



HTML/CSS

SANA EL ATCHIA

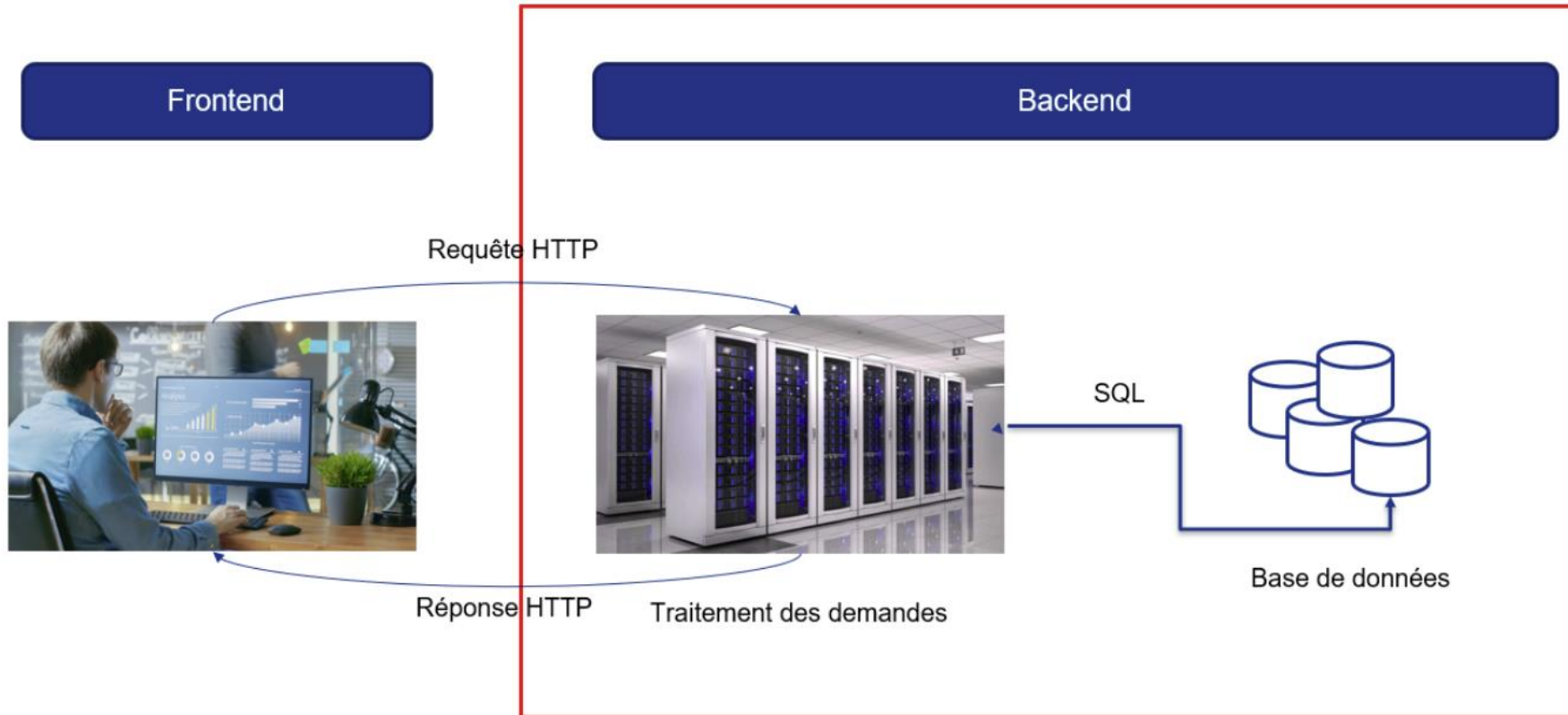


Comment réussir?

- La pratique
- Prendre des notes
- Faire les exercices
- Poser des questions !!



Architecture client-serveur

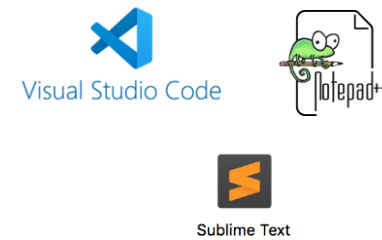


Outils de développement

Navigateur



Editeur



HTML vs CSS

HTML et **CSS** sont des langages informatiques qui permettent de créer des sites web.

Le rôle principal d'HTML est d'écrire le contenu : titre, sous-titre, formulaire, champ, citation, ...

Le rôle du CSS est de mettre en forme ce contenu : couleur, image de fond, police de caractère, bordures, ...



HTML



HTML + CSS

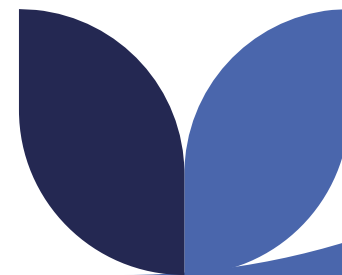
Introduction au langage HTML

- HTML (Hypertext Markup Language) est le langage de balisage conçu pour représenter les pages web.
- HTML est initialement dérivé du **SGML** (Standard Generalized Markup Language) dont la première publication date de 1986.
- C'est un langage composé des balises et des attributs



Les balises

- Une page HTML est composée d'un ensemble de balises qui sont invisibles par les visiteurs
- Les balises sont limités par des chevrons « < » « > »
- Le rôle d'une balise est d'indiquer la nature du texte qu'elle encadre
- On distingue deux types de balises :
 - Les Balises en paires** `<balise></balise>` : elles s'ouvrent et se ferment pour délimiter le contenu
 - Les balises orphelines** `<balise />` : elles permettent d'insérer un élément à un endroit précis



Les attributs

- Les attributs sont les options des balises
- Ils viennent de compléter les balises pour donner des informations supplémentaires
- L'attribut se place après le nom de la balise ouvrante et a le plus souvent une valeur, comme ceci :

```
<balise attribut="valeur">
```


Le Doctype avant tout

- Le **doctype** est le préambule d'une page HTML, il spécifie le type de document.
- Doctype HTML 4.01

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"  
"http://www.w3.org/TR/html4/strict.dtd">
```

- Doctype XHTML 1.1

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"  
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
```

- Doctype HTML 5

```
<!DOCTYPE html>
```



Structure de base d'une page HTML

Une page web est constituée de deux sections principales : un en-tête (`<head>`) et un corps (`<body>`).

```
<!DOCTYPE html>
<html>
  <head>
    <!-- En-tête de la page -->
    <meta charset="utf-8" />
    <title>Titre</title>
  </head>

  <body>
    <!-- Corps de la page -->
  </body>
</html>
```

Les commentaires

- Un **commentaire** en HTML est un texte qui ne change rien à l'affichage de la page.
- Un commentaire donne des indications sur le fonctionnement d'une page.
- Un commentaire est une balise HTML avec une forme bien spéciale :

```
<!-- Ceci est un commentaire -->
```

Comment rédiger des paragraphes?

- L'HTML propose la balise `<p></p>` pour organiser le texte en paragraphes

```
<!DOCTYPE html>
<html >
<head>
  <!-- En-tête de la page -->
  <meta charset="UTF-8">
  <title>Document</title>
</head>
<body>
  <!-- Corps de la page -->
  <p>Nous sommes un centre de formation.
  On dispose plusieurs prestations</p>
</body>
</html>
```

Sauter une ligne

- La balise **orpheline** `
` permet d'aller à la ligne suivante

```
<!DOCTYPE html>
<html >
<head>
  <!-- En-tête de la page -->
  <meta charset="UTF-8">
  <title>Document</title>
</head>
<body>
  <!-- Corps de la page -->
  <p>Nous sommes un centre de formation.<br />
  On dispose plusieurs prestations</p>
</body>
</html>
```

Comment structurer une page avec les titres

- Il y a six niveaux de titres :

`<h1></h1>`

Titre super important

`<h2></h2>`

Titre important

`<h3></h3>`

Titre un peu moins important (sous-titre)

`<h4></h4>`

Titre pas trop important

`<h5></h5>`

Titre pas important

`<h6></h6>`

Titre vraiment pas important du tout

Les titres qui s'affichent dans la page web.

