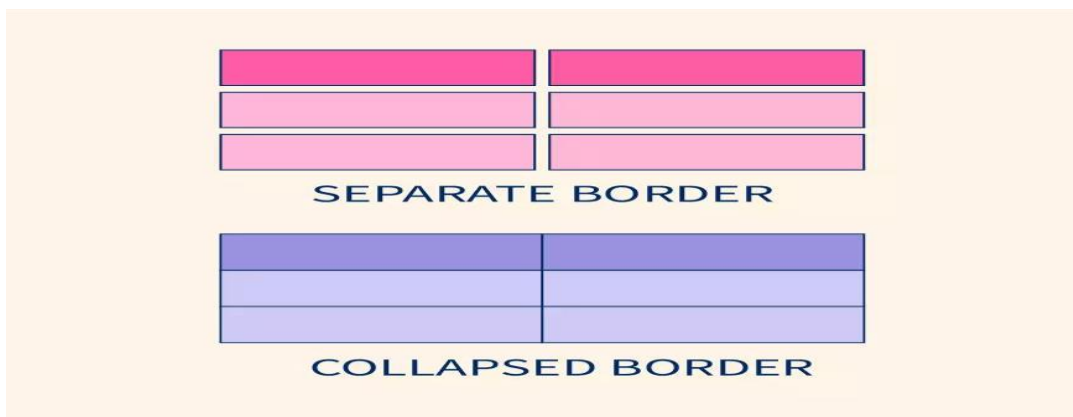
Colonne courte	Colonne avec un texte beaucoup plus long qui ne devrait pas élargir
123	Beaucoup de texte pour voir l'effet.

## 6. 2. border-collapse

La propriété **border-collapse** détermine si les bordures des cellules adjacentes dans un tableau sont fusionnées en une seule bordure ou séparées.


### Valeurs possibles :

- **separate** (valeur par défaut) : Les bordures des cellules sont séparées
- **collapse** : Les bordures adjacentes des cellules fusionnent en une seule bordure. Cela donne une apparence plus propre au tableau avec des bordures continues.



### Exemple d'utilisation :

html	CSS
<pre>&lt;table&gt;   &lt;tr&gt;     &lt;th&gt;Colonne courte&lt;/th&gt;     &lt;th&gt;Colonne avec un texte     beaucoup plus long qui ne devrait     pas élargir&lt;/th&gt;   &lt;/tr&gt;   &lt;tr&gt;     &lt;td&gt;123&lt;/td&gt;     &lt;td&gt; texte pour voir     l'effet.&lt;/td&gt;   &lt;/tr&gt; &lt;/table&gt;</pre>	<pre>table {   table-layout: auto;   border-collapse: collapse; } td, th {   border: 1px solid black; }</pre>



Colonne courte	Colonne avec un texte beaucoup plus long qui ne devrait pas élargir
123	Beaucoup de texte pour voir l'effet.

## XV. Propriétés des images :

La propriété **filter** en CSS permet d'appliquer des effets graphiques (comme un flou ou un changement de couleurs) aux éléments tels que des images, du texte, ou même des blocs. C'est un moyen simple de modifier l'apparence visuelle d'un élément sans avoir à modifier l'image ou l'élément lui-même.

### 1. invert()

La fonction **invert()** inverse les couleurs d'un élément. Cela signifie que les couleurs claires deviennent sombres et les couleurs sombres deviennent claires. La valeur est un pourcentage entre **0%** (aucune inversion) et **100%** (couleurs complètement inversées).



### Exemple d'utilisation :

#### html

```
<span> avant

</span>
<span> après

</span>
```

#### css

```
.f1{
  filter: invert(100%);
}
```



avant



après

## 2. grayscale()

La fonction **grayscale()** applique un effet de désaturation, transformant une image en niveaux de gris. La valeur est un pourcentage entre **0%** (pas de changement) et **100%** (image entièrement en niveaux de gris).



### Exemple d'utilisation :

#### html

```
<span> avant

</span>
<span> après

</span>
```

#### CSS

```
.f1{
  filter: grayscale(70%);
}
```



avant



après

## 3. blur()

La fonction **blur()** applique un flou à l'élément. La valeur spécifie le rayon du flou, exprimé en pixels.





