

# INTRODUCTION WEB

# Description du cours

- Le but de ce cours d'initiation au HTML et CSS est de permettre aux étudiants d'appréhender les bases du HTML et CSS afin de pouvoir ensuite créer de manière autonome des sites web.
- Le cours est une initiation, nous n'aurons pas le temps de voir en détail toutes les balises et positionnement mais des liens seront fournis afin de permettre aux étudiants d'approfondir leur expertise.

# Les bases d'un site web

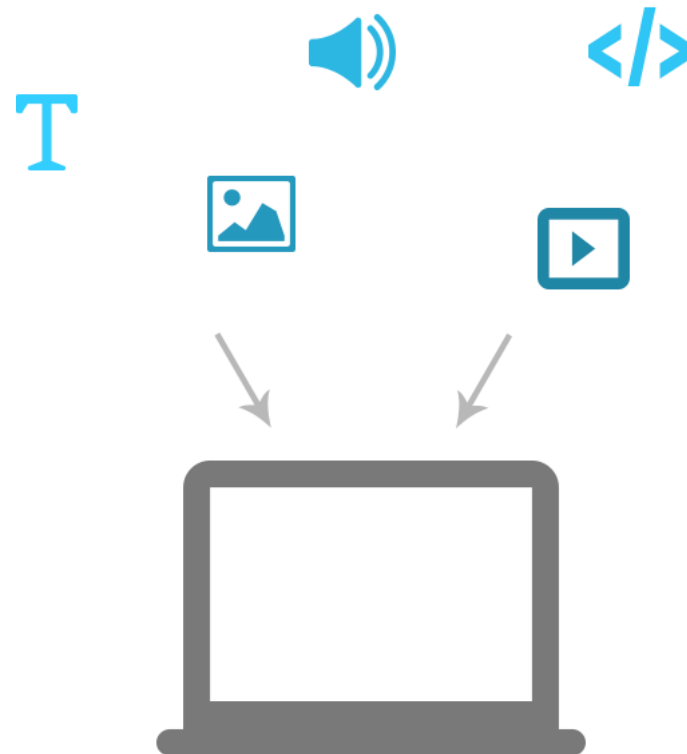
---

# Une page Web c'est ...

- Un fichier HTML est un format de fichier « texte » éditible dont les éléments ont du sens
- Au format .html
- Peut contenir du texte, des images, des liens, des médias, etc.
- Peut être liée à une autre page via des liens

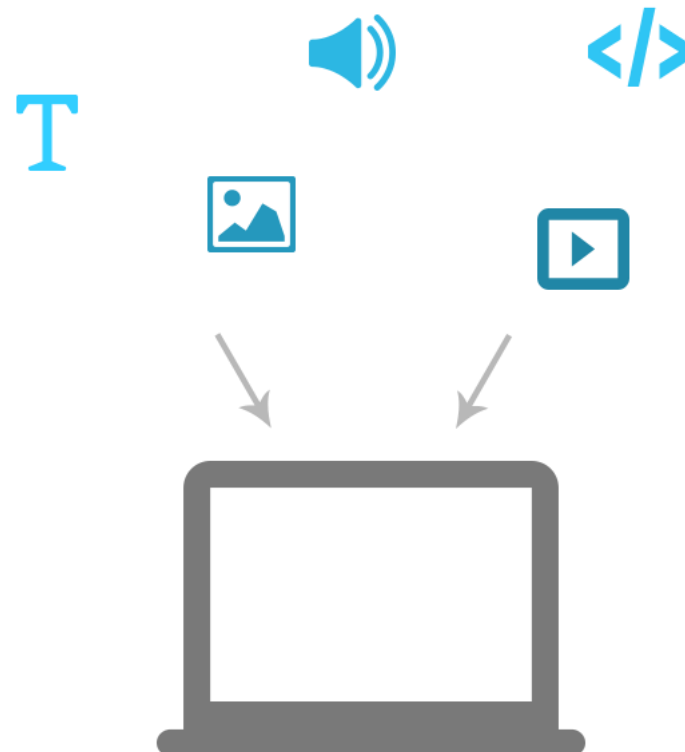
# Un site Web c'est ...

- Un ensemble de pages liées entre elles
- Accessible en ligne depuis n'importe où

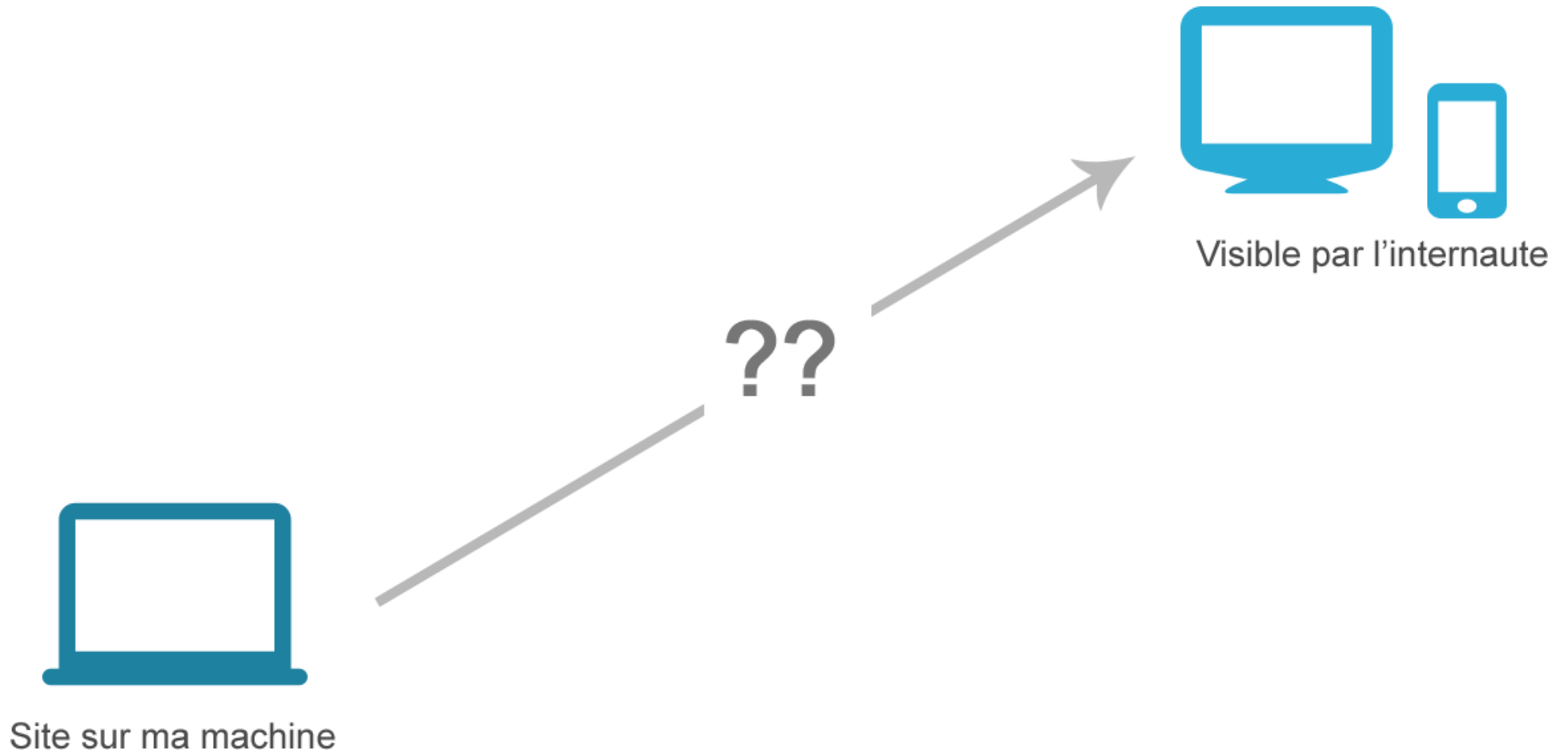


## Mais pas que ...

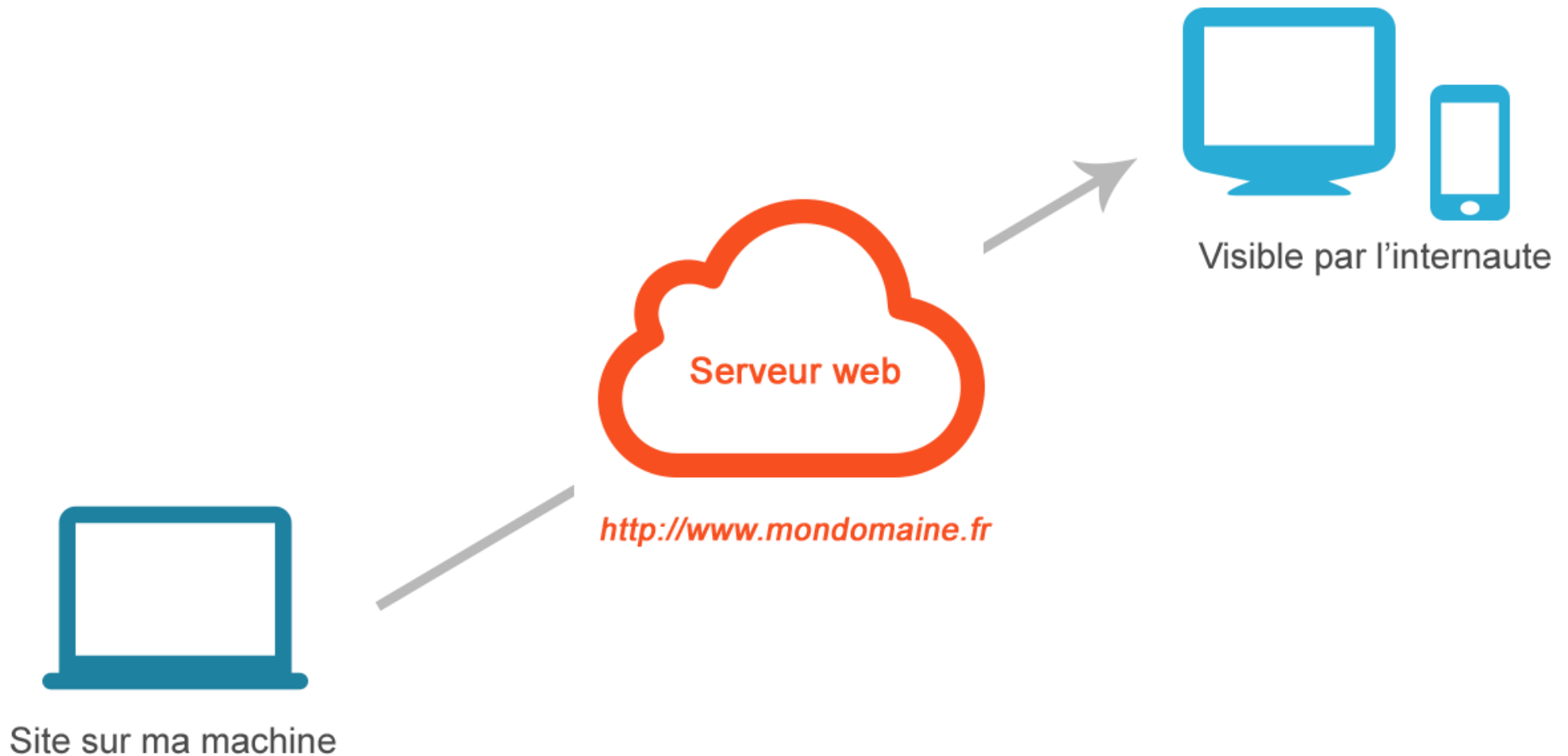
- Contenus textuels dans le fichier HTML, mais aussi images, sons, vidéos, CSS, JavaScript, polices d'écriture, etc. tous ces fichiers sont chargés dans le navigateur pour créer une page web.



# Mettre son site en ligne



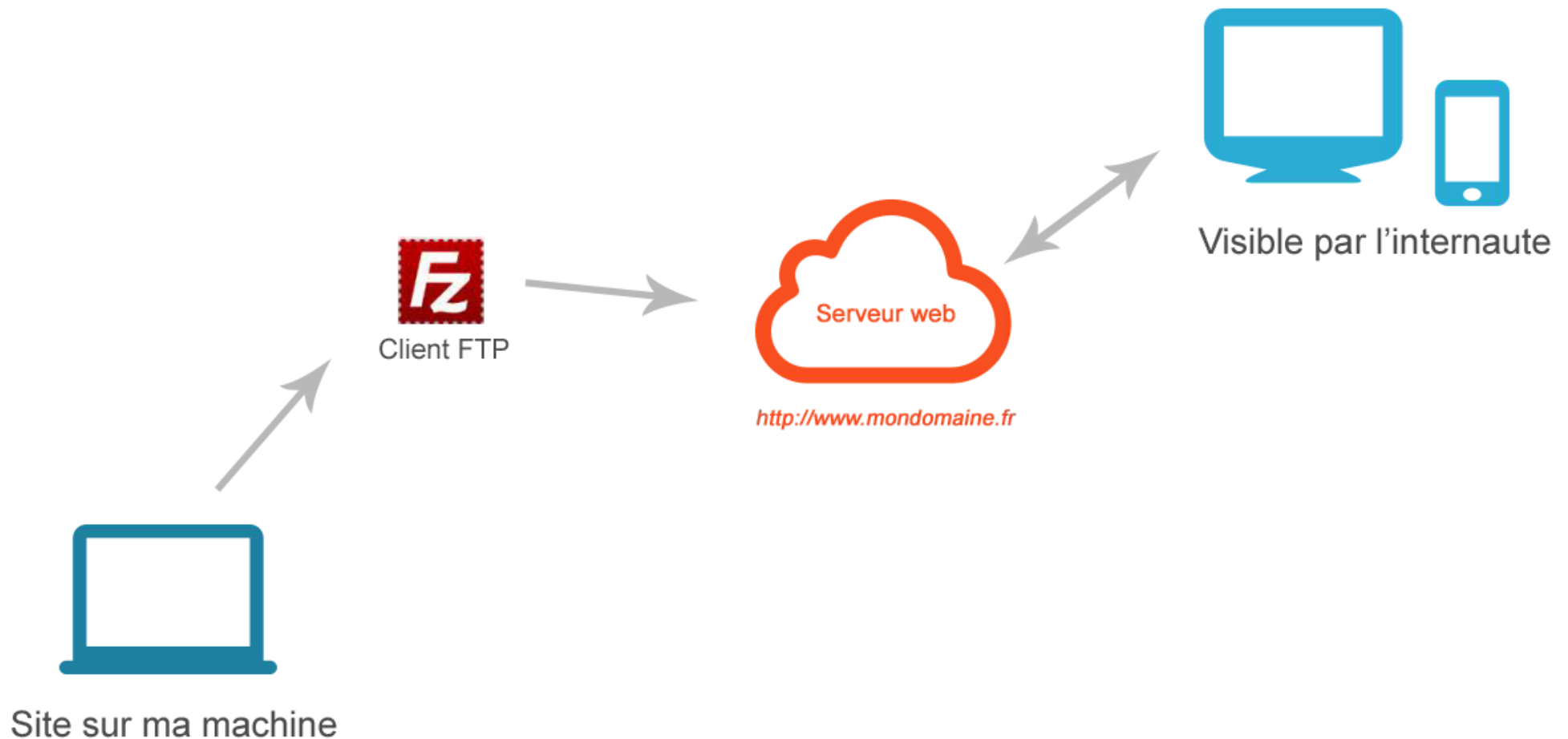
# Le principe de « serveur web »



Pour mettre en ligne on fait passer ses fichiers de la machine du créateur à un serveur web



# Transférer un fichier sur le serveur



Souvent à l'aide d'un client FTP. Votre hébergeur vous fournira les logins et mot de passe pour y accéder.