Remarque :

La balise `<title>` affiche le titre de la page dans la barre de titre du navigateur

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>Niveaux de titres</title>
  </head>

  <body>
    <h1>Titre super important</h1>
    <h2>Titre important</h2>
    <h3>Titre un peu moins important (sous-titre)</h3>

    <h4>Titre pas trop important</h4>
    <h5>Titre pas important</h5>
    <h6>Titre vraiment pas important du tout</h6>
  </body>
</html>
```

exemple

```
<body>
  <h1>Bienvenue à LA GRNADE CLASSE !</h1>

  <p>Nous sommes un centre de formation.<br />
  On dispose plusieurs prestations</p>

  <h2>Des formations </h2>

  <p>il y a des formations dans le domaine de l'informatique comme web designer et developpeur web</p>

  <p>Il y a aussi des formations fle et anglais</p>

  <h2>Petite enfance</h2>

  <p>Nous gardons des enfants entre 3ans et 6 ans<br />
  Rendez-vous sur le site ! </p>
</body>
```

Mettre en valeur le texte

comment donner de l'importance à certains mots du texte?

HTML vous propose différents moyens de mettre en valeur le texte de votre page.

- ``
- ``
- `<mark></mark>`

Les listes

Les listes nous permettent souvent de mieux structurer notre texte et d'ordonner nos informations.

Nous allons découvrir ici deux types de listes :

- ❑ Les listes non ordonnées ou listes à puces ;
- ❑ Les listes ordonnées ou listes numérotées, ou encore énumérations.



Liste non ordonnée

C'est un système qui nous permet de créer une liste d'éléments sans notion d'ordre.

```
<ul>  
  <li>web designer</li>  
  <li>developpeur web</li>  
  <li>fle</li>  
  <li>Anglais</li>  
</ul>
```

- web designer
- developpeur web
- fle
- Anglais

Liste ordonnée

Une liste ordonnée fonctionne de la même façon, seule une balise change : il faut remplacer `` par ``

```
<ol>
  <li>web designer</li>
  <li>developpeur web</li>
  <li>fle</li>
  <li>Anglais</li>
</ol>
```

1. web designer
2. developpeur web
3. fle
4. Anglais

Exercice

Ecrire en HTML, La liste de vos compétences dans le domaine informatique

- ☐ Ajoutez un titre de niveau 1.
- ☐ À la suite du titre, ajoutez un paragraphe indiquant "Voici mes compétences :".
- ☐ Ajoutez une liste ordonnée d'au moins cinq de vos compétences.
- ☐ Mettez en valeur la première compétence de la liste.

Liens

- Un lien est un texte sur lequel on peut cliquer pour se rendre sur une autre page
- Pour créer un lien, on utilise la balise `<a>` avec l'attribut `href`, pour indiquer l'adresse de la page cible
 - On peut faire un lien vers une autre page `page1.html`, en écrivant le nom du fichier

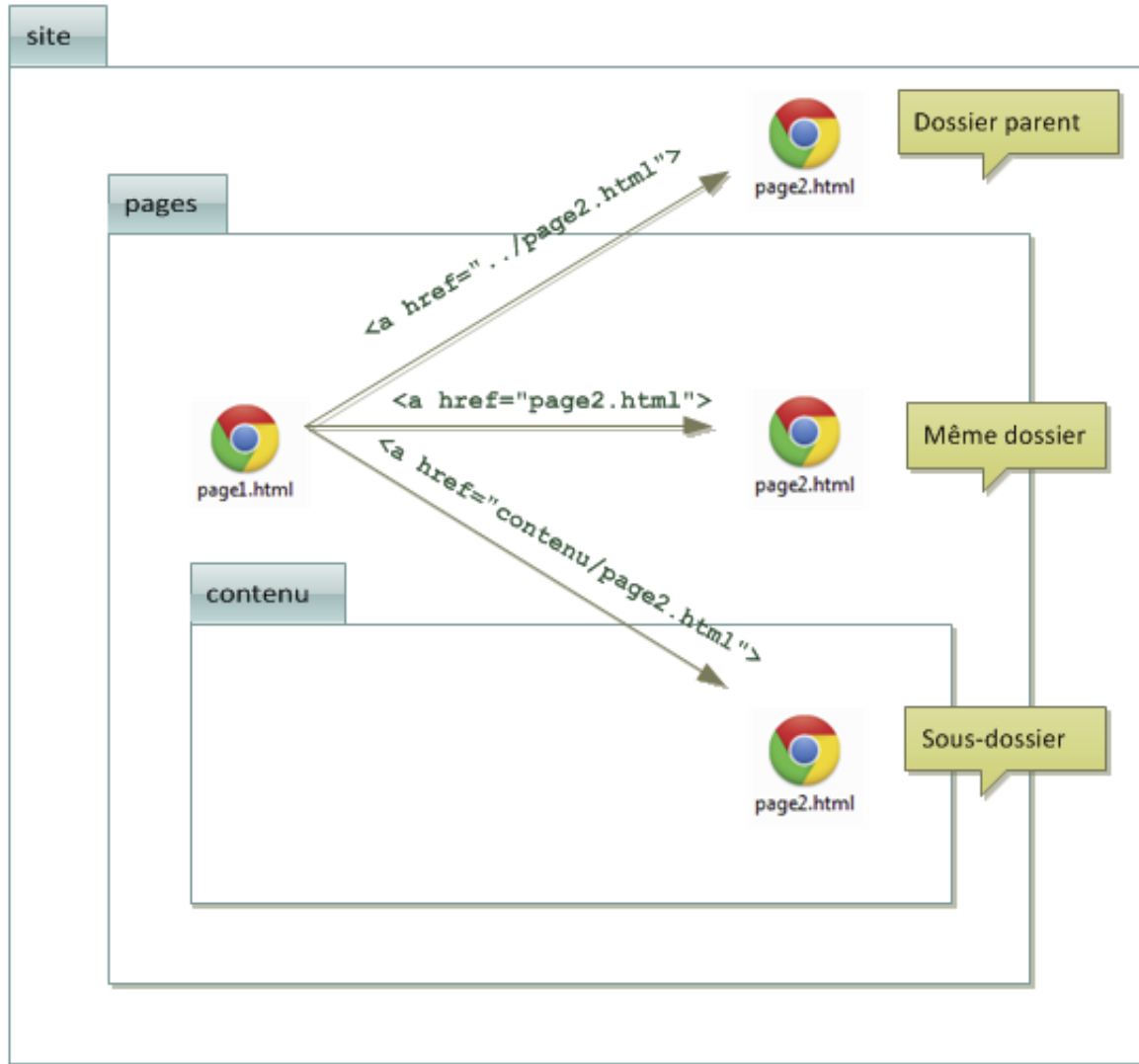
```
<a href="page1.html">la page 1</a>
```

- On peut faire un lien vers un site web

```
<a href="https://lagrandeclasse.fr/">Le site de LGC</a>
```



Liens



Ancre

- ❑ Le fonctionnement classique des liens hypertexte est de pointer vers une page différente de celle en cours de consultation,
- ❑ L'élément HTML `<a>` avec son attribut href, crée un lien hypertexte vers des pages web, des fichiers, des emplacements se trouvant dans la même page,..
- ❑ Ancre permet de créer un lien vers un endroit précis de la page courante

=> Pour créer un ancre, il suffit d'attribuer à l'élément vers lequel on veut pouvoir pointer un identifiant (avec l'attribut HTML id) et d'y associer un lien débutant par le caractère dièse #, suivi du nom de cet identifiant.

```
<div id="ancreID">test.....texte qui dépace la page</div>
```

```
<a href="#ancreID">haut de page</a>
```

Rmq : Avec la plupart des navigateurs, aucune ancre n'est nécessaire pour retourner en haut de page : le dièse # seul suffit à revenir au début du contenu.

```
<a href="#">haut de page</a>
```

Exercice - Ancre

Dans cet exercice on va attribuer des ancres à quelques endroits de la page et par la suite on va créer des liens qui pointent vers ces ancres afin de faciliter la navigation de l'utilisateur sur la page .

Pour cela créer une page composée de 4 chapitres précéder de 4 liens chaque lien envoie vers un chapitre précis , pour simplifier on va considérer que chaque chapitre est composé d'un titre et d'un paragraphe.



Image

- On insère une image avec la balise orpheline ``
- La balise doit être accompagnée de deux attributs obligatoires :
 - **src** : permet d'indiquer où se trouve l'image que l'on veut insérer,
 - **alt** : cela signifie « texte alternatif ». On doit toujours indiquer un texte alternatif à l'image, c'est-à-dire un court texte qui décrit ce que contient l'image.