### Exemple d'utilisation :

html	css
<pre>&lt;span&gt; avant &lt;img src="im1.png"&gt; &lt;/span&gt; &lt;span&gt; après &lt;img class="f1" src="im1.png"&gt; &lt;/span&gt;</pre>	<pre>.f1{   filter: blur(5px); }</pre>



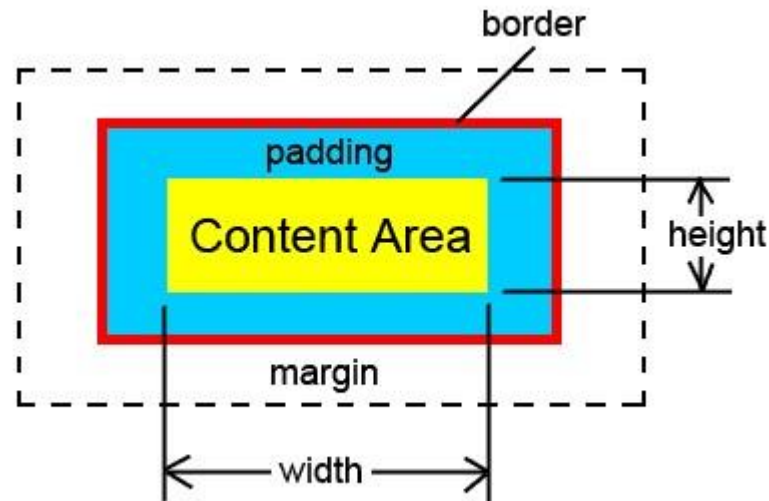
avant



après

## XVI. Propriétés des boîtes :

Les propriétés CSS des boîtes permettent de gérer la taille, la position, les marges, les bordures, le remplissage et d'autres aspects des éléments HTML.

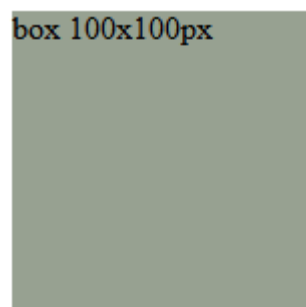


## 1. width et height

- ❖ **width** : définit la largeur d'un élément.
- ❖ **height** : définit la hauteur d'un élément.

### Exemple d'utilisation :

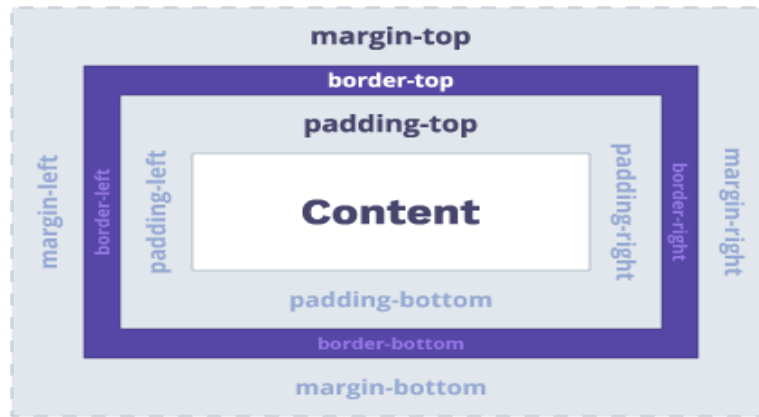
html	css
<code>&lt;div&gt; box 150x150 px&lt;/div&gt;</code>	<pre>div{     width :150px ;     height : 150px ;     background-color: #97A191; }</pre>



## 2. margin

La propriété **margin** ajoute de l'espace autour d'un élément. Elle peut être définie pour les quatre côtés individuellement ou globalement.





