مدن المهن و الكفاءات
ⵜⵉⵎⴷⵉⵏⵉⵏ ⵏ ⵜⵎⵙⵙⵓⵔⵉⵏ ⴰ ⵜⵓⵙⵙⵓⵔⵉⵏ
Cités des métiers et des compétences

مكتب التكوين المهني و إنعاش الشغل
Office de la Formation Professionnelle et de la Promotion du Travail

M104
Développer des sites statiques

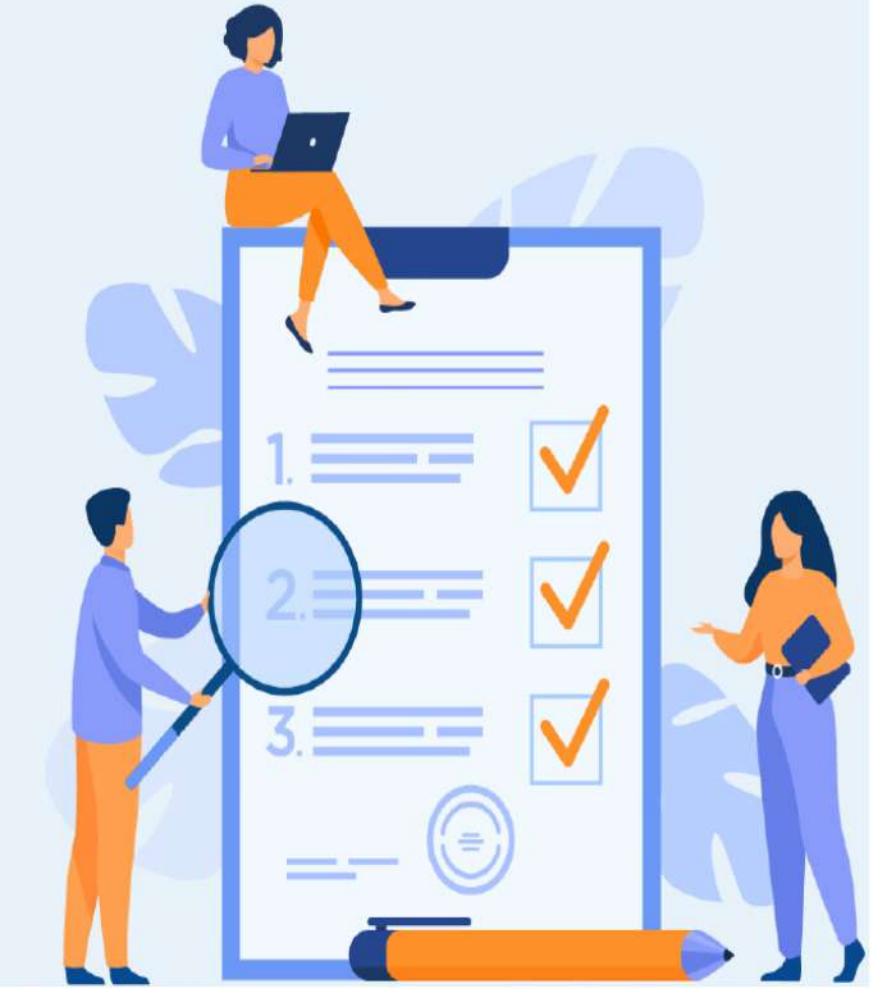
Mme Imane FRITET
imane.fritet@ofppt-edu.ma
Secteur : Digital et IA

PARTIE 3
Mettre en forme une page web avec les feuilles de style CSS

Dans cette partie, vous allez :

- ✓ Maitriser le CSS pour la mise en forme d'une page web
- ✓ Utiliser correctement des positionnements pour l'organisation d'une page web avec CSS
- ✓ Ajouter des animations
- ✓ Adapter des templates HTML/CSS à un site web statique

Mme Imane FRITET imane.fritet@ofppt-edu.ma Secteur : Digital et IA



CHAPITRE 1

Réaliser une page web statique

Ce que vous allez apprendre dans ce chapitre :

- ✓ Présenter les éléments CSS
- ✓ Gérer les types d'intégration du CSS
- ✓ Utiliser les différents CSS

01 - Introduire le CSS

1. Présentation du CSS
2. Codification des couleurs
3. Unités de mesure
4. Positions (center, left, right)
5. Fonts
6. Types d'intégration du CSS
7. Sélecteurs simples (element, class, id)
8. Sélecteurs complexes
9. Pseudo classes

Mme Imane FRITET imane.fritet@ofppt-edu.ma Secteur : Digital et IA

Présentation du CSS

Le **CSS** (**Cascading Style Sheets**) est un langage de feuilles de style utilisé pour définir la présentation visuelle des documents HTML et XML.

Le CSS permet de **séparer la structure d'une page** (le contenu, les titres, les paragraphes, etc.) **de sa mise en page** (les couleurs, les polices, les marges, etc.).

Cette séparation rend le code HTML plus propre et facilite la maintenance et la mise à jour du design.