```
<p>  
  Vous souhaitez voir l'image dans sa taille d'origine ? Cliquez dessus !<br />  
  <a href="logo-lgc.png">  
      
  </a>  
</p>
```



Figure

- Les figures se sont des éléments qui viennent enrichir le texte pour compléter les informations de la page,
- Les figures peuvent être de différents types :
 - images ;
 - codes source ;
 - citations ;
 - etc.
- La balise `<figure>` permet d'ajouter des figures
- Une figure est le plus souvent accompagnée d'une légende, Pour ajouter une légende, on utilise la balise `<figcaption>` à l'intérieur de la balise `<figure>`

```
<figure>  
    
  <figcaption>Logo de la grande classe</figcaption>  
</figure>
```



Exercice: création d'un CV

- Étape 1 : Créez une page HTML.
- Étape 2 : Ajoutez votre nom et votre prénom en titre principal.
- Étape 3 : Ajoutez une photo, sur laquelle on pourra cliquer pour avoir une version agrandie.
- Étape 4 : Ajoutez 3 sections avec un titre secondaire (Formations, Expériences, Compétences). Chaque section contient un paragraphe ou une liste à puce.



CSS



CSS

- C'est un langage qui vient pour compléter le HTML.
- CSS permet de gérer la mise en forme du site,
- Le HTML est né en 1991, et CSS en 1996

Où écrit-on le CSS ?

Il y a trois façon pour écrire le code en langage CSS:

- Dans un fichier .css
- Dans l'en-tête <head> du fichier HTML
- Directement dans les balises HTML via l'attribut style

Ajouter CSS dans un fichier .css

La balise `<link>` permet d'associer le fichier html au fichier appelé « style.css » chargé de la mise en forme

```
<head>  
  <meta charset="utf-8" />  
  <link rel="stylesheet" href="style.css" />  
  <title>Premiers tests du CSS</title>  
</head>
```

Ajouter CSS dans l'entête du fichier HTML

Insérer le code CSS directement dans la balise `<style>` à l'intérieur de l'entête `<head>` du fichier HTML

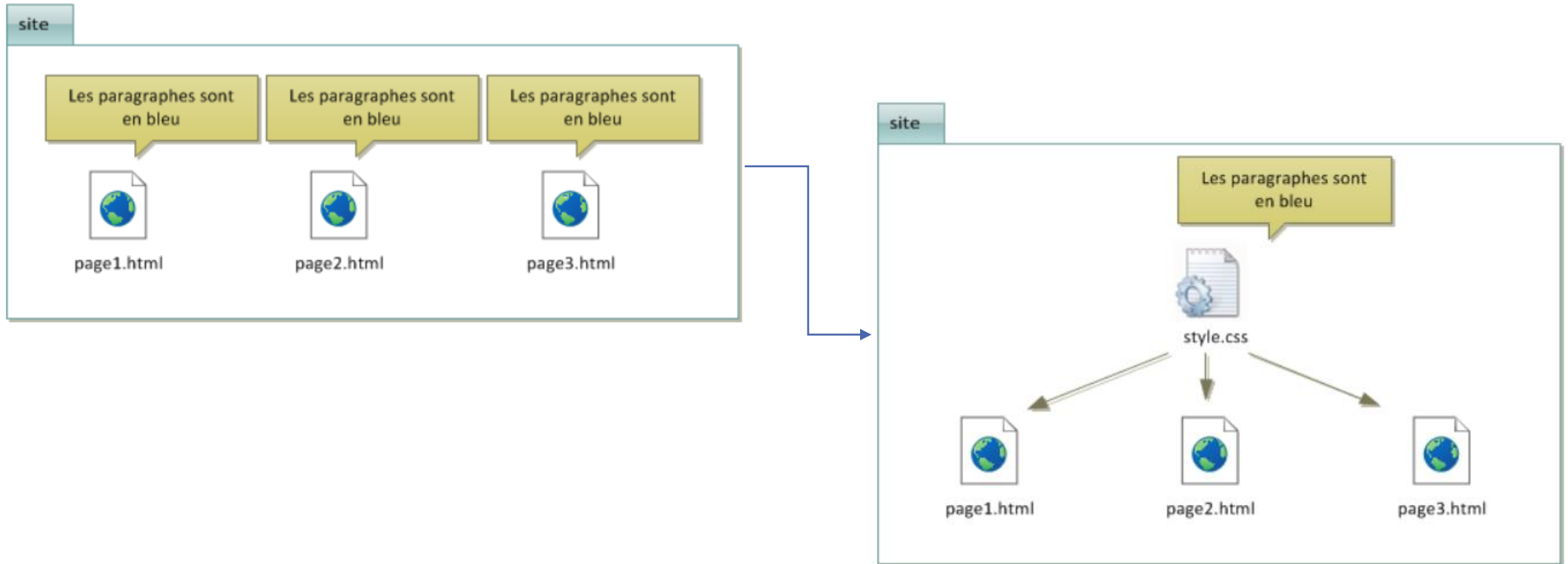
```
<head>
  <meta charset="utf-8" />
  <style>
    p
    {
      color: blue;
    }
  </style>
  <title>Premiers tests du CSS</title>
</head>
```


Ajouter CSS dans les balises HTML

Ajouter l'attribut **style** à n'importe quelle balise HTML afin d'intégrer CSS

```
<p style="color: blue;">Bonjour et bienvenue sur mon site !</p>
```

Pour appliquer CSS, il est recommandé de créer un fichier de style externe afin de l'utiliser dans des endroits différents



Appliquer un style

Un **sélecteur** : C'est ce qu'on utilise pour cibler les éléments de la page à modifier,

- Sélectionner une balise
- Class et id
- Les sélecteurs avancés

Appliquer un style

Sélectionner une balise

Un code CSS est composé de trois éléments :

- **Nom de balise** : On ajoute le nom de balise dont on veut modifier l'apparence.
- **Propriété CSS** : les « effets de style » de la page sont rangés dans des propriétés. (color, font-size,...)
- **Valeur** : pour chaque propriété CSS, on doit indiquer une valeur.

```
h1, em
{
    color: blue;
}
```

```
h1
{
    color: blue;
}

em
{
    color: blue;
}
```

```
balise1
{
    propriete1: valeur1;
    propriete2: valeur2;
    propriete3: valeur3;
}
```

```
balise2
{
    propriete1: valeur1;
    propriete2: valeur2;
    propriete3: valeur3;
    propriete4: valeur4;
}
```

```
balise3
{
    propriete1: valeur1;
```

Appliquer un style

Class et id

- On peut utiliser les attributs **class** et/ou **id** sur toutes les balises
- La valeur de la class/id est un nom qui sert à identifier la balise

- **Attribut class** : on peut l'utiliser sur plusieurs balises

- Html : `<h1 class="coloration">Bienvenue à LA GRNADE CLASSE !</h1>`

- CSS :

```
.coloration{  
  color: blue;  
}
```

- **Attribut id** : le nom de l'id est unique dans toute la page html

- Html : `<h1 id="coloration">Bienvenue à LA GRNADE CLASSE !</h1>`

- CSS :

```
#coloration{  
  color: blue;  
}
```

C/C : Deux balises peuvent avoir le même **nom** avec l'attribut **class**,

Le **nom** d'**id** être unique dans une page HTML

Balises HTML universelles

- ❖ ` ` : C'est une balise de type **inline**, c'est-à-dire une balise que l'on place au sein d'un paragraphe de texte pour sélectionner certains mots uniquement.
- ❖ `<div> </div>` :
 - C'est une balise de type **block**, qui entoure un bloc de texte.
 - On l'utilise souvent pour designer la page,

On peut utiliser ces 2 balises pour faire le style à un texte qui n'est pas entouré par l'une une balise :

```
<h1 >Bienvenue à <span id="coloration">LA GRNADE CLASSE</span> !</h1>
```



Appliquer un style

Les sélecteurs avancés

- ❖ **Sélecteur universel *** : permet de sélectionner toutes les balises sans exception

```
*  
{  
}
```

- ❖ **A B** Une balise contenue dans une autre :

```
h3 em  
{  
}
```

```
<h3>Titre avec <em>texte important</em></h3>
```

- ❖ **A + B** une balise qui en suit une autre : Sélectionner la première balise B situé après A

```
h3 + p  
{  
}
```

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

- ❖ **A[attribut]** : une balise qui possède un attribut

- ❖ **A[attribut="Valeur"]** : une balise, un attribut et une valeur exacte

Mise en forme Texte

- **Taille**
- **Police**
- **Alignement**

Mise en forme Texte - Taille

On modifie la taille du texte avec la propriété CSS **font-size** :

- **Taille absolue** : méthode précise (px, cm, mm, ...)
- **Taille relative** : Méthode souple (en pourcentage, em, ex, ...)
 - **xx-small** : minuscule;
 - **x-small** : très petit;
 - **small** : petit;
 - **large** : grand;
 - **x-large** : très grand;

```
font-size: small;
```

Mise en forme Texte - **Police**

On change la police du texte avec **font-family**

- Arial ;
- Arial Black ;
- Comic Sans MS ;
- Courier New ;
- Georgia ;
- Impact ;
- Times New Roman ;
- Trebuchet MS ;
- Verdana.

Texte en Arial

Texte en Arial Black

Texte en Comic Sans MS

Texte en Courier New

Texte en Georgia

Texte en Impact

Texte en Times New Roman

Texte en Trebuchet MS

Texte en Verdana

