



Module:

CSS

**Département Informatique, Ecole Supérieur de Technologie -Guelmim-
Université Ibn Zohr**

Année Universitaire: 2022/2023



Plan de cours

II. CSS: *Cascading Style Sheets*

- 1) Les bases de CSS.**
- 2) Les Propriétés de Type « Font- ».**
- 3) Les Propriétés de Type « Text- ».**
- 4) Le Modèle des Boîtes.**

Les bases de CSS: Définition

- CSS est l'abréviation de **Cascading Style Sheets, Feuilles de style en cascade** en français.
- CSS est utilisé pour appliquer des styles à un contenu, c'est-à-dire le mettre en forme.
- Avec CSS, on pourra changer la couleur ou la taille d'un texte, positionner tel contenu à tel endroit de notre page web ou ajouter des bordures ou des ombres autour d'un contenu.

Les bases de CSS: Sélecteurs, propriétés et valeurs

- Pour rappel, le CSS sert à modifier l'apparence de nos pages web en appliquant des styles au contenu en HTML.
- **Un sélecteur**, tout d'abord, va servir à déterminer à quel(s) élément(s) HTML ou à quel type d'éléments on souhaite appliquer un style particulier. Si l'on souhaite appliquer un style particulier à tous nos paragraphes, par exemple, on utilisera le sélecteur « p ».
- **Une propriété**, va nous servir à modifier le style d'un élément en ciblant un critère bien particulier comme la taille d'un texte, sa police ou sa couleur par exemple.
- **Une valeur**, enfin, va venir compléter une propriété et va en déterminer le comportement. Pour la propriété servant à changer la couleur d'un texte par exemple, la valeur va être la nouvelle couleur à appliquer.

Les bases de CSS: Sélecteurs, propriétés et valeurs

- Voici ci-dessous une illustration concrète de ce que l'on vient de dire :
 - **Sélecteur**: Dans cet exemple, nous utilisons le sélecteur simple « **p** », ce qui signifie que nous souhaitons appliquer un style particulier à tous les paragraphes de nos pages.
 - **Propriété**: Nous utilisons les propriétés « **color** » et « **font-size** ». Cela signifie donc que nous travaillerons sur la couleur et la taille de nos paragraphes.
 - **Valeur**: Nous indiquons que nous voulons que tous nos paragraphes s'affichent en bleu grâce à la valeur « **blue** » et que notre texte ait une taille de 16px avec la valeur « **16px** ».

Sélecteur

propriété

```
premier_pas_CSS.css
1 p{
2   color: blue;
3   font-size: 16px;
4 }
```

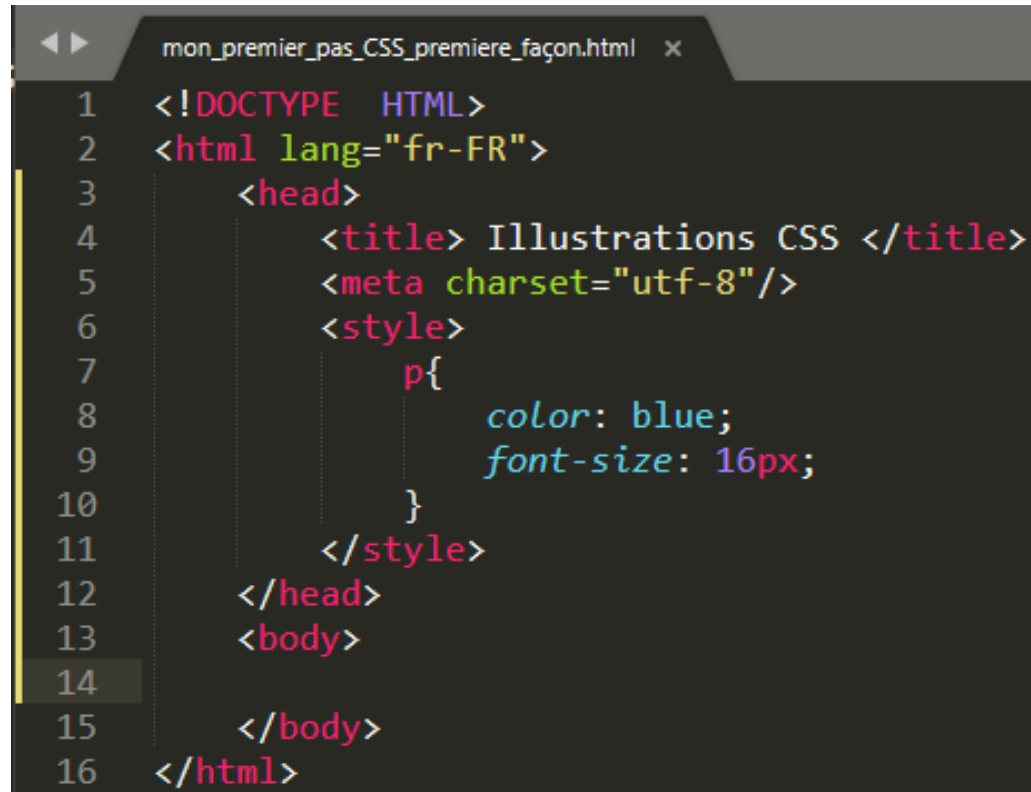
Valeur

Les bases de CSS: Où écrire le CSS ?

- Nous avons trois possibilités pour écrire notre CSS. L'une d'elles est préférable aux deux autres et nous allons immédiatement voir pourquoi.
- Nous pouvons écrire le CSS :
 - A l'intérieur de l'élément **head** de notre document HTML ;
 - Dans la balise ouvrante des éléments de notre fichier HTML ;
 - Dans un fichier portant l'extension **.css** séparé.
- Pour des raisons de performances du code, de clarté et d'économie de temps, je vous recommande vivement d'utiliser la dernière méthode dès que cela est possible.

Les bases de CSS: Où écrire le CSS ?

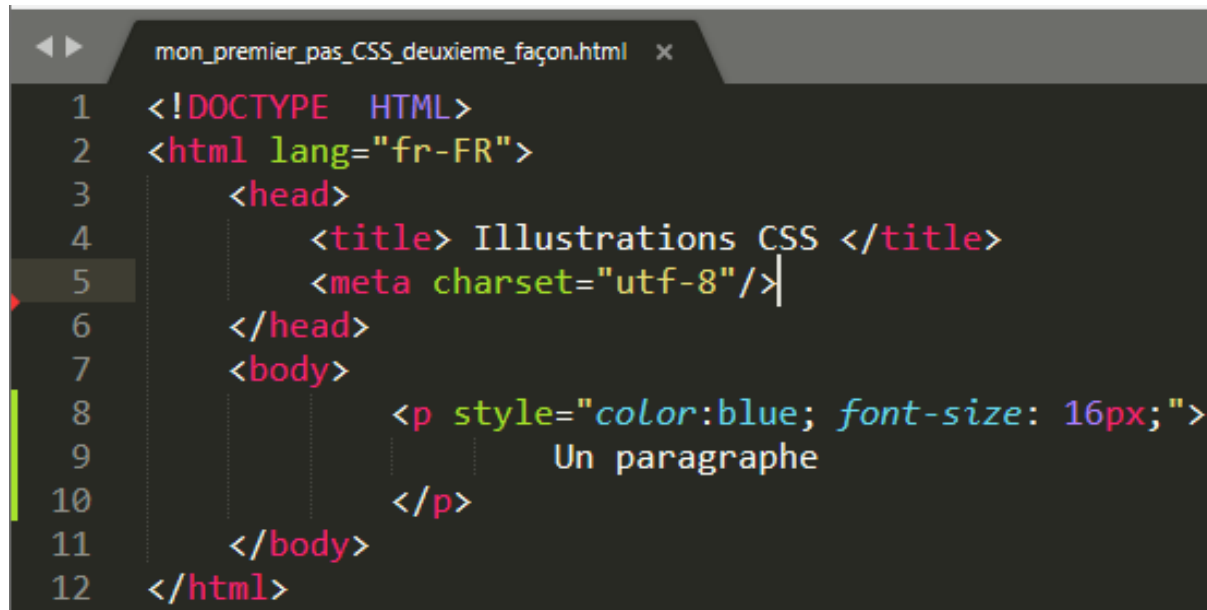
- **Première façon:** A l'intérieur de l'élément **head** de notre document HTML
 - Il suffit d'insérer un élément **style** dans notre élément **head** et de placer nos déclarations CSS à l'intérieur de cet élément **style** comme ceci :



```
mon_premier_pas_CSS_premiere_façon.html x
1  <!DOCTYPE HTML>
2  <html lang="fr-FR">
3      <head>
4          <title> Illustrations CSS </title>
5          <meta charset="utf-8"/>
6          <style>
7              p{
8                  color: blue;
9                  font-size: 16px;
10             }
11          </style>
12      </head>
13      <body>
14
15      </body>
16  </html>
```

Les bases de CSS: Où écrire le CSS ?

- **Deuxième façon:** Dans la balise ouvrante des éléments de notre fichier HTML
 - Nous allons devoir utiliser un attribut style et lui affecter en valeur nos propriétés CSS.
 - On doit respecter la syntaxe du CSS à l'intérieur de l'attribut style en utilisant les deux-points et les points virgules.



```
1 <!DOCTYPE HTML>
2 <html lang="fr-FR">
3   <head>
4     <title> Illustrations CSS </title>
5     <meta charset="utf-8"/>
6   </head>
7   <body>
8     <p style="color:blue; font-size: 16px;">
9       Un paragraphe
10    </p>
11  </body>
12 </html>
```


Les bases de CSS: Où écrire le CSS ?

