

CSS

Cascading Style Sheets

“Feuille de style”



- Qu'est ce qu'un fichier CSS et comment le lier au HTML
- Appliquer une propriété CSS à une balise
- Cibler des éléments
 - avec **class** et **id**
 - dans un texte avec
- Changer l'apparence du texte
 - Type / Taille / Style de Police
 - l'Alignement des textes
 - la couleur du texte / l'arrière-plan du texte
- Créer des bordures et arrondir des angles
- Ajouter une ombre portée (sur un élément / sur du texte)
- Créer des apparences dynamiques
 - Pseudo Classe
 - Sélecteurs avancés
- Structure d'une page
- Modèle de boîtes (Flexbox)
- La propriété Display
- Les positions
- Responsives design avec les media queries

Un fichier CSS

CSS : Langage utilisé pour **définir la présentation visuelle / la mise en forme des éléments écrits en HTML.**
Cela se code dans un fichier à part souvent nommé **style.css**.

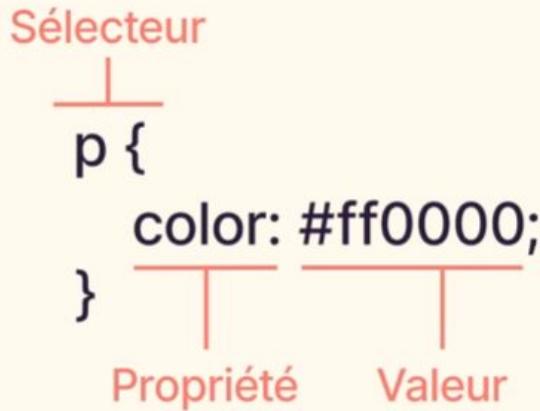
Il faut donc lier ces deux fichiers index.html et style.css :

```
4  <head>
5    <meta charset="UTF-8">
6    <link rel="stylesheet" href="style.css">
7    <title>Entraînement-ESG</title>
8  </head>
```

Remarque : *link* est une balise orpheline.

Lorsque vous cliquerez sur votre fichier HTML pour l'ouvrir dans le navigateur, ce dernier aura l'instruction d'appeler le fichier CSS associé pour appliquer les propriétés de style à la page web.

Appliquer une propriété CSS à une balise HTML



Sélecteur : nom des balises dont on veut modifier l'apparence ; toujours suivi d'accolade { ... } ; jamais de chevrons < ... > ; on peut mettre plusieurs sélecteurs au même style, séparés par une virgule.

Propriétés : désignent les paramètres qui doivent être mis en forme ; ex. : font-size, width, background... ; Il peut y en avoir plusieurs ; Chaque propriété est suivie par : et finit par un ;

Valeur : pour chaque propriété on doit indiquer une valeur



The screenshot shows a code editor with two files and a browser preview.

index.html:

```
1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5   <meta charset="UTF-8">
6   <link rel="stylesheet" href="style.
7     <title>Entrainement-ESG</title>
8 </head>
9
10 <body>
11   <h1>Entrainement ESG</h1>
12
13   <p>Lorem, ipsum dolor sit amet
14   consectetur adipisicing elit.
    Adipisci, quos qui? Enim
```

style.css:

```
16 p, h1{
17   color: darkcyan;
18 }
19 }
```

Browser Preview:

The browser title is "Entrainement-ESG". The page content includes a large heading "Entrainement ESG" and a paragraph of placeholder text "Lorem, ipsum dolor sit amet consectetur adipisicing elit. Adipisci, quos qui? Enim". The text color is dark cyan, as defined in the CSS.

Ce style est attribué à TOUS les paragraphes et le h1

On **commente** également en CSS comme en HTML

CTRL + /

L'attribut CLASS et ID

Pour cibler UN élément on va rajouter l'attribut **CLASS** (ou l'attribut **ID**) dans le HTML

=> sur l'HTML : l'attribut *class* est toujours suivi du signe = puis des guillemets / quotes s'écrit à l'intérieur des chevrons <p class="firstParagraph">....</p>

=> sur le CSS le nom de la *class* est précédé d'un point .

```
<h1>Entrainement ESG</h1>

<p class="myFirstParagraph">Lorem, ipsum dolor sit amet
consectetur adipisicing elit. Adipisci, quo qui? Enim
exercitationem dolor natus
    consectetur odio voluptas numquam maxime aut, labore
    magnam soluta laudantium voluptates repudiandae amet
    asperiores ab.
    Officia quibusdam itaque, rem quo voluptatibus odit
    voluptates nisi magnam ab fuga voluptas, earum velit neque
    dolorem adipisci cupiditate et repellat quea. Iste a sequi
    eos temporibus est inventore quea?
</p>
<p>Lorem ipsum dolor sit amet consectetur, adipisicing elit.
Exercitationem aut maiores nostrum dignissimos deleniti!
Consequatur impedit dolorum dignissimos amet, qui officia quo
iure ipsam tempore ad, laborum, nemo illum eaque?</p>
<H2>Les bases</H2>
```

```
15 }
16 p, h1{
17     color: darkcyan;
18 }
19
20 .myFirstParagraph{
21     color: #fff;
22     background-color: #fd4efc;
23 }
24 }
```

Entrainement ESG

Lore, ipsum dolor sit amet consectetur adipisicing elit. Adipisci, quo qui? Enim exercitationem dolor natus consectetur odio voluptas numquam maxime aut, labore magnam soluta laudantium voluptates repudiandae amet asperiores ab. Officia quibusdam itaque, rem quo voluptatibus odit voluptates nisi magnam ab fuga voluptas, earum velit neque dolorem adipisci cupiditate et repellat quea. Iste a sequi eos temporibus est inventore quea?

Lore ipsum dolor sit amet consectetur, adipisicing elit. Exercitationem aut maiores nostrum dignissimos deleniti! Consequatur impedit dolorum dignissimos amet, qui officia quo iure ipsam tempore ad, laborum, nemo illum eaque?

Les bases



L'attribut `class` peut être utilisé plusieurs fois dans la même page.

ex: permet de donner à tous les boutons, la même forme, couleur, taille...

FAIRE UN DON

L'`id` fonctionne comme la `class` MAIS ne peut être utilisé qu'une seule fois dans le code

s'utilise sur des éléments unique comme une ancre ou en JAVASCRIPT

l'id dans le CCS c est un `#` et non un `.` comme la class



Cas pratique

```
<h2 class="titre"> Les bases</h2>
<p class="paragraphe">Lorem ipsum dolor sit amet consectetur
adipisicing elit. Veniam quas perferendis ipsa?
    Delectus nemo, repellendus ipsam corrupti eaque
    aspernatur sit sapiente, sunt nesciunt ipsum tempora
    maxime,
    nulla vel cum repudiandae? </p>
<figure>
    
</figure>
```

```
37
38     .titre{
39         color: #686767;
40         font-family: 'Lobster', sans-serif;
41     }
42
43     .paragraphe{
44         color: #2e38bb;
45         font-family: 'Times New Roman', Times, serif;
46     }
47     #logo{
48         border: 3px solid #2e38bb;
49     }
50 }
```

Rq : on peut cumuler les class au sein d'une même balise, séparées par un espace (class="classe_1 classe_2").



Les bases

Lorem ipsum dolor sit amet consectetur adipisicing elit. Veniam quas perferendis ipsa? Delectus nemo, repellendus ipsam corrupti eaque aspernatur sit sapiente, sunt nesciunt ipsum tempora maxime, nulla vel cum repudiandae?



Cibler un élément dans un texte

Pour cibler juste un mot ou un groupe de mots dans le paragraphe, c'est avec la balise dans le même style

balise de type *inline* (comme <a>) à l'inverse de la balise de type *block* (comme <div> ou de <p>)

```
<p class="myFirstParagraph">Lorem, ipsum dolor sit amet  
consectetur adipisicing elit. Adipisci, quos qui? Enim  
exercitationem dolor natus  
consectetur <span>odio voluptas numquam</span> maxime aut,  
labore magnam soluta laudantium voluptates repudiandae amet  
asperiores ab.  
Officia quibusdam <em>itaque, rem quo voluptatibus </em>  
odit voluptates nisi magnam ab fuga voluptas, earum velit  
neque  
dolorem adipisci <strong>cupiditate et repellat</strong>  
quae. Iste a sequi eos temporibus est inventore quae?  
</p>
```

```
24 .myFirstParagraph{  
25     color: #525151;  
26 }  
27 .myFirstParagraph span,  
28 .myFirstParagraph em{  
29     color: #c41616;  
30 }  
31  
32  
33  
34  
35
```

Lorem, ipsum dolor sit amet consectetur adipisicing elit.
Adipisci, quos qui? Enim exercitationem dolor natus
consectetur *odio voluptas numquam* maxime aut, labore
magnam soluta laudantium voluptates repudiandae amet
asperiores ab. Officia quibusdam *itaque, rem quo voluptatibus*
odit voluptates nisi magnam ab fuga voluptas, earum velit
neque dolorem adipisci *cupiditate et repellat* quae. Iste a sequi
eos temporibus est inventore quae?

Lorem ipsum dolor sit amet consectetur, adipisicing elit.
Exercitationem aut maiores nostrum dignissimos delenit!
Consequatur impedit dolorum dignissimos amet, qui officia
quos iure ipsam tempore ad, laborum, nemo illum eaque?