```
font-family: police;
```

Télécharger des nouveaux polices

Accéder à google font :

<https://fonts.google.com/>

Choisir la police que voulez ajouter sur la page web

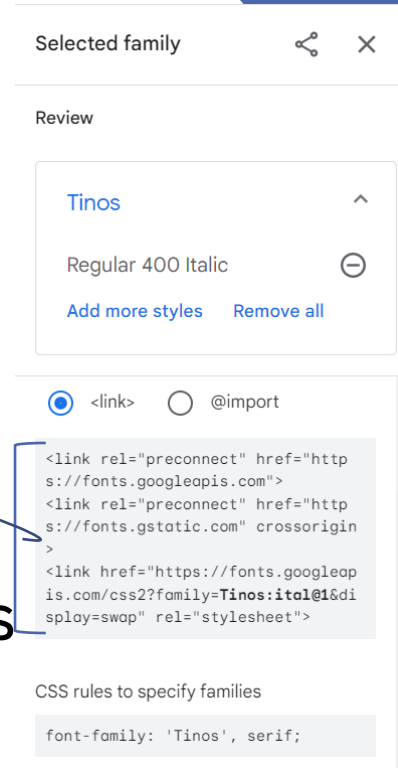
Light 300

Whereas recognition of the inherent digi

Light 300

Copier la balise <link> et la coller dans votre html, entre les balises <head>.

Attention, vous devez appeler les polices d'écriture avant le CSS qui les utilise.



Mise en forme Texte

De nombreuses propriétés de mise en forme du texte existent :

- Texte en italique :

font-style : *normal /italic;*

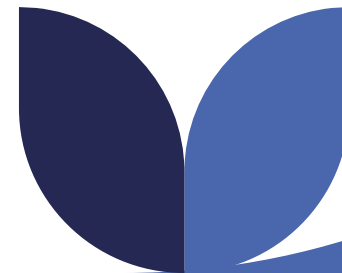
- Texte en gras :

font-weight : **bold/ normal;**

- Texte souligné :

text-decoration: underline/line-through/overline/none,

- `underline` : souligné ;
- `line-through` : barré ;
- `overline` : ligne au-dessus ;
- `none` : normal (par défaut).



Mise en forme Texte - **Alignement**

Le texte peut être aligné avec **text-align** :

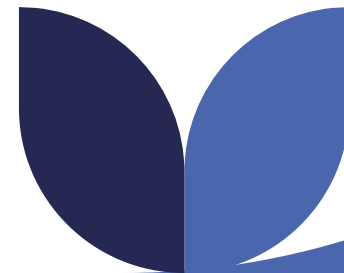
left : Le texte sera aligné à gauche(par défaut)

center : Le texte sera centré;

right : Le texte sera aligné à droite;

justify : Le texte sera justifié;

NB : On applique l'alignement que sur des blocs (div,p,h1...)



Couleur

Il y a plusieurs façon pour indiquer la couleur :

- ❖ **Ecrire son nom en anglais**








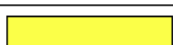



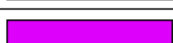

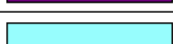
- ❖ **Forme hexadécimale** : c'est une combinaison de lettres et de chiffres qui indiquent une couleur.

On doit toujours commencer par écrire un dièse (#), suivi de six lettres ou chiffres allant de 0 à 9 et de A à F.

Exemple : #000000 correspond à la couleur noire et #FFFFFF à la couleur blanche

- ❖ **RGB** s'écrit Red-Green-Blue, ce qui s'abrège en « RGB ». Pour choisir une couleur, on doit définir une quantité de rouge, de vert et de bleu.

```
rgb(240,96,204);
```

white	
silver	
gray	
black	
red	
maroon	
lime	
green	
yellow	
olive	
blue	
navy	
fuchsia	
purple	
aqua	
teal	

Couleur du texte

On change la couleur du texte avec la propriété **color**

```
h1
{
  color: maroon;
}
```

```
p
{
  color: #FFFFFF;
}
```

```
p
{
  color: rgb(240,96,204);
}
```

Bordure standard

On applique une bordure à un élément avec la propriété **border**,

On peut utiliser jusqu'à trois valeurs pour modifier l'apparence de la bordure :

- La largeur : indiquer la largeur de bordure (en px)
- La couleur : indique la couleur de la bordure (en hexadécimal, en rgb,nom couleur)

- Le type de bordure :

none : pas de bordure (par défaut)

solid : un trait simple;

Dotted : pointillés;

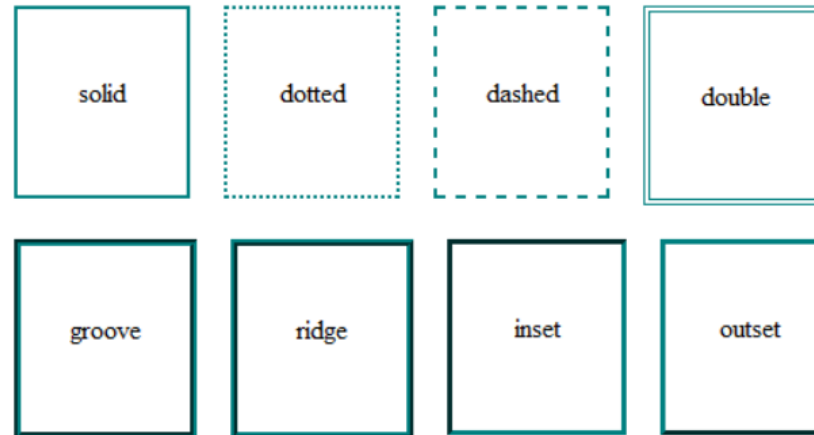
Dashed : tirets;

Double : bordure double;

Groove : en relief;

Inset: effet 3D global enfoncé;

Outset : effet 3D global surélevé



```
h1
{
  border: 3px blue dashed;
}
```


Bordure standard

On peut appliquer une bordure à un élément sur un seul côté :

- ❖ **Border-top** : bordure du haut;
- ❖ **Border-bottom** : bordure du bas;
- ❖ **Border-left** : bordure de gauche;
- ❖ **Border-right** : bordure de droite

Bordure arrondie

- La propriété **border-radius** permet d'arrondir les angles d'un élément.
- Elle prend comme valeur sa taille en pixel : `border-radius: 10px;`
- Elle peut aussi prendre la forme de l'arrondi pour chaque coin. Dans ce cas, indiquez quatre valeurs :

```
border-radius: 10px 5px 10px 5px;
```

Les ombres

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Cras ullamcorper sodales elit, sit amet pellentesque lectus aliquet quis. Etiam sem ipsum, rhoncus eu aliquam nec, mattis consectetur tortor. Mauris non lectus magna, vel interdum elit. Sed fermentum commodo commodo. Fusce imperdiet vestibulum neque, id pulvinar urna ultricies ullamcorper. Donec euismod, ipsum vehicula pretium tempor, mauris odio pellentesque metus, et ultrices arcu mauris sit amet leo. Curabitur ac scelerisque sem.

Box-shadow : Les ombres des boîtes s'appliquent sur tout un bloc

Text-shadow : L'ombre du texte

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Cras ullamcorper sodales elit, sit amet pellentesque lectus aliquet quis. Etiam sem ipsum, rhoncus eu aliquam nec, mattis consectetur tortor. Mauris non lectus magna, vel interdum elit. Sed fermentum commodo commodo. Fusce imperdiet vestibulum neque, id pulvinar urna ultricies ullamcorper. Donec euismod, ipsum vehicula pretium tempor, mauris odio pellentesque metus, et ultrices arcu mauris sit amet leo. Curabitur ac scelerisque sem.

Les deux propriétés prennent quatre valeurs dans l'ordre suivant :

1. Le décalage horizontal de l'ombre.
2. Le décalage vertical de l'ombre.
3. L'adoucissement du dégradé.
4. La couleur de l'ombre.