# Le client FTP

- FTP : File Transfer Protocol, protocole de communication destiné à l'échange de fichiers sur un réseau TCP / IP.
- Permet de « mettre en ligne » son site web
- Une sorte d'explorateur sur un serveur à distance
- Ex : [Filezilla](#)



# Le HTML - généralités

---

# HTML : définition

- Hypertext Markup Language (HTML)
- Langage de balises qui permet de structurer des pages
- Différentes versions depuis 1989
- Aujourd'hui :
  - XHTML (2000 – 2006)
  - HTML5

```
1 <!DOCTYPE html>
2
3 <html lang="fr">
4 <head>
5
6 <meta charset="UTF-8" />
7
8 <title>Quelle différence y a-t-il entre HTML et XHTML ? - Alsacreations</title>
9
10 <link rel="shortcut icon" href="http://www.alsacreations.com/favicon.ico" />
11 <link rel="icon" type="image/x-icon" href="http://www.alsacreations.com/favicon.ico" />
12 <link rel="icon" type="image/png" href="http://www.alsacreations.com/favicon.png" />
13
14 <meta name="keywords" content="xhtml, validation, html, tableless, doctype, différence" />
15
16 <meta name="geo.placename" content="Strasbourg, Alsace, France" />
17
18 <link rel="apple-touch-icon" href="http://www.alsacreations.com/apple-touch-icon.png" />
19 <link rel="stylesheet" type="text/css" href="http://www.alsacreations.com/css/styles20120827.min.css" />
20 <link rel="search" type="application/opensearchdescription+xml" title="Alsacreations"
21 href="http://www.alsacreations.com/opensearch.xml" />
22
23 <meta name="viewport" content="width=device-width" />
24
25 <!--[if lte IE 6]><link type="text/css" rel="stylesheet" href="http://www.alsacreations.com/css/ie6fix.css"
26 media="all" /><![endif-->
27 <!--[if lte IE 7]><style type="text/css">#footer, #page { zoom: 1; }</style><![endif-->
28
29 <link rel="stylesheet" title="Zenburn" href="/js/highlight/styles/zenburn.css"><link rel="alternate"
30 type="application/rss+xml" href="/rss/actualites.xml" title="Actualités" />
31 <link rel="alternate" type="application/rss+xml" href="/rss/commentaires.xml" title="Commentaires" />
32 <link rel="alternate" type="application/rss+xml" href="/rss/apprendre.xml" title="Apprendre (dernières
33 publications)" />
34
35 <script type="text/javascript">
36
37 var _gaq = _gaq || [];
38 _gaq.push(['_setAccount', 'UA-1456483-4']);
39 _gaq.push(['_trackPageview']);
40
41 (function() {
42   var ga = document.createElement('script'); ga.type = 'text/javascript'; ga.async = true;
43   ga.src = ('https:' == document.location.protocol ? 'https://ssl' : 'http://www') + 'google-
```

# Le W3C

- W3C : World Wide Web Consortium (1994)
  - TBLee, fondateur du W3C et inventeur du HTML
  - Chargé de promouvoir la compatibilité des technologies dans les navigateurs
  - 378 entreprises membres qui peuvent faire des propositions (Microsoft, Apple, Mozilla, Opera, Adobe, etc.)
  - Proposent un validateur <http://validator.w3.org/>

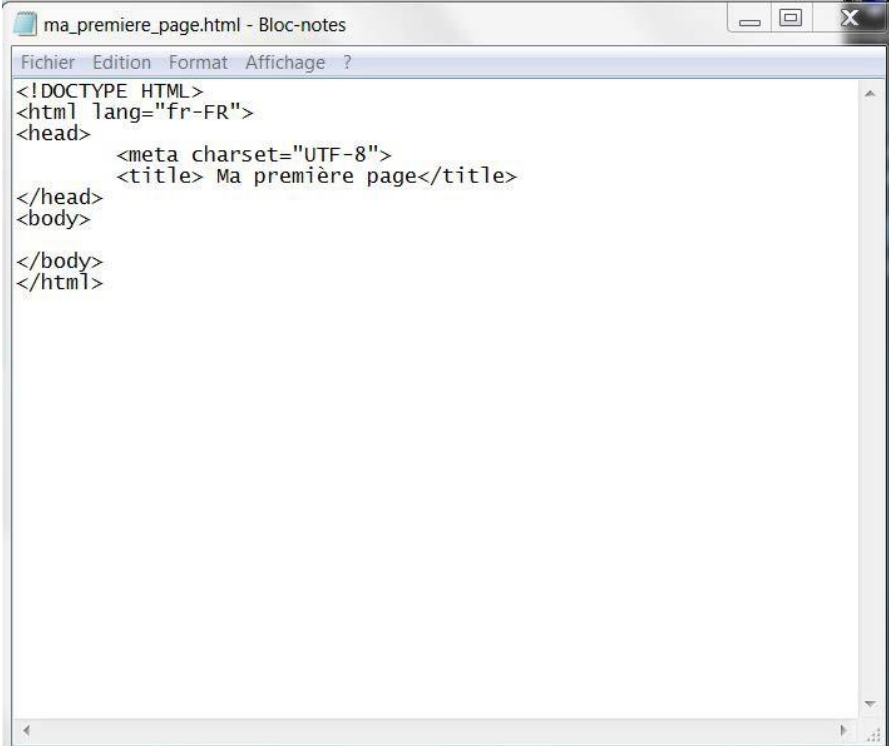
# Le WHATWG

- WHATWG : Web Hypertext Application Technology Working Group  
(2004)
  - Collaboration non officielle de développeurs de navigateurs
  - Tentative de réponse à la lenteur des standards du W3C
  - Mozilla Foundation, Opera, Apple, etc.

# Générer du HTML avec un éditeur

Blocnote ou TextEdit, si vous n'avez rien d'autre sous la main :

- Pas de coloration syntaxique
- Nécessite de connaître tout le langage



```
<!DOCTYPE HTML>
<html lang="fr-FR">
<head>
  <meta charset="UTF-8">
  <title> Ma première page</title>
</head>
<body>
</body>
</html>
```

# Brackets.io

- <http://brackets.io/>
- Éditeur spécialisé
- Auto-complétion, retour à la ligne, coloration syntaxique, plugins etc.
- Gratuit pour Mac et Windows – open source

 Brackets [Contribute](#) [Blog](#) [Support](#)



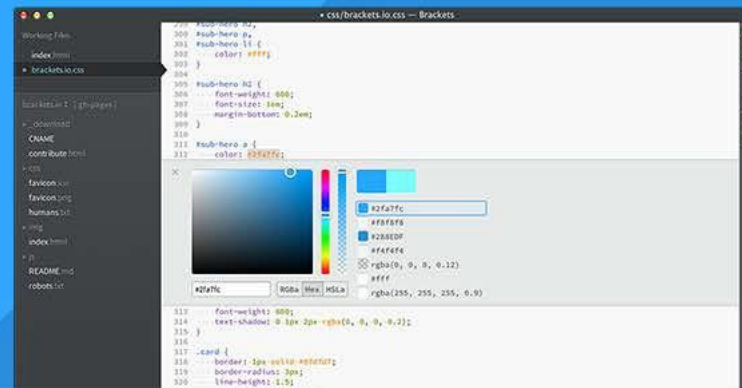
## Code the Web

Brackets is an open source code editor for web designers and front-end developers.

Download Brackets

Release 39 (WIN)

[Other Downloads](#) [Requirements](#) [Installation](#)



### Live HTML Development

As you code, HTML changes are instantly pushed to browser without having to save or reload the page.  
[See it in action.](#)

### JS Debugging with Theseus

Theseus makes inspecting variables and control flow easy, even in asynchronous code.

[Learn more.](#)

### Linux: New & Improved

Over the last few releases we've made progress on Linux (Debian/Ubuntu) support for

[Brackets.](#)

[Check it out.](#)

### Effective Development

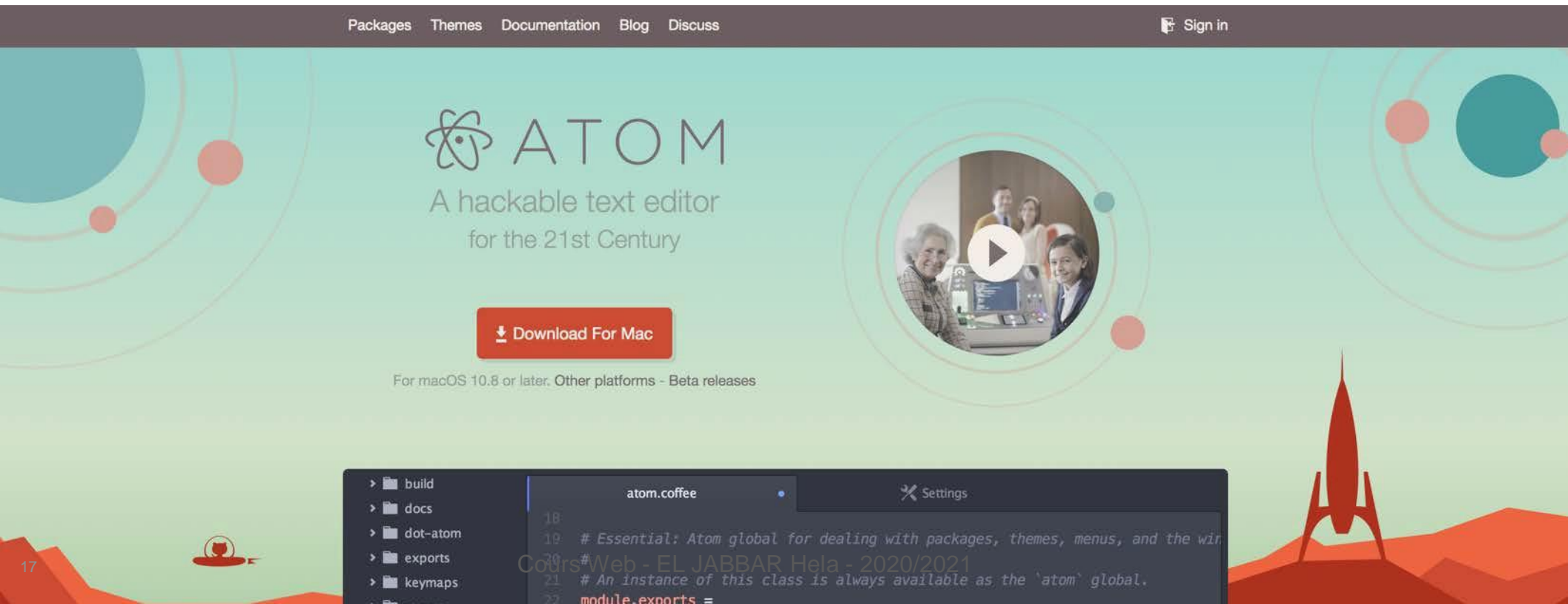
Imagine the ability to fluently edit external styles and code without even leaving your document!

[See how it looks.](#)



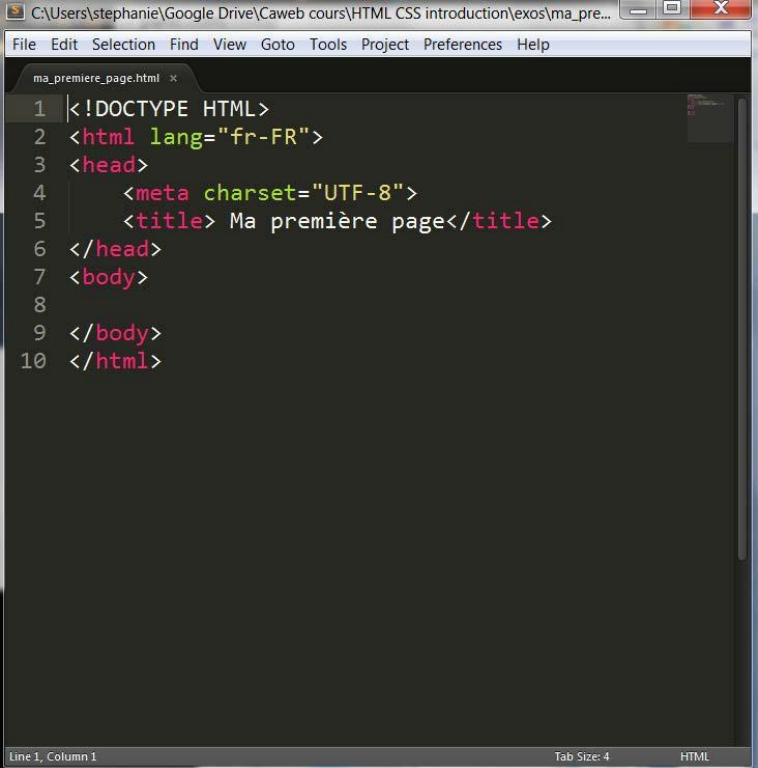
# Atom

- <https://atom.io/>
- Éditeur spécialisé
- Auto-complétion, retour à la ligne, coloration syntaxique, plugins etc.
- Gratuit pour Mac et Windows – open source



# Les éditeurs de texte plus avancés et spécialisés

- <http://www.sublimetext.com/>
- Éditeur spécialisé
- Auto-complétion, retour à la ligne, colorisation syntaxique
- Plugins, snippets, mode « sans distraction »
- 30 jours essai Windows/Mac/ Linux puis payant



```
1 <!DOCTYPE HTML>
2 <html lang="fr-FR">
3 <head>
4     <meta charset="UTF-8">
5     <title> Ma première page</title>
6 </head>
7 <body>
8
9 </body>
10 </html>
```

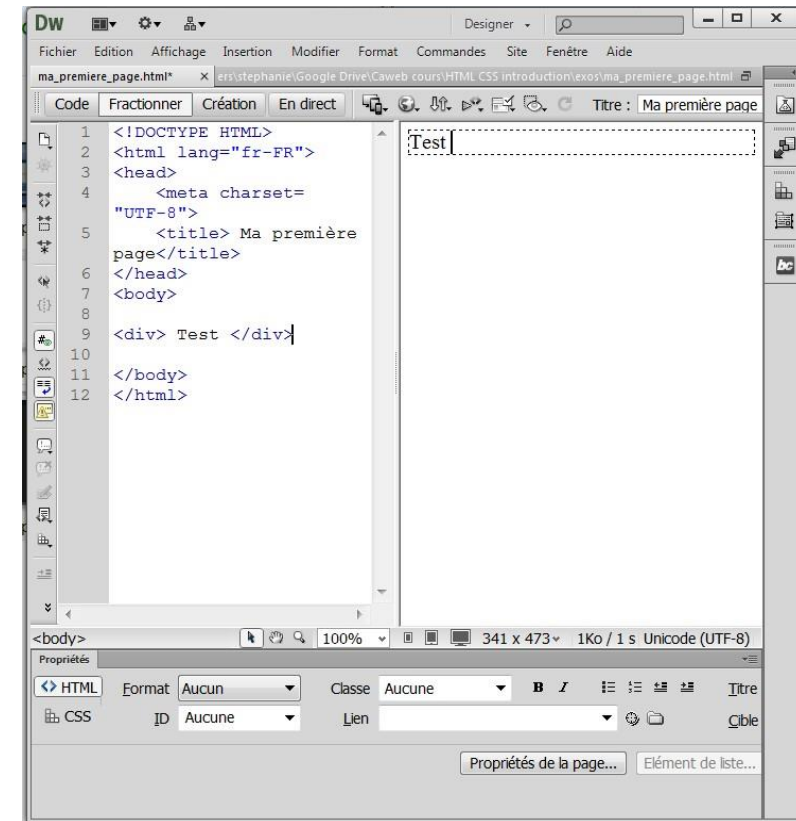
# D'autres éditeurs

- Coda : uniquement sur Mac (payant) <http://panic.com/coda/>
- Geany (Windows) gratuit <http://www.geany.org/Download/Releases>
- [Notepad ++](#)
- Komodo (Windows et Mac) <http://www.activestate.com/komodo-ide>
- Netbeans (Windows et Mac) gratuit <http://netbeans.org/>