Application d'un style à un code HTML

Il existe 3 manières pour introduire le style à un code HTML:

- **Dans les balises HTML, via l'attribut style.**
- **Dans la balise <style> du document HTML dans l'en-tête, après la balise title (dans la partie head).**
- **Dans un fichier CSS externe (.css).**

Il est généralement recommandé d'utiliser un fichier CSS externe pour une meilleure organisation et maintenance.

Syntaxe du CSS

La syntaxe du CSS (Cascading Style Sheets) est relativement simple et structurée.

Sélecteur : Le sélecteur cible un ou plusieurs éléments HTML auxquels vous souhaitez appliquer le style. Les sélecteurs peuvent être des balises HTML, des classes, des IDs, des attributs, etc.

```
selecteur {
  propriete: valeur;
}
```

Les déclarations de style sont composées de paires "propriété: valeur".
Une propriété est un attribut CSS tel que "color" (pour la couleur du texte), "font-size" (pour la taille de la police), "margin" (pour les marges), etc.
La valeur est la valeur que vous attribuez à cette propriété.

Exemple :

```
p {
  color: blue;
  font-size: 16px;
  margin-top: 10px;
}
```

Cela signifie que toutes les balises <p> dans le document HTML auront un texte de couleur bleue, une taille de police de 16 pixels et une marge supérieure de 10 pixels

01 - Introduire le CSS

1. Présentation du CSS
2. Codification des couleurs
3. Unités de mesure
4. Positions (center, left, right)
5. Fonts
6. Types d'intégration du CSS
7. Sélecteurs simples (element, class, id)
8. Sélecteurs complexes
9. Pseudo classes

Mme Imane FRITET imane.fritet@ofppt-edu.ma Secteur : Digital et IA

Codification des couleurs (Rappel)

En CSS, il existe plusieurs façons de spécifier les couleurs. Voici les méthodes les plus couramment utilisées pour coder les couleurs :

Méthode	Exemple	Explication
Couleurs nommées	background-color : grey	La couleur est désignée par son nom en anglais. Il n'y a que 16 noms de couleurs normalisés par le W3C
Couleurs en hexadécimal	background-color : #808080	La couleur est désignée par son code hexadécimal : les deux premiers digits correspondent à la valeur de rouge, les deux suivants le vert et les deux derniers le bleu
Couleurs décomposées	background-color : RGB(128,128,128)	La couleur est définie par trois nombres décimaux qui peuvent prendre les valeurs de 0 à 255 indiquant respectivement le taux de rouge, le taux de vert et le taux de bleu.

01 - Introduire le CSS

1. Présentation du CSS
2. Codification des couleurs
3. Unités de mesure
4. Positions (center, left, right)
5. Fonts
6. Types d'intégration du CSS
7. Sélecteurs simples (element, class, id)
8. Sélecteurs complexes
9. Pseudo classes

En CSS, vous pouvez spécifier des unités de mesure pour définir les dimensions, les marges, les espacements et d'autres propriétés.

1. **Unités relatives :**

- **em :** L'unité em est relative à la taille de la police de l'élément parent. Par exemple, si la taille de la police de l'élément parent est de 16 pixels, 1em équivaut à 16 pixels. 0.5em serait donc équivalent à 8 pixels.
- **rem :** L'unité rem est similaire à em, mais elle est relative à la taille de la police de l'élément racine (généralement le <html>). Cela le rend plus prévisible, car il n'est pas influencé par les éléments parents.

2. Unités absolues :

- **px :** Les pixels (px) sont des unités absolues qui sont couramment utilisées pour définir la taille de police, la largeur, la hauteur, les marges, etc.
- **cm :** Le centimètre (cm) est une unité absolue qui correspond à une centième de mètre.
- **mm :** Le millimètre (mm) est une unité absolue qui correspond à un millième de mètre.
- **in :** L'inch (in) est une unité absolue qui correspond à une inch (pouce), soit environ 2,54 centimètres.
- **pt :** Le point (pt) est une unité utilisée principalement pour les dimensions de police et équivaut à 1/72 de pouce.
- **pc :** Le pica (pc) est une unité qui équivaut à 12 points.

Le pixel et le point sont les unités les plus utilisées.
1 in = 96 px
1 cm = 37,8 px
1 mm = 3,78 px.

3. Unités relatives au viewport :

Les unités relatives au viewport permettent aux éléments de s'adapter à la taille de la fenêtre du navigateur.

Elle sont essentielles pour mettre en place un design responsive :

- **vw :** L'unité vw représente une fraction de la largeur de la fenêtre (viewport). Par exemple, 1vw équivaut à 1% de la largeur de la fenêtre.
- **vh :** L'unité vh représente une fraction de la hauteur de la fenêtre. Par exemple, 1vh équivaut à 1% de la hauteur de la fenêtre.

4. Unités de pourcentage :

Les pourcentages (%) sont couramment utilisés pour définir des proportions relatives par rapport aux dimensions parentes. Par exemple, 50% signifie la moitié de la taille de l'élément parent.

01 - Introduire le CSS

1. Présentation du CSS
2. Codification des couleurs
3. Unités de mesure
4. Positions (center, left, right)
5. Fonts
6. Types d'intégration du CSS
7. Sélecteurs simples (element, class, id)
8. Sélecteurs complexes
9. Pseudo classes

En CSS, vous pouvez aligner le texte en utilisant différentes propriétés pour déterminer l'alignement horizontal (gauche, centré, droite) et l'alignement vertical (haut, milieu, bas) d'un élément.