```
p
{
  box-shadow: 6px 6px 0px black;
}
```

Les pseudo-formats

- ❑ Le CSS permet de modifier l'apparence des éléments de façon dynamique, c'est-à-dire que des éléments peuvent changer de forme une fois que la page a été chargée. Pour ce faire, on utilise les pseudo-formats
- ❑ Pseudo-format est un mot-clé qui peut être ajouté à un sélecteur afin d'indiquer l'état spécifique dans lequel l'élément doit être pour être ciblé par la déclaration.
- ❑ On va voir comment changer l'apparence en utilisant des pseudo-formats :
 - au survol ;
 - lors du clic ;
 - lors du focus (élément sélectionné) ;
 - lorsqu'un lien a été consulté.

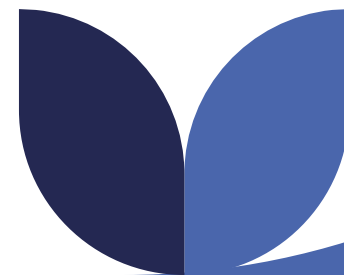
```
a:hover /* Apparence au survol des liens */  
{  
  text-decoration: underline;  
  color: green;  
}
```

Pseudo-formats

:hover : signifie « survoler » : Quand la souris est sur le lien (quand on pointe dessus).

:active : modifie l'apparence des liens au moment du clic,

:focus : permet de modifier l'apparence d'un élément sélectionné



Exercice

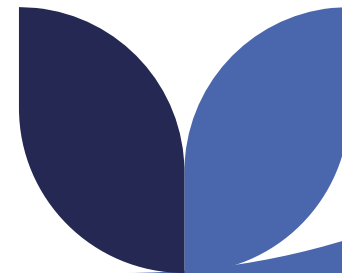
Ecrivez un article de votre choix en respectant les étapes suivantes :

1. Affichez le titre h1 en rouge.
2. Agrandissez le titre h1 pour lui donner une taille de 3em.
3. Centrez la figure (image) et sa description.
4. Ajoutez une bordure continue de la taille et de la couleur de votre choix autour de la liste à puces.
5. Ajoutez une ombre au cadre autour de la liste à puces.
6. Affichez les liens en gras lorsqu'on les survole.



TP: Mettre en forme votre CV

- Étape 1 : Récupérez le fichier HTML que vous aviez créé dans la première activité.
- Étape 2 : Ajoutez-y les effets de style suivants :
 - changer la couleur des textes ;
 - changer l'alignement des textes ;
 - définir la bordure d'un élément ;
 - définir l'ombre d'un élément.



Type de balises

On distingue deux principaux types de balises en HTML :

- **Block** (<p>, <h1>, ...) : ces balises créent un retour à la ligne et occupent par défaut toute la largeur disponible. Elles se suivent de haut en bas,
- **Inline** (<a>, , ...) : ces balises délimitent du texte au milieu d'une ligne, Elles se suivent de gauche à droite

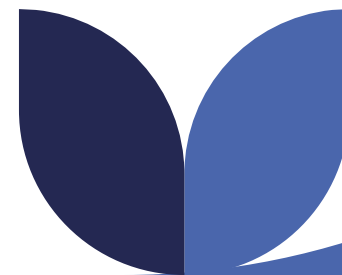


Dimensions d'un bloc

- Un bloc possède une largeur et une hauteur.
- En CSS, on dispose de deux propriétés :

width : C'est la largeur du bloc (en px ,pourcentage %)

height : C'est la hauteur du bloc (en px ,pourcentage %)



Marges d'un bloc/inline

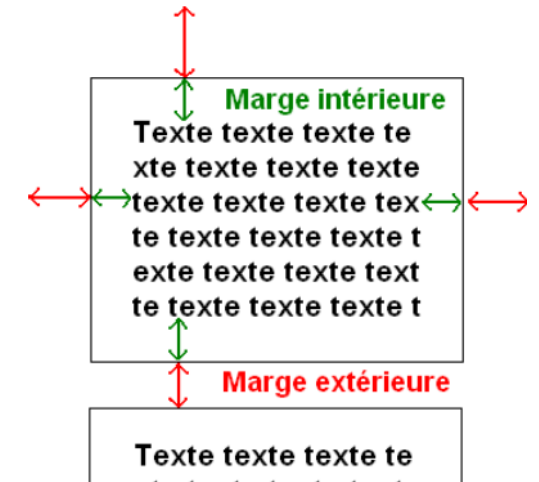
➤ Un bloc possède deux types de marges :

❖ **padding** : indique la taille de la marge intérieure (en px)

- `padding-top` : marge intérieure en haut ;
- `padding-bottom` : marge intérieure en bas ;
- `padding-left` : marge intérieure à gauche ;
- `padding-right` : marge intérieure à droite.

❖ **margin** : indique la taille de la marge extérieure (en px)

- `margin-top` : marge extérieure en haut ;
- `margin-bottom` : marge extérieure en bas ;
- `margin-left` : marge extérieure à gauche ;
- `margin-right` : marge extérieure à droite.



•NB:

`margin: 2px 0 3px 1px;` signifie « 2 px de marge en haut, 0 px à droite (le px est facultatif dans ce cas), 3 px en bas, 1 px à gauche ».

`margin: 2px 1px;` signifie « 2 px de marge en haut et en bas, 1 px de marge à gauche et à droite ».

Centrer un bloc - Horizontal

Pour centrer, il faut respecter les règles suivantes :

1. Donner une largeur au bloc (avec la propriété **width**) ;
2. Mettre les marges extérieures automatiques, comme ceci : **margin: auto;** .

```
width: 350px; /* On a indiqué une largeur (obligatoire) */  
margin: auto; /* On peut donc demander à ce que le bloc soit centré avec auto */
```

Propriété CSS : Overflow

Overflow définit comment gérer le dépassement du contenu d'un élément dans son bloc.
Elle possède quatre valeurs :

❖ **overflow: visible;**

Michaelmas term lately over,
and the Lord Chancellor
sitting in Lincoln's Inn Hall.
Implacable November
weather. As much mud in the
streets as if the waters had
but newly retired from the
face of the earth.

❖ **overflow: hidden;**

Michaelmas term lately over,
and the Lord Chancellor
sitting in Lincoln's Inn Hall.
Implacable November
weather. As much mud in the
streets as if the waters had
but newly retired from the

❖ **overflow: scroll;**

Michaelmas term lately
over, and the Lord
Chancellor sitting in
Lincoln's Inn Hall.
Implacable November
weather. As much mud in

❖ **overflow: auto;**

Michaelmas term lately
over, and the Lord
Chancellor sitting in
Lincoln's Inn Hall.
Implacable November
weather. As much mud in
the streets as if the waters

Tableau(HTML)

- ❖ **<table> </table>** : cette balise permet d'indiquer le début et la fin d'un tableau.
- ❖ Un tableau peut être divisé en trois sections:
 - <thead></thead>** : l'en-tête du tableau
 - <tbody></tbody>**: le corps du tableau
 - <tfoot></tfoot>** : le pied du tableau
- ❖ **<caption>** : définit le titre du tableau
- ❖ On peut fusionner des cellules horizontalement avec l'attribut **colspan** ou verticalement avec **rowspan** . Il faut indiquer combien de cellules doivent être fusionnées.

```
<table>
  <caption>Liste des stagiaires</caption>
  <thead>
    <tr>
      <th>Nom</th>
      <th>Age</th>
      <th>Code postal</th>
      <th>Commune</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>Solene</td>
      <td>33 ans</td>
      <td>75000</td>
      <td rowspan="2">Paris</td>
    </tr>
    <tr>
      <td>Etienne</td>
      <td>37 ans</td>
      <td>75000</td>
      <!--<td>Paris</td>-->
    </tr>
    <tr>
      <td>LUCIE</td>
      <td>26 ans</td>
      <td>92200</td>
      <td>Neuilly sur seine</td>
    </tr>
    <tr>
      <td>Nael</td>
      <td>40 ans</td>
      <td colspan="2">Aucune adresse</td>
    </tr>
  </tbody>
</table>
```

Tableau(CSS)

On change la position du titre avec la propriété CSS **caption-side** qui peut prendre deux valeurs :

- **top** : le titre sera placé au-dessus du tableau (par défaut) ;
- **bottom** : le titre sera placé en dessous du tableau.

```
caption{  
  caption-side: bottom;  
}
```

il existe une propriété CSS spécifique aux tableaux, **border-collapse** , qui signifie « coller les bordures entre elles ». Cette propriété peut prendre deux valeurs :

- **collapse** : les bordures seront collées entre elles, c'est l'effet qu'on recherche ici ;
- **separate** : les bordures seront dissociées (valeur par défaut).

```
table  
{  
  border-collapse: collapse;  
}  
td, th /* Mettre une bordure sur les td ET les th */  
{  
  border: 1px solid black;  
}
```

On peut ajouter une bordure aux cellules du tableau avec **border**.

Fusionner les éléments d'un tableau

Fusionner les cellules d'un tableau html en utilisant les propriétés "**colspan**" et "**rowspan**".

1			
2	3	4	5
	6	7	
	8		

Formulaire

- ❑ Un formulaire est une zone interactive de la page HTML, dans laquelle les visiteurs peuvent saisir des informations.

<form> </form> : permet d'ajouter un formulaire en HTML.

- ❑ La balise <form> contient toujours deux attributs :

method : cet attribut indique par quel moyen les données vont être envoyées. Il existe deux solutions pour envoyer des données sur le Web :

method="get" : les informations seront envoyées dans l'adresse de la page (http://...)

method="post" : Les données saisies dans le formulaire ne transitent pas par la barre d'adresse ;

action : c'est l'adresse de la page ou du programme qui va traiter les informations

```
<form method="post" action="traitement.php">
```


Formulaire

Un formulaire est composé de :

- ☐ **Zone de saisie**
- ☐ **Libellés**
- ☐ **Éléments d'options**
- ☐ **Champs regroupés**

Zone de saisie

Il y a deux zones de texte différentes :

`<input />` : la zone de texte monoligne

`<textarea> </textarea>` : la zone de texte multiligne

Zone de saisie `<input />` (1/2)

`<input />` : la zone de texte monoligne : comme son nom l'indique, on ne peut y écrire qu'une seule ligne. Elle sert à saisir des textes courts, par exemple un pseudo ;

On peut ajouter des attributs à la balise `<input />` pour personnaliser son fonctionnement :

`size` : agrandir le champ

`maxlength` : limiter le nombre de caractères à saisir

`placeholder` : donner une indication sur le contenu du champ

`type` : donner le type du champ (text, mail, date, password, url, tel, number, color, search...)

`name` : donner le nom du champ (variable)

`id` : est utilisé pour identifier l'élément HTML pour pouvoir y accéder et le manipuler. Il est donc unique pour cet élément.