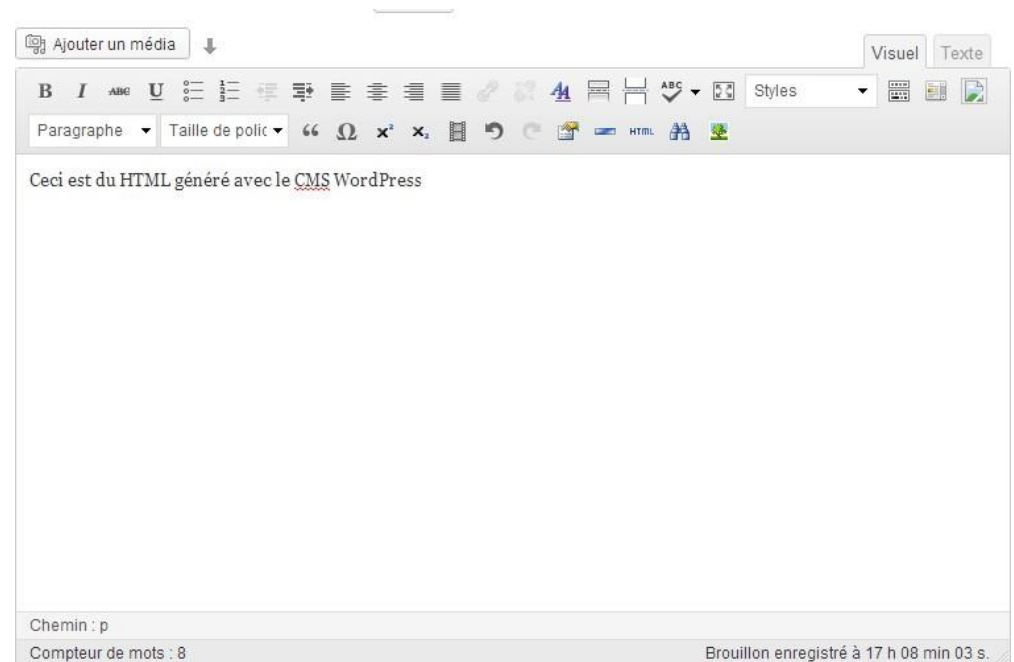
# Les éditeurs « WYSIWYG »

- What You See Is What You Get
- Permet de visualiser le rendu directement
- Code parfois pas toujours propre ou optimisé
- Exemple : Dreamweaver (payant), Blue Griffon (gratuit)



# Générer du HTML depuis un CMS

- CMS : Content Management System ou systèmes de gestion de contenu
- Développé dans un langage de programmation web (ex : PHP) et fait appel à une base de données (ex : SQL)
- Éditeurs « visuels » sans avoir besoin de notions de code
- Ex : WordPress, Drupal, Joomla, Prestashop, etc.

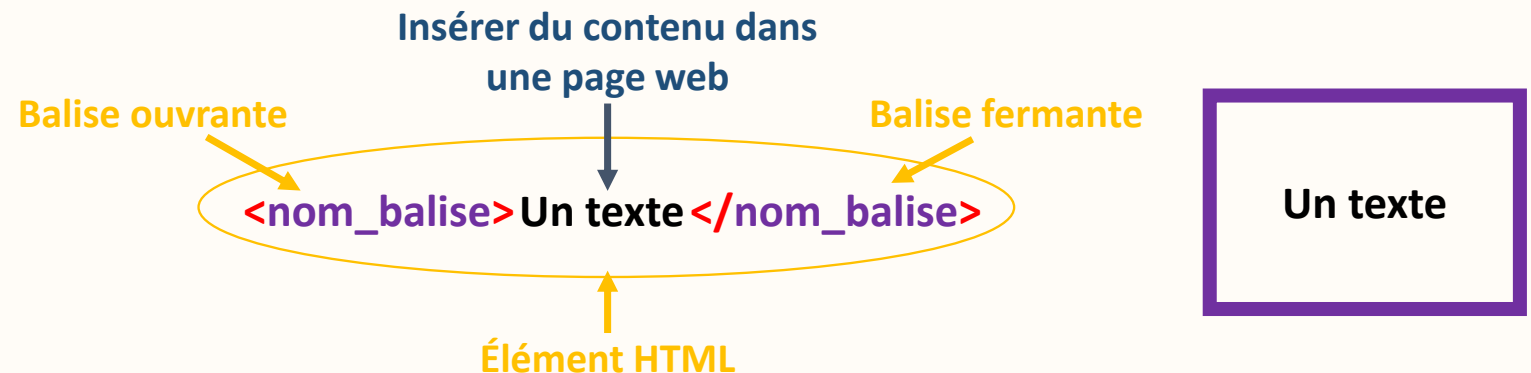


## Partie 1 - Le langage HTML

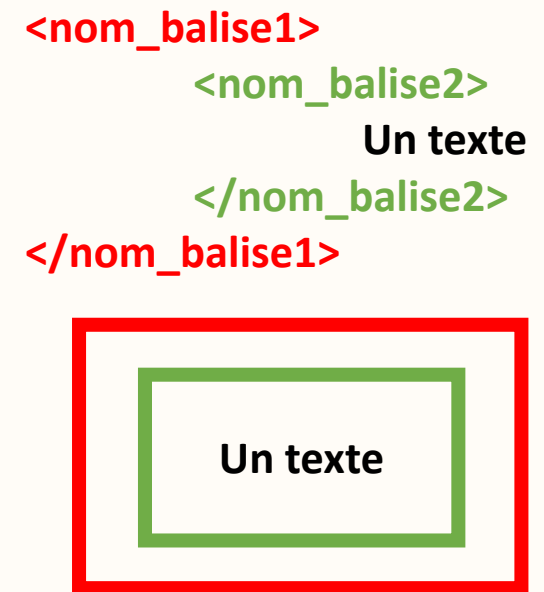
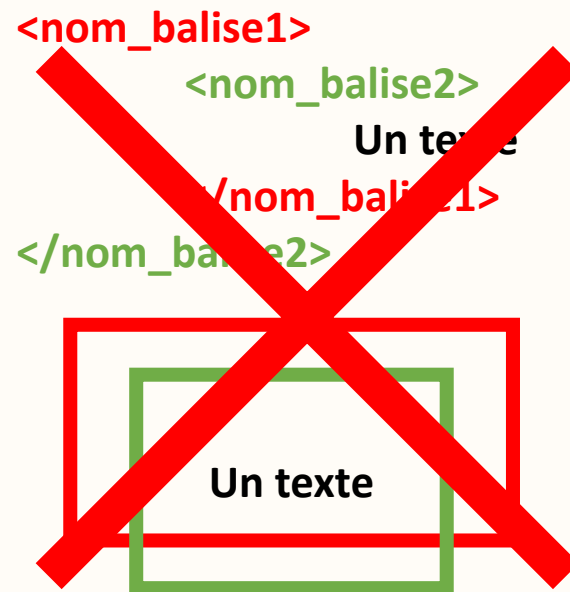
1. La Syntaxe du HTML
2. La structure d'une page HTML
3. Les commentaires
4. Les paragraphes
5. Les titres
6. La mise en valeur et de forme
7. Les balises universelles
8. L'élément de texte préformaté
9. Les codes informatiques
10. Les listes
11. Les liens
12. Les boutons
13. Les séparateurs
14. Les images et les figures
15. Les tableaux
16. Les formulaires
17. L'audio
18. La vidéo
19. Identification des balises
20. TP1 HTML

# La Syntaxe du HTML

## 1- Balises HTML:



## 2- Imbrication des balises HTML:



## Partie 1 - Le langage HTML

1. La Syntaxe du HTML
2. La structure d'une page HTML
3. Les commentaires
4. Les paragraphes
5. Les titres
6. La mise en valeur et de forme
7. Les balises universelles
8. L'élément de texte préformaté
9. Les codes informatiques
10. Les listes
11. Les liens
12. Les boutons
13. Les séparateurs
14. Les images et les figures
15. Les tableaux
16. Les formulaires
17. L'audio
18. La vidéo
19. Identification des balises
20. TP1 HTML

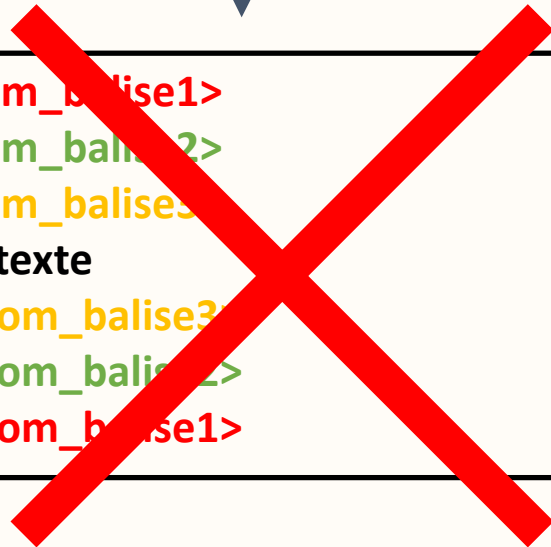
# La Syntaxe du HTML

## 3- L'indentation :

Code non indenté



```
<nom_balise1>  
<nom_balise2>  
<nom_balise3>  
Un texte  
</nom_balise3>  
</nom_balise2>  
</nom_balise1>
```



Code indenté



```
<nom_balise1>  
    <nom_balise2>  
        <nom_balise3>  
            Un texte  
        </nom_balise3>  
    </nom_balise2>  
</nom_balise1>
```

## 4- Déclaration HTML :

Un document HTML commence par une déclaration HTML qui indique qu'il s'agit bien d'une page web HTML :

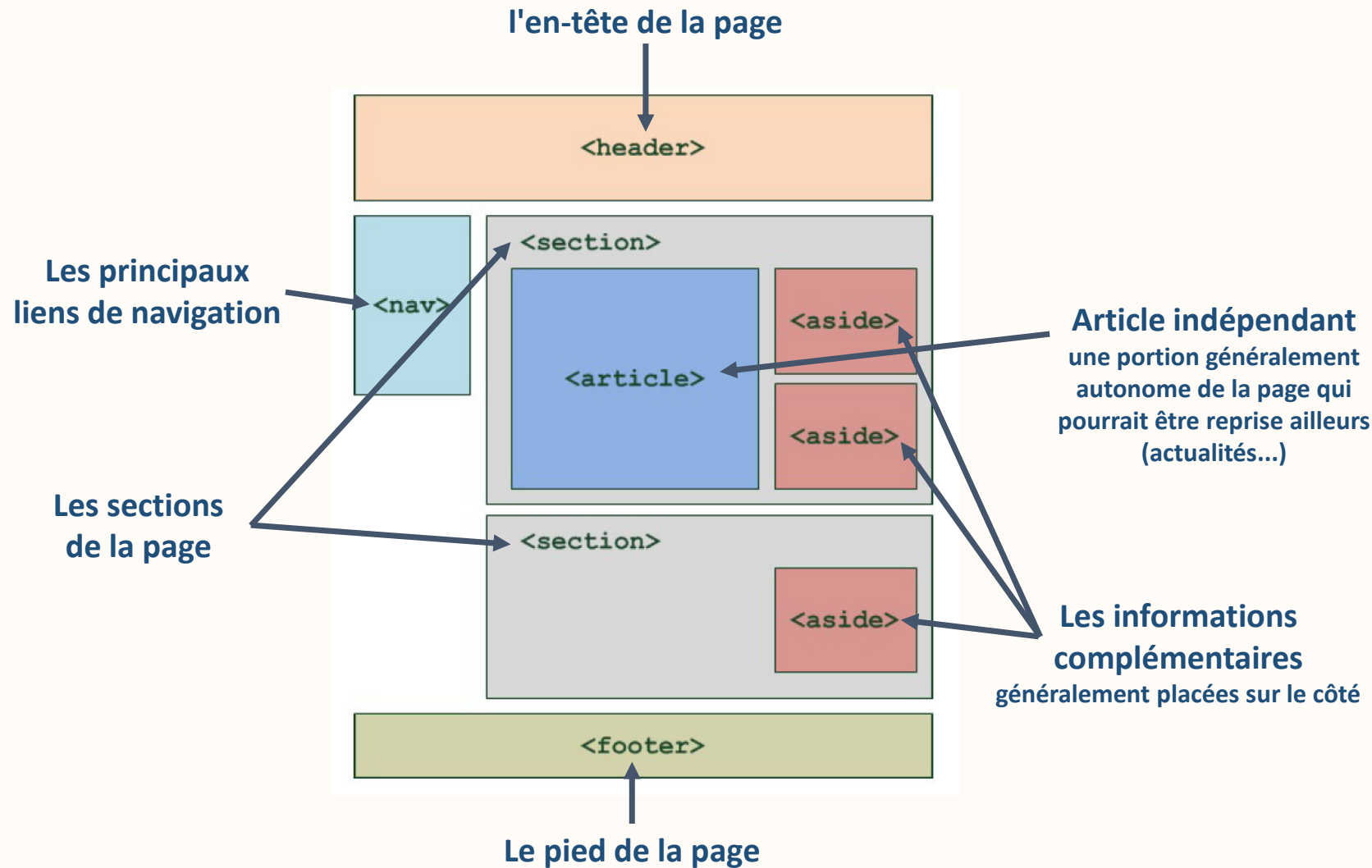
```
<!DOCTYPE html>
```

## Partie 1 - Le langage HTML

1. La Syntaxe du HTML
2. La structure d'une page HTML
3. Les commentaires
4. Les paragraphes
5. Les titres
6. La mise en valeur et de forme
7. Les balises universelles
8. L'élément de texte préformaté
9. Les codes informatiques
10. Les listes
11. Les liens
12. Les boutons
13. Les séparateurs
14. Les images et les figures
15. Les tableaux
16. Les formulaires
17. L'audio
18. La vidéo
19. Identification des balises
20. TP1 HTML

# La structure d'une page HTML

## 1- Structure générale:





## Partie 1 - Le langage HTML

1. La Syntaxe du HTML
- 2. La structure d'une page HTML**
3. Les commentaires
4. Les paragraphes
5. Les titres
6. La mise en valeur et de forme
7. Les balises universelles
8. L'élément de texte préformaté
9. Les codes informatiques
10. Les listes
11. Les liens
12. Les boutons
13. Les séparateurs
14. Les images et les figures
15. Les tableaux
16. Les formulaires
17. L'audio
18. La vidéo
19. Identification des balises
20. TP1 HTML

# La structure d'une page HTML

```
<!DOCTYPE html>
<html>
  <!--Balise de données générales du site-->
  <head>
    <meta charset="utf-8"> <!--Pour la prise en compte des caractères accentués -->
    <title>IFAA SUP</title> <!--Pour le titre de la page web-->
  </head>
  <!--Balise de contenu du site-->
  <body>
    <header>
      <!-- Placez ici le contenu de l'en-tête de la page -->
    </header>

    <nav>
      <!-- Placez ici les principaux liens de navigation de la page (menu) -->
    </nav>

    <section>
      <!-- Placez ici une section de la page-->

      <aside>
        <!-- Placez ici des informations complémentaires-->
      </aside>

      <article>
        <!-- Placez ici l'article indépendant de la page-->
      </article>
    </section>

    <footer>
      <!-- Placez ici le contenu du pied de page -->
    </footer>
  </body>
</html>
```

## Partie 1 - Le langage HTML

1. La Syntaxe du HTML
2. La structure d'une page HTML
3. Les commentaires
4. Les paragraphes
5. Les titres
6. La mise en valeur et de forme
7. Les balises universelles
8. L'élément de texte préformaté
9. Les codes informatiques
10. Les listes
11. Les liens
12. Les boutons
13. Les séparateurs
14. Les images et les figures
15. Les tableaux
16. Les formulaires
17. L'audio
18. La vidéo
19. Identification des balises
20. TP1 HTML

# Les commentaires en HTML

Ils permettent de décrire les lignes de code sans que les descriptions soient prises en compte lors de la visualisation de la page web.