■ Troisième façon: Dans un fichier portant l'extension .css séparé

- Nous allons devoir créer un fichier au format ".css" et enregistrer ce fichier dans le même dossier que votre fichier HTML dont vous souhaitez modifier le style.
- Nous allons maintenant devoir lier nos deux fichiers HTML et CSS. On va faire cela à l'aide d'un élément « **link** » que nous allons placer dans l'élément « **head** » de cette manière:

```
premier_pas_CSS.css
1  p{
2      color: blue;
3      font-size: 16px;
4  }
```

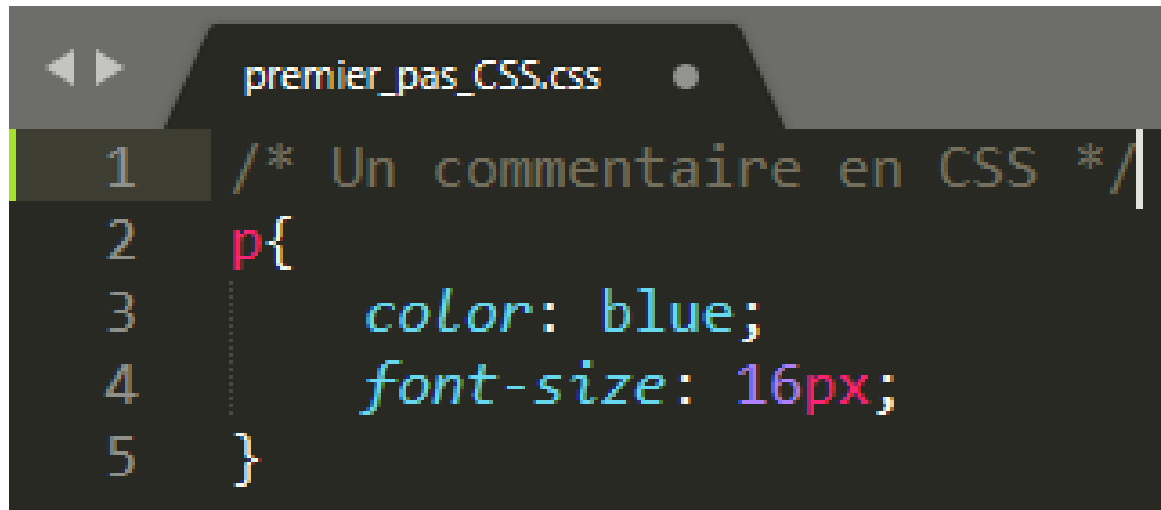
```
mon_premier_pas_CSS_troisieme_façon.html x
1  <!DOCTYPE HTML>
2  <html lang="fr-FR">
3      <head>
4          <title> Illustrations CSS </title>
5          <meta charset="utf-8"/>
6          <link rel="stylesheet" href="premier_pas_CSS.css"/>
7      </head>
8      <body>
9          <p> Un paragraphe </p>
10     </body>
11 </html>
```

Les bases de CSS: Où écrire le CSS ?

- **Troisième façon:** Dans un fichier portant l'extension **.css** séparé
 - L'élément « **link** » est représenté sous forme d'une balise orpheline et doit être accompagné de ses deux attributs « **rel** » et « **href** »
 - L'attribut « **rel** » sert à préciser le style du fichier lié. L'attribut « **href** » sert à faire le lien en soi.
 - Si le fichier avait été placé dans un dossier parent ou dans un sous-dossier par rapport à notre fichier HTML, il aurait fallu refléter cela dans la valeur de notre attribut « **href** ».

Les bases de CSS: Les commentaires en CSS

- Les fichiers CSS deviennent rapidement très long, donc si vous ne commentez pas efficacement vous risquez d'être très vite perdu.
- En CSS, on écrit les commentaires de cette manière:

A screenshot of a code editor window titled 'premier_pas_CSS.css'. The editor has a dark background with light-colored text. Line numbers 1 through 5 are visible on the left. The code is as follows:

```
1  /* Un commentaire en CSS */  
2  p{  
3      color: blue;  
4      font-size: 16px;  
5  }
```

Les bases de CSS: Sélecteurs simples et limitations

- Nous n'avons manipulé que des sélecteurs que l'on appelle « **simple** », car ils correspondent à des éléments HTML seuls et sans attributs (par exemple le sélecteur « **p** »).
- Le problème reste qu'on est quand même très limité avec des sélecteurs simples : comment faire pour appliquer un style différent à deux éléments de même type, deux paragraphe par exemple ?
- Pour cela que l'on a créé les attributs « **class** » et « **id** ».
- **Class** et **Id** sont deux attributs HTML qui ont été créés pour pouvoir appliquer différents styles à des éléments de même type.
- **Class** permet également de faire l'inverse et d'appliquer le même style à différents éléments choisis.

Les bases de CSS: Sélecteurs simples et limitations

- Premièrement on se place dans la balise ouvrante d'un élément HTML, on écrit le nom de notre attribut (class ou id), et on lui donne une valeur cohérente.
- Cette valeur ne devrait contenir ni de caractères spéciaux (accents et autres) ni d'espace.

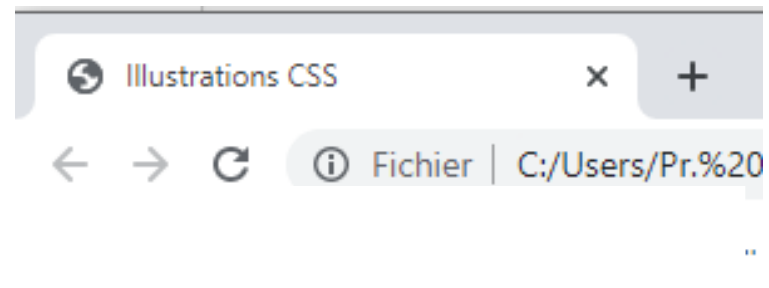
```
mon_premier_pas_CSS_class_id.html x
1  <!DOCTYPE HTML>
2  <html lang="fr-FR">
3      <head>
4          <title> Illustrations CSS </title>
5          <meta charset="utf-8"/>
6          <link rel="stylesheet" href="mon_premier_pas_CSS_class_id.css"/>
7      </head>
8      <body>
9          <p class="style_1"> Un paragraphe avec style 1 </p>
10         <p id="style_2">    Un paragraphe avec style 2</p>
11     </body>
12 </html>
```

Les bases de CSS: Sélecteurs simples et limitations

■ On retourne sur le fichier CSS:

- On va devoir commencer notre déclaration par un point (.) là où on a utilisé un attribut class.
- On va devoir commencer notre déclaration par un dièse (#) si l'on a utilisé l'attribut id.
- Après le point ou le dièse, on écrit la valeur de l'attribut en question pour former notre sélecteur.

```
mon_premier_pas_CSS_class_id.css x
1  /* Commentaire En CSS */
2  .style_1{
3      color: blue;
4  }
5
6  #style_2{
7      color: red;
8  }
```



Les bases de CSS: Sélecteurs simples et limitations

- Nous pouvons maintenant appliquer un style différent à chaque élément HTML grâce aux attributs **class** et **id**.
- Pourquoi avoir créé deux attributs pour faire la même chose ? En fait, il existe une différence notable entre **class** et **id** :
 - Un attribut **id** avec une valeur précise ne peut être utilisé qu'une fois dans une page.
 - Au contraire, l'attribut **class**, on peut utiliser plusieurs attributs **class** identiques.
 - Utilisation de l'attribut **class** est une des méthodes que nous utiliserons pour appliquer un même style à différents éléments HTML.

Les bases de CSS: Sélecteurs simples et limitations

- Nous pouvons maintenant appliquer un style différent à chaque élément HTML grâce aux attributs **class** et **id**.



The image shows a code editor on the left and a web browser on the right. The code editor displays an HTML file named 'mon_premier_pas_CSS_class_id_2.html' with the following content:

```
1 <!DOCTYPE HTML>
2 <html lang="fr-FR">
3   <head>
4     <title> Illustrations CSS </title>
5     <meta charset="utf-8"/>
6     <link rel="stylesheet" href="mon_premier_pas_CSS_class_id.css"/>
7   </head>
8   <body>
9     <h1 class="style_1"> Un titre avec style 1 </h1>
10    <p id="style_2"> Un paragraphe avec style 2 </p>
11    <p class="style_1"> Un paragraphe avec style 1</p>
12  </body>
13 </html>
```

The web browser on the right shows the rendered output of the HTML file. The title bar indicates the page is titled 'Illustrations CSS'. The address bar shows the file path 'C:/Users/Pr.%20H...'. The rendered content is as follows:

Un titre avec style 1

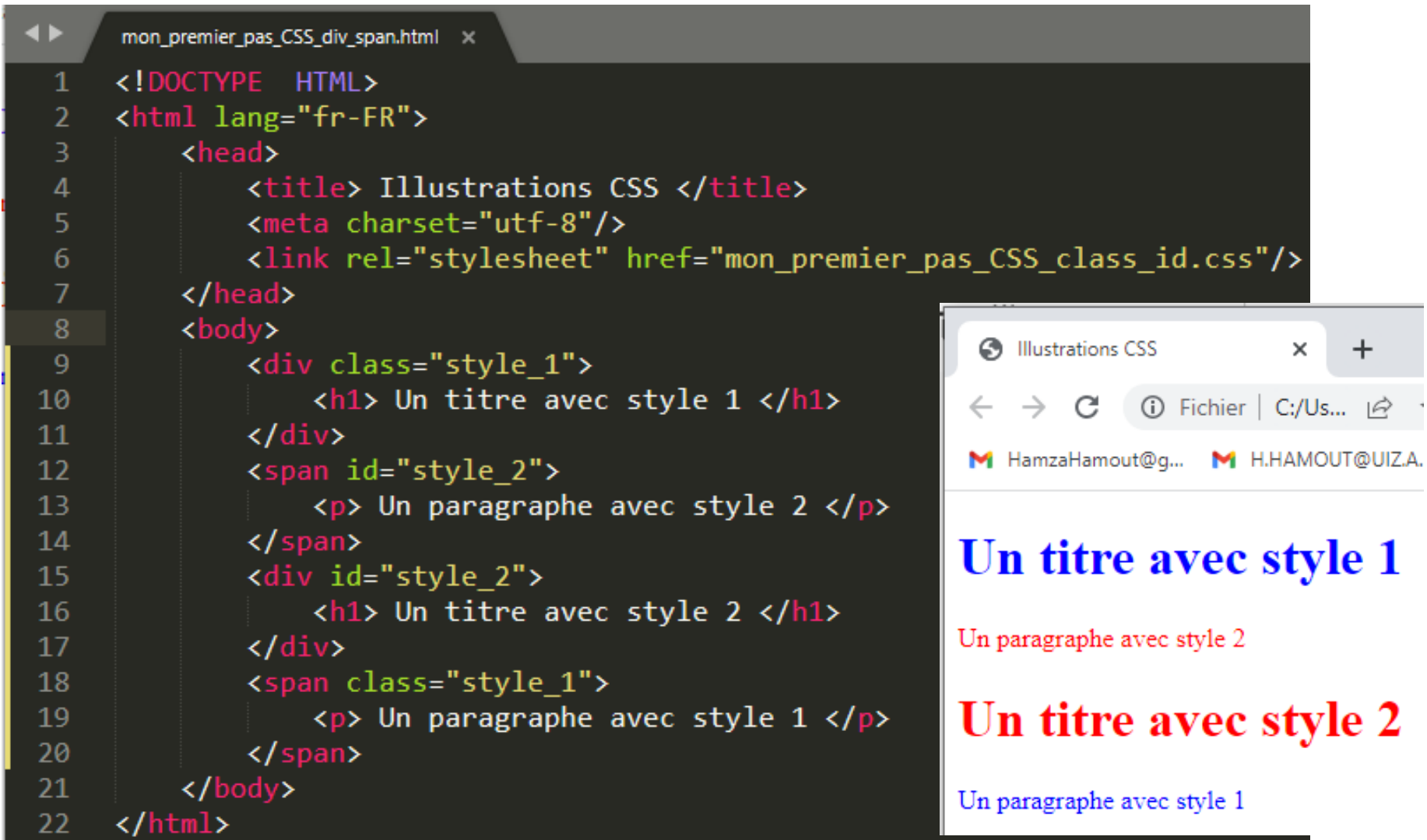
Un paragraphe avec style 2

Un paragraphe avec style 1

Les bases de CSS: Les éléments **div** et **span**

- Les éléments « **div** » et « **span** » ne possèdent aucune valeur sémantique, ils vont servir seulement comme des containers.
- Les éléments « **div** » et « **span** » sert à entourer des blocs de code et ainsi pouvoir attribuer des styles particuliers à ces blocs.
- L'utilisation des éléments « **div** » et « **span** » est très simple: il suffit d'entourer le bloc de code voulu avec une paire de balises ouvrante et fermante « **div** » ou « **span** »:
- Généralement, on attribuera une class ou un id à « **div** » et « **span** » afin de pouvoir différencier nos différents « **div** » et « **span** » dans notre page.
- Tout comme pour class et id, il existe une différence entre « **div** » et « **span** » : « **div** » est un élément de type block tandis que « **span** » est un élément de type inline.