Alignement horizontal (gauche, centré, droite) :

text-align : Cette propriété est utilisée pour aligner le texte horizontalement à l'intérieur d'un élément. Les valeurs courantes pour text-align sont :

- **left (gauche) :** Le texte est aligné à gauche.
- **center (centré) :** Le texte est centré horizontalement.
- **right (droite) :** Le texte est aligné à droite.
- **justify (justifié) :** Le texte est réparti sur toute la largeur de l'élément.

```css
h1 {
  text-align: center;
}
h2 {
  text-align: left;
}
h3 {
  text-align: right;
}
h4 {
  text-align: justify;
}
```

Alignement vertical (haut, milieu, bas) :

vertical-align : Cette propriété est principalement utilisée pour aligner du texte à l'intérieur d'éléments en ligne (par exemple, dans des cellules de tableau ou des éléments).

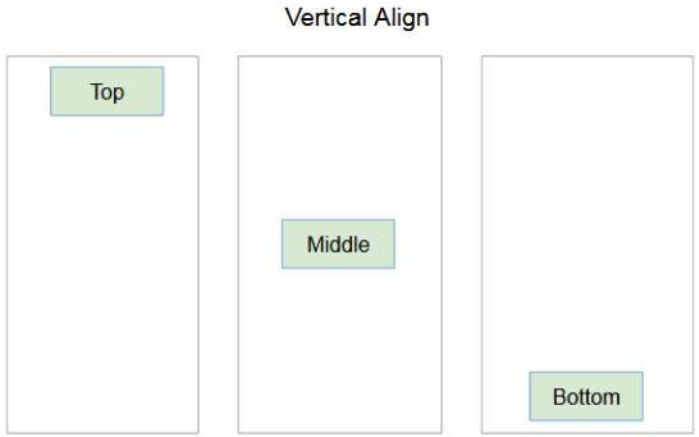
Les valeurs courantes pour vertical-align sont :

baseline : Alignement basé sur la ligne de base du texte.

top : Alignement en haut.

middle : Alignement au milieu.

bottom : Alignement en bas.



Mme Imane FRITET imane.fritet@ofppt-edu.ma Secteur : Digital et IA

Orientation du texte

Les propriétés **direction** et **unicode-bidi** sont utilisées pour changer la direction du texte d'un élément :

```
h1{
  direction : rtl;
  unicode-bidi : bidi-override;
}
```

01 - Introduire le CSS

1. Présentation du CSS
2. Codification des couleurs
3. Unités de mesure
4. Positions (center, left, right)
5. Fonts
6. Types d'intégration du CSS
7. Sélecteurs simples (element, class, id)
8. Sélecteurs complexes
9. Pseudo classes

Mme Imane FRITET imane.fritet@ofppt-edu.ma Secteur : Digital et IA

color : cette propriété définit la couleur du texte.

font-family : Cette propriété spécifie la police de caractères à utiliser pour le texte.

font-size : Cette propriété détermine la taille de la police du texte.

font-weight : Utilisée pour définir l'épaisseur (poids) de la police (par exemple, normal, bold, etc.).

text-align : Cette propriété contrôle l'alignement horizontal du texte (gauche, centré, droite, justifié).

text-transform : Cette propriété permet de transformer le texte en majuscules, en minuscules ou en capitales.

text-decoration : Utilisée pour ajouter ou supprimer la décoration du texte (soulignement, ligne par-dessus ou par-dessous, etc.).

line-height : Cette propriété détermine la hauteur de la ligne, c'est-à-dire l'espacement vertical entre les lignes de texte.

letter-spacing : Utilisée pour définir l'espacement entre les caractères dans le texte.

text-shadow : Vous permet d'ajouter une ombre au texte en contrôlant sa couleur, sa position horizontale et verticale, et son flou.

Propriétés du texte en CSS3 - Exemple

```css
p {
  color: blue;
  font-family: "Arial", sans-serif;
  font-size: 16px;
  font-weight: bold;
  text-align: center;
  line-height: 1.5;
}
a {
  text-decoration: none;
}
h1 {
  letter-spacing: 2px;
  text-transform: uppercase;
  text-shadow: 2px 2px 4px rgba(0, 0, 0, 0.5);
}
```

01 - Introduire le CSS

1. Présentation du CSS
2. Codification des couleurs
3. Unités de mesure
4. Positions (center, left, right)
5. Fonts
6. Types d'intégration du CSS
7. Sélecteurs simples (element, class, id)
8. Sélecteurs complexes
9. Pseudo classes

L'intégration du CSS dans une page web peut se faire de plusieurs manières, en fonction des besoins et des préférences du développeur.

1. **Intégration interne (Inline CSS) :** Dans ce cas, les styles CSS sont directement inclus dans les balises HTML au moyen de l'attribut style.