➤ Un commentaire sur une ligne :

```
<!-- Voici un commentaire -->
```

➤ Un commentaire sur plusieurs lignes :

```
<!--  
Une première ligne  
Une deuxième ligne  
Une troisième ligne  
....  
-->
```

## Partie 1 - Le langage HTML

1. La Syntaxe du HTML
2. La structure d'une page HTML
3. Les commentaires
4. Les paragraphes
5. Les titres
6. La mise en valeur et de forme
7. Les balises universelles
8. L'élément de texte préformaté
9. Les codes informatiques
10. Les listes
11. Les liens
12. Les boutons
13. Les séparateurs
14. Les images et les figures
15. Les tableaux
16. Les formulaires
17. L'audio
18. La vidéo
19. Identification des balises
20. TP1 HTML

# Les paragraphes

## 1- Balise de structuration du texte en paragraphes - `<p>...</p>`:

Elle permet d'organiser le texte en paragraphes.

`<p>`

HTML (HyperText Markup Language) : il a fait son apparition dès 1991 lors du lancement du Web. Son rôle est de gérer et organiser le contenu. C'est donc en HTML que vous écrirez ce que vous souhaitez que la page affiche : du texte, des liens, des images...

`</p>`

`<p>`

CSS (Cascading Style Sheets, aussi appelées Feuilles de style) : le rôle du CSS est de gérer l'apparence de la page web (agencement, positionnement, décoration, couleur, taille du texte...). Ce langage est venu compléter le HTML en 1996.

`</p>`

## 2- Balise de retour à la ligne - `<br/>`:

Les retours à la ligne dans les textes d'une page HTML sont considérés comme un espace simple. Pour insérer un retour à la ligne, il faut utiliser la balise `<br/>`.

`<p>`

Paragraphe 1.

`<br/>`

Paragraphe 2.

`</p>`

## Partie 1 - Le langage HTML

1. La Syntaxe du HTML
2. La structure d'une page HTML
3. Les commentaires
4. Les paragraphes
- 5. Les titres**
6. La mise en valeur et de forme
7. Les balises universelles
8. L'élément de texte préformaté
9. Les codes informatiques
10. Les listes
11. Les liens
12. Les boutons
13. Les séparateurs
14. Les images et les figures
15. Les tableaux
16. Les formulaires
17. L'audio
18. La vidéo
19. Identification des balises
20. TP1 HTML

# Les titres

On a six niveaux de titres selon l'importance du titre :

|                                       |                                                                                    |
|---------------------------------------|------------------------------------------------------------------------------------|
| <code>&lt;h1&gt;...&lt;/h1&gt;</code> | <code>&lt;!--Titre très important.--&gt;</code>                                    |
| <code>&lt;h2&gt;...&lt;/h2&gt;</code> | <code>&lt;!--Titre important.--&gt;</code>                                         |
| <code>&lt;h3&gt;...&lt;/h3&gt;</code> | <code>&lt;!--Titre un peu moins important (on peut dire "sous-titre".--&gt;</code> |
| <code>&lt;h4&gt;...&lt;/h4&gt;</code> | <code>&lt;!--Titre encore moins important.--&gt;</code>                            |
| <code>&lt;h5&gt;...&lt;/h5&gt;</code> | <code>&lt;!--Titre pas important.--&gt;</code>                                     |
| <code>&lt;h6&gt;...&lt;/h6&gt;</code> | <code>&lt;!--Titre vraiment pas important du tout.--&gt;</code>                    |

## Partie 1 - Le langage HTML

1. La Syntaxe du HTML
2. La structure d'une page HTML
3. Les commentaires
4. Les paragraphes
5. Les titres
6. La mise en valeur et de forme
7. Les balises universelles
8. L'élément de texte préformaté
9. Les codes informatiques
10. Les listes
11. Les liens
12. Les boutons
13. Les séparateurs
14. Les images et les figures
15. Les tableaux
16. Les formulaires
17. L'audio
18. La vidéo
19. Identification des balises
20. TP1 HTML

# La mise en valeur et de forme

## 1- La mise en valeur du texte :

|                                               |                                                                                        |
|-----------------------------------------------|----------------------------------------------------------------------------------------|
| <code>&lt;strong&gt;...&lt;/strong&gt;</code> | <code>&lt;!--Pour mettre bien en valeur une portion de texte (fort).--&gt;</code>      |
| <code>&lt;em&gt;...&lt;/em&gt;</code>         | <code>&lt;!--Pour mettre un peu en valeur une portion de texte.--&gt;</code>           |
| <code>&lt;mark&gt;...&lt;/mark&gt;</code>     | <code>&lt;!--Permet de faire ressortir visuellement une portion de texte.--&gt;</code> |

## 2- La mise en forme du texte :

|                                               |                                                                              |
|-----------------------------------------------|------------------------------------------------------------------------------|
| <code>&lt;!--Centrer un bloc:--&gt;</code>    |                                                                              |
| <code>&lt;center&gt;</code>                   | Le contenu à centrer                                                         |
| <code>&lt;/center&gt;</code>                  |                                                                              |
| <code>&lt;!--Mettre en gras:--&gt;</code>     |                                                                              |
| <code>&lt;b&gt;</code>                        | Le contenu à mettre en gras                                                  |
| <code>&lt;/b&gt;</code>                       |                                                                              |
| <code>&lt;!--Mettre en italique:--&gt;</code> |                                                                              |
| <code>&lt;i&gt;</code>                        | Le contenu à mettre en italique                                              |
| <code>&lt;/i&gt;</code>                       |                                                                              |
| <code>&lt;!--Souligner:--&gt;</code>          |                                                                              |
| <code>&lt;u&gt;</code>                        | Le contenu à souligner                                                       |
| <code>&lt;/u&gt;</code>                       |                                                                              |
| <code>&lt;!--Changer la police:--&gt;</code>  |                                                                              |
| <code>&lt;font&gt;</code>                     | Le contenu à qui on applique la police                                       |
| <code>&lt;/font&gt;</code>                    |                                                                              |
| <code>&lt;!--</code>                          |                                                                              |
| Attributs:                                    |                                                                              |
| <code>color="red"</code>                      | Indique la couleur à appliquer au texte. Ici en rouge.                       |
| <code>size=2</code>                           | Spécifie la taille de la police, valeur allant de 1 à 7.                     |
| <code>size=+2</code>                          | Augmente la taille de la police de façon relative. La taille varie de 1 à 7. |
| <code>face="Comic Sans MS"</code>             | Liste des polices de caractères à utiliser par ordre d'apparition.           |
| <code>--&gt;</code>                           |                                                                              |

## Partie 1 - Le langage HTML

1. La Syntaxe du HTML
2. La structure d'une page HTML
3. Les commentaires
4. Les paragraphes
5. Les titres
6. La mise en valeur et de forme
- 7. Les balises universelles**
8. L'élément de texte préformaté
9. Les codes informatiques
10. Les listes
11. Les liens
12. Les boutons
13. Les séparateurs
14. Les images et les figures
15. Les tableaux
16. Les formulaires
17. L'audio
18. La vidéo
19. Identification des balises
20. TP1 HTML

## Les balises universelles

Elles n'ont pas de sens mais servent à englober du HTML pour pouvoir lui donner une classe ou un identifiant pour lui appliquer du CSS quand aucune autre balise ne convient.

`<span>...</span>`

`<div>...</div>`

`<!--Balise en ligne (inline). Se met au milieu du texte.-->`

`<!--Balise en bloc (block). Se met sous forme de bloc.-->`

## Partie 1 - Le langage HTML

1. La Syntaxe du HTML
2. La structure d'une page HTML
3. Les commentaires
4. Les paragraphes
5. Les titres
6. La mise en valeur et de forme
7. Les balises universelles
8. L'élément de texte préformaté
9. Les codes informatiques
10. Les listes
11. Les liens
12. Les boutons
13. Les séparateurs
14. Les images et les figures
15. Les tableaux
16. Les formulaires
17. L'audio
18. La vidéo
19. Identification des balises
20. TP1 HTML

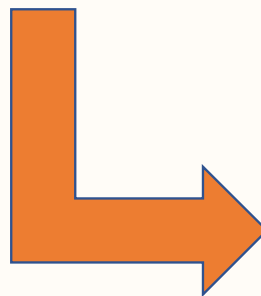
## L'élément de texte préformaté

La balise `<pre>...</pre>` permet de respecter la mise en page précise d'un texte de la façon dont elle a été définie dans le fichier HTML en prenant en compte les espaces, les tabulations, les retours à la ligne etc.

`<pre>`

```
L      TE
A      A
C  V
R A
QUE TU
PORTES
```

`</pre>`



```
L      TE
A      A
C  V
R A
QUE TU
PORTES
```

## Partie 1 - Le langage HTML

1. La Syntaxe du HTML
2. La structure d'une page HTML
3. Les commentaires
4. Les paragraphes
5. Les titres
6. La mise en valeur et de forme
7. Les balises universelles
8. L'élément de texte préformaté
- 9. Les codes informatiques**
10. Les listes
11. Les liens
12. Les boutons
13. Les séparateurs
14. Les images et les figures
15. Les tableaux
16. Les formulaires
17. L'audio
18. La vidéo
19. Identification des balises
20. TP1 HTML

# Les codes informatiques

On utilise la balise `<code>...</code>` pour les programmes informatiques.

```
<code>
```

Un programme informatique

```
</code>
```



## Partie 1 - Le langage HTML

1. La Syntaxe du HTML
2. La structure d'une page HTML
3. Les commentaires
4. Les paragraphes
5. Les titres
6. La mise en valeur et de forme
7. Les balises universelles
8. L'élément de texte préformaté
9. Les codes informatiques
- 10. Les listes**
11. Les liens
12. Les boutons
13. Les séparateurs
14. Les images et les figures
15. Les tableaux
16. Les formulaires
17. L'audio
18. La vidéo
19. Identification des balises
20. TP1 HTML

# Les listes

## 1- Liste non ordonnée :

Permet de créer une liste à puces non ordonnée.

```
<ul>
  <li>Elément 1 de la liste</li> <!--Délimite un élément de la liste (une puce).-->
  <li>Elément 2 de la liste</li> <!--Délimite un élément de la liste (une puce).-->
  <li>Etc...</li> <!--Délimite un élément de la liste (une puce).-->
</ul>
<!--
Attributs de la balise <ul type="..."> :
    type=          Définit le style de puce utilisé pour la liste. Il peut prendre les valeurs:
                    "disc"      Une puce sous forme d'un disque (par défaut).
                    "circle"    Une puce sous forme d'un cercle
                    "square"    Une puce sous forme d'un carré.
                    "none"      Les éléments de la liste ne seront pas marqués.
-->!
```

## Partie 1 - Le langage HTML

1. La Syntaxe du HTML
2. La structure d'une page HTML
3. Les commentaires
4. Les paragraphes
5. Les titres
6. La mise en valeur et de forme
7. Les balises universelles
8. L'élément de texte préformaté
9. Les codes informatiques
- 10. Les listes**
11. Les liens
12. Les boutons
13. Les séparateurs
14. Les images et les figures
15. Les tableaux
16. Les formulaires
17. L'audio
18. La vidéo
19. Identification des balises
20. TP1 HTML

# Les listes

## 2- Liste ordonnée :

Permet de créer une liste à puces ordonnée.

```
<ol>
  <li>Elément 1 de la liste</li> <!--Délimite un élément de la liste (une puce).-->
  <li>Elément 2 de la liste</li> <!--Délimite un élément de la liste (une puce).-->
  <li>Etc...</li> <!--Délimite un élément de la liste (une puce).-->
```

```
</ol>
```

```
<!--
```

Attributs de la balise `<ol type="..." start="..." >` :

|                         |                                                                     |
|-------------------------|---------------------------------------------------------------------|
| <code>type=</code>      | Définit le type du marqueur des éléments de la liste.               |
|                         | Il peut prendre les valeurs:                                        |
| <code>"1"</code>        | Les éléments seront numérotés avec des numéros (par défaut).        |
| <code>"A"</code>        | Les éléments seront numérotés avec des lettres majuscules.          |
| <code>"a"</code>        | Les éléments seront numérotés avec des lettres minuscules.          |
| <code>"I"</code>        | Les éléments seront numérotés avec des chiffres romains majuscules. |
| <code>"i"</code>        | Les éléments seront numérotés avec des chiffres romains minuscules. |
| <code>start="11"</code> | Permet de préciser la valeur de départ de la numérotation (ici 11). |

```
-->!
```

## Partie 1 - Le langage HTML

1. La Syntaxe du HTML
2. La structure d'une page HTML
3. Les commentaires
4. Les paragraphes
5. Les titres
6. La mise en valeur et de forme
7. Les balises universelles
8. L'élément de texte préformaté
9. Les codes informatiques
- 10. Les listes**
11. Les liens
12. Les boutons
13. Les séparateurs
14. Les images et les figures
15. Les tableaux
16. Les formulaires
17. L'audio
18. La vidéo
19. Identification des balises
20. TP1 HTML

# Les listes

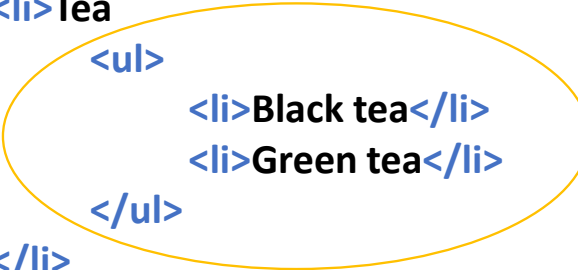
## 3- Listes imbriquées :

On peut mettre des listes à l'intérieur des listes :

```
<ul>
  <li>Coffee</li>

  <li>Tea
    <ul>
      <li>Black tea</li>
      <li>Green tea</li>
    </ul>
  </li>

  <li>Milk</li>
</ul>
```



## Partie 1 - Le langage HTML

1. La Syntaxe du HTML
2. La structure d'une page HTML
3. Les commentaires
4. Les paragraphes
5. Les titres
6. La mise en valeur et de forme
7. Les balises universelles
8. L'élément de texte préformaté
9. Les codes informatiques
10. Les listes
11. Les liens
12. Les boutons
13. Les séparateurs
14. Les images et les figures
15. Les tableaux
16. Les formulaires
17. L'audio
18. La vidéo
19. Identification des balises
20. TP1 HTML

# Les liens

## 1- Un lien vers un autre site :

```
<a href="https://www.ifaa-sup.fr/">IFAA SUP</a>
```

```
<!--
```

Attributs:

href="www.x.com"

Pour indiquer l'URL vers quelle page amener.

title="Le site de l'IFAA"

Pour afficher l'infobulle quand on pointe l'url.

target="\_blanc"

Pour forcer l'ouverture du lien dans un nouveau onglet.

```
-->
```

## 2- Ouvrir la messagerie avec l'adresse indiquée :

```
<a href="mailto:xxx@yahoo.fr">Contactez moi</a>
```

## 3- Rediriger tout en haut de la page Web :

```
<a href="#">Aller vers le haut</a>
```

## 4- Un lien vers une ancre :

On l'utilise quand on a une page très longue et qu'on veut aller directement à la portion définie par l'id "ancree".

```
<a href="#ancree">Aller vers le haut</a>
```

On donne l'identifiant à la balise du portion de texte vers où on veut se rediriger :

```
<h2 id="ancree">Titre vers où se rediriger</h2>
```

## Partie 1 - Le langage HTML

1. La Syntaxe du HTML
2. La structure d'une page HTML
3. Les commentaires
4. Les paragraphes
5. Les titres
6. La mise en valeur et de forme
7. Les balises universelles
8. L'élément de texte préformaté
9. Les codes informatiques
10. Les listes
- 11. Les liens**
12. Les boutons
13. Les séparateurs
14. Les images et les figures
15. Les tableaux
16. Les formulaires
17. L'audio
18. La vidéo
19. Identification des balises
20. TP1 HTML

# Les liens

## 5- Un lien vers une autre page du site :

- Si la page se trouve sur le même dossier, alors on donne directement le nom de la page:

```
<a href="page.html">Page demandée</a>
```

- Si la page ne se trouve pas sur le même dossier alors on précise le chemin où se trouve la page:

- la page se trouve dans un dossier à coté de la page appelante :

```
<a href="dossier/page.html">Page demandée</a>
```

- la page se trouve dans un dossier dans le dossier parent de la page appelante :

```
<a href="../dossier/page.html">Page demandée</a>
```

## Partie 1 - Le langage HTML

1. La Syntaxe du HTML
2. La structure d'une page HTML
3. Les commentaires
4. Les paragraphes
5. Les titres
6. La mise en valeur et de forme
7. Les balises universelles
8. L'élément de texte préformaté
9. Les codes informatiques
10. Les listes
11. Les liens
- 12. Les boutons**
13. Les séparateurs
14. Les images et les figures
15. Les tableaux
16. Les formulaires
17. L'audio
18. La vidéo
19. Identification des balises
20. TP1 HTML

## Les boutons

- L'élément HTML « button » est utilisé afin de créer un contrôle interactif ayant la forme d'un bouton et qui pourra être utilisé dans un formulaire ou dans le document.

```
<button>bouton</button>
```

- Par défaut, les boutons HTML sont mis en forme avec les styles natifs provenant du système d'exploitation mais leur apparence peut être adaptée grâce à CSS.