Type de POLICE

propriété font-family (change la manière d'écrire le texte / l'apparence du texte)

Par défaut, il y en a déjà plusieurs intégrées dans le navigateur

```
p {  
    font-family: ;  
}  
    'Courier New', Courier, monospace  
    'Franklin Gothic Medium', 'Arial Narrow', Arial, san-serif  
    'Gill Sans', 'Gill Sans MT', Calibri, 'Trebuchet MS'...  
    'Lucida Sans', 'Lucida Sans Regular', 'Lucida Grande'...  
    'Segoe UI', Tahoma, Geneva, Verdana, sans-serif  
    'Times New Roman', Times, serif  
    'Trebuchet MS', 'Lucida Sans Unicode', 'Lucida Grand...'...  
    Arial, Helvetica, sans-serif  
    Cambria, Cochin, Georgia, Times, 'Times New Roman', ...  
    Georgia, 'Times New Roman', Times, serif  
    Impact, Haettenschweiler, 'Arial Narrow Bold', sans-serif  
    Verdana, Geneva, Tahoma, sans-serif
```



Attention avec les polices avec SÉRIF, à intégrer avec parcimonie



Pour une autre police : <https://fonts.google.com/>

Chercher une police qui vous convient (ex. : LOBSTER) puis le style (regular 400)

En haut à gauche, cliquer sur “View Selected Family”

Copier le lien

Dans le head de l'index.html

Dans le CSS, pour le sélecteur concerné



The screenshot shows the Google Fonts interface with the following elements:

- Selected family:** Lobster, Regular 400
- Review:** Lobster, Regular 400
- Options:** Add more styles, Remove all
- Link Type:** <link> (selected)
- Generated CSS:**

```
<link rel="preconnect" href="https://fonts.googleapis.com">
<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
<link href="https://fonts.googleapis.com/css2?family=Lobster&display=swap" rel="stylesheet">
```
- CSS rules to specify families:**

```
font-family: 'Lobster', sans-serif;
```

The screenshot shows a code editor with two files open: `index.html` and `style.css`, and a browser window displaying the result.

index.html:

```
index.html M ...
index.html > html > head
4  <head>
5    <meta charset="UTF-8">
6    <!-- style css -->
7    <link rel="preconnect" href="https://fonts.googleapis.com">
8    <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
9    <link href="https://fonts.googleapis.com/css2?family=Lobster&display=swap" rel="stylesheet">
10   <link rel="stylesheet" href="style.css">
11   <title>Entrainement-ESG</title>
12 </head>
13
14 <body>
15   <h1>Entrainement ESG</h1>
16
```

style.css:

```
# style.css M ...
# style.css > h1
16
17 h1{
18   color: darkcyan;
19   font-family: 'Lobster', sans-serif;
20 }
21
22
23
24
25
26
27
28
29
```

Browser Preview:

The browser window title is "Entrainement-ESG". The page content is "**Entrainement ESG**". The browser toolbar includes icons for back, forward, search, and refresh, along with network tabs for "enfants", "ecole", "banque", "ChatGPT", and "GoStudent".



Conseils : pas plus de 3 polices différentes

Taille de POLICE

propriété font-size

avec une unité absolue : le pixel (px) ou une unité relative : rem, em

```
<h2 class="titre"> Les bases</h2>
<p class="paragraphe">Lorem ipsum
dolor sit amet consectetur
adipisicing elit. Veniam quas
perferendis ipsa?
    Delectus nemo, repellendus ipsam
    corrupti eaque aspernatur sit
    sapiente, sunt nesciunt ipsum
    tempora maxime,
    nulla vel cum repudiandae? </p>
<figure>
    
</figure>
```

```
37
38     .titre{
39         color: #686767;
40         font-family: 'Lobster', sans-serif;
41         font-size: 3rem;
42     }
43
44     .paragraphe{
45         color: #2e38bb;
46         font-family: 'Times New Roman',
47         Times, serif;
48         font-size: 1.5rem;
49     }
50     #logo{
51         border: 3px solid #2e38bb;
52     }
```

Les bases

Lorem ipsum dolor sit amet consectetur adipisicing elit. Veniam quas perferendis ipsa? Delectus nemo, repellendus ipsam corrupti eaque aspernatur sit sapiente, sunt nesciunt ipsum tempora maxime, nulla vel cum repudiandae?



unités absolues :

pixels -> taille fixe indépendante de l'environnement d'affichage

unités relatives :

%, rem, em, s'ajustent en fonction de l'environnement
flexibilité pour s'adapter aux différentes tailles d'écran



```
1 <p class="elem1">Élément 1 : 1em</p>
2 <p class="elem2">Élément 2 : 1.3em</p>
3 <p class="elem3">Élément 3 : 2em</p>
```

... et qu'on y associe le CSS ci-dessous :

```
1 .elem1 {
2   font-size: 1em;
3 }
4
5 .elem2 {
6   font-size: 1.3em;
7 }
8
9 .elem3 {
10  font-size: 2em;
11 }
```

... on obtient alors le résultat suivant :

Elément 1 : 1em

Elément 2 : 1.3em

Elément 3 : 2em

Style de POLICE

Mettre en Italique un mot, un groupe de mot : propriété **font-style** avec deux valeurs

font-weight : italic;
font-weight : normal;

```
37 .titre{  
38   color: #686767;  
39   font-family: 'Lobster', sans-serif;  
40   font-size: 3rem;  
41  
42 }  
43 .paragraphe{  
44   color: #2e38bb;  
45   font-family: 'Times New Roman', Times, serif;  
46   font-size: 1.5rem;  
47   font-style: italic;  
48  
49 }  
50 #logo{  
51   border: 3px solid #2e38bb;  
52 }  
53 }
```

Les bases

Lorem ipsum dolor sit amet consectetur adipisicing elit. Veniam quas perferendis ipsa? Delectus nemo, repellendus ipsam corrupti eaque aspernatur sit sapiente, sunt nesciunt ipsum tempora maxime, nulla vel cum repudiandae?



Mettre en Gras : propriété **font-weight** avec 3 valeurs

font-weight : bold; (le texte sera en gras)
font-weight : normal; (le texte sera écrit normalement)
font-weight : thin; (le texte est plus fin)

On peut aussi écrire une valeur numérique de 100 à 900

Thin (100)
Extra-Light (200)
Light (300)
Regular (400)
Medium (500)
SemiBold (600)
Bold (700)
ExtraBold / Heavy (800)
Black (900)

Souligner du texte :

La propriété CSS **text-decoration** avec 3 valeurs

underline : souligné;

line-through : barré;

none : normal; (par défaut)

Ceci est mon texte souligné

~~Ceci est mon texte barré~~

Ceci est mon texte normal



Aligner du texte

propriété **text-align** avec 4 valeurs
left : le texte sera aligné à gauche;
center : le texte sera centré;
right : le texte sera aligné à droite;
justify : le texte sera “justifié”

le texte prend toute la largeur sans laisser d'espace à
la fin des lignes

Texte aligné à gauche

Texte centré

Texte aligné à droite

 Lorem ipsum dolor sit amet consectetur
 adipisicing elit. Veniam quas preferendis ipsa?
 Delectus nemo, repellendus ipsam corrupti
 eaque aspernatur sit sapiente, sunt nesciunt
 ipsum tempora maxime, nulla vel cum
 repudiandae?

```
42 }  
43 }  
44 .paragraphe{  
45   color: #2e38bb;  
46   font-family: 'Times New Roman', Times, serif;  
47   font-size: 1.5rem;  
48   text-align:justify  
49 }  
50 }
```



L'alignement ne fonctionne que sur des balises block (100% de la largeur),
aligner une balise *inline* est IMPOSSIBLE comme <a>

Cas pratique

- Importer deux polices de Google Font : Montserrat et Manrope
- Assigner les propriétés de Polices suivantes :
 - Le titre H1
 - 3.5em
 - en couleur #A5B4FC
 - police Montserrat
 - 1 paragraphe
 - 1.1em
 - une liste Ordonnée
 - 1em
 - police Manrope
 - un lien
 - enlever le soulignement
 - 1em
 - police Manrope



Changer la couleur du texte

Propriété **color** : de nombreux **noms de couleurs** sont supportés par les navigateurs

```
color: blue;
color: red;
color: yellow;
color: purple;
```

Mais aussi

sous forme **hexadécimale** (6 chiffres précédés d'un #)

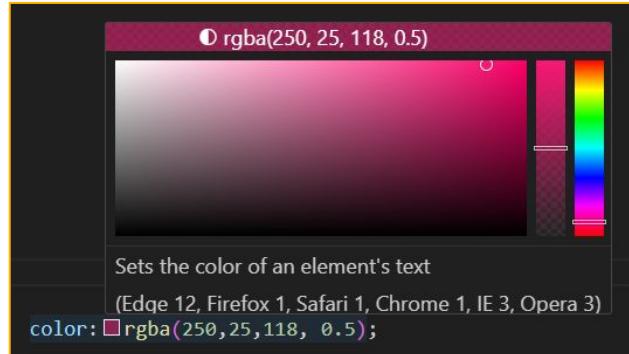
```
color: #2e38bb;
color: #c41616;
color: #d6c80a;
color: #4b1256;
```

en **notation RGB** (Red Green Blue)

```
color: rgb(46, 55, 186);
color: rgb(196, 23, 23);
color: rgb(214, 201, 10);
color: rgb(73, 18, 84);
```

On peut ajouter une notion
d'**opacité** en ajoutant **a**.

```
color: rgba(250,25,118, 0.5);
```



Changer l'arrière-plan du texte

propriété background-color

s'utilise comme la propriété color

avec des noms de couleur, des notations hexadécimale, des méthodes RGB ou RGBA

```
<body>
  <h1>L'héritage</h1>
  <p>Lorem, ipsum dolor sit amet consectetur adipisicing elit. Iure
aliquid eaque accusamus recusandae reiciendis
  exercitationem nisi atque repellat. Assumenda rerum autem
  corrupti ullam praesentium, amet repudiandae sed eius
  officiis saepe?
  Fuga suscipit facere sapiente! Asperiores, nulla soluta illo
  quae maiores esse nobis laboriosam, ut placeat,
  facere minus voluptates autem sit impedit quaerat. In laborum
  beatae, culpa sapiente consequuntur earum
  repellendus.</p>
  <h2>titres et paragraphes</h2>
  <p>Lorem ipsum dolor, sit amet consectetur adipisicing elit.
Tempore, veritatis dolores quod labore corrupti
  laboriosam ullam dignissimos suscipit quos ipsam! Ducimus
tempore aperiam veniam totam culpa, inventore iste
  similique eaque!
```

```
1
2   body{
3     background-color: □black;
4     color: ■white;
5   }
```



Le *body* est en noir avec écriture blanche,
mais les *<h1>*, *<p>*, *<h2>* ont pris la couleur 

La balise *body* contient les titres et les paragraphes

En CSS lorsque vous appliquez un style à une balise,
toutes les balises à l'intérieur prendront ce style

C'est l'héritage
les balises à l'intérieur héritent de ces propriétés

L'héritage

Lorem ipsum dolor sit amet consectetur adipisicing elit. Iure aliquid eaque accusamus recusandae reiciendis exercitationem nisi atque repellat. Assumenda rerum autem corrupti ullam praesentium, amet repudiandae sed eius officiis saepe? Fuga suscipit facere sapiente! Asperiores, nulla soluta illo quae maiores esse nobis laboriosam, ut placeat, facere minus voluptates autem sit impedit quaerat. In laborum beatae, culpa sapiente consequuntur earum repellendus.

titres et paragraphes

Lorem ipsum dolor, sit amet consectetur adipisicing elit. Tempore, veritatis dolores quod labore corrupti laboriosam ullam dignissimos suscipit quos ipsam! Ducimus tempore aperiam veniam totam culpa, inventore iste similique eaque! Ex alias assumenda ut accusantium, dolorem dolorum at vitae. Eum voluptatibus pariatur minus corporis doloremque, adipisci dolorum eius tempora tempore maiores dolore laboriosam ratione hic sed laborum est sit fugiat!