Par exemple :

```
<p style="color: blue; font-size: 16px;">Ceci est un paragraphe.</p>
```

2. **Intégration interne (Internal CSS) :** Les styles CSS sont inclus dans l'en-tête de la page HTML à l'intérieur d'une balise <style>. Cela permet de définir des styles pour plusieurs éléments de la page.

Par exemple :

```
<style>
  p {
    color: blue;
    font-size: 16px;
  }
</style>
```


Mme Imane FRITET imane.fritet@ofppt-edu.ma Secteur : Digital et IA

3. **Intégration externe (External CSS) :** Les styles CSS sont placés dans un fichier CSS externe distinct. Ce fichier est ensuite lié à la page HTML en utilisant la balise <link> dans l'en-tête de la page HTML.

Par exemple :

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="stylesheet" type="text/css" href="css/styles.css">
    <title>Document</title>
</head>
```

Dans cet exemple, les styles CSS sont placés dans le fichier styles.css qui se situe dans le dossier css.

01 - Introduire le CSS

1. Présentation du CSS
2. Codification des couleurs
3. Unités de mesure
4. Positions (center, left, right)
5. Fonts
6. Types d'intégration du CSS
7. Sélecteurs simples (element, class, id)
8. Sélecteurs complexes
9. Pseudo classes

Mme Imane FRITET imane.fritet@ofppt-edu.ma Secteur : Digital et IA

Les sélecteurs en CSS sont utilisés pour cibler des éléments HTML spécifiques en fonction de leur type, de leur classe ou de leur ID.

1. Sélecteur d'élément :

Le sélecteur d'élément cible un type d'élément HTML spécifique, comme les balises <p>, <h1>, <a>, etc

Exemple:

```
p {
  /* Styles pour tous les paragraphes */
}

a {
  /* Styles pour tous les liens hypertexte */
}
```

2. Sélecteur de classe :

Le sélecteur de classe cible des éléments avec une classe spécifique définie dans l'attribut class.

```
.mon-style {
  /* Styles pour tous les éléments ayant la classe "mon-style" */
}
```

2. Sélecteur de classe :

Exemple:

```
<table border="1">
    <tr><th>Nom</th><th>Prénom</th><th>Age</th></tr>
    <tr class="impaire "><td>MANSOURI</td><td>Hassan</td><td>23</td></tr>
    <tr><td>SAFIR</td><td>Laila</td><td>22</td></tr>
    <tr class="impaire"><td>BICHRI</td><td>Karim</td><td>21</td></tr>
    <tr><td>HAMIM</td><td>Mohamed</td><td>24</td></tr>
</table>
```

Nom	Prénom	Age
MANSOURI	Hassan	23
SAFIR	Laila	22
BICHRI	Karim	21
HAMIM	Mohamed	24

Le code CSS suivant signifie que tous les éléments qui appartiennent à la classe "impaire" auront un arrière-plan jaune claire.

```
.impaire
{
    background-color : rgba(255, 255, 0, 0.2);
}
```


Mme Imane FRITET imane.fritet@ofppt-edu.ma Secteur : Digital et IA

3. Sélecteur d'ID:

Le sélecteur d'ID cible un élément spécifique avec un ID unique défini dans l'attribut id.

Dans un document, il ne doit y avoir qu'un seul élément pour un identifiant donné.

```
#mon-element {
  /* Styles pour l'élément avec l'ID "mon-element" */
}
```

3. Sélecteur d'ID:

Exemple :

```
<table border="1">
    <tr><th>Nom</th><th>Prénom</th><th>Age</th></tr>
    <tr><td>MANSOURI</td><td>Hassan</td><td>23</td></tr>
    <tr><td>SAFIR</td><td>Laila</td><td>22</td></tr>
    <tr id="sel"><td>BICHRI</td><td>Karim</td><td>21</td></tr>
    <tr><td>HAMIM</td><td>Mohamed</td><td>24</td></tr>
</table>
```

Le code CSS suivant signifie que l'élément de id="sel" aura un arrière plan jaune.

```
#sel
{
    background-color : rgba(255, 255, 0, 0.2);
}
```

Nom	Prénom	Age
MANSOURI	Hassan	23
SAFIR	Laila	22
BICHRI	Karim	21
HAMIM	mohamed	24

Mme Imane FRITET imane.fritet@ofppt-edu.ma Secteur : Digital et IA

01 - Introduire le CSS

1. Présentation du CSS
2. Codification des couleurs
3. Unités de mesure
4. Positions (center, left, right)
5. Fonts
6. Types d'intégration du CSS
7. Sélecteurs simples (element, class, id)
8. Sélecteurs complexes
9. Pseudo classes

Mme Imane FRITET imane.fritet@ofppt-edu.ma Secteur : Digital et IA

1. Sélecteur universel :

Le sélecteur universel cible tous les éléments HTML de la page.