### Syntaxe :

Syntaxe	Exemple	Effet
1 valeur	<code>margin: 20px;</code>	20px en haut, droite, bas, gauche
2 valeurs	<code>margin: 20px 50px;</code>	Haut et Bas = 20px Droite et Gauche = 50px
3 valeurs	<code>margin: 10px 30px 50px;</code>	Haut = 10px Droite et Gauche = 30px Bas = 50px
4 valeurs	<code>margin: 5px 10px 15px 20px;</code>	Haut = 5px Droite = 10px Bas = 15px Gauche = 20px
Propriétés individuelles	<code>margin-top: 10px;</code> <code>margin-right: 20px;</code> <code>margin-bottom: 30px ;</code> <code>margin-left: 40px;</code>	Définit chaque côté séparément
Valeurs spéciales	<code>margin: auto ;</code>	centre un bloc horizontalement
	<code>margin: 0;</code>	aucune marge

### 3. padding

La propriété `padding` ajoute de l'espace à l'intérieur de l'élément, entre son contenu et sa bordure.

#### Exemple d'utilisation :

**html**

```
<p> un paragraphe avec
espacement intérieur à gauche
</p>
```

**css**

```
p{
  width:300px;
  border: 2px solid red;
  padding-left : 100px ;
}
```

un paragraphe avec espacement intérieur à gauche

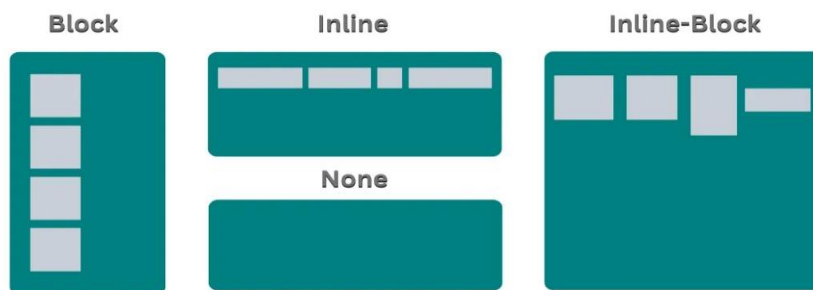
#### 4. Display :

En CSS, la propriété **display** détermine la manière dont un élément est rendu à l'écran :

- ❖ **none** : l'élément est totalement masqué et n'occupe aucun espace dans la page.
- ❖ **block** : l'élément s'affiche sous forme de bloc, prend toute la largeur disponible et commence toujours sur une nouvelle ligne (ex. : <div>, <p>).
- ❖ **inline** : l'élément s'intègre directement dans le flux du texte, à la suite des autres, sans retour à la ligne. On ne peut pas lui appliquer de dimensions (width, height) (ex. : <span>, <a>).
- ❖ **inline-block** : l'élément reste sur la même ligne que les autres, mais il conserve les possibilités de mise en forme d'un bloc (dimensions, marges, espacements).



#### CSS Display Example



#### Exemple d'utilisation :

html	CSS
<pre>&lt;div&gt; box 1 &lt;/div&gt; &lt;div&gt; box 2 &lt;/div&gt;</pre>	<pre>div{ display : inline-block ; width :200px ; background-color : #FAB9C2 ; height :150px ; }</pre>

box 1

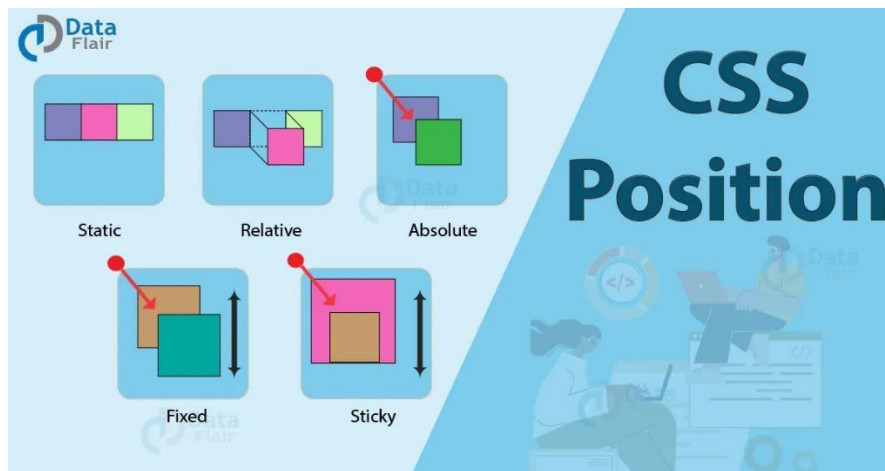
box 2

## 5. Position :

Elle permet de contrôler l'**emplacement** d'un élément dans la page.