C'est de là que vient le nom CSS *Cascading Style Sheets* (feuille de style en cascade) :
les propriétés sont héritées en cascade.

Une image de fond en background

Mettre une image de fond : pas forcément sur la page entière

propriété **background-image**

l'adresse de l'image peut être en absolu (<https://...>) ou en relatif (*images.png*)



L'adresse de l'image doit être indiquée par rapport au fichier CSS

la propriété **background** : super propriété

background-image => permet de définir le chemin de l'image en absolu ou relatif

background-repeat=> permet de répéter ou non l'image

background-attachment => associé à la valeur **fixed** : rendre l'image fixe au scroll

background-size => associé à la valeur **cover** : redimensionne l'image afin de s'adapter à l'élément qui la contient

background-position => associé aux valeurs **top, bottom, left, center, right** : dit où doit se trouver l'image



`background: une couleur;`

`background:` accepte une couleur (couleur, hexadécimal, RGB)



jouer sur la transparence avec la propriété `opacity`

valeur de 1 : c'est l'élément par défaut, aucune différence

valeur de 0: l'élément est totalement transparent



`opacity: 0.8;` (on voit 80%)

`opacity: 0.6;` (on voit 60%)

`opacity: 0.4;` (on voit 40%)

`opacity: 0.2;` (on voit 20%)

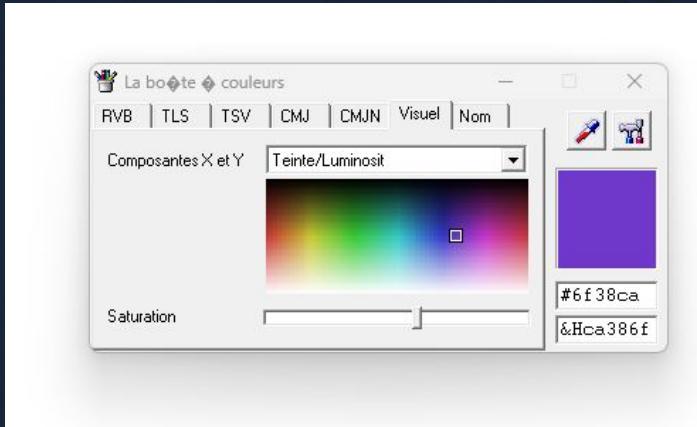


Bonjour et bienvenue sur mon site ! J'ai ajouté beaucoup de contenu en Lorem Ipsum. Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

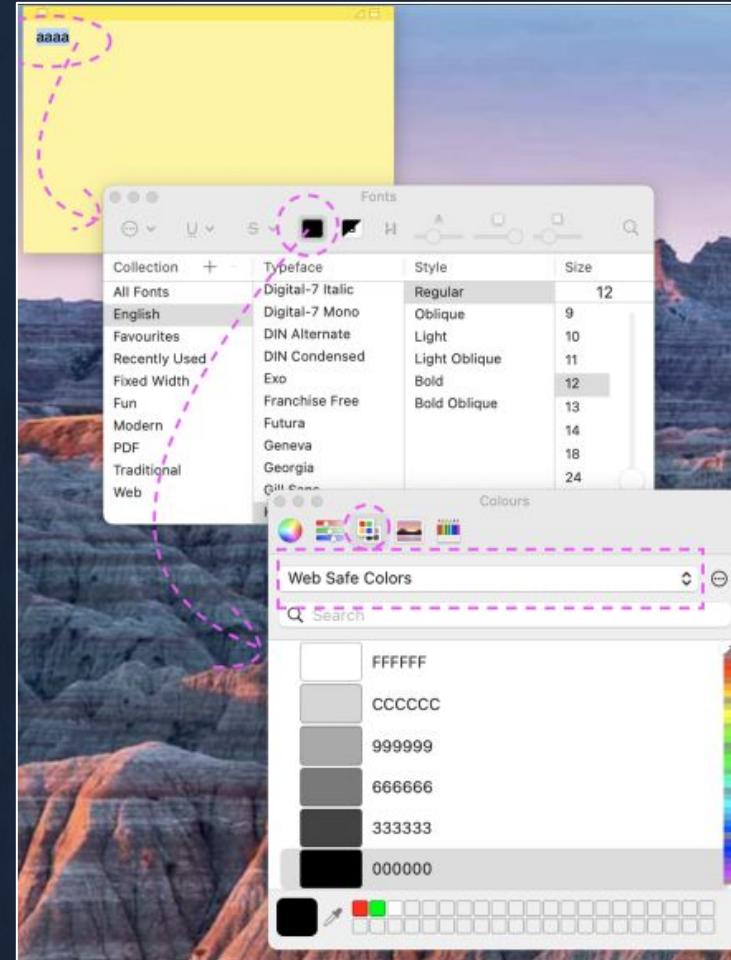
La boîte à couleurs

voici sur PC

<https://www.commentcamarche.net/telecharger/photo-graphisme/13501-la-boite-a-couleurs/>



Voici sur Mac



background: un dégradé;

```
.background h2{  
    background: linear-gradient(90deg, #8360c3, #2ebf91);  
    text-align: center;  
    padding: 50px;  
}
```

Ceci est le contenu de mon deuxième paragraphe

Dégradé

Voici un dégradé linéaire, à 90 degrés, en partant de la couleur #8360c3 pour arriver à la couleur #2ebf91

<https://cssgradient.io/>



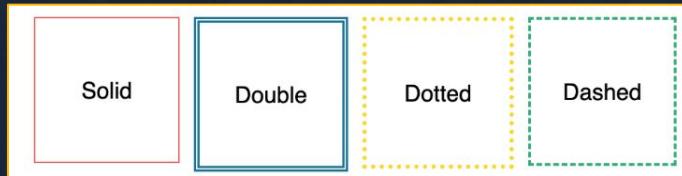
Créer des bordures

Propriété border : c'est une super propriété

border-width: la largeur de la bordure avec une valeur en pixel (2px)

border-color: la couleur de la bordure

border-style: le type de bordure solid, double, dotted, dashed



on peut mettre des bordures différentes
en fonction du côté avec:

border-top

border-right

border-bottom

border-left



Pour paramétrier chaque détail:

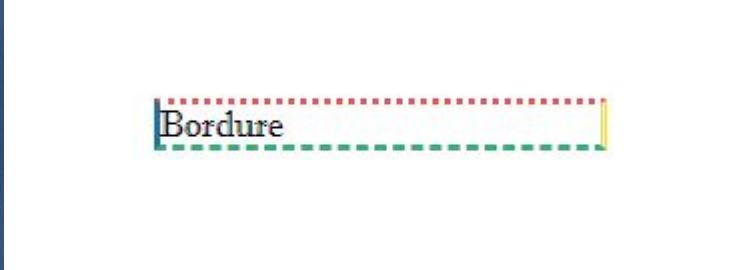
border-top-width

border-bottom-color

La largeur, la couleur et le type de bordure peuvent se cumuler

```
<section>
    <p class="border">Bordure</p>
</section>
```

```
80  /* section bordure */
81  .border{
82      border-top: 3px #EB5353 dotted;
83      border-right: 3px #F9D923 double;
84      border-bottom: 3px #36AE7C dashed;
85      border-left: 3px #187498 solid;
86      width:200px;
```



Bordure

A screenshot of a web browser window displaying a single paragraph element. The paragraph contains the text "Bordure". It features a complex border consisting of four distinct layers: a solid blue border on the left, a dashed red border on the top, a double yellow border on the right, and a dotted green border on the bottom. This visual representation demonstrates how multiple border properties can be applied simultaneously to create a compound effect.

Arrondir les angles

propriété **border-radius**

Indiquer la taille de l'arrondi en pixels

- 1 seule valeur: les 4 angles auront le même arrondi

```
<section>
  <p class="border">Bordure</p>
  <p class="borderRadius">Bordure</p>

</section>
```

89
90 .boundingBox {
91 font-size: 25px;
92 background-color: skyblue;
93 border-radius: 10px;
94 padding: 50px;
95 margin: 2rem;
96 }
97



- 4 valeurs: les 4 angles auront leur propre valeur

```
90     .boundingBox {  
91       font-size: 25px;  
92       background-color: skyblue;  
93       border-radius: 10px 30px 0px 90px;  
94       padding: 50px;  
95       margin: 2rem;  
96     }  
97
```



Ajouter une ombre portée sur un élément

propriété **box-shadow**

Effet de profondeur intéressant / de relief

ex: un bouton avec un relief indique clairement au visiteur que le bouton est “cliquable”

S'applique à tous les éléments

Prend 4 valeurs :

- décalage horizontal de l'ombre
- décalage vertical de l'ombre
- adoucissement du dégradé
- couleur de l'ombre

```
.boxShadow {  
    font-size: 25px;  
    background-color: skyblue;  
    border-radius: 8px;  
    padding: 40px;  
    margin: 2rem;  
    box-shadow: 6px 6px 0px #000000;  
}
```

Bordure

ombre horizontal de 6px, ombre vertical de 6px, couleur: noir

en passant la troisième valeur de 0 à 25px (pour adoucir):

```
.boxShadow {  
    font-size: 25px;  
    background-color: skyblue;  
    border-radius: 8px;  
    padding: 40px;  
    margin: 2rem;  
    box-shadow: 6px 6px 25px #8f8f8f;  
}
```

Bordure

Voici quelques exemples d'ombres
<https://getcsscan.com/css-box-shadow-examples>



Ajouter une ombre à un texte

propriété **text-shadow** : ajoute des ombres aux lettres

effet vintage garanti

fonctionne comme la propriété **box-shadow**

```
.textShadow{  
    font-size: 50px;  
    padding-left:5%;  
    text-shadow: 3px 3px 0px rgba(23, 245, 15, 0.502)  
}
```