Les bases de CSS: Les éléments div et span



The image shows a code editor on the left and a web browser on the right. The code editor displays the HTML code for a file named 'mon_premier_pas_CSS_div_span.html'. The code includes a DOCTYPE declaration, a lang attribute set to 'fr-FR', a title 'Illustrations CSS', a meta charset declaration, and a link to a CSS file. The body contains four elements: a div with class 'style_1' containing an h1 'Un titre avec style 1', a span with id 'style_2' containing a p 'Un paragraphe avec style 2', a div with id 'style_2' containing an h1 'Un titre avec style 2', and a span with class 'style_1' containing a p 'Un paragraphe avec style 1'. The web browser on the right shows the rendered output: 'Un titre avec style 1' in blue, 'Un paragraphe avec style 2' in red, 'Un titre avec style 2' in red, and 'Un paragraphe avec style 1' in blue.

```
1 <!DOCTYPE HTML>
2 <html lang="fr-FR">
3   <head>
4     <title> Illustrations CSS </title>
5     <meta charset="utf-8"/>
6     <link rel="stylesheet" href="mon_premier_pas_CSS_class_id.css"/>
7   </head>
8   <body>
9     <div class="style_1">
10       <h1> Un titre avec style 1 </h1>
11     </div>
12     <span id="style_2">
13       <p> Un paragraphe avec style 2 </p>
14     </span>
15     <div id="style_2">
16       <h1> Un titre avec style 2 </h1>
17     </div>
18     <span class="style_1">
19       <p> Un paragraphe avec style 1 </p>
20     </span>
21   </body>
22 </html>
```

Illustrations CSS

Un titre avec style 1

Un paragraphe avec style 2

Un titre avec style 2

Un paragraphe avec style 1

Les bases de CSS: Les éléments de type block et inline

- En HTML, tout élément est soit de type block, soit de type inline.
- Pour des raisons de mise en page, il est essentiel de bien comprendre les différences entre les éléments de type block et les éléments de type inline.

Les éléments de type block	Les éléments de type inline
Commencent sur une nouvelle ligne.	S'insèrent dans une ligne.
Occupent toute la largeur disponible.	Occupent seulement la largeur nécessaire.
Peuvent être imbriqués les uns dans les autres et contenir des éléments de type inline.	Peuvent être imbriqués les uns dans les autres mais ne peuvent pas contenir d'éléments de type block.

Les bases de CSS: Les éléments de type block et inline

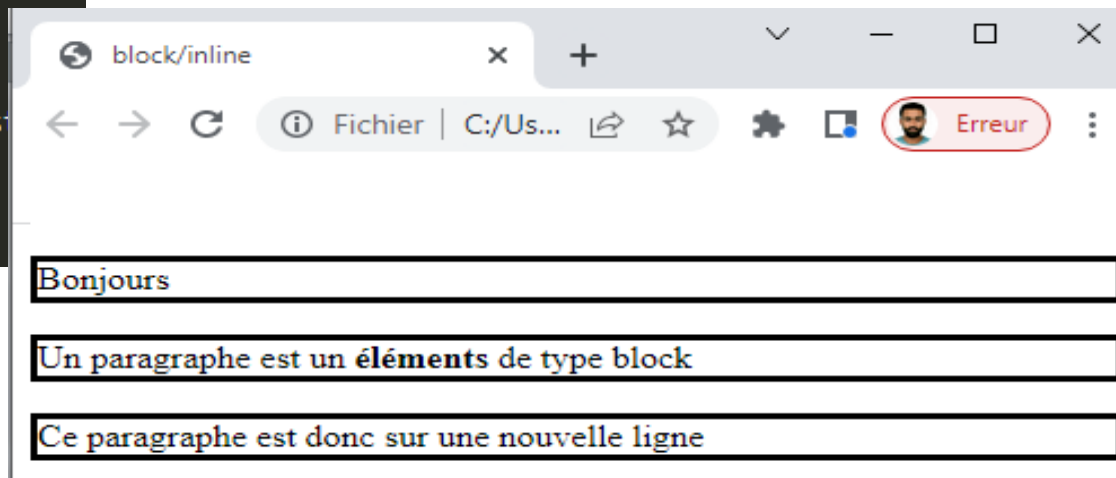
- Voyons ensemble quelques exemples d'éléments de type inline ou block pour que vous puissiez observer leur comportement.

Les éléments de type block	Les éléments de type inline
p	em
h1, h2, h3, h4, h5, h6	strong
Ol, ul, dl	mark
li	a
table	img

Les bases de CSS: Les éléments de type block et inline

- Illustrons ce que nous venons de dire en regardant par exemple de plus près le comportement des éléments `p` et `em` :

```
mon_premier_pas_CSS_block_inline.html •  
  
<head>  
  <title> block/inline </title>  
  <meta charset="utf-8"/>  
  <link rel="stylesheet" href="mon_premier_pas_CSS_class_id.css"/>  
</head>  
<body>  
  <p class="style_3"> Bonjours </p>  
  <p class="style_3"> Un paragraphe est un  
  <strong> éléments </strong> de type  
  block  
  </p>  
  <p class="style_3"> Ce paragraphe est  
  donc sur une nouvelle ligne </p>  
</body>  
</html>
```



Les bases de CSS: Les sélecteurs avancés

- Les sélecteurs avancés sont l'une des grandes forces du CSS. Grâce à eux, nous allons pouvoir cibler du contenu très précisément et assez simplement.

Ce sélecteur sert à	Il s'écrit comme cela
Sélectionneur tous les éléments: sélecteur universel.	*
Sélectionner deux éléments A et B.	A, B
Sélectionner un élément B contenu dans un élément A.	A B
Sélectionner le premier élément B suivant immédiatement un élément A.	A + B

Les bases de CSS: Les sélecteurs avancés

- Les sélecteurs avancés sont l'une des grandes forces du CSS. Grâce à eux, nous allons pouvoir cibler du contenu très précisément et assez simplement.

Ce sélecteur sert à	Il s'écrit comme cela
Sélectionner tous les éléments A possédant un attribut particulier.	A[nom de l'attribut]
Sélectionner tous les éléments A possédant un attribut particulier qui contient une « valeur » .	A[nom de l'attribut*= « valeur »]
Sélectionner tous les éléments A possédant un attribut particulier avec « valeur » précise.	A[nom de l'attribut= « valeur »]
Sélectionner tous les éléments A possédant un attribut particulier qui se termine par « valeur ».	A[nom de l'attribut\$= « valeur »]
Sélectionner tous les éléments A possédant un attribut particulier qui commence par « valeur ».	A[nom de l'attribut^= « valeur »]

Les bases de CSS: Les sélecteurs avancés

- Les sélecteurs avancés sont l'une des grandes forces du CSS. Grâce à eux, nous allons pouvoir cibler du contenu très précisément et assez simplement.

```
mon_premier_pas_CSS_selecteur_avancee.html x
1  <!DOCTYPE HTML>
2  <html lang="fr-FR">
3      <head>
4          <title> block/inline </title>
5          <meta charset="utf-8"/>
6          <link rel="stylesheet" href="
            mon_premier_pas_CSS_selecteur_avancee.css"/>
7      </head>
8      <body>
9          <h2> Bonjours </h2>
10         <h3> ESTG </h3>
11         <ul>
12             <li><a href="#internal">Lien interne</a></li>
13             <li><a href="http://example.com">Lien d'exemple</a></li>
14             <li><a href="#InSensitive">Lien interne insensible à la casse</a>
15             </li>
16             <li><a href="http://example.org">Lien vers example.org</a></li>
17         </ul>
18         <p> paragraphe contient <strong> une balise </strong> </p>
19         <p> paragraphe suivie par une titre </p>
20         <h1> Je suis le titre après un paragraphe </h1>
21     </body>
22 </html>
```


Les bases de CSS: Les sélecteurs avancés

- Les sélecteurs avancés sont l'une des grandes forces du CSS. Grâce à eux, nous allons pouvoir cibler du contenu très précisément et assez simplement.

```
mon_premier_pas_CSS_selecteur_avancee.css
1
2 *{
3     border: 0.5px dotted ;
4 }
5 a {
6     color: blue;
7 }
8 h2, h3{
9     color: blue;
10 }
11
12 a[href^="#"] {
13     background-color: gold;
14 }
15
16 a[href*="example"] {
17     background-color: silver;
18 }
19
20 a[href*="insensitive" i] {
21     color: cyan;
22 }
```

```
mon_premier_pas_CSS_selecteur_avancee.css
23
24 a[href*="cAsE" s] {
25     color: pink;
26 }
27
28 a[href$=".org"] {
29     color: red;
30 }
31
32 p strong {
33     background-color: green;
34 }
35
36 p+h1{
37     background-color: yellow;
38 }
39
```

Les bases de CSS: Les sélecteurs avancés

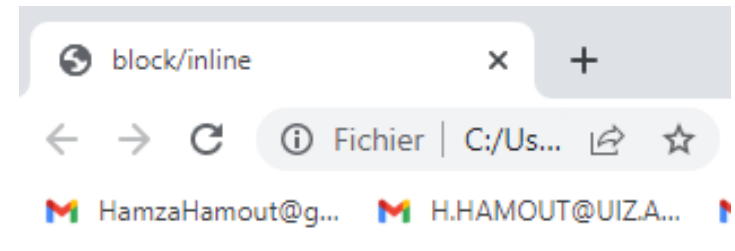
- Les sélecteurs avancés sont l'une des grandes forces du CSS. Grâce à eux, nous allons pouvoir cibler du contenu très précisément et assez simplement.