```
* {
  /* Styles pour tous les éléments de la page */
}
```

```
*
{
    color : brown;
}
```

2. Sélecteur d'attribut :

Le sélecteur d'attribut en CSS permet de cibler des éléments HTML en fonction de leurs attributs. Il est utilisé pour appliquer des styles à des éléments qui possèdent un attribut spécifique, ou à des éléments dont la valeur d'un attribut correspond à une valeur donnée.

Sélection d'éléments avec un attribut spécifique :

```
[attribut] {
  /* Styles appliqués à tous les éléments avec l'attribut "attribut" */
}
```

Exemple: Styles appliqués à tous les éléments ayant un attribut « href »

```
[href] {
  color: blue;
  text-decoration: underline;
}
```


Mme Imane FRITET imane.fritet@ofppt-edu.ma Secteur : Digital et IA

Sélection d'éléments avec un attribut et une valeur spécifiques :

```
[attribut="valeur"] {
  /* Styles appliqués à tous les éléments ayant l'attribut "attribut" avec une valeur égale à "valeur" */
}
```

Exemple: Styles appliqués à tous les boutons ayant un attribut "type" avec la valeur "button"

```
[type="button"] {
  color : red;
}
```

Sélection d'éléments dont la valeur de l'attribut commence par une chaîne donnée :

```
[attribut^="début"] {
  /* Styles appliqués à tous les éléments ayant un attribut "attribut" dont la valeur commence par "début" */
}
```

Exemple: Styles appliqués à tous les éléments ayant un attribut "src" dont la valeur commence par "https"

```
[src^="https"] {
  background-color: #f44336;
  color: white;
  text-align: center;
  text-decoration: none;
}
```

Sélection d'éléments dont la valeur de l'attribut se termine par une chaîne donnée :

```
[attribut$="fin"] {
  /* Styles appliqués à tous les éléments ayant un attribut
 "attribut" dont la valeur se termine par "fin" */
}
```

Exemple: Styles appliqués à tous les éléments ayant un attribut "src" dont la valeur se termine par ".jpg"

```
[src$=".jpg"] {
  width: 150px;
  height: auto;
}
```

Sélection d'éléments dont la valeur de l'attribut contient une chaîne donnée :

```
[attribut*="contient"] {
  /* Styles appliqués à tous les éléments ayant un attribut
"attribut" dont la valeur contient "contient" */
}
```

Exemple: Styles appliqués à tous les éléments ayant un attribut "title" dont la valeur contient "info

```
[title*="info"] {
  color: blue;
  text-align: center;
}
```

Sélection d'éléments dont la valeur de l'attribut est égale à l'une des valeurs spécifiées :

```
[attribut~="valeur"] {
  /* Styles appliqués à tous les éléments ayant un attribut "attribut" avec une valeur qui inclut "valeur" en tant que mot clé (séparé par des espaces) */
}
```

Exemple: Styles appliqués à tous les éléments ayant un attribut "class" avec la valeur "important" parmi d'autres classes.

```
[class~="important"] {
  background-color: #f44336;
  color: white;
  text-align: center;
  font-size : 100px;
}
```

Combinateur de voisin direct "+":

Le combinateur de voisin direct +, permet de cibler un élément qui est immédiatement suivi par un autre élément du même type.

```
element1 + element2 {
  /* Styles appliqués à l'élément2 qui suit immédiatement l'élément1 */
}
```

Exemple 1 : Cibler un paragraphe immédiatement suivi d'un titre h2 :

```
<h2>Titre 1</h2>
<p>Paragraphe 1</p>
<h2>Titre 2</h2>
<p>Paragraphe 2</p>
```

```
h2 + p {
  font-weight: bold;
  color: red;
}
```

Titre 1

Paragraphe 1

Titre 2

Paragraphe 2

Dans cet exemple, le sélecteur h2 + p cible les paragraphes (<p>) qui sont immédiatement suivis d'un titre de niveau 2 (<h2>) et leur applique un style de texte en gras et une couleur rouge.

Mme Imane FRITET imane.fritet@ofppt-edu.ma Secteur : Digital et IA

Combinateur de voisin direct "+":

Exemple 2 : Cibler une liste non ordonnée immédiatement suivie d'une liste ordonnée

```
<ul>
  <li>Élément 1</li>
  <li>Élément 2</li>
</ul>
<ol>
  <li>Élément 1</li>
  <li>Élément 2</li>
</ol>
```

```
ul + ol {
  color: red;
}
```

Dans cet exemple, le sélecteur ul + ol cible la liste ordonnée (<ol>) qui suit immédiatement une liste non ordonnée (<ul>) et lui applique une couleur de texte rouge.

- Élément 1
- Élément 2

1. Élément 1
2. Élément 2

Sélecteurs de voisins généraux "~":

Les sélecteurs de voisins généraux (tilde, ~) permettent de cibler tous les éléments qui sont du même type et qui suivent un élément spécifique, sans être nécessairement adjacents. Ils sont utiles lorsque vous souhaitez **appliquer des styles à plusieurs éléments similaires qui suivent un élément de référence**.

```
element1 ~ element2 {
  /* Styles appliqués à tous les éléments2
de même type qui suivent élément1 */
}
```

Sélecteurs de voisins généraux "~":

Exemple 1 : Cibler tous les paragraphes suivant un titre h2.