TEXT-shadow

<https://codepen.io/chaplin16/pen/oNdWWNd>

Créer des apparences dynamiques

Les éléments peuvent changer de forme une fois que la page est chargée

→ les pseudo-classes

:hover s'applique lorsque l'utilisateur survole un élément avec sa souris

```
<!-- pseudo classe -->
<section>
    <article class="pseudoClasse">
        
        <p>Lorem ipsum, dolor sit amet consectetur adipisicing elit. Recusandae unde officiis tempora autem aliquam. Sint commodi alias, sit cupiditate deserunt sapiente minus, voluptatem repellendus quas consectetur neque praesentium. Alias, reprehenderit?
        </p>
        <a href="https://www.google.fr/">Lien Google</a>
    </article>
</section>

</body>
</html>
```

```
/* pseudo classe */
.pseudoClasse{
    width:320px;
    border: 2px solid #f50404;
    text-align:center;
    border-radius: 5px;
}
.pseudoClasse img:hover{
    filter: grayscale(100%);
}

.pseudoClasse p:hover{
    color: #rgb(252, 4, 4);
}

.pseudoClasse a{
    text-decoration: none;
}
.pseudoClasse a:hover{
    color: #rgb(248, 1, 137);
    text-decoration: overline;
}
```



*Lorem ipsum, dolor sit amet consectetur
 adipisicing elit. Recusandae unde officiis
tempora autem aliquam. Sint commodi alias,
 sit cupiditate deserunt sapiente minus,
voluptatem repellendus quas consectetur
neque praesentium. Alias, reprehenderit?*

Lien Google

modification au passage de la souris, grâce au :hover

l'image passe en
blanc-gris



*Lorem ipsum, dolor sit amet consectetur
 adipiscing elit. Recusandae unde officiis
 tempora autem aliquam. Sint commodi alias,
 sit cupiditate deserunt sapiente minus,
 voluntatem repellendus quis consecetur
 neque praesentium. Alias reprehenderit;*

Lien Google

le paragraphe
change de couleur



*Lorem ipsum, dolor sit amet consectetur
 adipisicing elit. Recusandae unde officiis
 tempora autem aliquam. Sint commodi alias,
 sit cupiditate deserunt sapiente minus,
 voluptatem repellendus quis consequatur
 neque praesentium. Alias, reprehenderit?*

Lien Google

le lien est surligné,
passe en rose



Latin Text
Lorem ipsum, dolor sit amet consectetur adipisciing elit. Recusandae unde officiis tempora autem aliquam. Sint commodi alias, sit cupiditate deserunt sapiente minus, voluptatem repellendus quas consecetur neque praesentium. Alias reprehenderit?

Lion Goods

:active applique un style particulier au moment du clic

en pratique elle n'est utilisée que sur les liens hypertexte car le style ne se maintient que très peu de temps
pas forcément toujours visible

```
.pseudoClasse a:active{  
    background-color: #36AE7C;  
}
```



:focus applique un style lorsque l'élément est sélectionné

comme dans un input de formulaire lorsque l'on clique dessus pour avoir la main

```
input:focus{  
    border: 4px solid red;  
    background-color: #c0c0c0;  
}
```



:visited applique un style lorsque l'élément a déjà été sélectionné

par défaut, le navigateur colorie le lien déjà visité en violet

si vous ne le souhaitez pas alors:

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



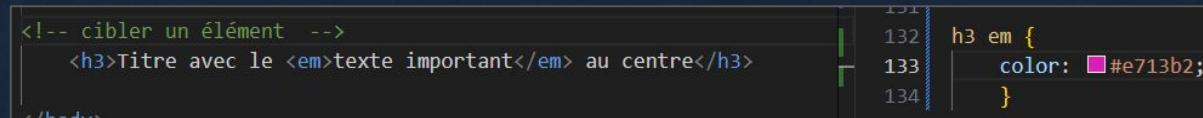
il existe de nombreux sélecteurs,
une grande quantité de propriétés CSS
utilisez <https://caniuse.com/>



propose des tables de compatibilité avec navigateurs et versions

Sélecteurs avancés

- Sélecteur universel : * => sélectionne toutes les balises
- Pour cibler une balise contenue dans une autre balise : on les accumule sans virgule



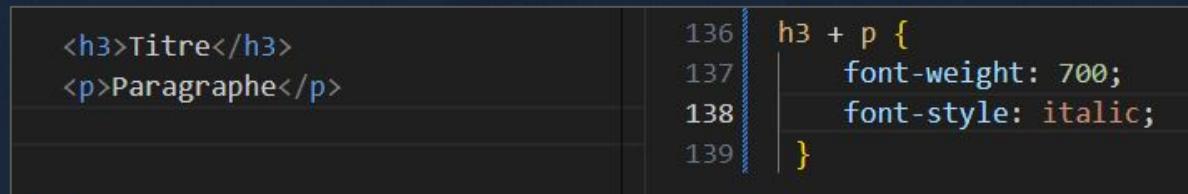
```
<!-- cibler un élément -->
<h3>Titre avec le <em>texte important</em> au centre</h3>
```

```
131
132 h3 em {
133   color: #e713b2;
134 }
```

Titre avec le **texte important** au centre

→ je cible toutes les balises em dans les balises h3

- Pour cibler une balise qui en suit une autre



```
<h3>Titre</h3>
<p>Paragraphe</p>
```

```
136 h3 + p {
137   font-weight: 700;
138   font-style: italic;
139 }
```

Titre
Paragraphe

→ je cible la première balise p située après un titre h3

- Pour cibler une balise qui a un attribut particulier

```
<a href="http://site.com" title="Infobulle">mon site</a>
```

```
143 a[title] {  
144     background-color: #rgb(238, 255, 1);  
145 }
```



→ je cible tous les liens hypertexte <a> qui ont un attribut *title*

- Pour cibler les balises du même genre qui sont les enfants directs

```
<div>  
    <h3>Titre</h3>  
    <p>Paragraphe 1</p>  
    <p>Paragraphe 2</p>  
</div>
```

```
145  
146 h3 ~ p {  
147     color: #rgb(255, 0, 0);  
148 }  
149
```

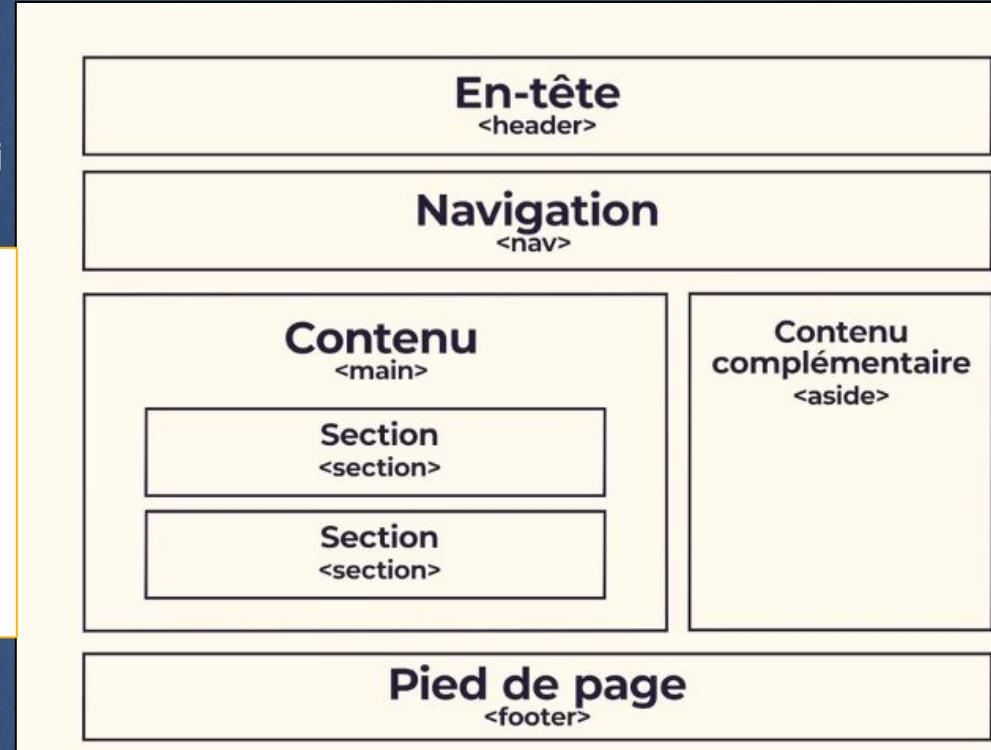
→ je cible tous les enfants directs <p> des h3

Il en existe de nombreux, découvrez les sur MDN
https://developer.mozilla.org/fr/docs/Web/CSS/CSS_selectors

Structurez une page

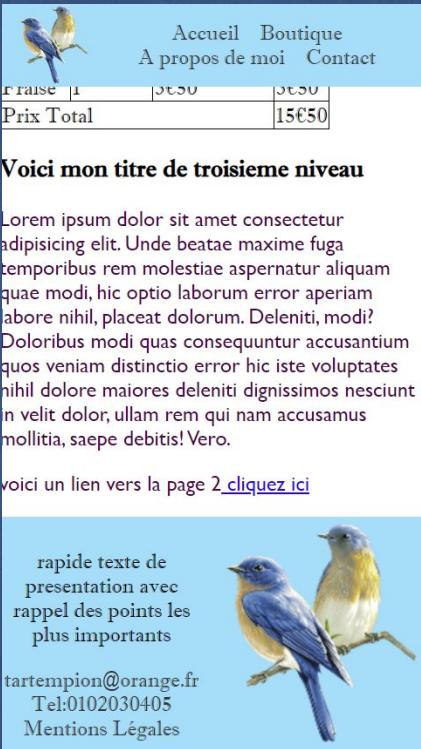
à l'intérieur du body:
différentes balises pour délimiter des contenus qui
constitue une page web.

- <header> : en-tête ;
- <footer> : pied de page ;
- <nav> : liens principaux de navigation ;
- <main> : contenu principal de la page ;
- <section> : section de page ;
- <aside> : informations complémentaires ;
- <article> : article indépendant.



Voici le squelette du site web: convention / ce que l'internaute à l'habitude de voir
ATTENTION: ce n'est pas de la mise en page
JUSTE cela indique au navigateur le rôle de chacun

<header> présent sur toutes pages (banner, logo..)
<footer> présent sur toutes les pages (liens de contact, mentions légales, politique de confidentialité...)