```
<input type="text" name="username" id="username"/>
```

Zone de saisie `<input />` (2/2)

- `<input type="email" />` : pour une zone de saisie d'e-mail
- `<input type="url" />` : pour une zone de saisie d'URL (adresse internet)
- `<input type="number" />` : pour une zone de saisie de nombre, dispose d'attributs spéciaux :
 - min : valeur minimale autorisée
 - max : valeur maximale autorisée
 - step : indique la valeur de déplacement, si le champ se déplace de un en un, de deux en deux, de trois en trois
- `<input type="color" />` : pour une zone de saisie de couleur
- `<input type="date" />` : pour une zone de saisie de date
- `<input type="search" />` : pour une zone de recherche



Zone de saisie

<textarea> </textarea> : la zone de texte multiligne : cette zone de texte permet d'écrire une quantité importante de texte sur plusieurs lignes,

On peut ajouter des attributs à la balise <textarea /> pour personnaliser son fonctionnement :

rows : indique le nombre de lignes de texte qui peuvent être affichées simultanément,

cols : indique le nombre de colonnes de texte qui peuvent être affichées simultanément,

name : donner le nom du champ (variable)

id : est utilisé pour **identifier** l'élément HTML pour pouvoir y accéder et le manipuler. Il est donc unique pour cet élément.

```
<textarea rows="5" cols="12" name="commentaire" id="commentaire"></textarea>
```

Libellés

- ❑ `<label></label>` : Le libellé vient pour expliquer à l'utilisateur ce qu'il doit écrire dans une zone de texte
- ❑ Pour lier le label au champ, il faut lui donner un attribut `for` qui a la même valeur que l'id du champ

```
<form method="post" action="traitement.php">
  <p>
    <label for="username">username</label> : <input type="text" name="username" id="username" />
  </p>
</form>
```

Les éléments d'options

Ce sont les éléments qui demandent aux utilisateurs de faire un choix parmi une liste de possibilités :

- Les cases à cocher :

```
<input type="checkbox" name="choix" />
```

- Les zones d'options :

```
<p>  
  Sur quel continent habitez-vous ?<br />  
  <input type="radio" name="continent" value="europe" id="europe" /> <label for="europe">Europe</label><br />  
  <input type="radio" name="continent" value="afrique" id="afrique" /> <label for="afrique">Afrique</label><br />  
  <input type="radio" name="continent" value="asie" id="asie" /> <label for="asie">Asie</label><br />  
  <input type="radio" name="continent" value="amerique" id="amerique" /> <label for="amerique">Amérique</label><br />  
  <input type="radio" name="continent" value="australie" id="australie" /> <label for="australie">Australie</label>  
</p>
```

- Les listes déroulantes

```
<label for="pays">Dans quel pays habitez-vous ?</label><br />  
<select name="pays" id="pays">  
  <option value="france">France</option>  
  <option value="espagne">Espagne</option>  
  <option value="italie">Italie</option>  
  <option value="royaume-uni">Royaume-Uni</option>  
  <option value="canada">Canada</option>  
  <option value="etats-unis">États-Unis</option>  
  <option value="chine">Chine</option>  
  <option value="japon">Japon</option>  
</select>
```



```
<p>
  <label for="pays">Dans quel pays habitez-vous ?</label><br />
  <select name="pays" id="pays">
    <optgroup label="Europe">
      <option value="france">France</option>
      <option value="espagne">Espagne</option>
      <option value="italie">Italie</option>
      <option value="royaume-uni">Royaume-Uni</option>
    </optgroup>
    <optgroup label="Amérique">
      <option value="canada">Canada</option>
      <option value="etats-unis">Etats-Unis</option>
    </optgroup>
    <optgroup label="Asie">
      <option value="chine">Chine</option>
      <option value="japon">Japon</option>
    </optgroup>
  </select>
</p>
```


Regrouper les champs

On peut regrouper les champs au sein de plusieurs balise `<fieldset>`, chaque `<fieldset>` peut contenir une légende `<legend>`

```
<form method="post" action="traitement.php">

  <fieldset>
    <legend>Vos coordonnées</legend> <!-- Titre du fieldset -->

    <label for="nom">Quel est votre nom ?</label>
    <input type="text" name="nom" id="nom" />

    <label for="prenom">Quel est votre prénom ?</label>
    <input type="text" name="prenom" id="prenom" />

    <label for="email">Quel est votre e-mail ?</label>
    <input type="email" name="email" id="email" />

  </fieldset>

  <fieldset>
    <legend>Votre souhait</legend> <!-- Titre du fieldset -->

    <p>
      Faites un souhait que vous voudriez voir exaucé :

      <input type="radio" name="souhait" value="riche" id="riche" /> <label for="riche">Etre
riche</label>
      <input type="radio" name="souhait" value="celebre" id="celebre" /> <label
for="celebre">Etre célèbre</label>
      <input type="radio" name="souhait" value="intelligent" id="intelligent" /> <label
for="intelligent">Etre <strong>encore</strong> plus intelligent</label>
      <input type="radio" name="souhait" value="autre" id="autre" /> <label for="autre">Autre...
</label>
    </p>

    <p>
      <label for="precisions">Si "Autre", veuillez préciser :</label>
      <textarea name="precisions" id="precisions" cols="40" rows="4"></textarea>
    </p>
  </fieldset>
</form>
```

Contraintes sur le formulaire

Sélectionner automatiquement un champ :

```
<input type="text" name="prenom" id="prenom" autofocus />
```

Rendre un champ obligatoire :

```
<input type="text" name="prenom" id="prenom" required />
```

Bouton d'envoi

```
<input type="submit" value="Envoyer" />
```

Réinitialisation du formulaire

```
<input type="reset" value="Effacer">
```

Exercice

Créer le formulaire d'inscription suivant en HTML

Formulaire d'inscription

Nom	<input type="text" value="saisir votre nom"/>
Prenom	<input type="text" value="saisir votre prenom"/>
Email	<input type="text" value="saisir votre Email"/>
Sexe	Masculin <input checked="" type="radio"/> Féminin <input type="radio"/>
Photo	<input type="button" value="Choisir un fichier"/> Aucun fichier choisi
Pays	<input type="text" value="France"/>
Les langages préférés	<div><div>Java</div><div>C</div><div>C++</div><div>PHP</div></div>
Domaines d'activités	<input type="checkbox"/> Informatique <input type="checkbox"/> Gestion <input type="checkbox"/> Pédagogie
<input type="button" value="Envoyer"/> <input type="button" value="Annuler"/>	

Exercice

Créer un formulaire contenant un champ de texte pour saisir un **login** et un champ **password** pour le mot de passe **de huit caractères au maximum**, chacun étant précédé d'un label explicatif. Ajouter un bouton d'envoi et un bouton de réinitialisation.



Exercice

Créer le formulaire suivant en HTML/CSS

Formulaire de participation à la compétition de jeux vidéo

Complétez le formulaire. Les champs marqué par * sont **obligatoires**

Informations personnelles

Nom *

EL ATCHIA SANA

Age*

xx

Sexe

Femme

Pourquoi voulez-vous participer à cette compétition ?

Quel framework choisir pour votre projet web ?

☐ React☐ Angular☐ VueJs☐ Symfony☐ Spring☐ Laravel

Contact

Téléphone

06xxxxxxxx

Email *

nom.prenom@lagrandeclasse.fr

Soumettre

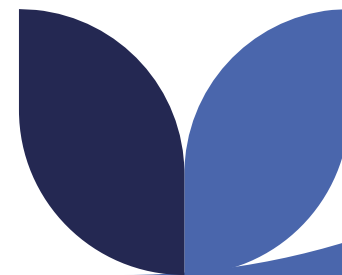
Attribut de modèle (Pattern)

Pattern (expression régulière) est une chaîne de caractères formalisée qui définit un modèle.

Fonctionnement : Comparaison de la valeur d'entrée à une expression régulière

Exemple :

[a-zA-Z0-9]+ : un modèle correspond à une chaîne de n'importe quelle longueur, tant que la chaîne ne contient que des lettres minuscules (a-z), des lettres majuscules (A-Z) ou des chiffres (0-9).



Comment écrire une expression régulière ?

Répétiteurs (1/)

Ces symboles agissent comme des répéteurs et indiquent à l'ordinateur que le caractère précédent doit être utilisé plus d'une fois

Exemple :

***** : Il indique à l'ordinateur de faire correspondre le caractère précédent (ou ensemble de caractères) **pour 0 ou plusieurs fois** (jusqu'à l'infini).

+ : Il indique à l'ordinateur de répéter le caractère précédent (ou ensemble de caractères) **au moins une ou plusieurs fois** (jusqu'à l'infini).

{...} : elles indiquent à l'ordinateur de répéter le caractère (ou l'ensemble de caractères) précédent **autant de fois que la valeur à l'intérieur de cette parenthèse**.



Comment écrire une expression régulière ?

- **[premier-dernier]** – Plage de caractères : correspond à n'importe quel caractère unique dans la plage du premier au dernier.

[a-zA-z] N'importe quelle lettre de a à z ou de A à Z

- **Le symbole d'échappement : ** Si vous voulez faire correspondre le '+' réel, '.' etc, ajoutez une barre oblique inverse (\) avant ce caractère.



Les différents formats audio

Voici les quatre principaux formats :

MP3 : le format le plus populaire et le plus utilisé de nos jours, il a même donné son nom à des baladeurs. C'est également un des formats les plus compatibles.

WMA : un format également très populaire et compatible avec un grand nombre de navigateurs. Ce format existe avec deux formes de compression.

AAC : un format développé par Apple, donc très répandu sur iTunes. Il est très utilisé sur iPhone, iPod et iPad.

WAV : un format non-compressé, donc déconseillé pour vos pages web. Ce format permet néanmoins de stocker des fichiers en grande qualité qui pourront ensuite être convertis.



Insérer des éléments audios

<audio></audio> permet d'ajouter des audios à une page HTML,

<audio src=""></audio> : ne permet que télécharger les information générales de l'audio dans le navigateur,

Pour afficher l'audio, il faut lui ajouter l'attribut **controls** qui permet d'ajouter les boutons **Lecture**, **pause** et la **barre de défilement**