## Partie 1 - Le langage HTML

1. La Syntaxe du HTML
2. La structure d'une page HTML
3. Les commentaires
4. Les paragraphes
5. Les titres
6. La mise en valeur et de forme
7. Les balises universelles
8. L'élément de texte préformaté
9. Les codes informatiques
10. Les listes
11. Les liens
12. Les boutons
- 13. Les séparateurs**
14. Les images et les figures
15. Les tableaux
16. Les formulaires
17. L'audio
18. La vidéo
19. Identification des balises
20. TP1 HTML

# Les séparateurs

Ils permettent d'insérer une ligne horizontale de séparation.

```
<hr width=800 size=5 color="blue" align="center" noshade />
```

```
<!--
```

Attributs:

|              |                                                                                                         |
|--------------|---------------------------------------------------------------------------------------------------------|
| width=800    | Définit la largeur de la ligne.                                                                         |
| color="blue" | Change la couleur de la ligne. Les couleurs sont en anglais.                                            |
| size=5       | L'épaisseur de la ligne. On doit donner une couleur pour ne pas avoir un rectangle au lieu d'une ligne. |
| align=       | Positionne la ligne dans la fenêtre. Il peut prendre les valeurs:                                       |
| "left"       | Positionne la ligne à gauche (par défaut).                                                              |
| "right"      | Positionne la ligne à droite.                                                                           |
| "center"     | Positionne la ligne au centre.                                                                          |
| noshade      | Permet de supprimer l'effet de relief du trait.                                                         |

```
-->
```

## Partie 1 - Le langage HTML

1. La Syntaxe du HTML
2. La structure d'une page HTML
3. Les commentaires
4. Les paragraphes
5. Les titres
6. La mise en valeur et de forme
7. Les balises universelles
8. L'élément de texte préformaté
9. Les codes informatiques
10. Les listes
11. Les liens
12. Les boutons
13. Les séparateurs
- 14. Les images et les figures**
15. Les tableaux
16. Les formulaires
17. L'audio
18. La vidéo
19. Identification des balises
20. TP1 HTML

# Les images et les figures

## 1- Les images :

```

```

```
<!--
```

### Attributs:

src="img.jpg"

Il permet d'indiquer où se trouve l'image (url de l'image).

alt="Paysage"

Le texte alternatif qui va remplacer l'image s'elle est introuvable.

title="Paysage"

Le texte qui s'affiche au survole de l'image

hspace=5

Permet de laisser de l'espace à gauche et à droite de l'image (horizontal). Ici 5 pixels autour de l'image.

vspace=5

Permet de laisser de l'espace en haut et en bas de l'image (vertical). Ici 5 pixels autour de l'image.

width=800

Définit la largeur de l'image.

height=800

Définit la hauteur de l'image.

```
-->
```

L'image doit être placée dans un bloc afin qu'elle ne se mette pas au milieu du texte :

```
<div>
```

```

```

```
</div>
```



## Partie 1 - Le langage HTML

1. La Syntaxe du HTML
2. La structure d'une page HTML
3. Les commentaires
4. Les paragraphes
5. Les titres
6. La mise en valeur et de forme
7. Les balises universelles
8. L'élément de texte préformaté
9. Les codes informatiques
10. Les listes
11. Les liens
12. Les boutons
13. Les séparateurs
- 14. Les images et les figures**
15. Les tableaux
16. Les formulaires
17. L'audio
18. La vidéo
19. Identification des balises
20. TP1 HTML

# Les images et les figures

## 2- Insertion d'une figure (image) avec une légende :

```
<figure>  
      
    <figcaption>la légende du figure ou de l'image qui représente la figure</figcaption>  
</figure>
```

## Partie 1 - Le langage HTML

1. La Syntaxe du HTML
2. La structure d'une page HTML
3. Les commentaires
4. Les paragraphes
5. Les titres
6. La mise en valeur et de forme
7. Les balises universelles
8. L'élément de texte préformaté
9. Les codes informatiques
10. Les listes
11. Les liens
12. Les boutons
13. Les séparateurs
14. Les images et les figures
- 15. Les tableaux**
16. Les formulaires
17. L'audio
18. La vidéo
19. Identification des balises
20. TP1 HTML

# Les tableaux

## 1- Les tableaux non structurés :

```
<table>
  <!--La balise <caption> permet d'ajouter un titre au tableau : -->
  <caption>Titre</caption>
  <!-- La balise <tr> permet d'ajouter la ligne d'entête du tableau (on utilise <th> pour les cellules) : -->
  <tr>
    <th>Texte de la cellule</th> <!--Produit une cellule d'entête du tableau-->
  </tr>
  <!--Ajouter une ligne du contenu du tableau (on utilise <td> pour les cellules) : -->
  <tr>
    <td>Texte de la cellule</td> <!--Produit une cellule simple du tableau-->
  </tr>
</table>
<!--
Attributs:
    align=          Positionne la table dans la page horizontalement. Il peut prendre les valeurs:
                     "left"      - Positionne la table à gauche (par défaut).
                     "center"    - Positionne la table au centre.
                     "right"     - Positionne la table à droite.
    bgcolor="red"    Donne une couleur d'arrière plan à la table.
    border=2         Définit l'épaisseur de la bordure de la table avec ses lignes et ses colonnes.
    cellpadding=5    Définit la taille de l'espace entre le contenu d'une cellule et sa bordure.
    cellspacing=5    Définit la taille de l'espace (vertical et horizontal) entre deux cellules.
    width=800        Définit la largeur du tableau.
    height=200       Définit la hauteur du tableau.
-->
```

## Partie 1 - Le langage HTML

1. La Syntaxe du HTML
2. La structure d'une page HTML
3. Les commentaires
4. Les paragraphes
5. Les titres
6. La mise en valeur et de forme
7. Les balises universelles
8. L'élément de texte préformaté
9. Les codes informatiques
10. Les listes
11. Les liens
12. Les boutons
13. Les séparateurs
14. Les images et les figures
15. Les tableaux
16. Les formulaires
17. L'audio
18. La vidéo
19. Identification des balises
20. TP1 HTML

# Les tableaux

➤ Disposer le texte dans sa cellule <th> et <td>:

```
<th align="center" valign="center">Centre le titre horizontalement et verticalement dans la cellule.</th>  
<td align="center" valign="center">Centre le texte horizontalement et verticalement dans la cellule.</td>  
<!--
```

Attributs:

|               |                                                                                   |
|---------------|-----------------------------------------------------------------------------------|
| align=        | Positionne le texte dans la cellule horizontalement. Il peut prendre les valeurs: |
| "left"        | - Positionne le texte à gauche (par défaut).                                      |
| "center"      | - Positionne le texte au centre.                                                  |
| "right"       | - Positionne le texte à droite.                                                   |
| valign=       | Positionne le texte dans la cellule verticalement. Il peut prendre les valeurs:   |
| "top"         | - Positionne le texte en haut.                                                    |
| "middle"      | - Positionne le texte au centre (par défaut).                                     |
| "bottom"      | - Positionne le texte au bas.                                                     |
| colspan="nbr" | Permet la fusion des cellules en colonne de nombre 'nbr'.                         |
| rowspan="nbr" | Permet la fusion des cellules en ligne de nombre 'nbr'.                           |

-->

## Partie 1 - Le langage HTML

1. La Syntaxe du HTML
2. La structure d'une page HTML
3. Les commentaires
4. Les paragraphes
5. Les titres
6. La mise en valeur et de forme
7. Les balises universelles
8. L'élément de texte préformaté
9. Les codes informatiques
10. Les listes
11. Les liens
12. Les boutons
13. Les séparateurs
14. Les images et les figures
- 15. Les tableaux**
16. Les formulaires
17. L'audio
18. La vidéo
19. Identification des balises
20. TP1 HTML

# Les tableaux

## 2- Les tableaux structurés :

```
<table>
  <!--La balise <caption> permet d'ajouter un titre au tableau : -->
  <caption>Titre</caption>

  <!--La balise <thead> permet de définir l'en-tête du tableau (la partie haute) :-->
  <thead>
    <tr>
      <th>Texte de la cellule</th> <!--Produit une cellule d'entête du tableau-->
    </tr>
  </thead>

  <!--La balise <tbody> permet de définir le corps du tableau (la partie centrale) :-->
  <tbody>
    <tr>
      <td>Texte de la cellule</td> <!--Produit une cellule simple du tableau-->
    </tr>
  </tbody>

  <!--La balise <tfoot> permet de définir le pied du tableau (la partie basse) :-->
  <tfoot>
    <tr>
      <td>Texte de la cellule</td> <!--Produit une cellule simple du tableau-->
    </tr>
  </tfoot>
</table>
```

## Partie 1 - Le langage HTML

1. La Syntaxe du HTML
2. La structure d'une page HTML
3. Les commentaires
4. Les paragraphes
5. Les titres
6. La mise en valeur et de forme
7. Les balises universelles
8. L'élément de texte préformaté
9. Les codes informatiques
10. Les listes
11. Les liens
12. Les boutons
13. Les séparateurs
14. Les images et les figures
15. Les tableaux
- 16. Les formulaires**
17. L'audio
18. La vidéo
19. Identification des balises
20. TP1 HTML

# Les formulaires

## 1- La balise principale du formulaire :

La balise `<form>...</form>` est la balise principale qui englobe le formulaire :

```
<form method="post" action="page.php" enctype="multipart/form-data">
  <!--Les champs de saisie :-->
  <input />
  ....

  <!--Le bouton d'envoi du formulaire :-->
  <input type="submit" name="submit" value="Envoyer" />
</form>
<!--
```

### Attributs:

|                                            |                                                                                                                                                                             |
|--------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>method=</code>                       | Indique par quel moyen les données vont être envoyées.<br>Elle peut prendre les valeurs:                                                                                    |
| <code>"get"</code>                         | Les données transiteront par l'URL qu'on peut récupérer grâce à l'array <code>\$_GET</code> . Il vaut mieux avoir un URL inférieur à 256 caractères.                        |
| <code>"post"</code>                        | Les données ne transiteront pas par l'URL, l'utilisateur ne les verra donc pas passer dans la barre d'adresse. On peut les récupérer grâce à l'array <code>\$_POST</code> . |
| <code>action="page.php"</code>             | Définit la page qui sera chargée lors de l'envoi du formulaire. Elle recevra les données saisies et les traitera.                                                           |
| <code>enctype="multipart/form-data"</code> | On l'ajoute si le formulaire propose aux visiteurs d'envoyer un fichier.                                                                                                    |

-->

## Partie 1 - Le langage HTML

1. La Syntaxe du HTML
2. La structure d'une page HTML
3. Les commentaires
4. Les paragraphes
5. Les titres
6. La mise en valeur et de forme
7. Les balises universelles
8. L'élément de texte préformaté
9. Les codes informatiques
10. Les listes
11. Les liens
12. Les boutons
13. Les séparateurs
14. Les images et les figures
15. Les tableaux
- 16. Les formulaires**
17. L'audio
18. La vidéo
19. Identification des balises
20. TP1 HTML

# Les formulaires

## 2- Les champs `<input />` :

`<input />`

`<!--`

Attributs:

`name="xxx"`

Le nom du champs qu'on va utiliser pour récupérer la saisie de l'utilisateur dans la partie traitement.

`id="xxx"`

L'identifiant qui identifie la balise, pour faire le lien avec le label correspondant ou/et pour lui appliquer mise en forme.

`type="..."`

Le type du champ. Le type de la saisie utilisateur dans le champ.

`size="1px"`

Permet d'agrandir le champ.

`maxlength=8`

Sert à limiter le nombre de caractères que l'on peut saisir.

`value="val"`

Sert à préremplir le champ avec une valeur par défaut val.

`placeholder="val"`

Donne une indication sur le contenu du champ. Elle disparaîtra dès que le visiteur aura cliqué sur le champ.

`required="required"`

Pour rendre le champ du formulaire obligatoire.

`autofocus`

Permet de placer automatiquement le curseur dans le champ du formulaire.

`disabled`

Permet de rendre le champ inactif.

`-->`

## Partie 1 - Le langage HTML

1. La Syntaxe du HTML
2. La structure d'une page HTML
3. Les commentaires
4. Les paragraphes
5. Les titres
6. La mise en valeur et de forme
7. Les balises universelles
8. L'élément de texte préformaté
9. Les codes informatiques
10. Les listes
11. Les liens
12. Les boutons
13. Les séparateurs
14. Les images et les figures
15. Les tableaux
- 16. Les formulaires**
17. L'audio
18. La vidéo
19. Identification des balises
20. TP1 HTML

# Les formulaires

## 3- Les types des champs <input /> :

### A- Le champ de saisi de texte :

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

### B- Le champ de saisi de mot de passe :

```
<input type="password" />
```

### C- Le champ de saisi d'email :

```
<input type="email" />
```

### D- Le champ de saisi d'url :

```
<input type="url" />
```

### E- Le champ de saisi de numéro de téléphone :

```
<input type="tel" />
```

### F- Le champ de recherche :

```
<input type="search" />
```

## Partie 1 - Le langage HTML

1. La Syntaxe du HTML
2. La structure d'une page HTML
3. Les commentaires
4. Les paragraphes
5. Les titres
6. La mise en valeur et de forme
7. Les balises universelles
8. L'élément de texte préformaté
9. Les codes informatiques
10. Les listes
11. Les liens
12. Les boutons
13. Les séparateurs
14. Les images et les figures
15. Les tableaux
- 16. Les formulaires**
17. L'audio
18. La vidéo
19. Identification des balises
20. TP1 HTML

# Les formulaires

## 3- Les types des champs <input /> :

### G- Le champ de saisi de nombre entier :

```
<input type="number" />
```

```
<!--
```

Attributs:

min=0      Permet d'indiquer la valeur minimale autorisée.

max=10     Permet d'indiquer la valeur maximale autorisée.

step="2"    Le pas de déplacement (par exemple si on indique un pas de 2, on aura 0, 2, 4,...).

```
-->
```

### H- Le champ range de sélection de nombre par un curseur :

```
<input type="range" />
```

### I- Le champ de sélection de la couleur :

```
<input type="color" />
```

### J- Le champ de sélection de la date :

```
<input type="date" />
```

### K- Le champ de sélection de l'heure :

```
<input type="time" />
```



## Partie 1 - Le langage HTML

1. La Syntaxe du HTML
2. La structure d'une page HTML
3. Les commentaires
4. Les paragraphes
5. Les titres
6. La mise en valeur et de forme
7. Les balises universelles
8. L'élément de texte préformaté
9. Les codes informatiques
10. Les listes
11. Les liens
12. Les boutons
13. Les séparateurs
14. Les images et les figures
15. Les tableaux
- 16. Les formulaires**
17. L'audio
18. La vidéo
19. Identification des balises
20. TP1 HTML

# Les formulaires

## 3- Les types des champs `<input />` :

### L- Le champ de sélection de la date et de l'heure :

```
<input type="datetime" />
```

### M- Le champ de sélection de la date et de l'heure sans gestion du décalage horaire :

```
<input type="datetime-local" />
```

### N- Le champ de sélection de la semaine :

```
<input type="week" />
```

### O- Le champs de sélection du mois :

```
<input type="month" />
```

### P- Les cases à cocher :

Les case à cocher d'une même sélection doivent avoir le même nom « name », mais valeurs « value » différentes.