```
<h2>Titre 1</h2>
<p>Paragraphe 1</p>
<p>Paragraphe 2</p>
<h2>Titre 2</h2>
<p>Paragraphe 3</p>
<p>Paragraphe 4</p>
```

```
h2 ~ p {
  font-weight: bold;
  color: red;
}
```

Dans cet exemple, le sélecteur h2 ~ p cible tous les paragraphes (<p>) qui suivent un titre de niveau 2 (<h2>) et leur applique un style de texte en gras et une couleur rouge.
Les paragraphes suivant les deux titres h2 sont affectés.

Mme Imane FRITET imane.fritet@ofppt-edu.ma Secteur : Digital et IA

Sélecteurs de voisins généraux "~":

Exemple 2 : Cibler toutes les cellules <td> d'une table qui suivent une cellule spécifique.

```
<table>
  <tr>
    <td>Cellule 1</td>
    <td>Cellule 2</td>
    <td>Cellule 3</td>
  </tr>
</table>
```

```
td:first-child ~ td {
  background-color: lightgray;
}
```

Cellule 1 Cellule 2 Cellule 3

Dans cet exemple, le sélecteur td:first-child ~ td cible toutes les cellules (<td>) qui suivent la première cellule dans la première ligne d'une table et leur applique une couleur de fond gris clair.

Sélecteurs enfant ">":

Les sélecteurs d'enfant (>) permettent de cibler les éléments enfants directs d'un élément parent spécifique. Contrairement au sélecteur général (~), le sélecteur d'enfant cible uniquement les éléments qui sont des enfants directs de l'élément parent, sans tenir compte des éléments descendants plus profonds.

```
parent > enfant {
  /* Styles appliqués à l'enfant qui est un enfant direct du parent */
}
```

Sélecteurs enfant ">":

**Exemple 1 : Cibler les paragraphes (<p>) qui sont des enfants directs d'une **

```
<div>
  <p>Paragraphe 1</p>
  <p>Paragraphe 2</p>
</div>
<p>Paragraphe 3 (pas un enfant direct de la div)</p>
```

```
div > p {
  color: blue;
}
```

Paragraphe 1

Paragraphe 2

Paragraphe 3 (pas un enfant direct de la div)

Dans cet exemple, le sélecteur div > p cible uniquement les paragraphes (<p>) qui sont des enfants directs de la , et leur applique une couleur de texte bleue.
Le paragraphe qui n'est pas un enfant direct de la div n'est pas affecté.

Sélecteurs enfant ">":

**Exemple 2 : Cibler les éléments <li> qui sont des enfants directs d'une liste non ordonnée **

```
<ul>
  <li>Élément 1</li>
  <li>Élément 2</li>
</ul>
<li>Élément 3 (pas un enfant direct de la liste)</li>
```

```
ul > li {
  font-weight: bold;
  color: green;
}
```

- **Élément 1**
- **Élément 2**

- Élément 3 (pas un enfant direct de la liste)

Dans cet exemple, le sélecteur ul > li cible uniquement les éléments de liste (<li>) qui sont des enfants directs de la liste non ordonnée (<ul>) et leur applique un style de texte en gras et une couleur de texte verte.
L'élément de liste qui n'est pas un enfant direct de la liste n'est pas stylisé.

Mme Imane FRITET imane.fritet@ofppt-edu.ma Secteur : Digital et IA

Sélecteurs enfant ">":

**Exemple 3 : Cibler les éléments qui sont des enfants directs de la division **

```
<span>Ici, est en rouge.</span>
<p>Voici un paragraphe.</p>
<code>Un peu de code.</code>
<div>
  <span>Et un autre span bleu.</span>
  <code>Encore du code</code>
  <span>Ici aussi, c'est bleu</span>
</div>
```

```
span {
  color : red;
}
div > span{
  color : blue;
}
```

Dans cet exemple, la première règle de style cible tous les éléments et définit leur couleur de texte en rouge.

La deuxième règle de style utilise le sélecteur d'enfant (div > span) pour cibler uniquement les éléments qui sont des enfants directs de la balise . Ces éléments sont donc colorés en bleu.

Ici, est en rouge.

Voici un paragraphe.

Un peu de code.
Et un autre span bleu. Encore du code Ici aussi, c'est bleu

PARTIE 1

Sélecteurs descendant " " (espace) :

Les sélecteurs descendants, indiqués par un espace (``), permettent de cibler des éléments HTML qui sont des descendants d'un autre élément, sans nécessairement être des enfants directs. Ces sélecteurs sont utilisés pour appliquer des styles à des éléments qui sont imbriqués dans une structure HTML.

```
element1 element2 {
  /* Styles appliqués à tous les éléments2
qui sont des descendants d'élément1 */
}
```

Sélecteurs descendant " " (espace) :

Exemple 1 : Cibler tous les éléments <p> qui sont des descendants de l'élément ayant la classe "conteneur".