### Valeurs communes :

- ❖ **static** (par défaut) : l'élément suit le flux normal du document.
- ❖ **relative** : l'élément est positionné par rapport à sa position normale.
- ❖ **absolute** : l'élément est retiré du flux et positionné par rapport à son conteneur positionné le plus proche.
- ❖ **fixed** : l'élément est positionné par rapport à la fenêtre et ne bouge pas lors du défilement.
- ❖ **sticky** : permet à un élément de rester positionné normalement dans la page, mais de devenir fixe (collant) lorsqu'on fait défiler la page .



La propriété **position** est utilisé souvent avec les propriétés CSS :

- ❖ **top** : décale l'élément vers le bas à partir du haut.
- ❖ **bottom** : décale l'élément vers le haut à partir du bas.
- ❖ **left** : décale l'élément vers la droite à partir de la gauche.
- ❖ **right** : décale l'élément vers la gauche à partir de la droite.

Remarque : Ces propriétés fonctionnent seulement si position ≠ static.

< Abidi />  
LYCÉE MONJI  
SLIM SBIBA

**Exemple d'utilisation :**

html	css
<pre>&lt;div&gt;   &lt;p&gt; un paragraphe &lt;/p&gt; &lt;/div&gt;</pre>	<pre>div{   width :200px ;   background-color : #FAB9C2 ;   height :150px ; } p{   background-color : red ;   position :relative ;   left : 180px ; }</pre>

↓


**6. float :**


La propriété **float** en CSS permet de positionner un élément à gauche ou à droite de son conteneur, de façon que le texte et les autres éléments s'enroulent autour.

**Valeurs principales :**

- ❖ left : l'élément flotte à gauche.
- ❖ right : l'élément flotte à droite.
- ❖ none : valeur par défaut, pas de flottement.
- ❖ **Exemple d'utilisation :**

html	css
<pre>&lt;div&gt;   &lt;p&gt; un paragraphe &lt;/p&gt; &lt;/div&gt;</pre>	<pre>div{   width :300px ;   border : 2px solid red;   height :40px ; } p{float :right ;}</pre>

↓



## 7. overflow

La propriété **overflow** contrôle comment le contenu débordant d'un élément est géré.

**Valeurs :**

- ❖ **visible** (par défaut) : le contenu déborde de la boîte.
- ❖ **hidden** : le contenu qui dépasse est masqué.
- ❖ **scroll** : des barres de défilement sont ajoutées si nécessaire.
- ❖ **auto** : des barres de défilement apparaissent seulement si le contenu déborde.
- ❖ **clip** : semble fonctionner exactement comme overflow: hidden. La différence est que vous pouvez utiliser d'autres propriétés pour faire défiler quelque chose avec **hidden**, mais pas si l'élément a **clip**.

### uide to CSS Overflow



**Exemple d'utilisation :**

html	CSS
<pre>&lt;div&gt;   Contenu trop grand pour le   container &lt;/div&gt;</pre>	<pre>div{   width :50px ;   background-color : #FAB9C2 ;   height :50px ;   overflow :clip ; }</pre>



Contenu  
trop  
grand

## 8. box-shadow

La propriété **box-shadow** ajoute une ombre autour d'un élément.

**Syntaxe :**

```
box-shadow: offsetX offsetY taille color;
```

**Exemple d'utilisation :**

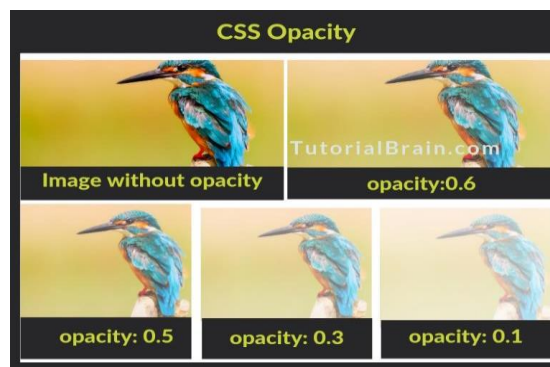
html	CSS
<pre>&lt;div&gt;   Div avec un ombre rouge &lt;/div&gt;</pre>	<pre>div{   width :200px ;   border : 2px solid black ;   height :50px ;   box-shadow : 2px 2px 3px red ; }</pre>

↓

Div avec un ombre rouge

**9. Opacity**

La propriété **opacity** définit la transparence d'un élément. La valeur varie de **0** (entièrement transparent) à **1** (entièrement opaque).