Option 1 `<input type="checkbox" name="zoneCases" value="case1" /> <br/>`

Option 2 `<input type="checkbox" name="zoneCases" value="case2" /> <br/>`

`<!--`

Attributs:

`checked="checked"` La case sera cochée par défaut.

`value="case1"` Pour donner une valeur à la case à cocher. C'est la valeur qu'on récupère lors des traitements des formulaires.

`-->`

## Partie 1 - Le langage HTML

1. La Syntaxe du HTML
2. La structure d'une page HTML
3. Les commentaires
4. Les paragraphes
5. Les titres
6. La mise en valeur et de forme
7. Les balises universelles
8. L'élément de texte préformaté
9. Les codes informatiques
10. Les listes
11. Les liens
12. Les boutons
13. Les séparateurs
14. Les images et les figures
15. Les tableaux
16. Les formulaires
17. L'audio
18. La vidéo
19. Identification des balises
20. TP1 HTML

# Les formulaires

## 3- Les types des champs `<input />` :

### Q- Les boutons radios :

Ils permettent de sélectionner une option parmi plusieurs options.

Les boutons radios d'une même sélection doivent avoir le même nom, mais valeurs (value) différentes.

```
Choix 1 <input type="radio" name="selection1" value="radio1" /> <br/>
```

```
Choix 2 <input type="radio" name="selection1" value="radio2" /> <br/>
```

```
<!--
```

Attributs:

`checked="checked"` L'option sera sélectionnée par défaut.

`value="radio1"` Pour donner une valeur à l'option. C'est la valeur qu'on récupère lors des traitements des formulaires.

```
-->
```

### R- Le champ caché :

Il permet de créer une variable avec une valeur sans que le champ soit visible par l'utilisateur.

```
<input type="hidden" name="nomDeLaVariable" value="valeurDeLaVariable" />
```

## Partie 1 - Le langage HTML

1. La Syntaxe du HTML
2. La structure d'une page HTML
3. Les commentaires
4. Les paragraphes
5. Les titres
6. La mise en valeur et de forme
7. Les balises universelles
8. L'élément de texte préformaté
9. Les codes informatiques
10. Les listes
11. Les liens
12. Les boutons
13. Les séparateurs
14. Les images et les figures
15. Les tableaux
- 16. Les formulaires**
17. L'audio
18. La vidéo
19. Identification des balises
20. TP1 HTML

# Les formulaires

## 3- Les types des champs `<input />` :

### S- Le champ de sélection de fichier :

```
<input type="file" />
```

### T- Les boutons de traitement du formulaire :

```
<!--Bouton d'envoi des données du formulaire qui permet de rediriger vers la page indiquée dans l'attribut action du formulaire :-->
```

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

```
<!--Bouton qui remet à zéro le formulaire en supprimant toutes les données saisies:-->
```

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

```
<!--Bouton d'envoi présenté sous forme d'image:-->
```

```
<input type="image" value="Envoyer" src="img.jpg" /> <!--L'attribut "src" indique l'image.-->
```

```
<!--Bouton sans effet par défaut. En général, ce bouton est géré en Javascript pour exécuter des actions sur la page.-->
```

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

```
<!--
```

Attributs:

value="Envoyer" C'est le texte qui sera afficher sur le bouton.

```
-->
```

## Partie 1 - Le langage HTML

1. La Syntaxe du HTML
2. La structure d'une page HTML
3. Les commentaires
4. Les paragraphes
5. Les titres
6. La mise en valeur et de forme
7. Les balises universelles
8. L'élément de texte préformaté
9. Les codes informatiques
10. Les listes
11. Les liens
12. Les boutons
13. Les séparateurs
14. Les images et les figures
15. Les tableaux
- 16. Les formulaires**
17. L'audio
18. La vidéo
19. Identification des balises
20. TP1 HTML

# Les formulaires

## 4- Le champ de saisie de texte (sur plusieurs lignes) :

Il permet d'insérer une zone de texte.

```
<textarea name="message" rows="10" cols="100"></textarea>
```

```
<!--
```

Attributs:

rows="10" Pour donner le nombre de lignes de la zone de texte.

cols="50" Pour donner le nombre de colonnes de la zone de texte.

```
-->
```

## Partie 1 - Le langage HTML

1. La Syntaxe du HTML
2. La structure d'une page HTML
3. Les commentaires
4. Les paragraphes
5. Les titres
6. La mise en valeur et de forme
7. Les balises universelles
8. L'élément de texte préformaté
9. Les codes informatiques
10. Les listes
11. Les liens
12. Les boutons
13. Les séparateurs
14. Les images et les figures
15. Les tableaux
- 16. Les formulaires**
17. L'audio
18. La vidéo
19. Identification des balises
20. TP1 HTML

# Les formulaires

## 5- Les listes déroulantes :

```
<select name="choix">
  <!--On utilise la balise <option>...</option> pour ajouter un élément à la liste:-->
  <option value="choix1" selected="selected">Choix 1</option>
  <!--
  Attributs:
    value                La valeur de l'option qui sera récupérée lors des traitements
    selected="selected"  Pour que l'option soit sélectionnée par défaut.
  -->
  <option value="choix2">Choix 2</option> <!--Une autre option.-->
</select>
```

➤ On peut grouper les options d'une liste déroulante avec la balise `<optgroup>...</optgroup>`:

```
<select name="Choix">
  <!--Groupe 1: L'attribut « label » permet de spécifier le titre du groupe.-->
  <optgroup label="Choix1">
    <option value="Choix11">Choix_1_1</option>
    <option value="Choix12">Choix_1_2</option>
  </optgroup>
  <!--Groupe 2:-->
  <optgroup label="Choix2">
    <option value="Choix21">Choix_2_1</option>
    <option value="Choix22">Choix_2_2</option>
  </optgroup>
</select>
```

## Partie 1 - Le langage HTML

1. La Syntaxe du HTML
2. La structure d'une page HTML
3. Les commentaires
4. Les paragraphes
5. Les titres
6. La mise en valeur et de forme
7. Les balises universelles
8. L'élément de texte préformaté
9. Les codes informatiques
10. Les listes
11. Les liens
12. Les boutons
13. Les séparateurs
14. Les images et les figures
15. Les tableaux
- 16. Les formulaires**
17. L'audio
18. La vidéo
19. Identification des balises
20. TP1 HTML

# Les formulaires

## 6- Les libellés :

Ils permettent d'ajouter un texte devant le champs :

```
<!--
```

Ici on ajoute un label.

On donne à l'attribut « for » la valeur de l'attribut « id » du champ qu'on veut lier avec le label.

```
-->
```

```
<label for="aaa">Titre du champ :</label>
```

```
<!--Ici on lie le label avec ce champs dont l'attribut « id » a la valeur « aaa » :-->
```

```
<input type="text" id="aaa" />
```

## Partie 1 - Le langage HTML

1. La Syntaxe du HTML
2. La structure d'une page HTML
3. Les commentaires
4. Les paragraphes
5. Les titres
6. La mise en valeur et de forme
7. Les balises universelles
8. L'élément de texte préformaté
9. Les codes informatiques
10. Les listes
11. Les liens
12. Les boutons
13. Les séparateurs
14. Les images et les figures
15. Les tableaux
- 16. Les formulaires**
17. L'audio
18. La vidéo
19. Identification des balises
20. TP1 HTML

# Les formulaires

## 7- Regrouper les champs d'un formulaire :

```
<!--Groupe 1:-->
<fieldset>
    <!--Le titre (légende) du groupe de champs 1:-->
    <legend>Informations essentielles</legend>

    <!--Les éléments du formulaire qui feront partis du groupe 1:-->
    <input type="email" />
    <!--....-->
</fieldset>

<!--Groupe 2:-->
<fieldset>
    <!--Le titre (légende) du groupe de champs 2:-->
    <legend>Informations secondaires</legend>

    <!--Les éléments du formulaire qui feront partis du groupe 2:-->
    <input type="text" />
    <!--....-->
</fieldset>
```

## Partie 1 - Le langage HTML

1. La Syntaxe du HTML
2. La structure d'une page HTML
3. Les commentaires
4. Les paragraphes
5. Les titres
6. La mise en valeur et de forme
7. Les balises universelles
8. L'élément de texte préformaté
9. Les codes informatiques
10. Les listes
11. Les liens
12. Les boutons
13. Les séparateurs
14. Les images et les figures
15. Les tableaux
16. Les formulaires
17. L'audio
18. La vidéo
19. Identification des balises
20. TP1 HTML

# L'audio

```
<audio src="audio.mp3" controls></audio>
```

```
<!--
```

### Attributs:

controls

Ajoute les boutons de contrôles ("Lecture", "Pause", ...etc).

width=300

Permet de modifier la largeur de la fenêtre de lecture de l'audio.

loop

Permet de jouer la musique en boucle.

autoplay

Permet de jouer la musique dès le chargement de la page.

preload=

Indique si la musique sera préchargée au chargement de la page ou non. Il peut prendre les valeurs:

"auto"

- le navigateur décide s'il doit précharger toute la musique, uniquement les métadonnées ou rien du tout (par défaut).

"metadata"

- charge uniquement les métadonnées (durée, etc.).

"none"

- pas de préchargement. Utile si on veut pas gaspiller de bande passante sur le site.

```
-->
```

- Proposer plusieurs formats au navigateur en lui permettant de prendre le format compatible (indispensable si on veut que l'audio soit lu par tous les navigateurs) :

```
<audio controls>
```

```
<source src="audio.mp3"></source> <!--Format 1-->
```

```
<source src="audio.ogg"></source> <!--Format 2-->
```

```
</audio>
```



## Partie 1 - Le langage HTML

1. La Syntaxe du HTML
2. La structure d'une page HTML
3. Les commentaires
4. Les paragraphes
5. Les titres
6. La mise en valeur et de forme
7. Les balises universelles
8. L'élément de texte préformaté
9. Les codes informatiques
10. Les listes
11. Les liens
12. Les boutons
13. Les séparateurs
14. Les images et les figures
15. Les tableaux
16. Les formulaires
17. L'audio
- 18. La vidéo**
19. Identification des balises
20. TP1 HTML

## La vidéo

```
<video src="sintel.webm" poster="sintel.jpg" width="600" controls ></video>
```

```
<!--
```

### Attributs:

|                  |                                                                                                                                                                                                                               |
|------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| controls         | Ajoute les boutons de contrôles ("Lecture", "Pause", ...etc).                                                                                                                                                                 |
| width=300        | Permet de modifier la largeur de la fenêtre de lecture de la vidéo.                                                                                                                                                           |
| height=300       | Permet de modifier la hauteur de la fenêtre de lecture de la vidéo.                                                                                                                                                           |
| loop             | Permet de jouer la vidéo en boucle.                                                                                                                                                                                           |
| autoplay         | Permet de jouer la vidéo dès le chargement de la page.                                                                                                                                                                        |
| poster="img.jpg" | L'image à afficher à la place de la vidéo lorsque celle-ci n'est pas lancée. Par défaut, le navigateur prendra la première image de la vidéo, mais c'est souvent une image noire ou une image peu représentative de la vidéo. |
| preload=         | Indique si la vidéo sera préchargée au chargement de la page ou non. Il peut prendre les valeurs:                                                                                                                             |
| "auto"           | - le navigateur décide s'il doit précharger toute la vidéo, uniquement les métadonnées ou rien du tout (par défaut).                                                                                                          |
| "metadata"       | - charge uniquement les métadonnées (durée, etc.).                                                                                                                                                                            |
| "none"           | - pas de préchargement. Utile si on veut pas gaspiller de bande passante sur le site.                                                                                                                                         |

```
-->
```

## Partie 1 - Le langage HTML

1. La Syntaxe du HTML
2. La structure d'une page HTML
3. Les commentaires
4. Les paragraphes
5. Les titres
6. La mise en valeur et de forme
7. Les balises universelles
8. L'élément de texte préformaté
9. Les codes informatiques
10. Les listes
11. Les liens
12. Les boutons
13. Les séparateurs
14. Les images et les figures
15. Les tableaux
16. Les formulaires
17. L'audio
- 18. La vidéo**
19. Identification des balises
20. TP1 HTML

## La vidéo

- Proposer plusieurs formats au navigateur en lui permettant de prendre le format compatible (indispensable si on veut que la vidéo soit lue par tous les navigateurs):

```
<video controls>
  <source src="video.mp4"></source> <!--Format 1-->
  <source src="video.ogv"></source> <!--Format 2-->
</video>
```

## Partie 1 - Le langage HTML

1. La Syntaxe du HTML
2. La structure d'une page HTML
3. Les commentaires
4. Les paragraphes
5. Les titres
6. La mise en valeur et de forme
7. Les balises universelles
8. L'élément de texte préformaté
9. Les codes informatiques
10. Les listes
11. Les liens
12. Les boutons
13. Les séparateurs
14. Les images et les figures
15. Les tableaux
16. Les formulaires
17. L'audio
18. La vidéo
- 19. Identification des balises**
20. TP1 HTML

# Identification des balises

Pour identifier une balise HTML parmi les autres balises, on a deux possibilités :

## 1- L'identifiant :

- On lui ajoutant un identifiant : Pour ajouter un identifiant à une balise, on lui ajoute l'attribut « id ».
- L'identifiant est unique. On ne peut pas donner le même identifiant à plus qu'une balise.

```
<balise id="nom_identifiant">...</balise>
```

## 2- La classe :

- On lui ajoute une classe : Pour ajouter une classe à une balise, on lui ajoute l'attribut « class ».
- La classe n'est pas unique et consiste à identifier un groupe de balises. On peut donner la même classe à plusieurs balises.
- Le nom de la classe ne doit pas comporter d'espaces, sinon il sera considéré comme plusieurs classes.

```
<balise class="nom_classe">...</balise>
```

## Partie 1 - Le langage HTML

1. La Syntaxe du HTML
2. La structure d'une page HTML
3. Les commentaires
4. Les paragraphes
5. Les titres
6. La mise en valeur et de forme
7. Les balises universelles
8. L'élément de texte préformaté
9. Les codes informatiques
10. Les listes
11. Les liens
12. Les boutons
13. Les séparateurs
14. Les images et les figures
15. Les tableaux
16. Les formulaires
17. L'audio
18. La vidéo
19. Identification des balises
20. TP1 HTML

# TP1 HTML



# LE LANGAGE CSS

Module : Technologie Web

# Définition et avantages



Cascading Style Sheets (**CSS**) : Feuille de style en cascade

- Langage de feuilles de style pour documents HTML.
- Syntaxe différente du HTML.
- Permet une séparation claire de la structure et du style.
- Définition de styles génériques pour les balises.

# Définition et avantages



- Faciliter la modification de la présentation ou du contenu (possibilité de travailler en équipe plus efficacement).
- Coût de développement et de maintenance allégé(code plus simple).
- Code source allégé.
- Possibilité de présentations différentes selon : le profil utilisateur, la résolution d'écran, etc.
- Uniformiser la présentation sur plusieurs pages (un seul fichier CSS à modifier pour que les changements se répercutent sur l'ensemble du site).