```html
<div class="conteneur">
  <p>Paragraphe 1 dans un conteneur</p>
  <ul>
    <li>Élément 1</li>
    <li>Élément 2</li>
  </ul>
  <p>Paragraphe 2 dans un conteneur</p>
</div>
<p>paragraphe 3</p>
```

```css
.conteneur p {
  color: blue;
}
```

Paragraphe 1 dans un conteneur

- Élément 1
- Élément 2

Paragraphe 2 dans un conteneur

paragraphe 3

Dans cet exemple, le sélecteur descendant **.conteneur p** cible tous les éléments <p> qui sont des descendants de l'élément ayant la classe "conteneur" ( ). Ces éléments <p> deviennent bleus en conséquence.
Le paragraphe qui n'est pas un sélecteur descendant **.conteneur p** n'est pas affecté.

Mme Imane FRITET imane.fritet@ofppt-edu.ma Secteur : Digital et IA

Sélecteurs descendant " " (espace) :

Exemple 2 : Cibler tous les éléments <p> qui sont des descendants de l'élément ayant la classe "conteneur".

```html
<ul>
  <li>
    <div>Élément 1</div>
    <ul>
      <li>Sous-élément A</li>
      <li>Sous-élément B</li>
    </ul>
  </li>
  <li>
    <div>Élément 2</div>
    <ul>
      <li>Sous-élément A</li>
      <li>Sous-élément B</li>
    </ul>
  </li>
</ul>
```

```css
li {
  list-style-type : disc;
}
li li {
  list-style-type : circle;
}
```

- Élément 1
  - Sous-élément A
  - Sous-élément B
- Élément 2
  - Sous-élément A
  - Sous-élément B

Dans cet exemple, le sélecteur simple **li** cible tous les éléments <li> et définit leur type de liste (indiqué par la propriété list-style-type) comme "disc".
Le sélecteur descendant **li li** cible spécifiquement les éléments <li> qui sont des descendants directs d'autres éléments <li>, c'est-à-dire les éléments de la sous-liste. Elle modifie leur type de liste pour "circle".

Mme Imane FRITET imane.fritet@ofppt-edu.ma Secteur : Digital et IA

01 - Introduire le CSS

1. Présentation du CSS
2. Codification des couleurs
3. Unités de mesure
4. Positions (center, left, right)
5. Fonts
6. Types d'intégration du CSS
7. Sélecteurs simples (element, class, id)
8. Sélecteurs complexes
9. Pseudo classes

Mme Imane FRITET imane.fritet@ofppt-edu.ma Secteur : Digital et IA

Les pseudo-classes en CSS sont des sélecteurs spéciaux qui vous permettent de définir un état spécial d'un élément.

Elles sont utilisées pour appliquer des styles lorsque les éléments répondent à des interactions telles que le survol, le focus, etc.

Syntaxe

```
selector:pseudo-class {
  property: value;
}
```

Pseudo-Classes les plus courantes

:hover est utilisée pour sélectionner un élément lorsque l'utilisateur survole cet élément avec la souris.

```
a:hover {
  color: red;
}
```

:active est utilisée pour sélectionner un élément lorsque l'utilisateur clique dessus.

```
button:active {
  background-color: green;
}
```

:focus est utilisée pour sélectionner un élément lorsqu'il a le focus, généralement après avoir été cliqué ou après avoir été atteint à l'aide de la tabulation.

```
input:focus {
  border: 4px solid red;
}
```

```
<input type="text">
```

:first-child et :last-child sont utilisées pour sélectionner le premier et le dernier enfant d'un élément parent respectivement.

```
p:first-child {
  color: red;
}
p:last-child {
  color: blue;
}
```

```
<div>
  <p>This is some text.</p>
  <p>This is some text.</p>
  <p>This is some text.</p>
</div>
```

This is some text.

This is some text.

This is some text.

:nth-child(n) est utilisée pour sélectionner les éléments basés sur leur position parmi les enfants d'un élément parent. Vous pouvez spécifier un nombre n pour cibler un enfant particulier.

```
<ul>
  <li>First list item</li>
  <li>Second list item</li>
  <li>Third list item</li>
  <li>Fourth list item</li>
  <li>Fifth list item</li>
</ul>
```

```
li:nth-child(4) {
  background: lightgreen;
}
```

- First list item
- Second list item
- Third list item
- Fourth list item
- Fifth list item

```
li:nth-child(odd) {
  background-color: #f0f0f0;
}
li:nth-child(even) {
  background-color: #e0e0e0;
}
```

- First list item
- Second list item
- Third list item
- Fourth list item
- Fifth list item

Mme Imane FRITET imane.fritet@ofppt-edu.ma Secteur : Digital et IA

Pseudo-Classes Spécifiques

Certaines pseudo-classes sont spécifiques à certains éléments HTML, comme les liens (**:link, :visited**), les formulaires (**:checked, :required**), etc. Elles sont utilisées pour cibler ces éléments spécifiques dans certaines conditions.

:link et :visited sont utilisées pour cibler les liens non visités et les liens visités respectivement.

```
<a href="https://www.google.com/">Google</a>
<a href="https://cmc.ac.ma/">CMC</a>
```

Google

CMC

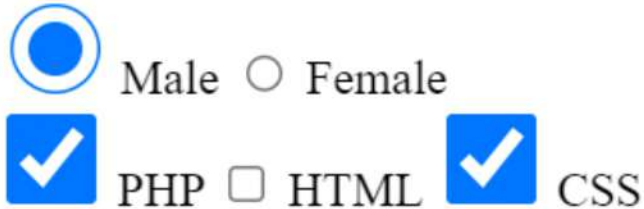
```
a:link {
  color: green;
}

a:visited {
  color: red;
}
```

Pseudo-Classes Spécifiques

:checked est utilisée pour cibler les éléments de formulaire tels que les cases à cocher et les boutons radio lorsqu'ils sont cochés.