**Exemple d'utilisation :**

html	CSS
<pre>&lt;div&gt;   Div opaque &lt;/div&gt; &lt;div class= "t"&gt;   Div transparente &lt;/div&gt;</pre>	<pre>div{   width :200px ;   background-color :red ;   height :50px ; } .t{Opacity : 0.7 ;}</pre>

↓

Div opaque

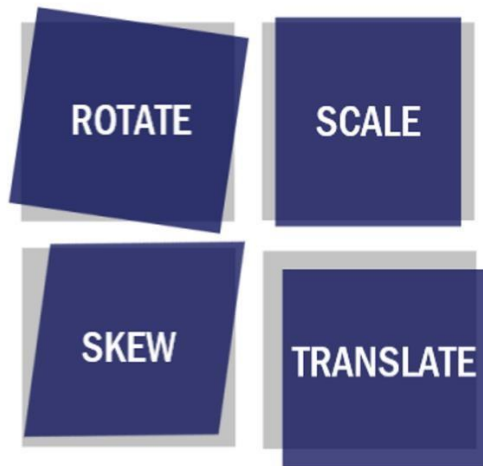
Div transparente

LYCÉE MONJI

SLIM SBIBA

## XVII. Transformation

Les **transformations CSS** permettent de modifier l'apparence des éléments en les faisant pivoter, déplacer, redimensionner ou déformer. Les fonctions principales utilisées pour ces transformations sont : **rotate**, **skew**, **scale**, et **translate**. Voici les définitions et exemples pour chacune d'elles.



### 1. rotate() : Rotation d'un élément

La fonction **rotate()** permet de faire pivoter un élément autour de son point d'origine. La valeur est exprimée en degrés (**deg**).

#### Exemple d'utilisation :

html	css
<pre>&lt;div&gt;   Une division &lt;/div&gt;</pre>	<pre>div{ border :4px solid red ;} div:hover{   transform : rotate(45deg) ; }</pre>

↓

Sans passage de souris	Lorsque le souris entre



## 2. skew() : Inclinaison d'un élément

La fonction **skew()** permet d'incliner un élément le long de l'axe X ou Y. Les valeurs sont exprimées en degrés.

### Exemple d'utilisation :

html	css
<pre>&lt;div&gt;   Une division &lt;/div&gt;</pre>	<pre>div{ border :4px solid red ;} div:hover{   transform : skew(20deg) ; }</pre>

↓

Sans passage de souris	Lorsque le souris entre
	



## 3. scale() : Redimensionnement d'un élément

La fonction **scale()** modifie la taille d'un élément en fonction des facteurs de redimensionnement donnés. Un facteur de **1** signifie que l'élément conserve sa taille d'origine.

### Exemple d'utilisation :

html	css
<pre>&lt;div&gt;   Une division &lt;/div&gt;</pre>	<pre>div{ border :4px solid red ;} div:hover{   transform : scale(1.5) ; }</pre>