En général : le menu de navigation => en haut ou sur le côté

IMPERATIF: gardez le menu toujours au même endroit pour toutes les pages

SINON l'internaute sera perdu et frustré

```
<body>
  <header>
    
    <nav>
      <a href="#">Accueil</a>
      <a href="#">Boutique</a>
      <a href="#">A propos de moi</a>
      <a href="#">Contact</a>
    </nav>
  </header>
  <main>
```



exemple de header



Accueil Qu suis-je Pourquoi consulter Mes prestations Tarifs Contact

```
</body>
</html>
```

```
</div>
<div class="footer_contact">
    <p>
        rapide texte de présentation avec rappel des points
        les plus importants
    </p>
    <a href="mailto:contact@creationweb.fr"
        class="mailto">tartempion@orange.fr</a>
    <a href="tel:0102030405" class="tel">Tel:0102030405</a>
    <a href="mentionsLegales.html">Mentions Légales</a>
</div>


```

```
</footer>
```

ex du footer d'amazon

The screenshot shows the footer of an Amazon page with several columns of links:

- Pour mieux nous connaître**: Apropos d'Amazon, Carrières, Durabilité, Amazon Science.
- Gagnez de l'argent**: Vendez sur Amazon, Vendez sur Amazon Business, Vendez sur Amazon Handmade, Amazon pour les start-ups, Protégez et développez votre marque, Devenez Partenaire, Expédié par Amazon, Faites la promotion de vos produits, Auto-publiez votre livre, Amazon Pay, [Voir plus : Gagner de l'argent avec nous](#).
- Moyens de paiement Amazon**: Carte Amazon Business Amex, Cartes de paiement, Paiement en plusieurs fois, Convertisseur de devises Amazon, Cartes cadeaux, Recharge en ligne, Recharge en point de vente.
- Besoin d'aide ?**: Voir ou suivre vos commandes, Tarifs et options de livraison, Amazon Prime, Retours et remplacements, Recyclage (y compris les équipements électriques et électroniques), Infos sur notre Marketplace, Application Amazon Mobile, Service Client, Accessibilité.

Below the footer, there's a navigation bar with links to various countries and regions, followed by sections for different Amazon services like Kindle Direct Publishing, Audible, AbeBooks, Amazon Web Services, Amazon Advertising, Amazon Business, and others.

En-tête
`<header>`

Navigation
`<nav>`

Contenu
`<main>`

Section
`<section>`

Section
`<section>`

**Contenu
complémentaire**
`<aside>`

Pied de page
`<footer>`

Modèles de boites

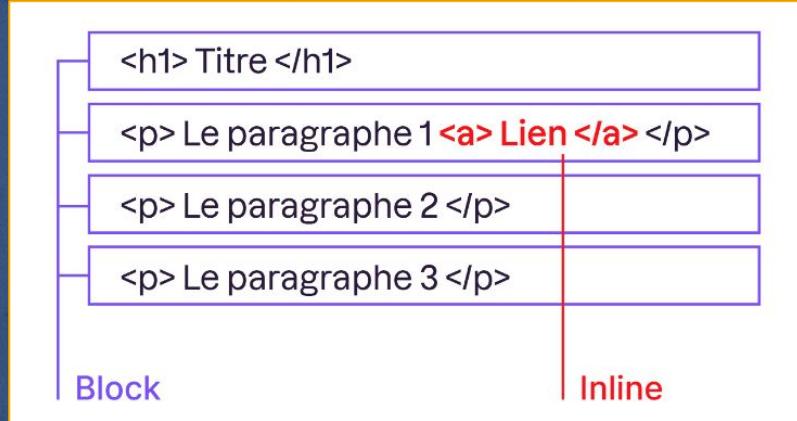
Type *block* et type *inline*

Type *block*: prend 100% de la largeur
retour à la ligne avant et après

ex: <div> <header> <p> <article> <h1> <h6> (visible via l inspecteur)

Type *inline*: se trouve obligatoirement à l'intérieur d'une balise block
prend la largeur et la hauteur de son contenu
ne revient pas à la ligne

ex: <a> (visible via l inspecteur)



Les blocs : les uns sous les autres
100% de leur largeur
peuvent imbriquer d'autres blocs ex: <main> contient des <section>

L'inline: elle vient s'insérer sur une même ligne
à l'intérieur d'une balise bloc

Remarque: les balises universelles <div> et

balises génériques sans valeur sémantique

MAIS

intérêt: on peut leur appliquer une class (ou un id)

on peut gérer leur largeur (width), leur hauteur (height)

```
<div>    block  
<span>  inline
```

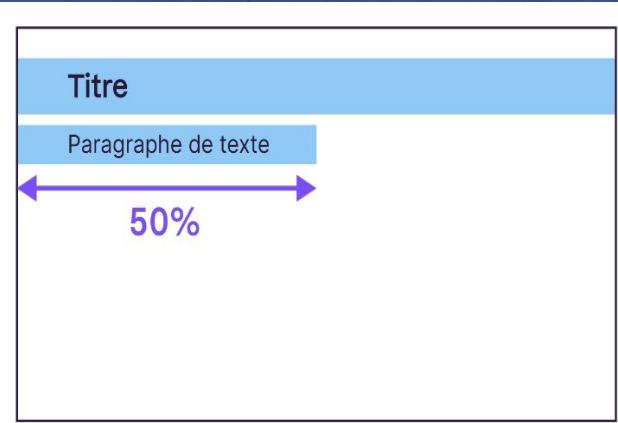
Ne pas abuser de ces balises, utilisez en premier les sémantiques et en dernier recours les **div** et **span**

Par exemple :

- `` : ici, il ne s'agit pas d'une bonne pratique, alors qu'il existe les balises `` ou `` qui servent à indiquer l'importance !
- `<div class="titre">` : idem, cela ne convient pas, puisqu'il existe des balises faites spécialement pour les titres (`<h1>` , `<h2>` ...).

Les unités : em - % - px - rem

```
p{  
    width:50%  
}
```



px : unité de mesure absolue
est définie pour être petite mais visible
c'est une taille fixe indépendante de l'élément parent
ex: une bordure de 1 pixel entoure ce texte

em : taille relative à la taille de la police de l'élément parent
taille de police de caractères large sur ordi
taille de police de caractères petite sur smartphone
=
taille en em proportionnelle
ex: $1.5\text{em} = 1.5 * \text{la police de la taille de la police de l'élément parent}$

rem: ressemble à l'unité **em**
unité de mesure basée sur la taille de police du html
unité constante dans tout le document
ex: $1.2\text{ rem} = 1.2 * \text{la taille de la police de la racine html}$



Les tailles de police de caractères sont définis en px, rem et em.

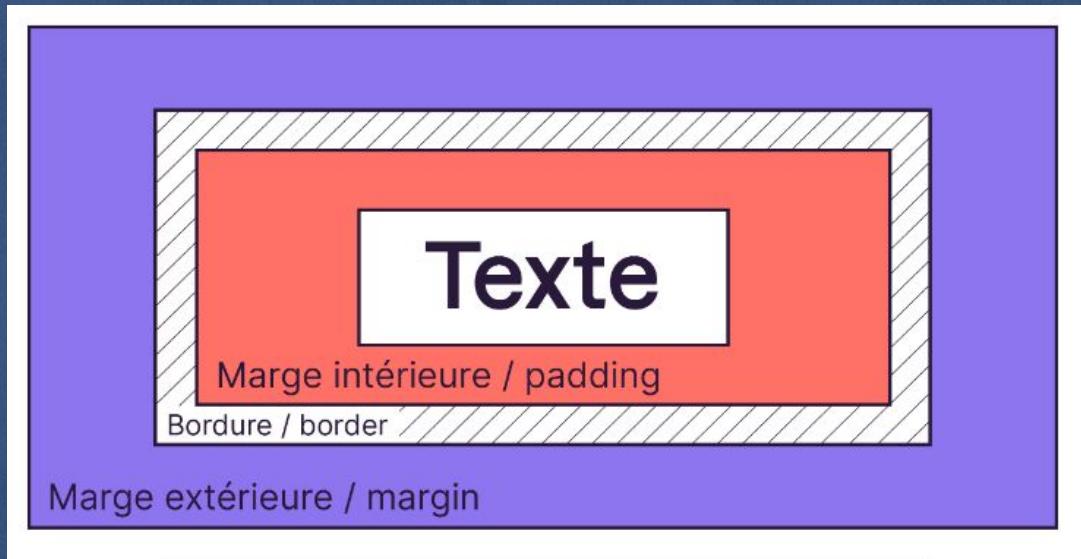
% : unité de mesure relative par rapport à un valeur de base qui est l'élément parent / pour un design adaptable au support

Définissez des marges avec *margin / padding*

2 propriétés pour modifier la marge

margin: taille de la marge extérieure

padding: taille de la marge intérieure



```
p{  
background-color: #f50404;  
margin:0; /* par defaut */  
padding:0; /* par defaut */  
}
```

L'héritage

Lorem, ipsum dolor sit amet consectetur adipisicing elit. Iure aliquid eaque accusamus recusandae reiciendis exercitationem nisi atque repellat. Assumenda rerum autem corrupti ullam praesentium, amet repudiandae sed eius officiis saepe?

titres et paragraphes

```
p{  
background-color: #f50404;  
margin-top:50px;  
margin-right: 30px;  
margin-bottom:60px;  
margin-left:15px;  
padding:0; /* par defaut */  
}
```

L'héritage

50px

15px
30px
60px

Lorem, ipsum dolor sit amet consectetur adipisicing elit. Iure aliquid eaque accusamus recusandae reiciendis exercitationem nisi atque repellat. Assumenda rerum autem corrupti ullam praesentium, amet repudiandae sed eius officiis saepe?