Les bases de CSS: La notion d'héritage

- L'héritage est une notion centrale et fondamentale du CSS. L'héritage signifie que tout élément HTML va hériter des styles de ses parents (c'est le fameux « cascading »).
- En HTML, si un élément A est inclus dans un élément B ; l'élément A s'appellera l'enfant et l'élément B sera le parent de l'élément A. Ainsi, si l'on applique un style à l'élément B, l'élément A en héritera automatiquement.

```
mon_premier_pas_CSS_notion_heritage.html x
1 <!DOCTYPE HTML>
2 <html lang="fr-FR">
3   <head>
4     <title> block/inline </title>
5     <meta charset="utf-8"/>
6     <link rel="stylesheet" href="
7       mon_premier_pas_CSS_class_id.css"/>
8   </head>
9   <body class="style_1">
10    <p> Bonjours </p>
11    <p> Un paragraphe est un <strong> éléments </strong>
12    de type block
13    </p>
14    <p> Ce paragraphe est donc sur une nouvelle ligne </p>
15  </body>
16 </html>
```



Les bases de CSS: La notion d'héritage

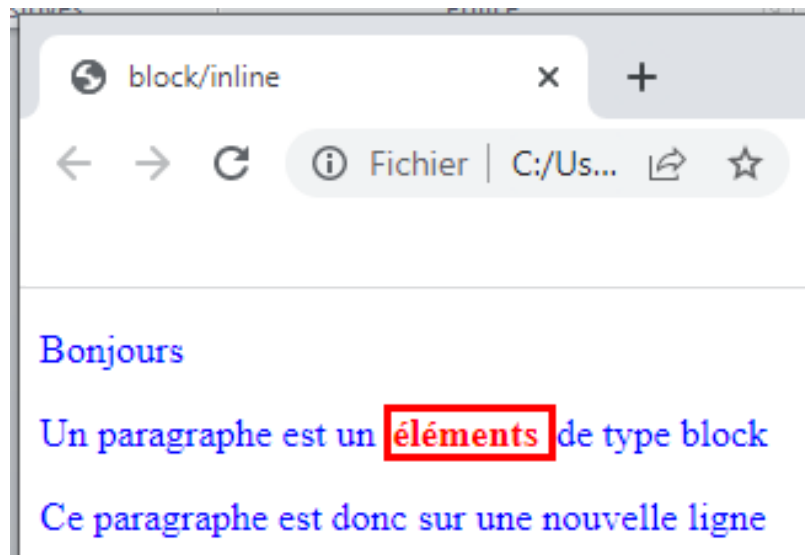
- Que se passe-t-il lorsque l'on donne deux ordres contradictoires à un élément parent et à son enfant en CSS (par exemple, donner une couleur rouge au parent et bleue à l'enfant) ?
- Le CSS possède ici sa logique et le style appliqué sera celui le plus proche de l'élément en question. Cela signifie que si on applique un style à un élément enfant, c'est bien ce style qui lui sera appliqué.

```
mon_premier_pas_CSS_notion_heritage_logique.html x mon_premier_pas_CSS_class_id.css x
1 <!DOCTYPE HTML>
2 <html lang="fr-FR">
3   <head>
4     <title> block/inline </title>
5     <meta charset="utf-8"/>
6     <link rel="stylesheet" href="
7       mon_premier_pas_CSS_class_id.css"/>
8   </head>
9   <body class="style_1">
10     <p> Bonjours </p>
11     <p> Un paragraphe est un <strong class="style_4">
12       éléments </strong> de type block
13     </p>
14     <p> Ce paragraphe est donc sur une nouvelle ligne </p>
15   </body>
16 </html>
```

```
mon_premier_pas_CSS_class_id.css x
1 /* Commentaire En CSS */
2 .style_1{
3   color: blue;
4 }
5
6 #style_2{
7   color: red;
8 }
9
10 .style_3{
11   border: solid;
12 }
13
14 .style_4{
15   border: solid;
16   color: red;
17 }
```

Les bases de CSS: La notion d'héritage

- Que se passe-t-il lorsque l'on donne deux ordres contradictoires à un élément parent et à son enfant en CSS (par exemple, donner une couleur rouge au parent et bleue à l'enfant) ?
- Le CSS possède ici sa logique et le style appliqué sera celui le plus proche de l'élément en question. Cela signifie que si on applique un style à un élément enfant, c'est bien ce style qui lui sera appliqué.



Les Propriétés de Type « Font- »: font-size

- Nous utiliserons la propriété « font-size » lorsque nous voudrions modifier la taille d'un texte.
- Cette propriété accepte deux type de valeurs: valeurs de type absolu exprimée en « **pixel** » ou en « **point** », ou valeurs de type relatif exprimée en « **em** », en « **ex** », ou en « **pourcentage** ».
- Une valeur de type absolu: va fixer la taille d'un texte définitivement.
 - Avantage de ce type de valeur est que vous pouvez contrôler précisément le rendu de votre texte et de votre page web.

Les Propriétés de Type « Font- »: font-size

- Nous utiliserons la propriété « font-size » lorsque nous voudrions modifier la taille d'un texte.
- Cette propriété accepte deux type de valeurs: valeurs de type absolu exprimée en « **pixel** » ou en « **point** », ou valeurs de type relatif exprimée en « **em** », en « **ex** », ou en « **pourcentage** ».
- Une valeur de type relatif: permet de s'adapter relativement aux préférences de vos visiteurs, c.-à-d. le texte pourra avoir des tailles différentes selon les règles faits par vos visiteurs sur leurs navigateurs.
 - Avantage de ce type de valeur, être capables de lire vos écrits sans difficulté. De plus, le texte peut également s'adapter relativement aux autres éléments de votre page web.

Les Propriétés de Type « Font- »: font-size

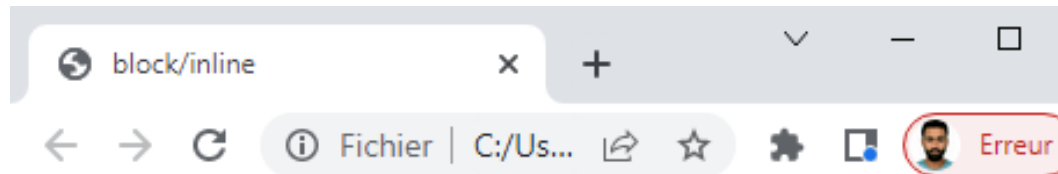
- Voici un exemple d'utilisation de la propriété « **font-size** », avec des valeurs relative et absolue. Notez qu'on utilisera les notations « **px** » pour **pixel**, « **pt** » pour **point** et « **%** » pour **pourcentage**.

```
mon_premier_pas_CSS_font_size.html x
<!DOCTYPE HTML>
<html lang="fr-FR">
  <head>
    <title> block/inline </title>
    <meta charset="utf-8"/>
    <link rel="stylesheet" href="premier_pas_CSS_font_size.css"/>
  </head>
  <body>
    <h1> font_size absolu en pixel </h1>
    <h2> font_size absolu en point </h2>
    <h3> font_size relatif en pourcentage </h3>
    <h4> font_size relatif en ex </h4>
    <p> font_size relatif en em </p>
  </body>
</html>
```

```
premier_pas_CSS_font_size.css •
1  /* Commentaire En CSS */
2  h1{ font-size: 15px;}
3
4  h2{font-size: 10pt;}
5
6  h3{font-size: 60%;}
7
8  h4{font-size: 5ex;}
9
10 p{font-size: 1em;}
```


Les Propriétés de Type « Font- »: font-size

- Voici un exemple d'utilisation de la propriété « **font-size** », avec des valeurs relative et absolue. Notez qu'on utilisera les notations « **px** » pour **pixel**, « **pt** » pour **point** et « **%** » pour **pourcentage**.



font_size absolu en pixel

font_size absolu en point

font_size relatif en pourcentage

font_size relatif en ex

font_size relatif en em

Les Propriétés de Type « Font- »: font-style

- La propriété « **font-style** » permet de fixer l'inclinaison d'un texte:
- La propriété « **font-style** » accepte quatre valeurs différentes:
 - **Normal**: valeur par défaut;
 - **Italic**: change le texte en italique;
 - **Oblique**: penche le texte;
 - **inherit**: hérite des propriétés de l'élément parent.

Les Propriétés de Type « Font- »: font-style

- Exemple: La propriété « **font-style** », Normal, Italic, Oblique:

```
mon_premier_pas_CSS_font_style.html x
1 <!DOCTYPE HTML>
2 <html lang="fr-FR">
3   <head>
4     <title> block/inline </title>
5     <meta charset="utf-8"/>
6     <link rel="stylesheet" href="premier_pas_CSS_font_style.css"/>
7   </head>
8   <body>
9     <h1> font_style : Normal </h1>
10    <h2> font_style italique </h2>
11    <h3> font_style oblique </h3>
12  </body>
13 </html>
```

```
premier_pas_CSS_font_style.css x
1 /* Commentaire En CSS */
2 h1{ font-style: normal;}
3
4 h2{font-style: italic; ;}
5
6 h3{font-style: oblique;}
7
```

Les Propriétés de Type « Font- »: font-style

- Exemple: La propriété « **font-style** », Normal, Italic, Oblique:

font_style : Normal

font_style italique

font_style oblique

- vous pourrez constater par vous mêmes qu'elles rendent un résultat visuellement quasi-identique:
 - **italique** est un état spécial d'une police, et qui n'est pas supporté par toutes les polices.
 - **oblique** correspond à une inclinaison forcée de la version normale d'une police, et peut donc être appliqué à toutes les polices.

Les Propriétés de Type « Font- »: font-style

■ Exemple: La propriété « **font-style** », inhérit:

- Nous avons dit que tout élément enfant héritait par défaut des styles de ses parents. A quoi sert donc la valeur inherit ? En fait, cette dernière sert à annuler un style dans un cas bien particulier.

```
mon_premier_pas_CSS_font_style_inherit.html x
<!DOCTYPE HTML>
<html lang="fr-FR">
  <head>
    <title> block/inline </title>
    <meta charset="utf-8"/>
    <link rel="stylesheet" href="
premier_pas_CSS_font_style_inherit.css"/>
  </head>
  <body>
    <p> Bonjour </p>
    <p> Je suis en oblique ! </p>
    <div>
      <p class="style_normal"> Je suis pas en
oblique </p>
      <p class="style_normal"> Moi non plus ! </p>
    </div>
  </body>
</html>
```

```
premier_pas_CSS_font_style_inherit.css x
/* Commentaire En CSS */
p{
  font-style: oblique;
}

div{
  font-style: normal;
}

.style_normal{
  font-style: inherit;
}
```

Les Propriétés de Type « Font- »: font-style

- Exemple: La propriété « **font-style** », inhérit:
 - Nous avons dit que tout élément enfant héritait par défaut des styles de ses parents. A quoi sert donc la valeur inherit ? En fait, cette dernière sert à annuler un style dans un cas bien particulier.

Bonjour

Je suis en oblique !

Je suis pas en oblique

Moi non plus !

Les Propriétés de Type « Font- »: font-weight

- La propriété « **font-weight** » permet de fixer le poids d'un texte. Cette propriété accepte six valeurs différentes :
 - **Normal**: la valeur par défaut;
 - **Lighter**: version allégée de la police ;
 - **Bold**: la police est en gras;
 - **Bolder**: la police est encore plus en gras;
 - Une centaine valeur compris entre 100 et 900, du plus léger au plus gras;
 - **Inherit**: hérite des styles de ses parents.
- **NB**: certaines polices et certains navigateurs ne supportent pas les valeurs inférieures à « **normal** » et supérieures à « **bold** ». Dans ce cas, la police s'affichera la valeur la plus proche supportée.