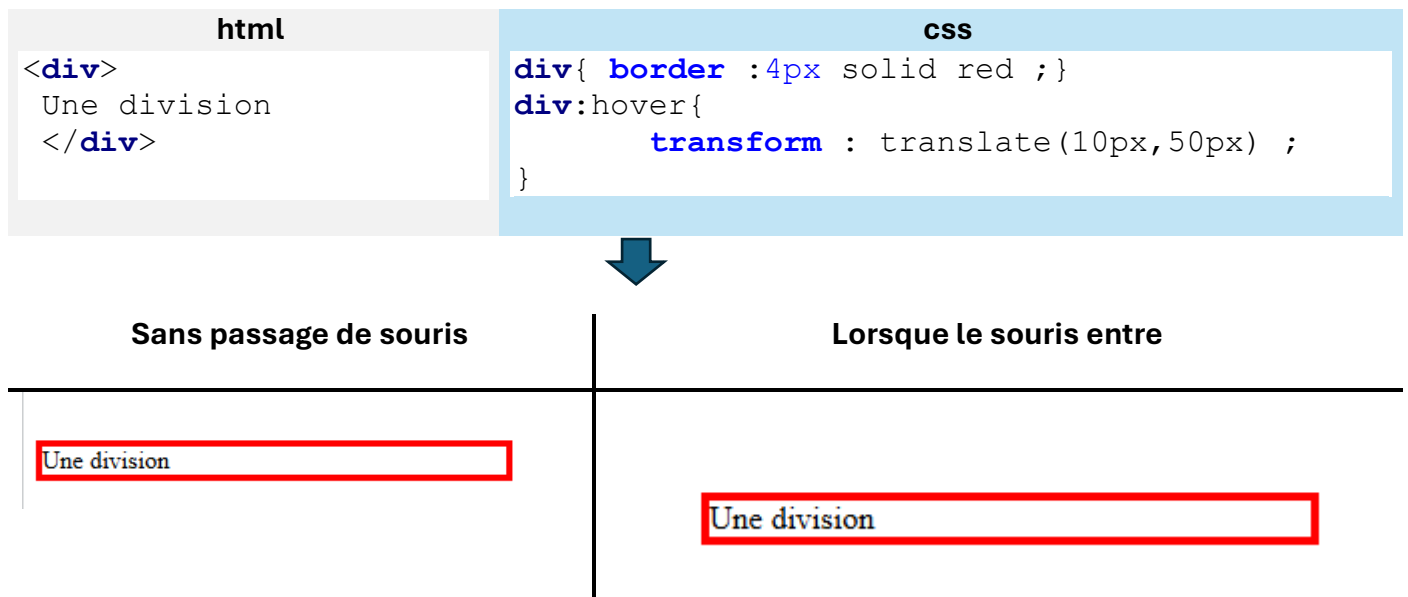
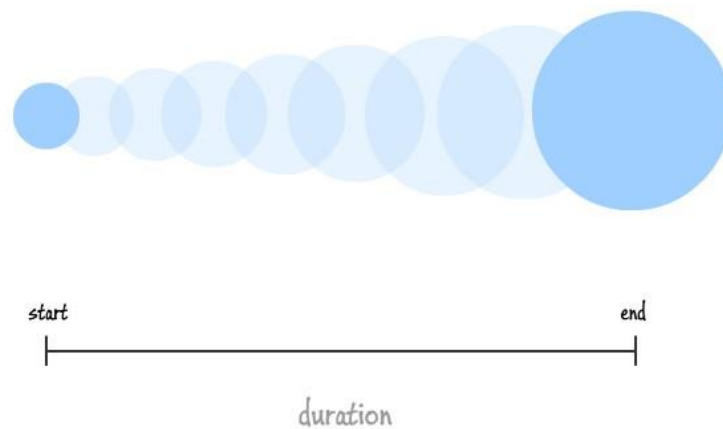
↓

Sans passage de souris	Lorsque le souris entre
	

## 4. translate() : Déplacement d'un élément

La fonction **translate()** déplace un élément sur l'axe X ou Y. Les valeurs sont exprimées en pixels (**px**) ou en pourcentages (%).



**Exemple d'utilisation :****XVIII. Les transitions**

Les **transitions en CSS** permettent d'ajouter des animations simples et fluides lorsque des propriétés d'un élément changent d'état, par exemple lorsqu'un élément est survolé, cliqué, ou lorsqu'une classe lui est ajoutée. Les transitions facilitent le passage progressif d'une propriété CSS à une autre.