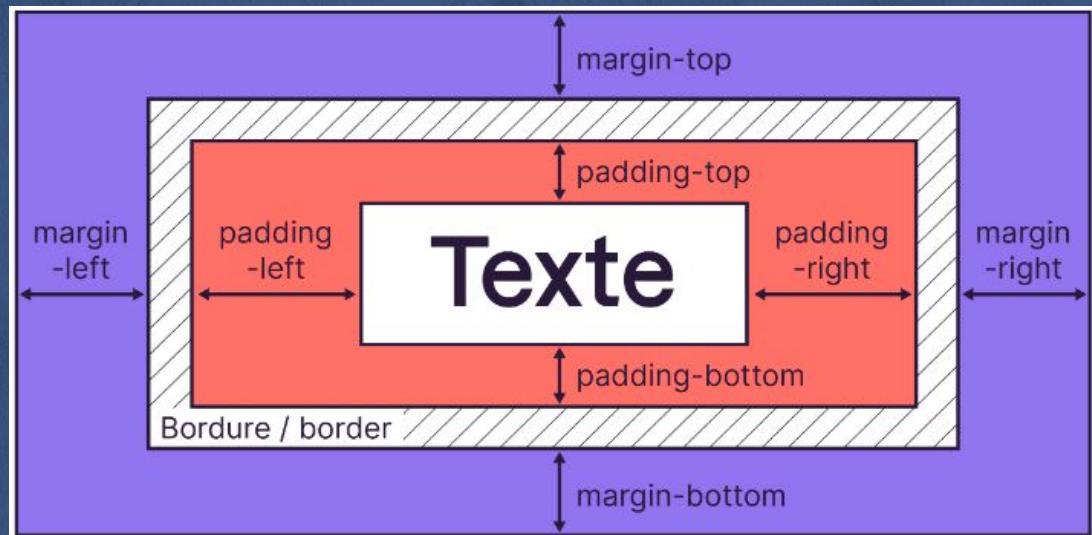
titres et paragraphes

p{
}
}

```
background-color: #f50404;  
margin:0px; /* par default */  
padding-top:15px;  
padding-right: 30px;  
padding-bottom:60px;  
padding-left:40px;
```

L'héritage

↓
Lorem, ipsum dolor sit amet
consectetur adipisicing elit. Iure
aliquid eaque accusamus recusanda
reiciendis exercitationem nisi atque
repellat. Assumenda rerum autem
corrupti ullam praesentium, amet
repudiandae sed eius officiis saepe?



titres et paragraphes

On peut condenser (refactoriser) autant pour le margin que pour le padding:

```
p{  
background-color: #f50404;  
margin-top:50px;  
margin-right: 30px;  
margin-bottom:60px;  
margin-left:15px;  
padding:0; /* par defaut */  
}
```

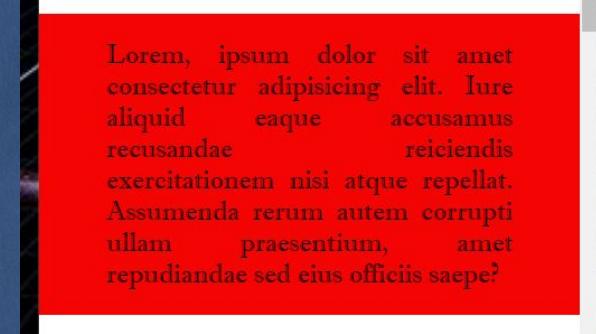
```
p{  
background-color: #f50404;  
margin:0px; /* par defaut */  
padding:15px 30px 60px 40px;  
}
```

haut droite bas gauche
top right bottom left



Et si les valeurs sont les mêmes en haut et bas / gauche et droite

```
p{  
background-color: #f50404;  
margin:0px;  
text-align: justify;  
padding:15px 40px;  
}
```



si les valeurs sont les mêmes à droite et à gauche

```
p{  
background-color: #f50404;  
margin:0px;  
text-align: justify;  
padding:5px 40px 70px;  
}
```

padding-right
&
padding-left: 40px;

padding-top: 5px;

↓

Lore*m*, ipsum dolor sit amet
consectetur adipisic*ing* elit. Iure
aliquid ea*que* accusamus
recusandae reiciendis
exercitationem nisi atque repellat.
Assumenda rerum autem corrupti
ullam praesentium, amet
repudiandae sed eius officiis saepe?

| padding-bottom: 70px;



la propriété *border* fonctionne pareil

Pour centrer des blocs:

on définit une width



le contenu va se centrer automatiquement

on indique margin: auto;

```
p{  
background-color: #f50404;  
padding:10px;  
width:150px;  
margin:auto;  
}
```



Mise en page avec Flexbox

Il faut: un conteneur ET des éléments dedans

```
<!-- section flexbox -->
<section class="container">
  <div class="element1"></div>
  <div class="element2"></div>
  <div class="element3"></div>
</section>

177 .element1{
178   background-color: #rgb(214, 241, 60);
179   width:50px;
180   height: 50px;
181 }
182 .element2{
183   background-color: #rgb(192, 8, 38);
184   width:50px;
185   height: 50px;;
186 }
187 .element3{
188   background-color: #rgb(187, 9, 211);
189   width:50px;
190   height: 50px;;
191 }
```



La propriété *display* avec flex

```
.container{  
    display:flex;  
}
```



Pour agencer ces éléments : *flex-direction*

cela permet de positionner les éléments de façon vertical / horizontal

4 valeurs sont acceptées:

row (par défaut, en ligne)

column (en colonne)

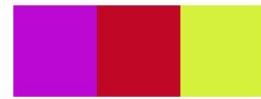
row-reverse (organisé sur une ligne mais en ordre inversé)

column-reverse (organisé sur une colonne mais en ordre inversé)

```
.container{  
  display:flex;  
  flex-direction: row;  
}
```



```
.container{  
  display:flex;  
  flex-direction: row-reverse;  
}
```



```
.container{  
  display:flex;  
  flex-direction: column;  
}
```



```
.container{  
  display:flex;  
  flex-direction: column-reverse;  
}
```



Retournez à la ligne avec *flex-wrap*

Par défaut, les **blocs** vont rester sur la ligne et s'écraser les uns contre les autres

=> on peut demander à ce que les éléments reviennent à la ligne quand plus de place

3 valeurs sont acceptées : **nowrap** (par défaut, les éléments se serrent pour tenir sur la ligne - par défaut)

wrap (les éléments passent à la ligne dès qu'il n'y a plus de place)

wrap-reverse (les éléments passent à la ligne mais à l'envers)

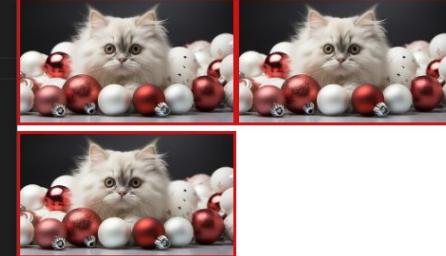
```
<section class="imgs">
  <div class="img1">
    
  </div>
  <div class="img2">
    
  </div>
  <div class="img3">
    
  </div>
</section>
```

```
177
178
179 .imgs{
180   display: flex;
181 }
182
183 .img1 img,
184 .img2 img,
185 .img3 img{
186   width:150px;
187   height: auto;
188   border: 2px solid #d11212;
189 }
190
191
```



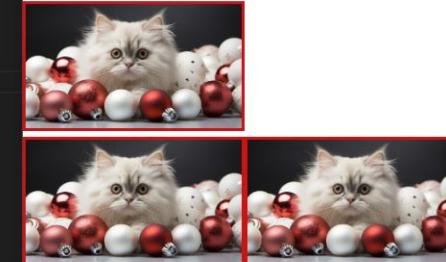
```
<section class="imgs">
  <div class="img1">
    
  </div>
  <div class="img2">
    
  </div>
  <div class="img3">
    
  </div>
</section>
```

```
177
178
179 .imgs{
180   display: flex;
181   flex-wrap: wrap;
182 }
183
184 .img1 img,
185 .img2 img,
186 .img3 img{
187   width:150px;
188   height: auto;
189   border: 2px solid #d11212;
190 }
191
192
```



```
<section class="imgs">
  <div class="img1">
    
  </div>
  <div class="img2">
    
  </div>
  <div class="img3">
    
  </div>
</section>
```

```
177
178
179 .imgs{
180   display: flex;
181   flex-wrap:wrap-reverse;
182 }
183
184 .img1 img,
185 .img2 img,
186 .img3 img{
187   width:150px;
188   height: auto;
189   border: 2px solid #d11212;
190 }
191
192
```



L'axe horizontal et l'axe vertical

Les éléments sont organisées de manière horizontale OU de manière verticale

Selon le choix que vous faites, cela deviendra *l'axe principal*
et l'axe restant: *l'axe secondaire*

Aligner les éléments sur l'axe principal

=> Aligner les éléments sur l'axe principal avec la propriété *justify-content*

Elle accepte 5 valeurs: *flex-start* (alignés au début - par défaut)

flex-end (alignés à la fin)

center (alignés au centre)

space-between (les éléments sont étirés sur tout l'axe avec de l'espace entre eux)

space-around (les éléments sont étirés sur tout l'axe mais il y a de l'espace aussi
sur les extrémités)

Aligner sur l'axe secondaire

=> Aligner les éléments sur l'axe secondaire avec la propriété *align-items*

Elle accepte 5 valeurs: stretch (les éléments sont étirés sur tout l'axe - par défaut)

flex-start (alignés au début)

flex-end (alignés à la fin)

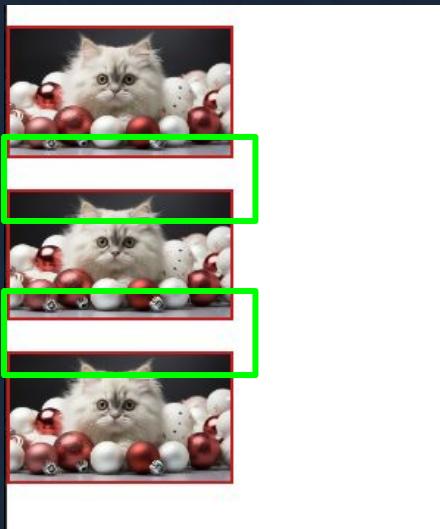
center (alignés au centre)

baseline (alignés sur la ligne de base (semblable à flex-start)

Si on veut mettre un espace entre ces éléments: c'est la propriété *gap*



column-gap: 1rem;



row-gap: 1rem;

```
.imgs{  
    display: flex;  
    flex-direction: column;  
    flex-wrap: wrap;  
    gap: 1rem;  
}
```

Cas pratique:

<https://codepen.io/Aurore-Ramos/pen/bGzmJbJ>

Encore plus simple, pour centrer horizontalement et verticalement avec un flex-box

```
/* header */
header{
    display: flex;
    background-color: #9bdffa;
}

nav {
    margin: auto;
}
```