Les Propriétés de Type « Font- »: font-weight

- La propriété « **font-weight** » permet de fixer le poids d'un texte. Cette propriété accepte six valeurs différentes :

```
mon_premier_pas_CSS_font_weight.html
<!DOCTYPE HTML>
<html lang="fr-FR">
  <head>
    <title> block/inline </title>
    <meta charset="utf-8"/>
    <link rel="stylesheet" href="
      premier_pas_CSS_font_weight.css"/>
  </head>
  <body>
    <p> Font-weight types </p><br>
    <div class="p1"> Je suis en Normal</div><br>
    <div class="p2"> Je suis en Lighter</div><br>
    <div class="p3"> Je suis en Blod</div><br>
    <div class="p4"> Je suis en Bolder</div><br>
    <div class="p5"> Je suis avec valeur 700</div>
    <div class="p1">
      <p class="p6"> je suis aussi en valeur 600 </p>
    </div>
  </body>
</html>
```

```
premier_pas_CSS_font_weight.css
/* Commentaire En CSS */
.p1{font-weight: normal; }
.p2{font-weight: lighter; }
.p3{font-weight: bold; }
.p4{font-weight: bolder; }
.p5{font-weight: 700; }
.p6{font-weight: inherit; }
p{ font-weight: bolder; }
```


Les Propriétés de Type « Font- »: font-weight

- La propriété « **font-weight** » permet de fixer le poids d'un texte. Cette propriété accepte six valeurs différentes :

Font-weight types

Je suis en Normal

Je suis en Lighter

Je suis en Blod

Je suis en Bolder

Je suis avec valeur 700

je suis aussi en valeur 600

Les Propriétés de Type : line-height

- La propriété « **line-height** » sert à fixer l'écartement, c'est-à-dire la distance entre deux lignes de texte.

```
mon_premier_pas_CSS_line_height.html x
<!DOCTYPE HTML>
<html lang="fr-FR">
  <head>
    <title> block/inline </title>
    <meta charset="utf-8"/>
    <link rel="stylesheet" href="
      premier_pas_CSS_line_height.css"/>
  </head>
  <body>
    <p class="p1"> Je suis paragraphe 1 </p>
    <p> Je suis paragraphe 2 </p>

    <p class="p2"> Je suis paragraphe 3 </p>
    <p> Je suis paragraphe 3</p>
  </body>
</html>
```

```
premier_pas_CSS_line_height.css x
/* Commentaire En CSS */
.p1{line-height: 15px; }

.p2{line-height: 40px; }
```

Je suis paragraphe 1

Je suis paragraphe 2

Je suis paragraphe 3

Je suis paragraphe 3

Les Propriétés de Type « Font- »: font-family

- La propriété « **font-family** » permet de définir une liste, ordonnée par priorité, de polices à utiliser pour mettre en forme le texte de l'élément ciblé.
- Dans tous les cas, nous déclarerons plusieurs polices afin de s'assurer qu'au moins une des polices mentionnées soit supportée par vos visiteurs.
- En effet, il existe toujours des versions de navigateurs et des ordinateurs ne supportant pas certaines polices, d'où tout l'intérêt d'en déclarer plusieurs.
- La première police déclarée sera le choix par défaut. Si elle n'est pas lue par votre visiteur, alors on utilisera la seconde et etc.
- Commencez par la police de votre choix et terminez toujours par une famille générique, pour laisser le navigateur choisir une police similaire dans la famille générique, si aucune autre police n'est disponible.

Les Propriétés de Type « Font- »: font-family

- Les polices suivantes sont les meilleures polices Web sécurisées pour HTML et CSS :
 - Arial (sans-serif);
 - Verdana (sans-serif);
 - Tahoma (sans-serif);
 - Trebuchet MS (sans-serif);
 - Times New Roman (serif);
 - Georgia (serif);
 - Garamond (serif);
 - Courier New (monospace);
 - Brush Script MT (cursive).

Les Propriétés de Type « Font- »: font-family

- Les polices suivantes sont les meilleures polices Web sécurisées pour HTML et CSS :

mon_premier_pas_CSS_font_family.html x

```
<!DOCTYPE HTML>
<html lang="fr-FR">
  <head>
    <title> block/inline </title>
    <meta charset="utf-8"/>
    <link rel="stylesheet" href="
premier_pas_CSS_font_family.css"/>
  </head>
  <body>
    <p class="p1"> Je suis paragraphe avec arial </p>
    <p class="p1_1"> Je suis paragraphe avec arial </p>
    <p class="p2"> Je suis paragraphe avec Tahoma </p>
    <p class="p3"> Je suis paragraphe avec Times New Roman </p>
    >
    <p class="p4"> Je suis paragraphe avec Brush SScript MT </p>
    >
  </body>
</html>
```

premier_pas_CSS_font_family.css x

```
/* Commentaire En CSS */
.p1{font-family: arial, verdana, sans-serif}

.p1_1{font-family: arial, verdana}

.p2{font-family: tahoma, verdana, sans-serif}

.p3{font-family: Times New Roman , verdana, serif}

.p4{font-family: Brush Script MT , verdana, serif}
```

Je suis paragraphe avec arial

Je suis paragraphe avec arial

Je suis paragraphe avec Tahoma

Je suis paragraphe avec Times New Roman

Je suis paragraphe avec Brush SScript MT

Les Propriétés de Type : color

- Il existe plusieurs façons de gérer la couleur d'un texte:
 - **Première façon:** de changer la couleur d'un texte : en attribuant un nom de couleur (en anglais) en valeur de la propriété « **color** ».
 - Cette méthode reste très limitée car nous ne pouvons choisir que parmi seize noms de couleurs qui sont les suivants :

black (#000000)	silver (#C0C0C0)	gray (#808080)	white (#FFFFFF)
maroon (#800000)	red (#FF0000)	purple (#800080)	fuchsia (#FF00FF)
green (#008000)	lime (#00FF00)	olive (#808000)	yellow (#FFFF00)
navy (#000080)	blue (#0000FF)	teal (#008080)	aqua (#00FFFF)

Les Propriétés de Type : color

- Il existe plusieurs façons de gérer la couleur d'un texte:
 - **Deuxième façon:** de changer la couleur d'un texte : en attribuant en hexadécimales à la valeur de la propriété « **color** ».
 - Cette méthode à pour le but de choisir très précisément la couleur voulue. Cette valeur devra commencer par un dièse, suivie de 6 symboles choisis entre **0** et **F**. Exemple: **Green: #008000**
 - **Les deux premiers symboles** vont définir l'intensité de rouge de notre couleur.
 - **Les deux deuxième symboles** vont définir l'intensité de vert de notre couleur.
 - **Les deux troisième symboles** vont définir l'intensité de bleu de notre couleur.

Les Propriétés de Type : color

- Il existe plusieurs façons de gérer la couleur d'un texte:
 - **Troisième façon:** est d'utiliser la valeur de type RGB, **R: Red, G: Green, B: Blue.** (**rgb(R_val, G_val, B_val)**).
 - Nous allons indiquer trois nombres compris entre **0** et **255** en valeur.
 - Le premier nombre correspond une nouvelle fois à l'intensité de rouge, le second à l'intensité de vert et le troisième à l'intensité de bleu que l'on veut utiliser pour former notre couleur finale.
 - Il existe de nombreuses similitudes entre les notations en hexadécimal et en RGB. On peut d'ailleurs tout à fait convertir l'hexadécimal en RGB, en partant du principe que **00** en hexa correspond à **000** en RGB et que **FF** en hexa correspond à **255** en RGB.

Les Propriétés de Type : opacity

- Cette propriété prend une valeur entre **0** totalement **transparent** et **1** totalement **opaque**.
- Si on utilise une valeur de type nom de couleur ou hexadécimale avec la propriété « **color** », on doit utiliser la propriété « **opacity** » pour fixer l'opacité des textes.
- Si nous avons utilisé une valeur de type RGB avec la propriété « color », nous pouvons également maîtriser l'opacité de notre texte d'une façon plus simple, en utilisant une valeur de type **RGBa**.
 - Il suffit de rajouter une valeur pour l'opacité de notre texte après les trois valeurs de notre propriété RGB: `rgb(R_val, G_val, B_val, O_val)`.

Les Propriétés de Type : opacity

```
mon_premier_pas_CSS_opacity.html x
<!DOCTYPE HTML>
<html lang="fr-FR">
  <head>
    <title> block/inline </title>
    <meta charset="utf-8"/>
    <link rel="stylesheet" href="premier_pas_CSS_opacity.css"/>
  </head>
  <body>
    <p class="p1"> Je suis paragraphe 1 </p>
    <p class="p2"> Je suis paragraphe 2 </p>
    <p class="p3"> Je suis paragraphe 3 </p>
    <p class="p4"> Je suis paragraphe 4 </p>
  </body>
</html>
```

```
premier_pas_CSS_opacity.css x
/* Commentaire En CSS */
.p1{color:red;
  opacity: 0.7;
}

.p2{color:red;
  opacity: 0.2;
}

.p3{color: rgb(200, 50, 10,0.7);}
.p4{color: rgb(200, 50, 10,0.2);}
```

Je suis paragraphe 1

Je suis paragraphe 2

Je suis paragraphe 3

Je suis paragraphe 4

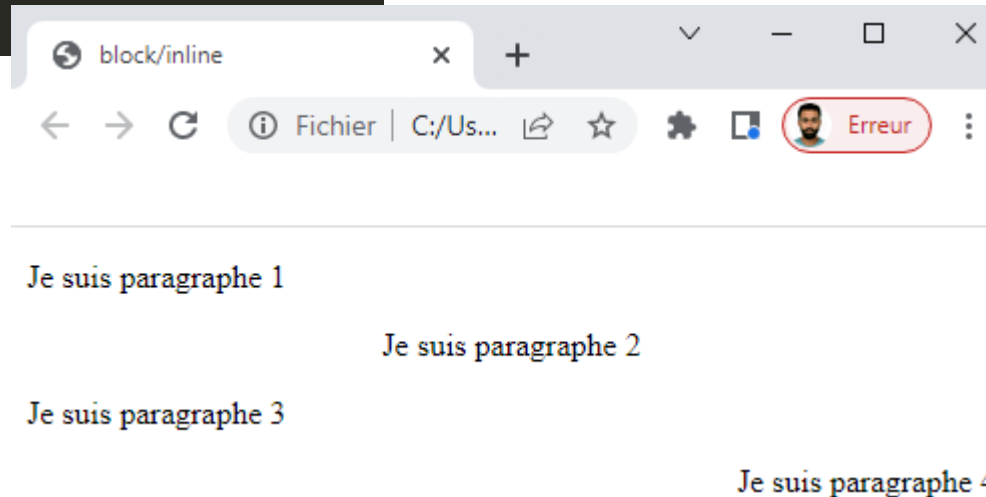
Les Propriétés de Type « Text- » text-align

- Pour modifier l'alignement d'un texte, nous allons utiliser la propriété « **text-align** ».
- Cette propriété peut prendre cinq valeurs différentes:
 - **Left**: le texte sera aligné sur la gauche, c'est la valeur par défaut;
 - **Center**: le texte sera centré;
 - **Right**: le texte sera aligné sur la droite;
 - **Justify**: le texte sera justifié;
 - **inherit**: hérite des propriété de l'élément parent.
- **NB**: Le centrage ou l'alignement se fait toujours par rapport à l'élément parent le plus proche du texte.