LYCÉE MONJI  
SLIM SBIBA

**Propriétés de transition :**

Propriété	Rôle	Exemple
<b>transition-property</b>	Définit quelle(s) propriété(s) CSS vont être animées.	transition-property: width;
<b>transition-duration</b>	Définit la durée de la transition (combien de temps elle dure).	transition-duration: 2s;
<b>transition-delay</b>	Définit le délai avant le début de la transition.	transition-delay: 1s;

**Syntaxe de propriété transition :**

**transition:** <property> <duration> <delay>;

**Exemple :**

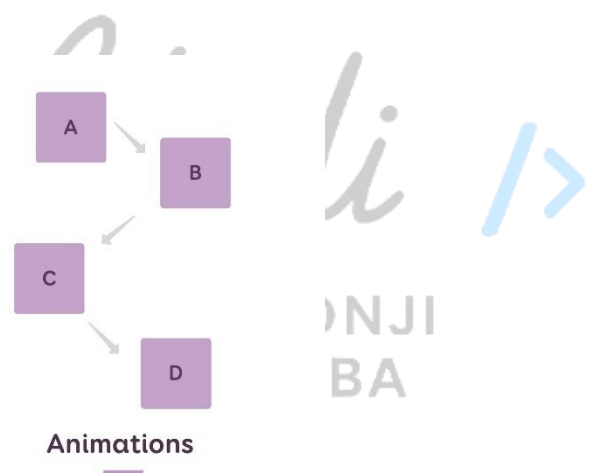
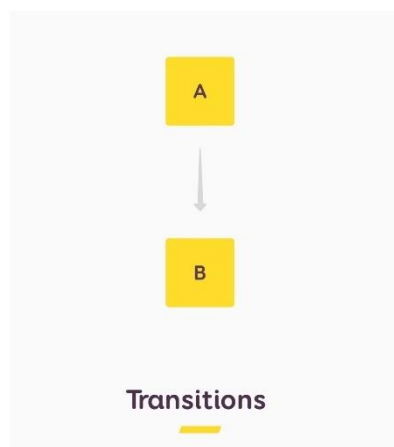
orsqu'on passe la souris sur le div, la couleur d'arrière-plan change vers le bleu. Ce changement commence après **1 seconde de retard** et la transition dure **3 secondes**.

**Méthode 1 :**

```
div:hover {
  background-color: blue;
  transition-property: background-color;
  transition-duration: 3s;
  transition-delay: 1s;
}
```

**Méthode 2 :**

```
div:hover {
  background-color: blue;
  transition: background-color 3s 1s;
}
```

**XIX. animations**

Les **animations en CSS** permettent de créer des mouvements complexes et dynamiques en spécifiant une série d'étapes à travers les valeurs des propriétés CSS. Contrairement aux transitions qui ne se produisent que lorsque l'état d'un élément change (comme lors d'un survol), les animations peuvent être exécutées en continu ou selon un cycle défini. Voici une description détaillée des principales propriétés d'animations CSS :

Propriété	Rôle	Exemple
<b>@keyframes + nom</b>	Définit les étapes de l'animation (nom qu'on réutilise dans animation-name).	@keyframes slide { from { left: 0; } to { left: 100px; } }
<b>animation-name</b>	Indique quel nom d'animation utiliser (défini avec @keyframes).	animation-name: slide;
<b>animation-duration</b>	Durée de l'animation (obligatoire).	animation-duration: 3s;
<b>animation-delay</b>	Délai avant le démarrage de l'animation.	animation-delay: 1s;
<b>animation-iteration-count</b>	Nombre de répétitions de l'animation (1, 2, infinite).	animation-iteration-count: infinite;
<b>animation-direction</b>	Sens de l'animation : (normal, reverse, alternate, alternate-reverse).	animation-direction: alternate;

### Syntaxe générale de propriété animation :

animation: <name> <duration> <delay> <iteration-count> <direction> ;

### Exemple :

Créer un carré rouge qui change de couleur en **4 étapes**. L'animation doit durer **4 secondes**, commencer après **1 seconde**, se répéter **à l'infini** et aller **dans les deux sens** à chaque cycle.

### Réponse :

div	keyframes
<pre>div { width: 100px; height: 100px; background-color: red; /* animation */ animation-name: moveColor; animation-duration: 4s; animation-delay: 1s; animation-iteration-count: infinite; animation-direction: alternate; }</pre>	<pre>@keyframes moveColor { 0% { background-color: red; } 33% { background-color: yellow;} 66% { background-color: green;} 100% { background-color: blue;} }</pre>