[Accueil](#) [Boutique](#) [A propos de moi](#) [Contact](#)

La propriété Display

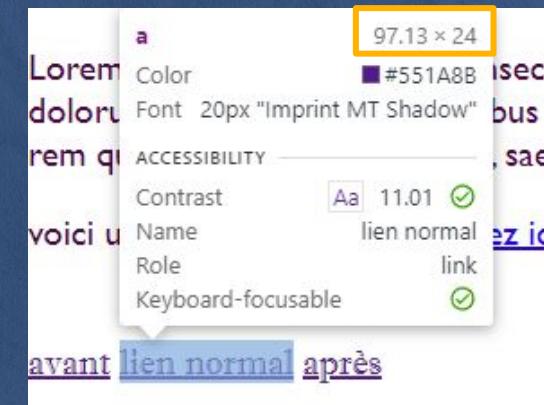
Permet de modifier le type de n'importe quel élément
ex: des liens (type:inline) en type block

→ il devient possible de modifier leurs dimensions

```
<a href="">avant</a>
<a class="changeType" href="#">lien en type
block via son display</a>
<a href="#">après</a>
```

```
171
172
173 .changeType{
174   display: block;
175 }
```

```
<a href="">avant</a>
<a href="#">lien normal</a>
<a href="#">après</a>
```



La propriété Display: none;

Si on veut masquer un élément de la page,
on lui attribue la propriété display avec la valeur **none**

ex: très pratique lorsqu'il faut cacher des éléments suivant la taille de l'écran

Il est possible de combiner inline et block pour tirer avantage des deux.
ex: display: inline-block;

Il existe de très nombreuses valeurs de display:
<https://developer.mozilla.org/fr/docs/Web/CSS/display>

Les positions des éléments

Jusqu'à maintenant, les éléments ne se superposent jamais :
ils sont dans le flux de la page

On peut sortir ces éléments du flux de la page avec la propriété *position* :

positionne avec précision des éléments dans la page

s'utilise sur des balises type inline et block mais la plupart du temps c'est sur des block
accepte 5 valeurs :

relative décale l'élément par rapport à sa position normale

absolu place un élément n'importe où dans la page

fixed / sticky (comportement semblables) un élément reste toujours à l'écran

La propriété *position* permet d'accéder à 4 propriétés **left, right, top, bottom**

Position relative

Permet d'effectuer des ajustements: l'élément est décalé par rapport à sa position initiale

voici un lien qui se trouve au milieu de la phrase et dont la position peut etre déplacée.

```
<p class="testPosition">  
    voici un lien qui se trouve<a href="page2.html"> au milieu  
    de la phrase</a> et dont la position peut etre déplacée.  
</p>
```

```
204  
205 .testPosition a {  
206     background-color: #rgb(231, 245, 37);  
207     position: relative;  
208     top: 10px;  
209     left: 10px;  
210 }
```

voici un lien qui se trouve au milieu de la phrase et dont la position peut etre déplacée.

Position absolute

Permet de placer un élément n'importe où dans la page

voici un lien qui se trouve au milieu de la phrase et dont la position peut être déplacée.

```
<p class="testPosition">  
    voici un lien qui se trouve<a href="page2.html"> au milieu  
    de la phrase</a> et dont la position peut être déplacée.  
</p>
```

```
204  
205     .testPosition a {  
206         background-color: #rgb(231, 245, 37);  
207         position: absolute;  
208         top: 100px;  
209         left: 10px;  
210     }
```



L'héritage

au milieu de la phrase

Lorem ipsum dolor sit amet consectetur
eius officiis saepe?

titres et paragraphes

Lorem ipsum dolor, sit amet consectetur
similique sequi



L'élément en position absolute va se positionner par rapport à la page

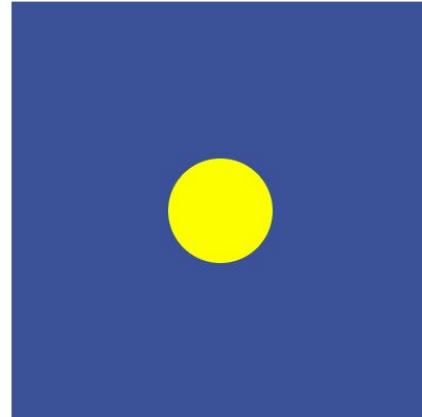
SAUF

si un autre l'élément est en **position relative**. Dans ce cas, il va se positionner par rapport à cet élément avec son *top*, *bottom*, *left*, *right*

```
<div class="testAbsolute">
  <div class="square">
    <div class="round"></div>
  </div>
</div>

207 .testAbsolute {
208   display: flex;
209   justify-content: center;
210   align-items: center;
211 }
212
213 .square{
214   background-color: #335399;
215   width:400px;
216   height: 400px;
217   position: relative;
218 }
219
220 .round{
221   background-color: yellow;
222   width: 100px;
223   height: 100px;
224   border-radius: 50%;
225   position: absolute;
226   top: 50%;
227   left: 50%;
228   transform: translate(-50%, -50%);
229 }
```

Doloribus modi quas consequuntur accusantium quos veniam distinctio error hic iste voluptates nihil dolore maiores deleniti dignissimos nesciunt in velit dolor, ullam rem qui nam accusamus mollitia, saepe debitisi! Vero.



Position Fixed

L'élément reste fixe à sa position spécifiée par rapport à la fenêtre du navigateur

souvent utilisé pour le header ou un icône de tel, whatsapp...
reste en permanence affiché sur l'écran



L'héritage

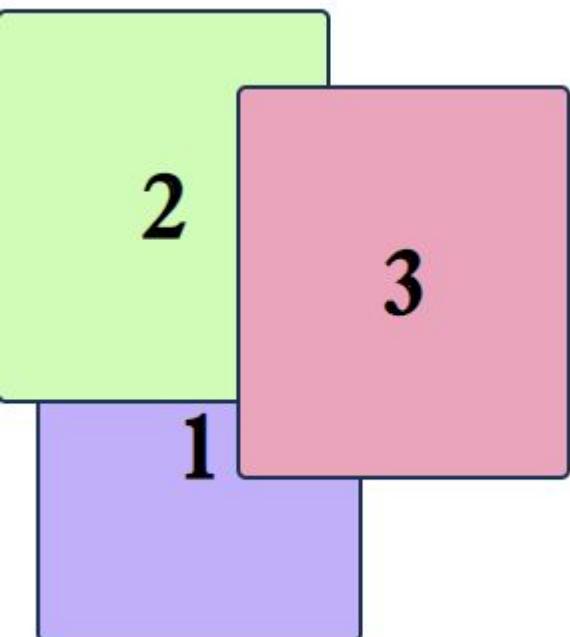
Ipsum, ipsum dolor sit amet consectetur adipisicing elit.
Iure aliquid eaque accusamus recusandae reiciendis



```
<header>
  
  <nav>
    <a href="#">Accueil</a>
    <a href="#">Boutique</a>
    <a href="#">A propos de moi</a>
    <a href="#">Contact</a>
  </nav>
</header>
<main>
```

```
13  /* header */
14  header{
15    display: flex;
16    justify-content: space-between;
17    align-items: center;
18    background-color: #9bdffa;
19    position: fixed;
20    top:0;
21    left:0;
22    right:0;
23    padding:0 15px;
24    z-index:10;
25  }
```

Prendre le dessus ou le dessous avec z-index



```
<div class="contener">
  <div class="box one">1</div>
  <div class="box two">2</div>
  <div class="box three">3</div>
</div>
```

```
.contener {
  position: relative;
  height: 300px;
  width: 300px;
  bottom: 100px;
}

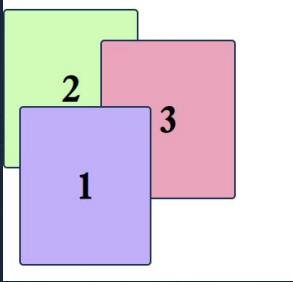
.one {
  background-color: #C2B0F9;
  left: 20px;
  top: 150px;
}

.two {
  background-color: #CBFCB9;
  left: 0px;
  top: 30px;
}

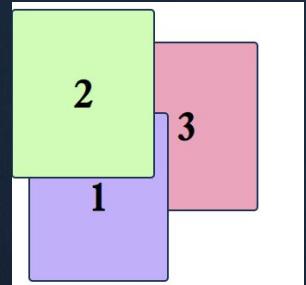
.three {
  background-color: #F2A3BB;
  left: 120px;
  bottom: 35px;
}

.box{
  position: absolute;
  border: 2px solid #123456;
  padding:70px;
  border-radius: 5px;
  font-size:45px;
  font-weight: 700;
}
```

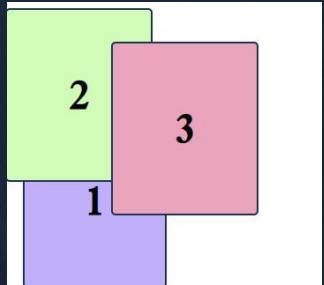
La propriété *z-index* permet de définir qui doit être placé devant avec une valeur plus élevée prendre le dessus



```
.one {  
    background-color: #C2B0F9;  
    left: 20px;;  
    top: 150px;  
    z-index:3;  
}  
  
.two {  
    background-color: #CBFCB9;  
    left: 0px;  
    top: 30px;  
}  
  
.three {  
    background-color: #F2A3BB;  
    left: 120px;  
    bottom: 35px;  
}
```



```
.one {  
    background-color: #C2B0F9;  
    left: 20px;;  
    top: 150px;  
    z-index:3;  
}  
  
.two {  
    background-color: #CBFCB9;  
    left: 0px;  
    top: 30px;  
    z-index:5;  
}  
  
.three {  
    background-color: #F2A3BB;  
    left: 120px;  
    bottom: 35px;  
}
```



```
.one {  
    background-color: #C2B0F9;  
    left: 20px;;  
    top: 150px;  
    z-index:3;  
}  
  
.two {  
    background-color: #CBFCB9;  
    left: 0px;  
    top: 30px;  
    z-index:5;  
}  
  
.three {  
    background-color: #F2A3BB;  
    left: 120px;  
    bottom: 35px;  
    z-index:8;  
}
```