Les Propriétés de Type « Text- » text-align

```
mon_premier_pas_CSS_texte_align.html x
<!DOCTYPE HTML>
<html lang="fr-FR">
  <head>
    <title> block/inline </title>
    <meta charset="utf-8"/>
    <link rel="stylesheet" href="
      mon_premier_pas_CSS_texte_align.css"/>
  </head>
  <body>
    <p class="style_1"> Je suis paragraphe 1 </p>
    <p class="style_2"> Je suis paragraphe 2 </p>
    <p class="style_3"> Je suis paragraphe 3 </p>
    <p class="style_4"> Je suis paragraphe 4 </p>
  </body>
</html>
```

```
mon_premier_pas_CSS_texte_align.css
/* Commentaire En CSS */
.style_1{ text-align: left;}
.style_2{text-align: center;}
.style_3{text-align: justify;}
.style_4{text-align: right;}
```



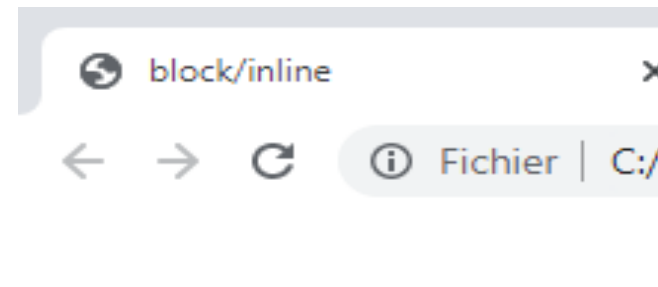
Les Propriétés de Type « Text- » text-decoration

- On peut modifier la décoration d'un texte grâce à la propriété « **text-decoration** » parmi six valeur pour cette propriété:
 - **Underline** : le texte sera souligné ;
 - **Overline** : une ligne apparaîtra au dessus du texte ;
 - **Line-through** : le texte sera barré ;
 - **Blink** : le texte clignotera (attention, ne fonctionne pas sur tous les navigateurs) ;
 - **Inherit** : hérite des propriété de l'élément parent.
 - **None** : pas de décoration, comportement par défaut.

Les Propriétés de Type « Text- » text-decoration

```
mon_premier_pas_CSS_texte_decoration.html x
<!DOCTYPE HTML>
<html lang="fr-FR">
  <head>
    <title> block/inline </title>
    <meta charset="utf-8"/>
    <link rel="stylesheet" href="
      mon_premier_pas_CSS_texte_decoration.css"/>
  </head>
  <body>
    <p class="style_1"> Je suis paragraphe 1 </p>
    <p class="style_2"> Je suis paragraphe 2 </p>
    <p class="style_3"> Je suis paragraphe 3 </p>
    <p class="style_4"> Je suis paragraphe 4 </p>
  </body>
</html>
```

```
mon_premier_pas_CSS_texte_decoration.css x
/* Commentaire En CSS */
.style_1{ text-decoration: underline;}
.style_2{text-decoration: overline;}
.style_3{text-decoration: line-through;}
.style_4{text-decoration: blink;}
```



Je suis paragraphe 1

Je suis paragraphe 2

~~Je suis paragraphe 3~~

Je suis paragraphe 4

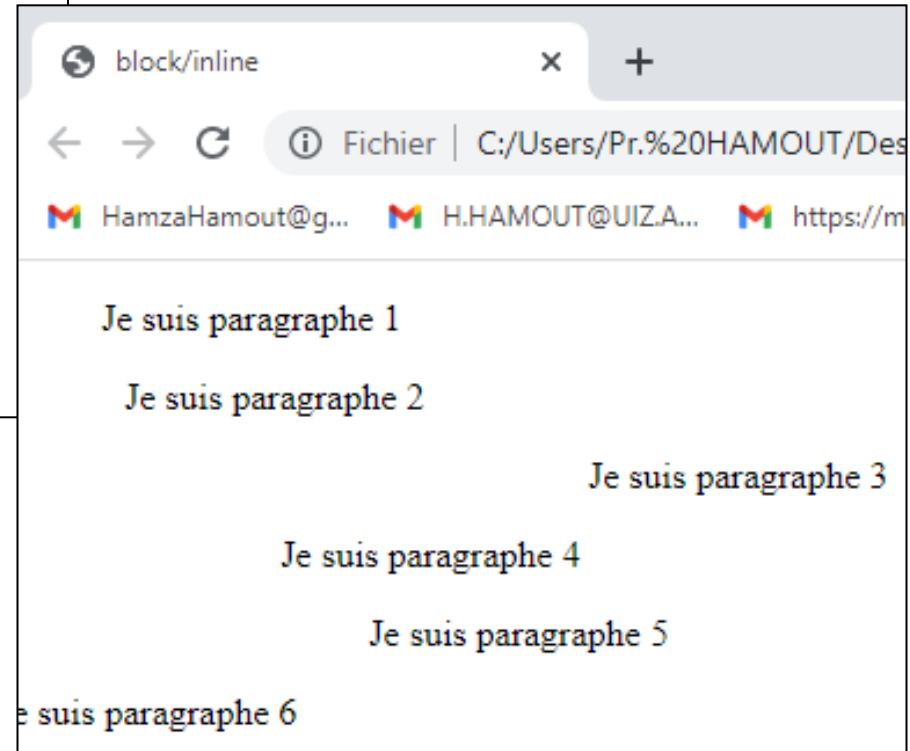
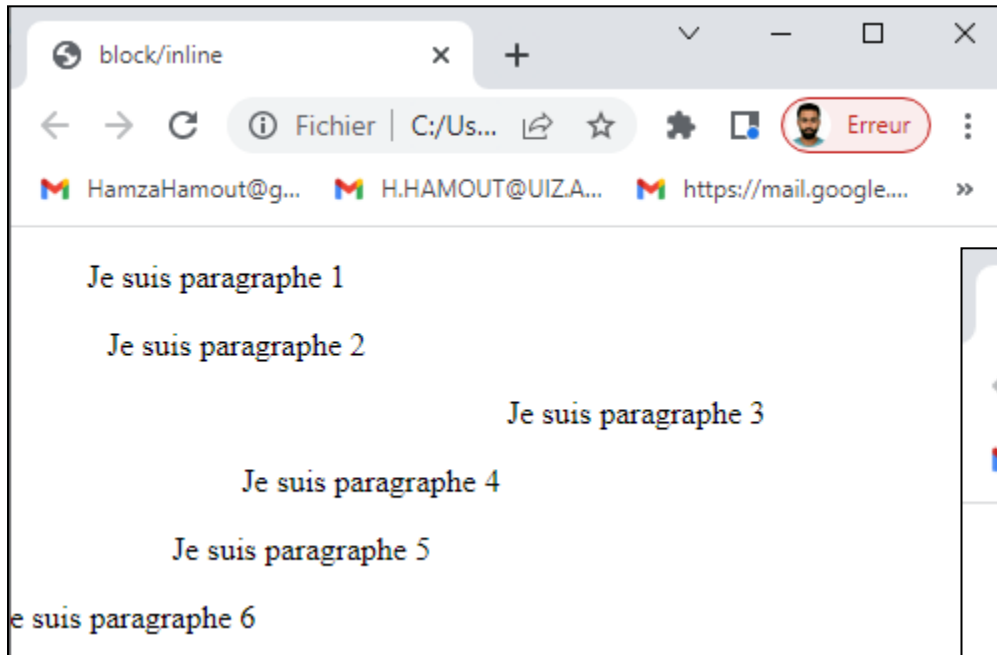
Les Propriétés de Type « Text- » text-indent

- La propriété « **text-indent** » sert à gérer l'indentation d'un texte, c'est le décalage d'un texte sur la droite généralement.
- Cette propriété accepte des valeurs de type absolu (**px**, **pt**) et relatif (**em**, **ex**, **%**). Vous pouvez également lui attribuer des valeurs négatives afin de décaler votre

```
mon_premier_pas_CSS_texte_indent.html x
<!DOCTYPE HTML>
<html lang="fr-FR">
  <head>
    <title> block/inline </title>
    <meta charset="utf-8"/>
    <link rel="stylesheet" href="
      mon_premier_pas_CSS_texte_indent.css"/>
  </head>
  <body>
    <p class="style_1"> Je suis paragraphe 1 </p>
    <p class="style_2"> Je suis paragraphe 2 </p>
    <p class="style_3"> Je suis paragraphe 3 </p>
    <p class="style_4"> Je suis paragraphe 4 </p>
    <p class="style_5"> Je suis paragraphe 5 </p>
    <p class="style_6"> Je suis paragraphe 6 </p>
  </body>
</html>
```

```
mon_premier_pas_CSS_texte_indent.css x
/* Commentaire En CSS */
.style_1{ text-indent: 30px;}
.style_2{text-indent: 30pt;}
.style_3{text-indent: 15em;}
.style_4{text-indent: 15ex;}
.style_5{text-indent: 15%;}
.style_6{text-indent: -15px;}
```

Les Propriétés de Type « Text- » text-indent



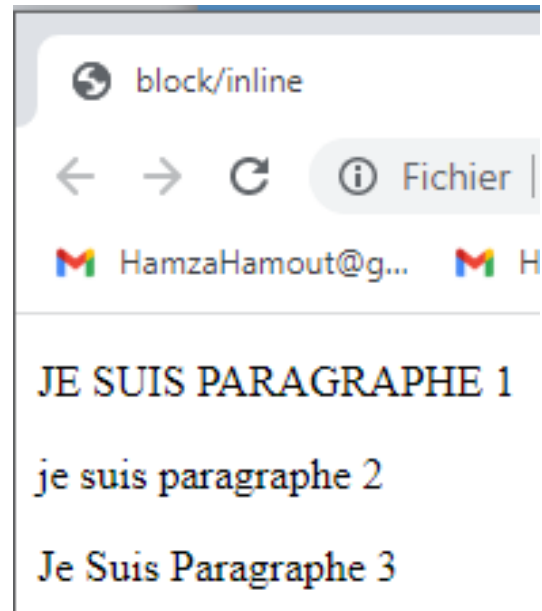
Les Propriétés de Type « Text- » text-transform

- On utilise la propriété « **text-transform** » pour modifier l'aspect des caractères d'un texte (majuscules ou minuscules).
- Nous pouvons choisir parmi cinq valeurs :
 - **Uppercase** : transforme tout le texte en majuscules ;
 - **Lowercase** : met tout le texte en minuscules ;
 - **Capitalize** : met uniquement la première lettre de chaque mot en majuscule ;
 - **None** : pas de transformation ;
 - **Inherit** : hérite des styles de l'élément parent.