# Définition et avantages



Le code CSS peut être écrit dans une balise:

```
<p style="text-align: right; color: black">
```

Les feuilles de style permettent de gérer la mise en page des divers éléments d'une page web </p>

Pour un paragraphe :

```
<p style="text-align: right; color: black" > Les feuilles de style permettent de gérer la mise en page des divers éléments d'une page Web </p>
```

```
<p style="text-align: right; color: black" > Bien sur, une page Web peut être composée de nombreux paragraphes, comme n'importe quel texte.</p>
```

```
<p style="text-align: right; color: black" > Mais alors, il devient fastidieux de devoir saisir à nouveau toutes les informations de style ! </p>
```



# Définition et avantages



- ❑ Mais si le site possède plusieurs pages ?
- ➔ créer une feuille de style externe qui sera liée à chacune des pages Web

```
<HTML>
<HEAD>
  <Meta charset="utf-8" />
  <TITLE>Le CSS</TITLE>
```

```
<link rel="stylesheet" type="text/css" href="fichier.css">
```

```
</HEAD>
<BODY>
  <H1>Introduction</H1>
  ...
</BODY>
</H1>
```

// Code CSS

Fichier.CSS

# Définition et avantages



- Dans l'en-tête <head> du fichier HTML

```
<html>
  <head>
    <meta charset="utf-8" />
    <style>
      p
      {
        color: blue;
      }
    </style>
    <title>Premiers tests du
    CSS</title>
  </head>
  <body>
    <h1>Mon super site</h1>
    <p>Bonjour et bienvenue sur
    mon site !</p>
    <p>Pour le moment, mon site
    est un peu <em>vide</em>.
    Patientez encore un peu
    !</p>
  </body>
</html>
```

- Directement dans les balises (non recommandé)

```
<html>
  <head>
    <meta charset="utf-8" />
    <title>Premiers tests du
    CSS</title>
  </head>
  <body>
    <h1>Mon super site</h1>
    <p style="color: blue;">Bonjour
    et bienvenue sur mon site
    !</p>
    <p>Pour le moment, mon site
    est un peu <em>vide</em>.
    Patientez encore un peu
    !</p>
  </body>
</html>
```

# Définition et avantages

Une règle CSS est caractérisée par deux principaux éléments :

- Un sélecteur de **balises**, permettant de préciser à quelles balises du document le style s'applique
- Une **déclaration** de style, définie entre accolades, permettant de préciser le style à appliquer aux balises sélectionnées. La déclaration est elle-même constituée :
  - d'une ou plusieurs **propriété(s)**, suivie du caractère « : » (double point),
  - d'une ou plusieurs valeur(s) associée(s) à chaque propriété, entourée de guillemets s'il s'agit de plusieurs mots ou séparés par des virgules s'il y en a plusieurs, suivie d'un point virgule

*balise { propriété\_de\_style: valeur; propriété\_de\_style: valeur }*

**Exemple :**    *H1 { font-family: Arial; font-style: italic }*

# Définition et avantages

## Notion d'héritage :

- Si on traduit les trois mots de « Cascading Style Sheets » on obtient Feuille de style en cascade.
- Exemple:

si vous définissez une police de type "arial" sur la balise <body>, l'ensemble des autres éléments du site prendra comme police "arial", inutile de le redéfinir à chaque éléments. Sauf, si une autre police de caractère est déclarée explicitement.

# Notion de classe

- Un sélecteur contextuel, ou classe, associe une règle particulière à un élément en fonction de sa situation.
- On appelle sélecteur simple les balises HTML classiques auxquelles on a attribué des caractéristiques de style.

```
.classe1 {  
    color: green;  
    font-size: 14px;  
    font-weight: bolder;  
}
```

```
.classe2 {  
    color: red;  
    font-size: 18px;  
}
```

```
<p class="classe1">  
un premier paragraphe  
</p>
```

un premier paragraphe

```
<p class="classe2">  
un deuxieme paragraphe  
</p>
```

un deuxieme paragraphe

# Notion des ID

- La déclaration d'un ID en CSS est identique à la notion de class à une exception près : On ne peut déclarer une seule fois un ID

```
#once {  
  margin: 20px;  
  font-size: 11px;  
  color: blue;  
}
```

```
<p id="once">  
  est correct  
</p>
```

Mais si on rencontre dans la même page

```
<p id="once">  
  ce n'est plus correct !  
</p>
```

- Remarque : On utilise un **#** pour les id et un **."** pour les classes



# Combinaison de selecteurs

- Il est possible de combiner les éléments HTML qui regroupent les mêmes caractéristiques

```
img, table {border:none;}
```

# CSS : Exemple 1

- A la balise <H1> on pourra lui rattacher la police Verdana, italique, de taille 16 pixels, de couleur Rouge. Chaque fois qu'on utilisera la balise <H1> dans un document, tout le texte situé dans les limites de la balise recevra cette mise en forme.

| HTML                                                                                                                                                                                      | CSS                                                                                           |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------|
| <pre>&lt;HTML&gt; &lt;BODY&gt; &lt;H1 style=" font-size: 16px; font-family: Verdana; font-style: italic; color:black;"&gt; Titre de niveau 1&lt;/H1&gt; &lt;/BODY&gt; &lt;/HTML&gt;</pre> | <pre>h1 { font-size : 16px; font-family : Verdana; font-style: italic; color : black; }</pre> |



# CSS : Exemple 2

```
<style type="text/css">

body{
    margin: 0px;
    background-color:Ivory ;
    width: 100%;
    height: 100%;
    font-size:12px;
    font-family:Verdana, Geneva, sans-serif;
    font-weight:normal;
}

a{
    color: #000000;
    text-decoration:none;
}

p {
    font-size: 110%;
    font-family: Helvetica, sans-serif;
}

img {border:none;}

</style>
```

# Les commentaires

- Il est possible de mettre des commentaires presque n'importe où dans une feuille de style CSS en les encadrant de `/*` et `*/`, comme dans le langage C.

```
/* style de la balise A (lient hypertexte) */  
a{  
    color: #000000;  
    text-decoration:none;  
}  
  
/* style de la balise P (paragraphe) */  
p {  
    font-size: 12px;  
    font-family: verdana, geneva, sans-serif;  
}  
  
... ..
```

# Quelques propriétés

- Mise en forme d'un texte en CSS

**color:** Gestion de la couleur du texte  
**letter-spacing:** Espace entre les lettres  
**text-align:** Alignement d'un texte  
**text-decoration:** Habilement d'un texte

**text-indent:** Espace avant un texte  
**text-transform:** Transformation du texte  
**white-space:** Controle des espaces blancs  
**word-spacing:** Espace entre les mots

- Gestion des différentes polices

**font-family:** Famille de police  
**font-size:** Taille de votre police de caractère  
**font-style:** Gras, italique, oblique...  
**font-variant:** Variante de police  
**font-weight:** Taille de la police

- Les bordures en CSS

**border-color:** Couleur de la bordure  
**border-style:** Apparence de la bordure  
**border-width:** Taille de la bordure

# Avantages (1/2)

- La structure du document et la présentation sont gérés dans des fichiers séparés.
- La conception d'un document se fait dans un premier temps sans se soucier de la présentation, ce qui permet d'être plus efficace.
- Dans le cas d'un site Internet, la présentation est uniformisée : Les documents (pages « html ») font référence à la (aux) même(s) feuille(s) de styles. Cette caractéristique permet de plus un relookage rapide.

# Avantages (2/2)

- Un même document peut donner le choix entre plusieurs feuilles de style (impression ou lecture à l'écran par exemple). Certains navigateurs web permettent d'accéder facilement à un choix de feuilles de style.
- Le code HTML est considérablement réduit en taille et en complexité, puisqu'il ne contient plus de balises de présentation.
- Réduire le temps de chargement des pages.

# Les sélecteurs (1/4)



## *1. Sélecteurs standards:*

- *Sélecteur universel* : tous les éléments HTML

Exemple : `*{color :blue ; }`

- *Style d'une catégorie de balises:*

Exemple : `h1 { color :#FF00FF ; }`

- *Style de plusieurs catégories de balises*

Exemple : `h1, h2, h3, p { color : green ; }`

- *Style pour balises imbriquées*

Exemple : `p em {color :red ; }`



# Les sélecteurs (2/4)



## Fichier.HTML

```
<HTML>
  <HEAD>
    <TITLE>Le CSS</TITLE>
```

```
<link rel="stylesheet" type= "text/css" href= "fichier.css">
```

```
</HEAD>
  <BODY>

    <H1>Introduction</H1>
      ...
    </BODY>
</HTML>
```

## Fichier.CSS

```
H1 {
  color : red ;
}
```

Introduction

...

# Les sélecteurs (3/4)



## 2. Sélecteur de classes:

❑ *Sélecteur défini par l'utilisateur*: permet de dissocier un style de balises particulières

Exemple :

- `style_rouge { color:red; }`
- `style_bleu { color:blue; }`
- `p.paprag_vert { color:green; } /* applicable seulement aux paragraphes */`

➤ Utilisation dans une page Web

Il faut faire un appel explicite à la classe de style désirée.

Exemple :

`<h1 class="style_rouge">Titre de niveau 1</h1>`



# Les sélecteurs(4/4)



## 3. *Sélecteur d'identificateurs*

- ❖ Un sélecteur de classe commence par un "." alors que le sélecteur d'identificateur commence par un "#".
- ❖ Le style de type classe peut être utilisé plusieurs fois dans un document html alors que le style de type identificateur ne peut être utilisé qu'une seule fois dans le document html.
- ❖ Le style de type classe s'utilise dans n'importe quelle balise avec l'attribut "class" alors que le style de type identificateur s'utilise avec l'attribut "id " .

# Pseudo-format



Le CSS nous permet aussi 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

- au survol: **a:hover{ }**
- lors du clic : **a:active { }** /\* Quand le visiteur clique sur le lien \*/
- lors du focus : **a:focus { }** /\* Quand le visiteur sélectionne le lien \*/
- lorsqu'un lien a été consulté :  
**a:visited{ }** /\* Quand le visiteur a déjà consulté la page \*/

## Syntaxe

**Selecteur : pseudoformat**

```
{ prop1: val1;  
prop2: val2;  
..}
```

# Les styles de mise en forme du texte (1/6)



❑ font-family : définit <nom> ou <famille> police précise (Arial, Times, Helvetica...)

⇒ **H3 {font-family: Arial}**

❑ font-style : définit le style de l'écriture normal ou italique ou oblique

⇒ **H3 {font-style: italic}**

❑ font-weight : définit l'épaisseur de la police normal ou bold ou bolder ou lighter ou valeur numérique soit (100 | 200 | 300 | 400 | 500 | 600 | 700 | 800 | 900)

⇒ **P {font-weight: bold}**

# Les styles de mise en forme du texte (2/6)



❑ **font-size** : définit la taille de la police en points (pt), inches (in), centimètres (cm), pixels (px) ou pourcentage (%)

⇒ *P {font-size: 12pt}*

❑ **font-variant** : définit une variante par rapport à la normale normal ou small-caps

⇒ *P {font-variant: small-caps}*

# Les styles de mise en forme du texte (3/6)



□ **text-align** : définit l'alignement du texte left ou center ou right

⇒ *H1 {text-align: center}*

□ **text-indent** : définit un retrait dans la première ligne d'un bloc de texte  
souvent utilisé avec <P>spécifié en inches (in) ou en centimètres (cm) ou en pixels (px)

⇒ *P {text-indent: 1cm}*

□ **text-decoration** : définit une décoration du texte, soit barré, clignotant, etc. blink ou underline ou line-through ou overline ou none

⇒ *A:visited {text-decoration: blink }*



# Les styles de mise en forme du texte (4/6)



❑ **text-transform** : définit la casse du texte (majuscule, minuscule)

❑ uppercase (met les caractères en majuscules) ou

❑ lowercase (met les caractères en minuscules) ou

❑ capitalize (met le premier caractère en majuscule)

⇒ *P {text-transform: uppercase}*

❑ **Color** : définit la couleur du texte par exemple en hexadécimal

⇒ *H3 {color: #000080}*

❑ **word-spacing** : définit l'espace entre les mots en points (pt), inches (in), centimètres (cm), pixels (px) ou pourcentage (%)

⇒ *P {word-spacing: 5pt}*

# Les styles de mise en forme du texte (5/6)



❑ **letter-spacing** : définit l'espace entre les lettres spécifié en points (pt), inches (in), centimètres (cm), pixels (px) ou pourcentage (%)

$\Rightarrow P \{letter-spacing: 2pt\}$

❑ **line-height** : définit l'interligne soit l'espace entre les lignes du texte en points (pt), inches (in), centimètres (cm), pixels (px) ou pourcentage (%)

$\Rightarrow P \{line-height: 10pt\}$

# Les styles de mise en forme du texte (6/6)



❑ **Width** : détermine la longueur d'un élément de texte ou d'une image en points (pt), inches (in), centimètres (cm), pixels (px) ou pourcentage (%)

⇒ ***H1 {width: 200px}***

❑ **height** : détermine la hauteur d'un élément de texte ou d'une image en points (pt), inches (in), centimètres (cm), pixels (px) ou pourcentage (%)

⇒ ***H1 {height: 100px}***

❑ **white-space** : espace ou blanc normal ou pre ou nowrap PRE

***{white-space: pre}***



# Les couleurs (1/2)

## □ Color

la propriété `color` permet de fixer la couleur du texte.

### Comment manipuler les couleurs ?

- liste de couleur standard
- Un code hexadécimal long `#00FF45`
- Un code hexadécimal court `#0F3`
- Un code rgb, de 0 à 255 `rgb(0,0,100)`
- un code rgb, en pourcentage `rgb(10%,10%,30%)`

### Liste des couleurs standard :

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

## Les couleurs (2/2)



### ❑ **background-color**

la propriété background-color permet de fixer la couleur de fond de la page.

```
body
{
background-color : rgb(0,0,0) ;
color : white
}
```

# FLOAT – CLEAR (1/2)



- ❑ **Float** : Une boîte flottante est **retirée du flux normal** placée le plus à droite (float: right) ou le plus à gauche (float: left) possible dans son conteneur.

## **Exemple : Boîte flottante simple**

### Code HTML:

```
<p class="jaune">Une boîte jaune flottant</p> <p  
class="verte">Une boîte verte doté d'un contenu plus long...</p>
```

### Code CSS:

```
.jaune { background-color: #ffff00;  
        float: right;  
        width: 100px;  
        margin: 0;  
    }  
.verte { background-color: #00ff00;.
```

# FLOAT – CLEAR (1/2)



- ❑ **Clear:** définit si un élément peut être à côté des éléments flottants qui le précèdent ou s'il doit être placé en dessous d'eux (se libérer).

## Les valeurs

Left: L'élément est déplacé vers le bas pour se libérer des éléments flottants à gauche.

Right : L'élément est déplacé vers le bas pour se libérer des éléments flottants à droite.

Both : L'élément est déplacé en dessous des éléments flottants de gauche et de droite

# Modèles d'affichage en C.S.S (1/3)



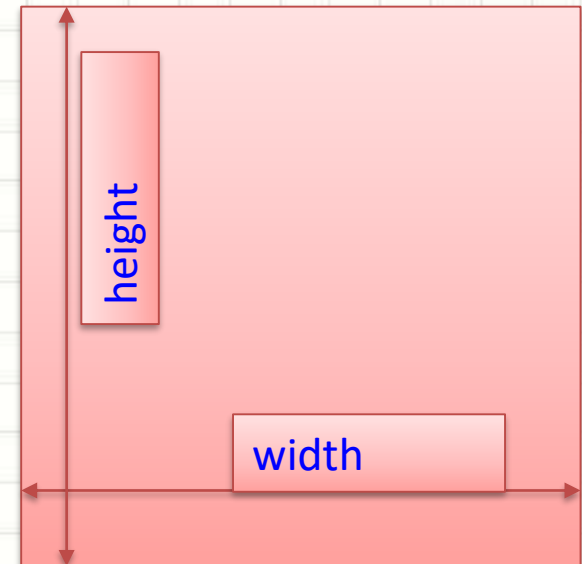
- Les éléments « bloc » : une balise de type block sur votre page web crée automatiquement un retour à la ligne avant et après
  - Les éléments blocs s'empilent les uns sur les autres
  - Un élément bloc peut contenir d'autres éléments blocs
  - Un élément bloc peut contenir des éléments en ligne*exemples* : p, h1, h2, h3, h4, h5, h6, div
  
- Les éléments « en ligne » : une balise de type inline se trouve obligatoirement à l'intérieur d'une balise block. Une balise inline ne crée pas de retour à la ligne
  - Les éléments en ligne se placent les uns à côté des autres
  - Un élément en ligne peut contenir d'autres éléments en ligne*exemples* : em, strong, b, i, a, span



# Modèles d'affichage en C.S.S (2/3)



- *width*
  - Permet de spécifier la largeur d'un bloc
- *height*
  - Permet de spécifier la hauteur d'un bloc
- *position*
  - **Absolute** Le bloc est placé de façon absolue
  - **Relative**: Le bloc est placé relativement à la position qu'il aurait du occuper



# Modèles d'affichage en C.S.S (3/3)



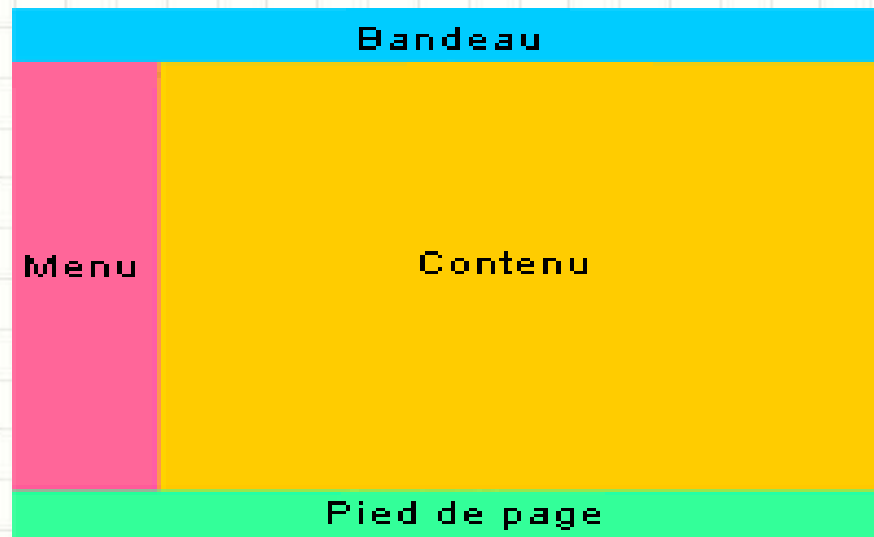
- Containers génériques.