Le Responsive design avec les media queries

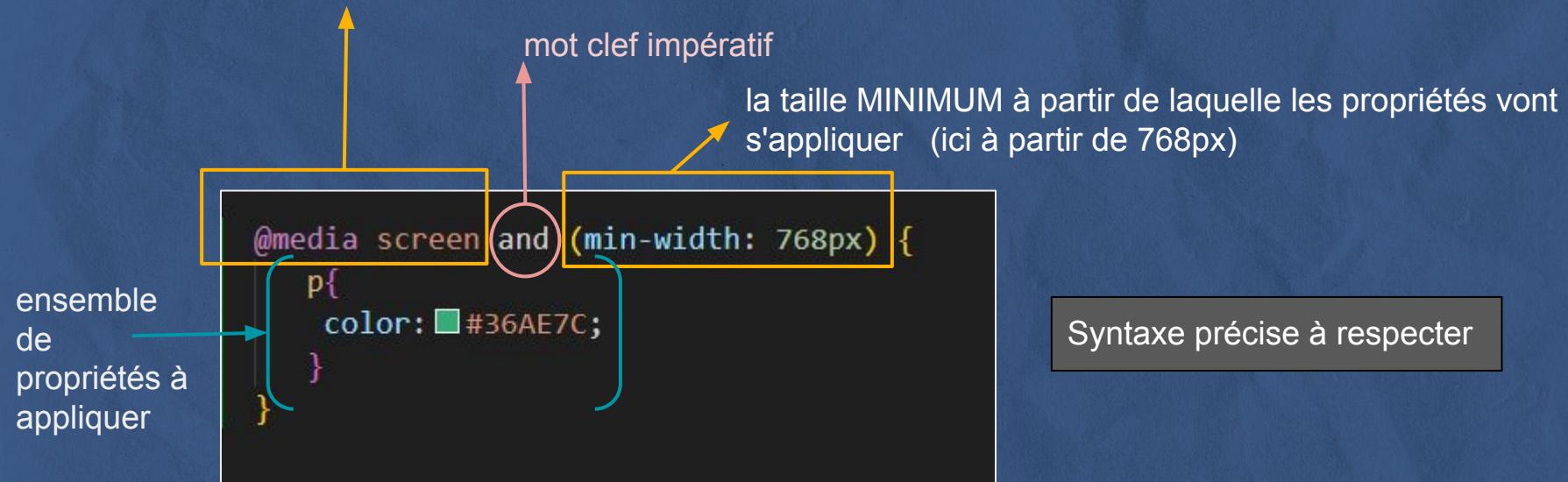
Les media queries sont des règles qui indiquent quand on doit appliquer des propriétés en CSS, en fonction de la taille de l'écran sur lequel est affiché le site web

=> on indique au navigateur comment se comporter en fonction de la taille de l'écran



Syntaxe d'une media query

type de media - le plus souvent *screen*



! @media all and (min-width: 481px) and (max-width: 768px){
 @media print and (max-width: 1279px){

Ces tailles sont appelés des *breakpoints* ou *point de rupture*
=> car c est à partir de ces points que le design change



Pour que ces média queries soit pris en compte :

il faut que `<meta name="viewport" content="width=device-width, initial-scale=1.0">`

```
@media screen and (min-width: 768px) {  
    p{  
        color: #36AE7C;  
    }  
}
```



L'héritage

Lorem, ipsum dolor sit amet consectetur adipisicing elit. Iure aliquid eaque accusamus recusandae reiciendis exercitationem nisi atque repellat. Assumenda rerum autem corrupti ullam praesentium, amet repudiandae sed eius officiis saepe?

titres et paragraphes

Lorem ipsum dolor, sit amet consectetur adipisicing elit. Tempore, veritatis dolores quod labore corrupti laboriosam ullam dignissimos suscipit quos ipsam! Ducimus tempore aperiam veniam totam culpa, inventore iste similique eaque!



Lorem ipsum dolor sit amet consectetur adipisicing elit. Veniam quas preferendis ipsa? Delectus nemo, repellendus ipsam corrupti eaque aspernatur sit sapiente, sunt nesciunt ipsum tempora maxime, nulla vel cum repudiandae?



L'héritage

Lorem, ipsum dolor sit amet consectetur adipisicing elit. Iure aliquid eaque accusamus recusandae reiciendis exercitationem nisi atque repellat. Assumenda rerum autem corrupti ullam praesentium, amet repudiandae sed eius officiis saepe?

titres et paragraphes

Lorem ipsum dolor, sit amet consectetur adipisicing elit. Tempore, veritatis dolores quod labore corrupti laboriosam ullam dignissimos suscipit quos ipsam! Ducimus tempore aperiam veniam totam culpa, inventore iste similique eaque!



Lorem ipsum dolor sit amet consectetur adipisicing elit. Veniam quas preferendis ipsa? Delectus nemo, repellendus ipsam corrupti eaque aspernatur sit sapiente, sunt nesciunt ipsum tempora maxime, nulla vel cum repudiandae?

Les minimum et Maximum

Très pratique car cela permet de définir des limites à ne pas dépasser:

min-width: largeur minimale;

max-width: largeur maximale;

min-height: hauteur minimale;

max-height: hauteur maximale;

```
main{  
    max-width:900px;  
    margin:auto;  
}
```

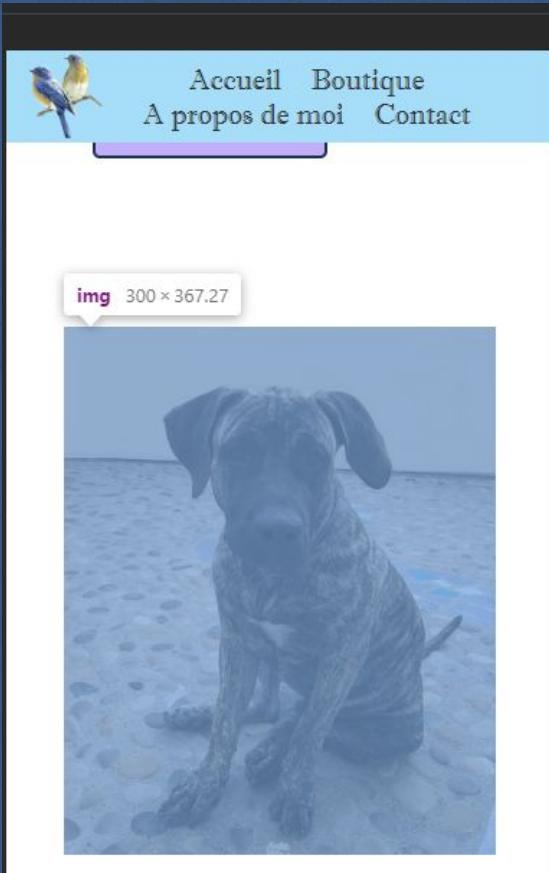
L'héritage

 Lorem, ipsum dolor sit amet consectetur adipisicing elit. Iure aliquid eaque accusamus recusandae reiciendis exercitationem nisi atque repellat. Assumenda rerum autem corrupti ullam praesentium, amet repudiandae sed eius officiis saepe?

titres et paragraphes

 Lorem ipsum dolor, sit amet consectetur adipisicing elit. Tempore, veritatis dolores quod labore corrupti laboriosam ullam dignissimos suscipit quos ipsam! Ducimus tempore aperiam veniam totam culpa,

Essayez avec une image



Filtrer

```
<html lang="fr">
  > <head> ... </head>
  > <body>
    > <header> ... </header> flex
    > <main>
      <!-- l héritage des styles -->
      > <section> ... </section>
        <!-- section cat-->
      > <section class="cat"> ... </section> flex
        <!-- form
        > <section> ...
          <!-- section
        > <div class=" ...
          <!-- section
        > <section class=" ...
          <!-- table
        > <table> ...
          <!-- section
        > <section> ...
          <!-- position
        > <div class=" ...
          <!-- min e
        > <section class=" ...
          
        </section>
      </main>
    > <footer> ... </footer> flex
  > element.style {
    .dogUlk img {
      width: 60%;
      min-width: 300px;
    }
  }
```

A small image of a dog, likely a Great Dane, sitting on a surface. This image is part of the HTML structure shown in the developer tools.



[Accueil](#) [Boutique](#) [A propos de moi](#) [Contact](#)



img 477 × 583.95

Coupez ce qui dépasse avec la propriété *overflow*

Si vous avez un paragraphe qui est trop long par rapport au cadre que vous souhaitez, cela donnera ça:

```
<section class="overflow">
  <p>
    Lorem, ipsum dolor sit amet
    consectetur adipisicing elit. Soluta
    in cupiditate adipisci! Accusamus,
    quae nam! Error ab molestias fuga
    necessitatibus, sunt voluptatum id
    voluptates eum pariatur officiis iste
    expedita consequatur?
    Quidem vitae dolore, repellendus
    voluptatem ea obcaecati iure dolores
    maiores animi officia, nulla, maxime
    culpa temporibus architecto veniam
    voluptate nostrum minima saepe aperiam
    necessitatibus ut vero accusantium.
    Omnis, esse commodi.
  </p>
</section>
```

```
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249

.overflow p{
  border:2px solid red;
  width:300px;
  margin: auto;
  height:200px;
}
```

>Lorem, ipsum dolor sit amet
consectetur adipisicing elit. Soluta
in cupiditate adipisci! Accusamus,
quae nam! Error ab molestias fuga
necessitatibus, sunt voluptatum id
voluptates eum pariatur officiis iste
expedita consequatur? Quidem
vitae dolore, repellendus
voluptatem ea obcaecati iure
dolores maiores animi officia, nulla,
maxime culpa temporibus
architecto veniam voluptate
nostrum minima saepe aperiam
necessitatibus ut vero accusantium.
Omnis, esse commodi.

La propriété *overflow* accepte 4 valeurs

- **visible**: si le texte dépasse des limites du paragraphe, il reste visible et sort volontairement du paragraphe.
- **hidden**: si le texte sort des limites, il sera caché.
- **scroll**: là encore le texte sera caché mais le navigateur va mettre en place un scroll vertical (barre de défilement) pour qu'on puisse lire l'ensemble.
- **auto**: c'est le mode “pilote automatique”. C'est le navigateur qui décide de mettre ou non les barres de défilement.

```
.overflow p{  
    border:2px solid red;  
    width:300px;  
    margin: auto;  
    height:200px;  
    overflow: auto;  
}
```