```
<audio src="C:\Users\lagra\Downloads\mozar.mp3" controls></audio>
```

Les différents formats vidéos

Voici les quatre principaux formats :

IFF : un format très utilisé pour les appareils Apple

AVI : un format très utilisé sous Windows

H.264 : un format puissant et pris en charge par la plupart des navigateurs

WebM : un format libre de droits proposé par Google

Insérer des éléments vidéos

<video></ video > permet d'ajouter une vidéo à une page HTML,

< video src=""></ video > : ne permet que télécharger les information générales de la vidéo dans le navigateur,

Pour afficher la vidéo, il faut lui ajouter l'attribut **controls** qui permet d'ajouter les boutons **Lecture, pause** et la **barre de défilement**

```
<video src="C:\Users\lagra\Downloads\htmlCours1.mp4" width="600"controls></video>
```

On peut ajouter l'attribut suivant à la balise <video> :

```
<video src="C:\Users\lagra\Downloads\htmlCours1.mp4" width="600" poster="C:\Users\lagra\OneDrive\Bureau\architecture.png" controls></video>
```

poster : image à afficher à la place de la vidéo tant que celle-ci n'est pas lancée. Par défaut, le navigateur prend la première image de la vidéo.

Média queries

Les média queries : Ce sont des règles à appliquer pour changer le design d'un site en fonction des caractéristiques de l'écran,

Grâce à cette technique, nous pourrons :

- Créer un design qui s'adapte automatiquement à l'écran de chaque visiteur !
- Changer l'apparence du site en fonction d'autres **critères** comme le **type d'écran** (smartphone, télévision, projecteur...), **l'orientation de l'écran** (portrait ou paysage), etc

Les règles disponibles

Il existe de nombreuses règles permettant de construire des media queries :

- **color** : gestion de la couleur (en bits/pixel) ;
- **height** : hauteur de la zone d'affichage (fenêtre) ;
- **width** : largeur de la zone d'affichage (fenêtre) ;
- **orientation** : orientation du périphérique (portrait ou paysage) ;
- **media** : type d'écran de sortie. Quelques-unes des valeurs possibles :
 - **screen** : écran « classique »,
 - **handheld** : périphérique mobile,
 - **print** : impression,
 - **tv** : télévision,
 - **projection** : projecteur,
 - **all** : tous les types d'écrans.

NB :

On peut rajouter le préfixe **min-** ou **max-** devant la plupart de ces règles.

Exemple :

min-width signifie « largeur minimale »,

max-height « hauteur maximale »

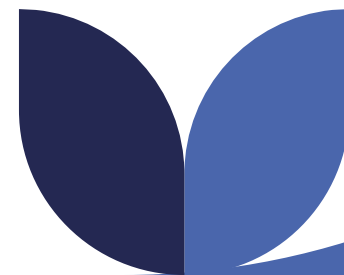
Combiner les règles entre eux

Les règles peuvent être combinées à l'aide des mots suivants :

- **only** : « uniquement »
- **and** : « et »
- **not** : « non »

Pour créer les media queries, vous devez respecter le formalisme suivant :

1. commencer par **@media** ;
2. indiquer le type d'écran ;
3. plusieurs conditions comme la largeur maximum de l'écran ;
4. puis le style **CSS** que vous souhaitez **appliquer** sous ces conditions préétablies



Appliquer une média query

Les media queries sont **des règles qui indiquent quand on doit appliquer des propriétés CSS.**

Il y a deux façons de les utiliser :

- En chargeant une feuille de style .css différente en fonction de la règle

```
<!-- ajouter un attribut media , dans lequel on va écrire la règle qui doit s'appliquer pour que le fichier soit chargé( ce qu'on appelle media query) -->  
<link rel="stylesheet" media="screen and (max-width: 1280px)" href="petite_resolution.css" />
```

```
<link rel="stylesheet" href="style.css" />
```

- En écrivant la règle directement dans le style.css habituel

```
/* Ajoutons cette instruction, dans le fichier style.css */  
@media screen and (max-width: 1280px)  
{  
    /* Rédigez vos propriétés CSS ici */  
}
```

Exemple d'application

- On crée une media query avec la directive `@media` suivie du type d'écran et d'une ou plusieurs conditions (comme la largeur maximale d'écran).
- Le style CSS qui suit sera activé uniquement si les conditions sont remplies.

```
p{
  font-family:Arial, Helvetica, sans-serif;
  /*font-style: italic; */
  /*font-weight: bold;*/
  text-decoration: underline;
  /*text-align:right;*/
  color:rgb(68, 0, 255);
  border :2px black solid;

  border-radius: 10px 4px 6px 10px;
  box-shadow: 6px 6px 0px black;
  text-shadow: 1px 1px 0px rgb(68, 0, 255);
}
```

```
/* Nouvelles règles si la fenêtre fait au plus 1024px de large */
@media screen and (max-width: 1024px)
{
  p
  {
    color: red;
    background-color: rgb(47, 140, 194);
    font-size: 1.2em;
  }
}
```

Exercice

Appliquer des média queries sur ces deux conditions :

1/Lorsque la page a une largeur comprise entre 640 px et 960 px , changer la couleur de fond du chapitre 1.



Créer une animation en CSS

- ❖ La structure d'une animation CSS
- ❖ Les étapes d'une animation
- ❖ Les propriétés d'une animation



Structure d'une animation CSS

Une animation CSS est toujours définie par une règle **@keyframes** suivie du nom de l'animation

```
/*fichier .css*/  
  
@keyframes nom_animation{  
    /*etapes de l'animation & les propriétés des différents éléments */  
}
```

Les étapes d'une animation CSS (1/2)

Une animation est un cycle durant lequel l'état d'un élément est modifié.
Dans une animation, on peut avoir au minimum deux états :

- **From** : c'est l'état initial qui correspond à l'instant 0% de l'animation,
- **To** : c'est l'état final qui correspond à l'instant 100% de l'animation,

```
@keyframes nom_animation{  
  /*etapes de l'animation & les propriétés des différents éléments */  
  From {  
    /*propriétés css */  
    background :■rgb(173, 202, 219) ;  
  }  
  To {  
    /*propriétés css */  
    background :■rgb(173, 202, 219) ;  
  }  
}
```



Les étapes d'une animation CSS (2/2)

Pour rajouter un **état intermédiaire**, il suffit de rajouter un **pourcentage**.