Les Propriétés de Type « Text- » text-transform

```
mon_premier_pas_CSS_texte_transform.html x
<!DOCTYPE HTML>
<html lang="fr-FR">
  <head>
    <title> block/inline </title>
    <meta charset="utf-8"/>
    <link rel="stylesheet" href="
      mon_premier_pas_CSS_texte_transform.css"/>
  </head>
  <body>
    <p class="style_1"> Je suis paragraphe 1 </p>
    <p class="style_2"> Je suis paragraphe 2 </p>
    <p class="style_3"> Je suis paragraphe 3 </p>
  </body>
</html>
```

```
mon_premier_pas_CSS_texte_transform.css x
/* Commentaire En CSS */
.style_1{ text-transform: uppercase;}
.style_2{text-transform: lowercase;}
.style_3{text-transform: capitalize;}
```



Les Propriétés de Type: letter-spacing & word-spacing

- Les propriétés « **letter-spacing** » et « **word-spacing** » permettent respectivement d'ajuster l'espace entre les lettres et entre les mots.
- Ces propriétés se comportent de manière similaire et acceptent des valeurs absolues (**px**, **pt**) et relatives (**ex**, **em**, **%**).

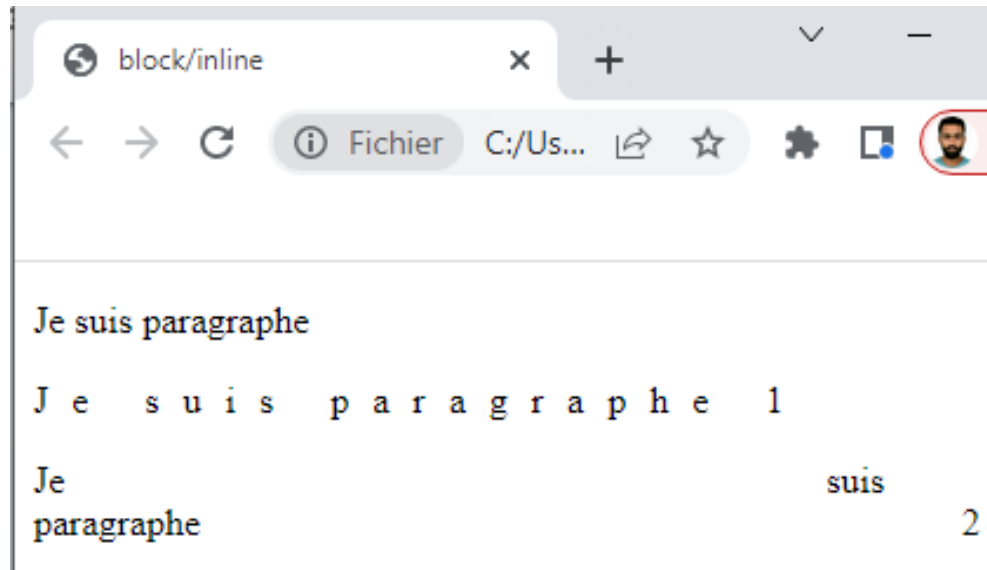
```
mon_premier_pas_CSS_texte_spacing.html x
<!DOCTYPE HTML>
<html lang="fr-FR">
  <head>
    <title> block/inline </title>
    <meta charset="utf-8"/>
    <link rel="stylesheet" href="
      mon_premier_pas_CSS_texte_spacing.css"/>
  </head>
  <body>
    <p> Je suis paragraphe </p>
    <p class="style_1"> Je suis paragraphe 1 </p>
    <p class="style_2"> Je suis paragraphe 2 </p>
  </body>
</html>
```

```
mon_premier_pas_CSS_texte_spacing.css x
/* Commentaire En CSS */
.style_1{ letter-spacing: 10px}

.style_2{word-spacing: 20em}
```

Les Propriétés de Type: letter-spacing & word-spacing

- Les propriétés « **letter-spacing** » et « **word-spacing** » permettent respectivement d'ajuster l'espace entre les lettres et entre les mots.
- Ces propriétés se comportent de manière similaire et acceptent des valeurs absolues (**px**, **pt**) et relatives (**ex**, **em**, **%**).



Les Propriétés de Type « Text- » text-shadow

- On peut ajouter des effets d'ombre à un texte en utilisant la propriété « **text-shadow** ».
- Cette propriété nécessite au minimum deux valeurs pour fonctionner. Cependant, dans la grande majorité des cas, nous en utiliserons quatre, dans l'ordre vu ci-dessous:
- Les trois premières valeurs correspondent à des longueurs et la dernière est la couleur de l'ombre :
 - 1ère valeur : déplacement horizontal de l'ombre ;
 - 2ème valeur : déplacement vertical de l'ombre ;
 - 3ème valeur : rayon de propagation (flou gaussien) de l'ombre ;
 - 4ème valeur : couleur de l'ombre (accepte les mêmes valeurs que la propriété « color »).

Les Propriétés de Type « Text- » text-shadow

mon_premier_pas_CSS_texte_shadow.html x

```
<!DOCTYPE HTML>
<html lang="fr-FR">
  <head>
    <title> block/inline </title>
    <meta charset="utf-8"/>
    <link rel="stylesheet" href="
      mon_premier_pas_CSS_texte_shadow.css"/>
  </head>
  <body>
    <p> Je suis paragraphe </p>
    <p class="style_1"> Je suis paragraphe 1 </p>
    <p class="style_2"> Je suis paragraphe 2 </p>
  </body>
</html>
```

mon_premier_pas_CSS_texte_shadow.css x

```
/* Commentaire En CSS */
.style_1{ text-shadow: 5px 3pt 4px green;}
.style_2{ text-shadow: 5px 3pt pink;}
```

block/inline

← → ↻ ⓘ Fichier

Je suis paragraphe

Je suis paragraphe 1

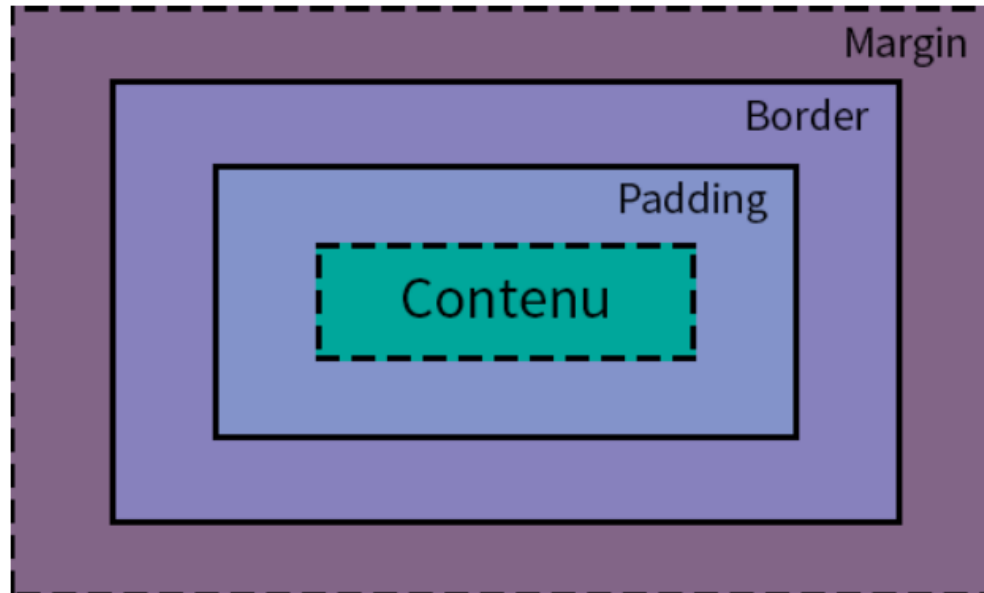
Je suis paragraphe 2

Le Modèle des Boîtes

- Le modèle des boîtes est un concept essentiel : « tout élément d'une page est une boîte rectangulaire et peut avoir un padding, une marge et des bordures.
- Tout élément, qu'il soit un élément de type block ou de type inline, est une boîte rectangulaire.
- Les propriétés permettant d'indiquer la **longueur**, la **largeur**, la **marge**, le **padding** et les bordures d'un élément forment ce qu'on appelle le **modèle des boîtes**.

Le Modèle des Boîtes

- La première boîte est définie par la longueur et la largeur d'un élément.
- La padding, ou marge intérieure, forme ensuite la seconde boîte.
- Les bordures qui constituent la troisième boîte.
- La marge extérieure vient former la quatrième et dernière boîte.



Le Modèle des Boîtes: Hauteur et largeur d'un élément

- Tout élément possède une **hauteur** et une **largeur** par défaut.

- **La hauteur:**

- La hauteur d'un élément est déterminée par son contenu. Par exemple, des paragraphes d'une ligne ou de deux lignes n'occuperont pas la même hauteur.
- Pour modifier la hauteur d'un élément, on utilise la propriété « **height** » à laquelle on attribue une valeur en **px**, **%** ou **égale** à auto dans la grande majorité des cas.
- En utilisant la valeur **auto**, on laisse le navigateur de nos visiteurs décider de la hauteur que doit prendre l'élément visé.

Le Modèle des Boîtes: Hauteur et largeur d'un élément

- Tout élément possède une **hauteur** et une **largeur** par défaut.
- **La largeur:**
 - La largeur par défaut d'un élément est avant tout déterminée par son type (block ou inline) puis par son contenu si l'élément est de type inline.
 - **Rappele:** Les éléments de type block occupent automatiquement toute la largeur disponible.
 - Pour modifier la largeur d'un élément, on utilise cette fois la propriété « **width** ». Cette propriété prend les mêmes types de valeurs que « **height** ».

Le Modèle des Boîtes: Hauteur et largeur d'un élément

```
mon_premier_pas_CSS_height_width.html x
<!DOCTYPE HTML>
<html lang="fr-FR">
  <head>
    <title> block/inline </title>
    <meta charset="utf-8"/>
    <link rel="stylesheet" href="
      mon_premier_pas_CSS_height_width.css"/>
  </head>
  <body>
    <p class="style_1"> Je suis paragraphe 1 </p>
    <p class="style_2"> Je suis paragraphe 2 </p>
    <p class="style_3"> Je suis paragraphe 3 </p>
  </body>
</html>
```

```
mon_premier_pas_CSS_height_width.css x
border:2px solid black;
height: 100px;
}

.style_3{
border:2px solid black;
width: 100px;
}
```