```
<input type="radio" value="male" name="gender" checked> Male
<input type="radio" value="female" name="gender"> Female<br>
<input type="checkbox" id="php" value="php" checked>
<label for="php">PHP</label>
<input type="checkbox" id="html" value="html">
<label for="html">HTML</label>
<input type="checkbox" id="css" value="css" checked>
<label for="css">CSS</label>
```

```
input:checked {
  height: 30px;
  width: 30px;
}
```



Pseudo-Classes Spécifiques

:required est utilisée pour cibler les éléments obligatoires de formulaire qui ont l'attribut required.

```
Last-name: <input type="text" required>
First-name: <input type="text">
```

```
input:required {
  background-color: lightgreen;
  border: 2px solid green;
}
```


```
input:required {
  background-color: lightgreen;
  border: 2px solid green;
}
```

Pseudo-Classes Spécifiques

::before est utilisé pour insérer du contenu avant le contenu de l'élément ciblé. Il est généré automatiquement par le navigateur et peut être utilisé pour ajouter des éléments visuels décoratifs.

::after est utilisé pour insérer du contenu après le contenu de l'élément ciblé.

```
<p>Lorem ipsum dolor sit amet,
  ut labore et dolore magna aliqua.</p>
```

Avant text: Lorem ipsum dolor sit amet, ut labore et dolore magna aliqua. Apres text.

```
p::before {
  content: "Avant text: ";
  font-weight: bold;
  color: #990011;
}
p::after {
  content: " Apres text.";
  font-weight: bold;
  color: #110099;
}
```

Sélecteur	Exemple	Description
: checked	input :checked	Sélectionne tout élément <input> coché du formulaire
: disabled	input :disabled	Sélectionne chaque élément <input> désactivé
: empty	p :empty	Sélectionne chaque élément <p> qui n'a pas d'enfant
: enabled	input :enabled	Sélectionne chaque élément <input> activé
: first-of-type	p :first-of-type	Sélectionne chaque élément <p> qui est le premier élément <p> de son parent
: in-range	input :in-range	Sélectionne les éléments <input> avec une valeur située dans une plage spécifiée
: invalide	input :invalid	Sélectionne tous les éléments <input> avec une valeur invalide
: last-child	p :last-child	Sélectionne tous les éléments <p> qui sont les derniers enfants de leurs parents
: last-of-type	p :last-of-type	Sélectionne chaque élément <p> qui est le dernier élément <p> de son parent
: not(sélecteur)	:not(p)	Sélectionne tous les éléments qui ne sont pas des paragraphes
: nth-child (n)	p :nth-child(2)	Sélectionne chaque élément <p> qui est le deuxième enfant de son parent
: nth-last-child (n)	p :nth-last-child(2)	Sélectionne chaque élément <p> qui est le deuxième enfant de son parent, à partir du dernier enfant
: nth-last-of-type(n)	p :nth-last-of-type(2)	Sélectionne chaque élément <p> qui est le deuxième élément <p> de son parent, à partir du dernier enfant
: nth-of-type (n)	p :nth-of-type(2)	Sélectionne chaque élément <p> qui est le deuxième élément <p> de son parent
: only-of-type	p :only-of-type	Sélectionne chaque élément <p> qui est le seul élément <p> de son parent
: only-child	p :only-child	Sélectionne chaque élément <p> qui est le seul enfant de son parent
: optional	input :optional	Sélectionne les éléments <input> sans attribut "required"
: out-of-range	input :out-of-range	Sélectionne les éléments <input> qui ont des valeurs en dehors d'une plage spécifiée

Sélecteur	Exemple	Description
:read-only	input :read-only	Sélectionne les éléments <input> avec un attribut "readonly" spécifié
:read-write	input :read-write	Sélectionne les éléments <input> sans attribut "readonly"
:required	input :required	Sélectionne les éléments <input> avec un attribut "required" spécifié
root	root	Sélectionnez l'élément racine du document
:target	#news :target	Sélectionne l'élément #news actuellement actif
:valid	input :valid	Sélectionne tous les éléments <input> avec une valeur valide
:link	a :link	Sélectionne tous les liens non visités
:visited	a :visited	Sélectionnez tous les liens visités
:active	a :active	Sélectionne le lien actif
:hover	a :hover	Sélectionne les liens sur lesquels la souris passe
:focus	input :focus	électionne l'élément <input> qui a le focus
:first-letter	p :first-letter	Sélectionne la première lettre de chaque élément <p>
:first-line	p :first-line	Sélectionne la première ligne de chaque élément <p>
:first-child	p :first-child	Sélectionne tous les éléments <p> qui sont les premiers enfants de leurs parents
:before	p :before	Insérer du contenu avant chaque élément <p>
:after	p :after	Insérer du contenu après chaque élément <p>
:lang(it)	p :lang(it)	Sélectionne chaque élément <p> avec une valeur d'attribut lang commençant par "it"