```
@keyframes nom_animation{  
  /*etapes de l'animation & les propriétés des différents éléments */  
  0% {  
    /*propriétés css */  
    background :■rgb(173, 202, 219) ;  
  }  
  50%{  
    background :■rgb(129, 173, 199) ;  
  }  
  100% {  
    /*propriétés css */  
    background :■rgb(93, 169, 212) ;  
  }  
}
```

- 50% signifie qu'une fois arrivé au 50% d'avancement de l'animation, l'élément prend un nouvel état.

Les propriétés d'une animation

Pour appliquer une animation sur un élément HTML (<div>,....),on utilise des propriétés comme :

- **Animation-delay** : correspond au délai avant que l'animation commence.

0s : l'animation commence immédiatement après le chargement de votre page.

- **Animation-duration** : indique le temps que doit durer l'animation

- **Animation-iteration-count** : détermine le nombre de fois que doit être répétée l'animation.

Infinite : pour que l'animation tourne en continu.

- **Animation-direction** :

normal : l'animation recommencera au début de chaque cycle au lieu de faire des allers-retours

- **Animation-name** : fait référence à l'animation qui doit être appliquée



Appliquer une animation sur une div

```
/*fichier .css*/

@keyframes nom_animation{
  /*etapes de l'animation & les propriétés des différents éléments */
  0% {
    /*propriétés css */
    background :■rgb(173, 202, 219) ;
  }
  50%{
    background :■rgb(129, 173, 199) ;
  }
  100% {
    /*propriétés css */
    background :■rgb(93, 169, 212) ;
  }
}

div{
  Animation-name : nom_animation ;
  Animation-delay : 2s ;
  Animation-duration : 3s ;
  Animation-iteration-count : infinite ;
}
```

Appliquer une animation sur une image

On utilise l'animation pour passer un élément d'une forme carrée à une forme ronde de façon progressive, tout en modifiant la couleur

CSS

```
@keyframes borderAnimation {  
  0% {  
    border-radius:0 0 0 0;  
    background:■rgb(94, 9, 221);  
  }  
  25% {  
    border-radius:10% 10% 10% 10%;  
    background:■rgb(105, 34, 212);  
  }  
  50% {  
    border-radius:25% 25% 25% 25%;  
    background:■rgb(129, 83, 199);  
  }  
  75% {  
    border-radius:40% 40% 40% 40%;  
    background:■rgb(160, 132, 202);  
  }  
  100% {  
    border-radius:50%;  
    background:■rgb(193, 184, 206);  
  }  
}  
img{  
  Animation-name : borderAnimation ;  
  Animation-delay : 2s ;  
  Animation-duration : 3s ;  
}
```

HTML

```

```

Appliquer une animation sur une image

On utilise l'animation pour passer un élément d'une forme carrée à une forme ronde de façon progressive, tout en modifiant la couleur

CSS

```
@keyframes borderAnimation {  
  0% {  
    border-radius:0 0 0 0;  
    background:■rgb(94, 9, 221);  
  }  
  25% {  
    border-radius:10% 10% 10% 10%;  
    background:■rgb(105, 34, 212);  
  }  
  50% {  
    border-radius:25% 25% 25% 25%;  
    background:■rgb(129, 83, 199);  
  }  
  75% {  
    border-radius:40% 40% 40% 40%;  
    background:■rgb(160, 132, 202);  
  }  
  100% {  
    border-radius:50%;  
    background:■rgb(193, 184, 206);  
  }  
}  
img{  
  Animation-name : borderAnimation ;  
  Animation-delay : 2s ;  
  Animation-duration : 3s ;  
}
```

HTML

```

```

Exercice

Ajouter un titre h1 au formulaire d'inscription
Ajouter une animation sur le titre ajouté

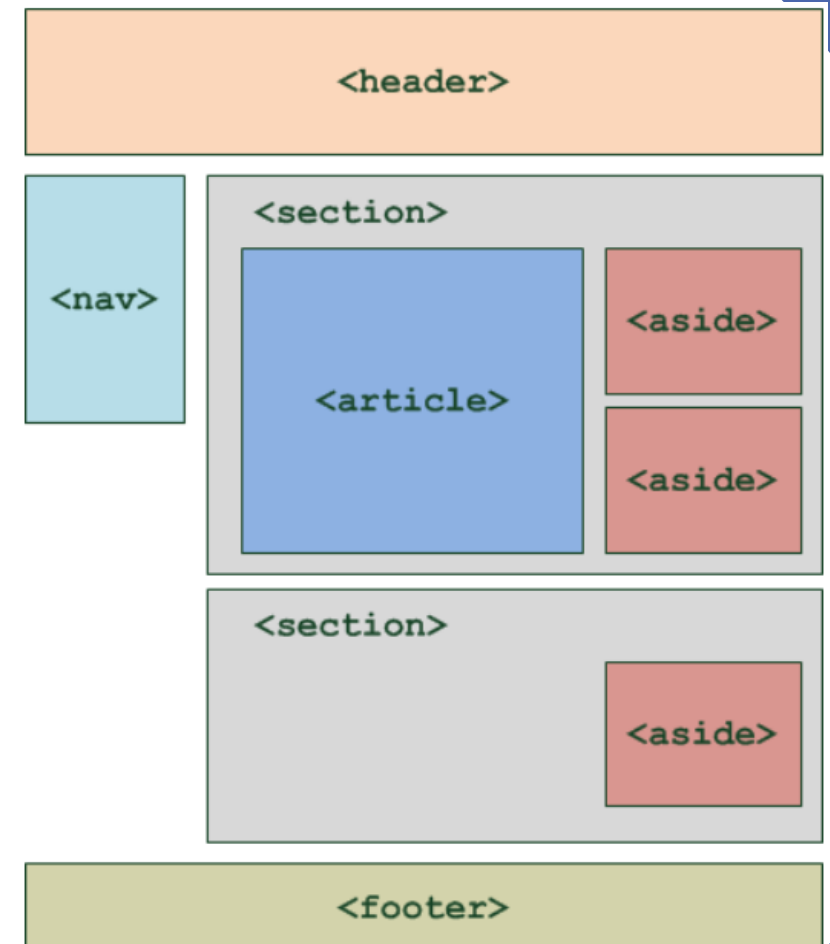
Exercice

Ajouter un titre h1 au formulaire d'inscription
Ajouter une animation sur le titre ajouté (voir Démo)

Structure d'une page

Une page web est constituée de :

- **<header>** : en-tête
- **<footer>**: pied de la page
- **<nav>** : regroupe les principaux liens de navigation,
- **<section>** : section de page,
- **<aside>** : informations complémentaires;
- **<article>** : article indépendant



Structure d'une page

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>Zozor - Le Site Web</title>
  </head>

  <body>
    <header>
      <h1>Zozor</h1>
      <h2>Carnets de voyage</h2>
    </header>

    <nav>
      <ul>
        <li><a href="#">Accueil</a></li>
        <li><a href="#">Blog</a></li>
        <li><a href="#">CV</a></li>
      </ul>
    </nav>

    <section>
      <aside>
        <h1>À propos de l'auteur</h1>
        <p>C'est moi, Zozor ! Je suis né un 23 novembre 2005.</p>
      </aside>
      <article>
        <h1>Je suis un grand voyageur</h1>
        <p>Bla bla bla bla (texte de l'article)</p>
      </article>
    </section>

    <footer>
      <p>Copyright Zozor - Tous droits réservés<br />
      <a href="#">Me contacter !</a></p>
    </footer>
  </body>
</html>
```

Remarques (1/2):

- La propriété **height** est applicable sur toutes les balises HTML qui ne sont pas « en ligne »
- La propriété **height** CSS ne spécifie que hauteur de la zone contenu, cela n'inclue pas les marges intérieures, ni les bordures,
- Pour le éléments de type **inline** ou de type display:inline, la propriété CSS **height** n'est pas prise en compte, la hauteur correspond à la valeur de line-height
- Si le conteneur(parent) a une hauteur (height) plus petite que son contenu alors vous devez spécifier au parent la propriété **overflow:hidden**.

Dans le cas où vous voulez que la hauteur du parent s'adapte au contenu, vous devez spécifier au parent la propriété **height:auto**



Remarques (2/2) :

box-sizing permet de spécifier si la valeur de **width** ou de **height** inclut la marge intérieure et la taille de la bordure,

```
box-sizing : border-box;  
box-sizing : padding-box;  
box-sizing : content-box;
```

Display permet de modifier le type de rendu d'un élément.

```
display : block;  
display : inline;  
display : inline-block;  
display : table-cell;  
display : list-item;  
display : flex;  
display : grid;
```

line-height permet de spécifier la **hauteur d'une ligne de texte** et de **gérer l'espace entre les lignes**

```
line-height : normal;  
line-height : 5px;  
line-height : 1.5;
```

SPAN permet de créer **des propriétés CSS** à un endroit précis

Exercice

Créer une barre de navigation **horizontal en HTML/CSS** en respectant les **remarques** ci-dessus





Merci