Je suis paragraphe 1

Je suis paragraphe 2

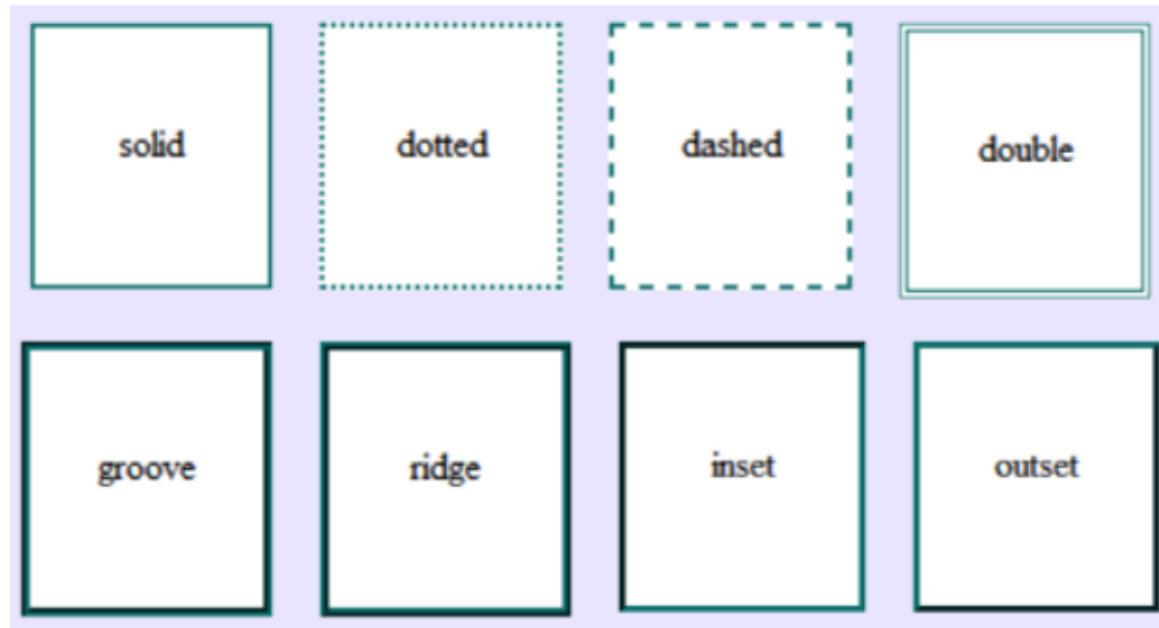
Je suis
paragraphe 3

Le Modèle des Boîtes: Les bordures et les bordures arrondies

- Il existe de nombreuses sortes de bordures dont certaines sont plus ou moins bien supportées par certains navigateurs.
- Pour créer des bordures et les personnaliser, nous allons avoir besoin de trois propriétés :
 - « **Border-style** », qui va définir le style de la bordure ;
 - « **border-radius** », créer des bordures arrondies;
 - « **Border-width** », qui va définir l'épaisseur de la bordure (valeur en **px**) ;
 - « **Border-color** », qui va définir la couleur de la bordure (accepte les mêmes valeurs que la propriété « **color** »).

Le Modèle des Boîtes: Les bordures et les bordures arrondies

- « **Border-style** », qui va définir le style de la bordure:
 - La propriété « **border-style** » peut prendre de nombreuses valeurs différentes.
 - Les valeurs les plus utilisées sont **solid**, **dotted** (pointillé) et **dashed** (tiret). Vous pouvez voir ci-dessous le résultat pour chaque style de bordure :



Le Modèle des Boîtes: Les bordures et les bordures arrondies

- « **border-radius** », créer des bordures arrondies:
 - La propriété « **border-radius** » va prendre une valeur : la taille de l'angle, en **px**.
 - Notez que l'on peut définir des angles différents pour chaque côté de nos bordures en utilisant les mots clefs « **border-top-left-radius** », « **border-top-right-radius** », « **border-bottom-left-radius** » et « **border-bottom-right-radius** ».

Le Modèle des Boîtes: Les bordures et les bordures arrondies

```
mon_premier_pas_CSS_border.html x
<!DOCTYPE HTML>
<html lang="fr-FR">
  <head>
    <title> block/inline </title>
    <meta charset="utf-8"/>
    <link rel="stylesheet" href="
      mon_premier_pas_CSS_border.css"/>
  </head>
  <body>
    <p class="style_1"> Je suis paragraphe 1 </p>
    <p class="style_2"> Je suis paragraphe 2 </p>
    <p class="style_3"> Je suis paragraphe 3 </p>
  </body>
</html>
```



Je suis paragraphe 1

Je suis paragraphe 2

Je suis paragraphe 3

```
mon_premier_pas_CSS_border.css x
/* Commentaire En CSS */
.style_1{
  border-width: 3px;
  border-style: dotted;
  border-color: blue;
  border-radius: 15px;
}

.style_2{
  border-width: 3px;
  border-style: double;
  border-color: red;
  border-bottom-right-radius: 13px;
}

.style_3{
  border-width: 3px;
  border-style: dashed;
  border-color: green;
  border-bottom-left-radius: 13px;
  border-top-right-radius: 17px;
}
```

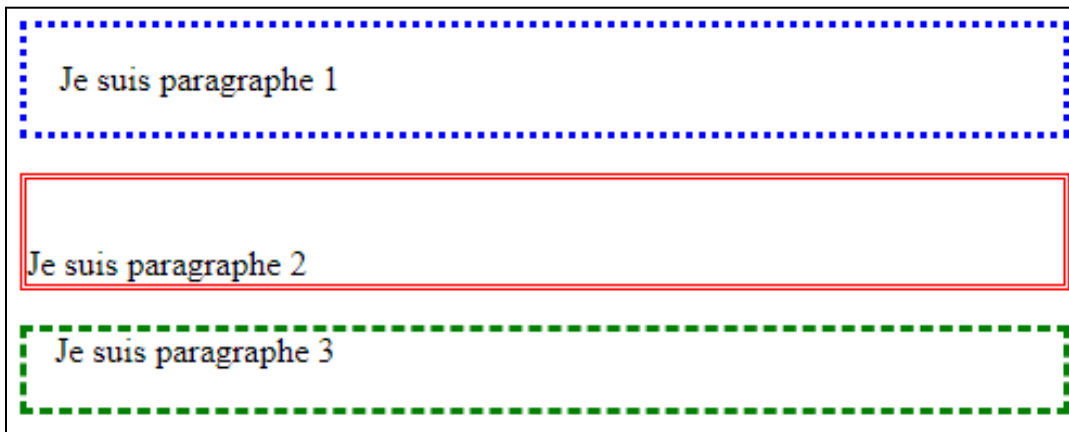
Le Modèle des Boîtes: Les marges intérieures

- Pour définir les marges intérieures d'un élément, nous utiliserons la propriété « **padding** ».
- La propriété « **padding** » va définir l'espace entre l'élément en soi et sa bordure. Cette propriété doit être utilisée uniquement dans ce but, et jamais pour positionner des éléments dans une page ou les uns par rapport aux autres.
- On donnera généralement une valeur en **px** à « **padding** ». Notez que l'on peut définir des espacements différents pour chaque marge intérieure de nos éléments en utilisant les propriétés « **padding-right** », « **padding-bottom** », « **padding-left** » et « **padding-top** ».

Le Modèle des Boîtes: Les marges intérieures

mon_premier_pas_CSS_padding.html

```
<!DOCTYPE HTML>
<html lang="fr-FR">
  <head>
    <title> block/inline </title>
    <meta charset="utf-8"/>
    <link rel="stylesheet" href="
      mon_premier_pas_CSS_padding.css"/>
  </head>
  <body>
    <p class="style_1"> Je suis paragraphe 1 </p>
    <p class="style_2"> Je suis paragraphe 2 </p>
    <p class="style_3"> Je suis paragraphe 3 </p>
  </body>
</html>
```



mon_premier_pas_CSS_padding.css

```
border-style: dotted;
border-color: blue;
padding: 15px;
}

.style_2{
  border-width: 3px;
  border-style: double;
  border-color: red;
  padding-top: 30px;
}

.style_3{
  border-width: 3px;
  border-style: dashed;
  border-color: green;
  padding-left: 13px;
  padding-bottom: 17px;
}
```

Le Modèle des Boîtes: Les marges extérieures

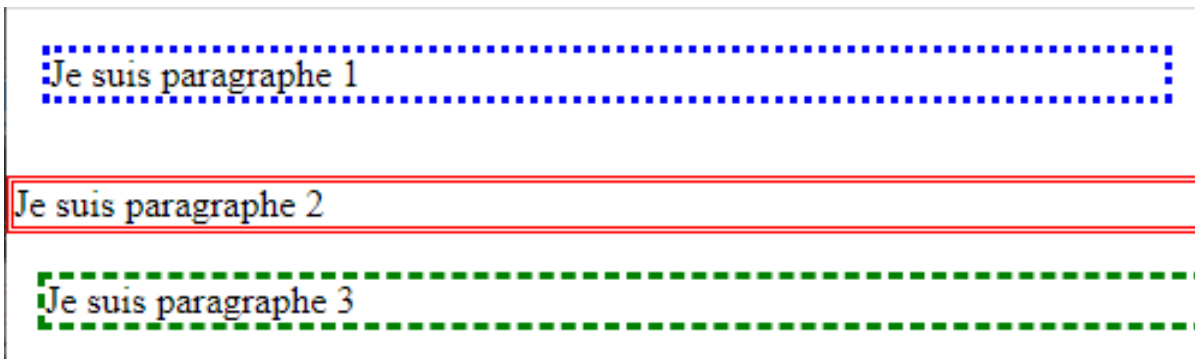
- Pour définir la taille des marges extérieures, c'est-à-dire de l'espace à l'extérieur des bordures d'un élément, nous allons utiliser la propriété « **margin** ».
- La propriété « **margin** » peut tout-à-fait être utilisée pour positionner des éléments dans une page ou les uns par rapport aux autres.
- Nous attribuerons généralement des valeurs en **px** ou en **%** à cette propriété.
- Tout comme la propriété « **padding** », nous allons pouvoir des marges différentes de chaque côté de nos éléments avec les propriétés « **margin-right** », « **margin-bottom** », « **margin-left** » et « **margin-top** ».

Le Modèle des Boîtes: Les marges extérieures

- Notez que les valeurs par défaut des marges intérieures et extérieures peuvent légèrement différer d'un navigateur à un autre. Cela peut impacter le design général de votre site pour certains de vos visiteurs.
- Afin de s'assurer que chaque visiteur verra un résultat conforme à nos attentes, il suffit d'appliquer un « **padding** » et une « **margin** » avec des valeurs égales à zéro à notre élément « **body** ». Ensuite, on précisera les différentes marges souhaitées à nos éléments enfants.

Le Modèle des Boîtes: Les marges extérieures

```
mon_premier_pas_CSS_margin.html x
<!DOCTYPE HTML>
<html lang="fr-FR">
  <head>
    <title> block/inline </title>
    <meta charset="utf-8"/>
    <link rel="stylesheet" href="
      mon_premier_pas_CSS_maring.css"/>
  </head>
  <body>
    <p class="style_1"> Je suis paragraphe 1 </p>
    <p class="style_2"> Je suis paragraphe 2 </p>
    <p class="style_3"> Je suis paragraphe 3 </p>
  </body>
</html>
```



```
mon_premier_pas_CSS_maring.css x
/* Commentaire En CSS */
body{
  padding: 0px;
  margin: 0px;
}
.style_1{
  border-width: 3px;
  border-style: dotted;
  border-color: blue;
  margin: 15px;
}
.style_2{
  border-width: 3px;
  border-style: double;
  border-color: red;
  margin-top: 30px;
}
.style_3{
  border-width: 3px;
  border-style: dashed;
  border-color: green;
  margin-left: 13px;
}
```

FIN.

MERCI DE VOTRE ATTENTION

DES QUESTIONS ?