`<div>` permet de structurer le document

`<span>` permet de changer la mise en page de quelques mots dans un texte

`<div>` et `<span>` s'utilisent toujours avec les classes et les ID.

## Exemple 1



# Exemple d'utilisation de DIV



```
<html>
<head>
<meta http-equiv="Content-Type"
content="text/html; charset=iso-8859-1">
<link rel="stylesheet" type="text/css"
href="styles.css">
<title>Positionnement CSS</title>
</head>
<body>
<div id="bandeau">
Ceci est le bandeau</div>
<div id="menu">
Ceci est le menu</div>
<div id="contenu">
Ceci est le contenu</div>
<div id="pied_page">Ceci est le pied de
page</div>
</body>
</html>
```

```
div {text-align:center;}

div#bandeau {
    width:600px;
    height:50px;
    background-color:#00CCFF;
}
div#menu { float:left;
    width:100px;
    height:400px;
    background-color:#FF6699;
}
div#contenu { float:left;
    width:500px;
    height:400px;
    background-color:#FFCC00;
}
div#pied_page {
    clear:both;
    width:600px;
    height:50px;
    background- color:#33FF99;
}
```



# Exemple d'utilisation de DIV

```
<body>
<div id="header">
  <h1>Header Text</h1>
</div>
<div id="navcol">
  <h4>Nav Column</h4>

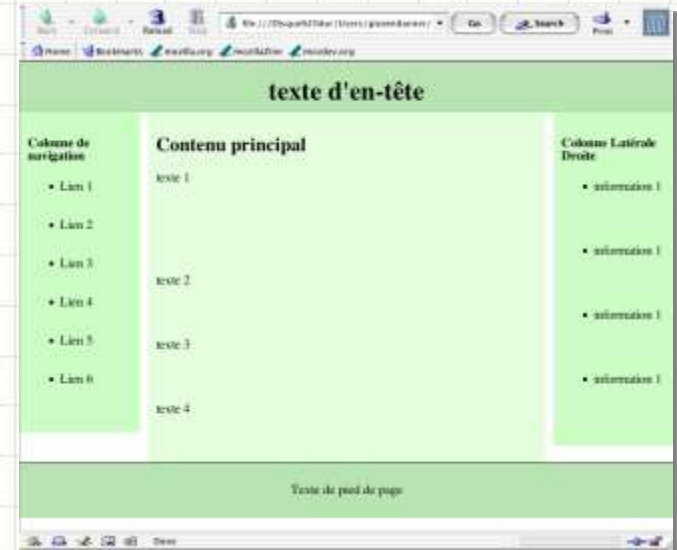
  <ul>
    <li>Let me not
    <li>Admit imped
    <li>Which alter
    <li>Or bends wi
    <li>Oh, no, it

    <li>Let me not
  </ul>
</div>
<div id="sidecol">
  <h4>Starboard Side Colu
  <ul>
    <li>Let me not
    <li>Admit imped

    <li>Which alter
    <li>Or bends wi
  </ul>
</div>
<div id="main">
  <h2>Main Content</h2>
  <p> That looks on tempe

  <p>That looks on tempes
  <p>If this be error and
  <p>That looks on tempes
</div>
<div id="foot">
  <p>Footer text goes her
</div>
</body>
```

## Exemple 2





```
1 body {  
2     margin: 0px;  
3     padding: 0px;  
4 }
```

```
5 div#header {  
6     text-align: center;  
7     background-color: #B8E4B2;  
8     height: 60px;  
9     margin: 0px;  
10    padding: 1px;  
11 }
```

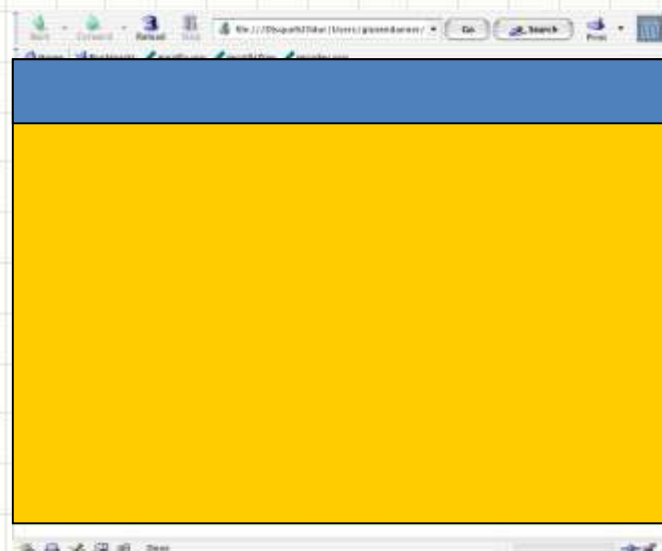
```
<div id="header">  
    <h1>Header Text</h1>  
</div>
```

```
12 div#navcol {  
13     padding: 10px;  
14     width: 130px;  
15     float: left;  
16     background-color: #CCFDC5;  
17 }
```

```
18 div#main {  
19     padding: 10px;  
20     margin-left: 160px;  
21     margin-right: 160px;  
22     background-color: #E3FFDE;  
23 }
```

```
24 div#sidecol {  
25     padding: 10px;  
26     width: 130px;  
27     float: right;  
28     background-color: #CCFDC5;  
29 }
```

```
30 div#foot {  
31     border-top: solid #000 1px;  
32     background-color: #B8E4B2;  
33     padding: 10px;  
34     text-align: center;  
35     clear: both;  
36 }
```



```

1  body {
2      margin: 0px;
3      padding: 0px;
4  }

```

```

5  div#header {
6      text-align: center;
7      background-color: #B8E4B2;
8      height: 60px;
9      margin: 0px;
10     padding: 1px;
11 }

```

```

12 div#navcol {
13     padding: 10px;
14     width: 130px;
15     float: left;
16     background-color: #CCFDC5;
17 }

```

```

18 div#main {
19     padding: 10px;
20     margin-left: 160px;
21     margin-right: 160px;
22     background-color: #E3FFDE;
23 }

```

```

24 div#sidecol {
25     padding: 10px;
26     width: 130px;
27     float: right;
28     background-color: #CCFDC5;
29 }

```

```

30 div#foot {
31     border-top: solid #000 1px;
32     background-color: #B8E4B2;
33     padding: 10px;
34     text-align: center;
35     clear: both;
36 }

```

```

<div id="navcol">
    <h4>Nav Column</h4>

    <ul>

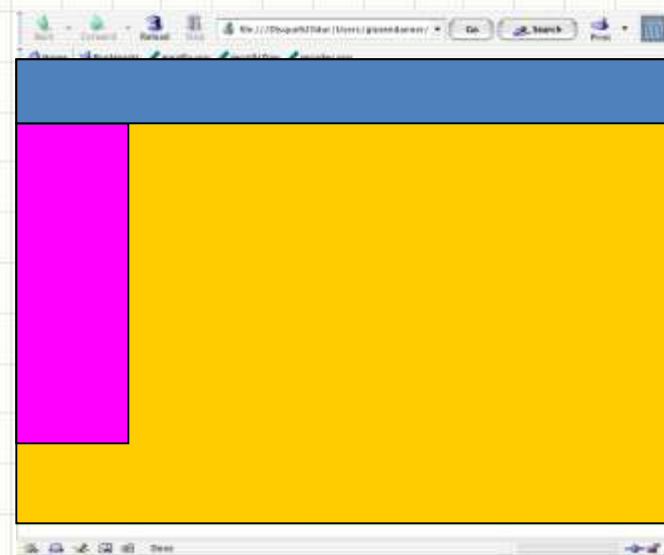
        <li>Let me not
        <li>Admit impe
        <li>Which alter
        <li>Or bends w
        <li>Oh, no, it

        <li>Let me not

    </ul>

</div>

```



```
1 body {  
2   margin: 0px;  
3   padding: 0px;  
4 }
```

```
5 div#header {  
6   text-align: center;  
7   background-color: #B8E4B2;  
8   height: 60px;  
9   margin: 0px;  
10  padding: 1px;  
11 }
```

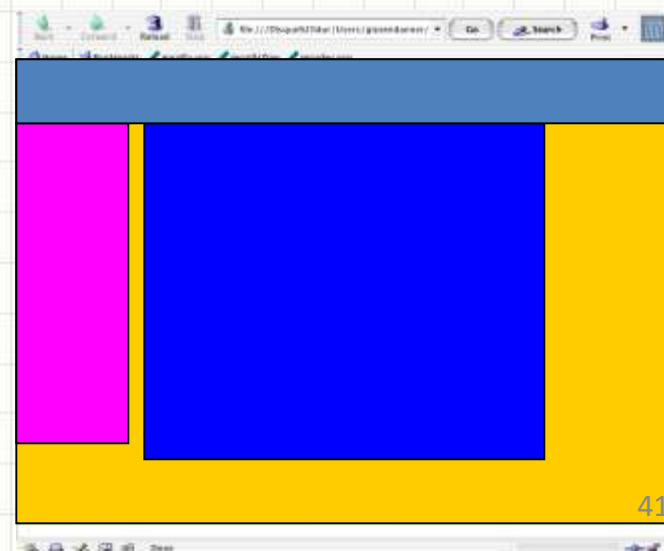
```
12 div#navcol {  
13   padding: 10px;  
14   width: 130px;  
15   float: left;  
16   background-color: #CCFDC5;  
17 }
```

```
18 div#main {  
19   padding: 10px;  
20   margin-left: 160px;  
21   margin-right: 160px;  
22   background-color: #E3FFDE;  
23 }
```

```
24 div#sidecol {  
25   padding: 10px;  
26   width: 130px;  
27   float: right;  
28   background-color: #CCFDC5;  
29 }
```

```
30 div#foot {  
31   border-top: solid #000 1px;  
32   background-color: #B8E4B2;  
33   padding: 10px;  
34   text-align: center;  
35   clear: both;  
36 }
```

```
<div id="main">  
  <h2>Main Content</h2>  
  <p> That looks on tempests and is never sha  
  
  <p>That looks on tempests and is never sha  
  <p>If this be error and upon me proved, wi  
  <p>That looks on tempests and is never sha  
</div>
```





```

1  body {
2      margin: 0px;
3      padding: 0px;
4  }

```

```

5  div#header {
6      text-align: center;
7      background-color: #B8E4B2;
8      height: 60px;
9      margin: 0px;
10     padding: 1px;
11 }

```

```

12 div#navcol {
13     padding: 10px;
14     width: 130px;
15     float: left;
16     background-color: #CCFDC5;
17 }

```

```

18 div#main {
19     padding: 10px;
20     margin-left: 160px;
21     margin-right: 160px;
22     background-color: #E3FFDE;
23 }

```

```

24 div#sidecol {
25     padding: 10px;
26     width: 130px;
27     float: right;
28     background-color: #CCFDC5;
29 }

```

```

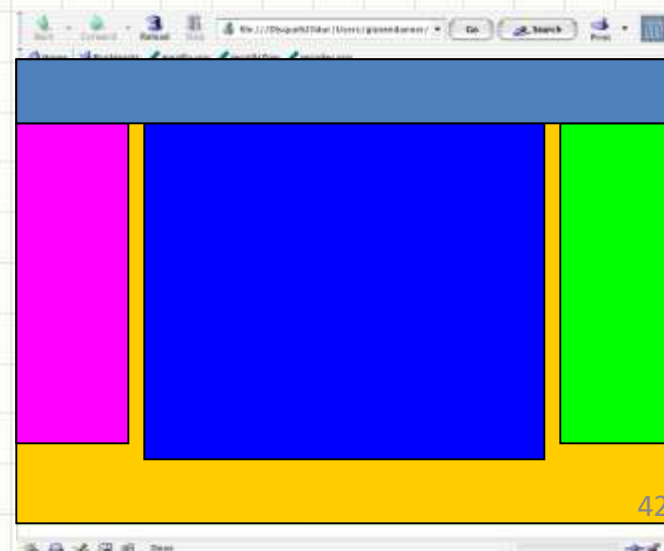
30 div#foot {
31     border-top: solid #000 1px;
32     background-color: #B8E4B2;
33     padding: 10px;
34     text-align: center;
35     clear: both;
36 }

```

```

<div id="sidecol">
    <h4>Starboard Side Column</h4>
    <ul>
        <li>Let me not to the ma
        <li>Admit impediments; l
        <li>Which alters when it
        <li>Or bends with the re
    </ul>
</div>

```



```
1 body {
2     margin: 0px;
3     padding: 0px;
4 }
```

```
5 div#header {
6     text-align: center;
7     background-color: #B8E4B2;
8     height: 60px;
9     margin: 0px;
10    padding: 1px;
11 }
```

```
12 div#navcol {
13     padding: 10px;
14     width: 130px;
15     float: left;
16     background-color: #CCFDC5;
17 }
```

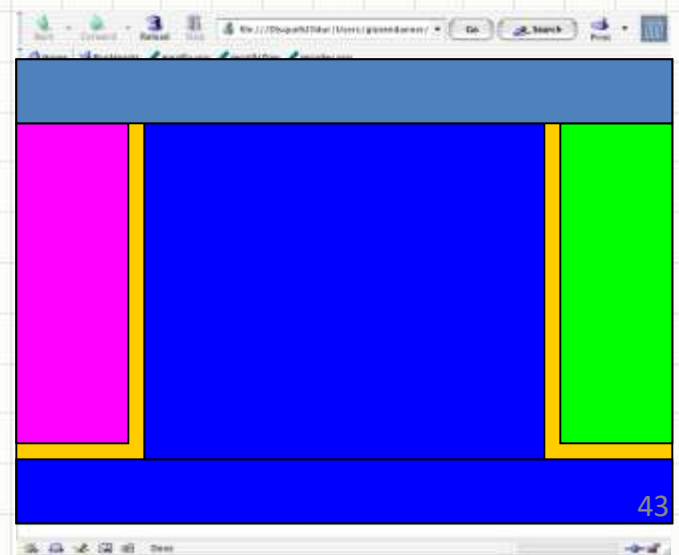
```
18 div#main {
19     padding: 10px;
20     margin-left: 160px;
21     margin-right: 160px;
22     background-color: #E3FFDE;
23 }
```

```
24 div#sidecol {
25     padding: 10px;
26     width: 130px;
27     float: right;
28     background-color: #CCFDC5;
29 }
```

```
30 div#root {
31     border-top: solid #000 1px;
32     background-color: #B8E4B2;
33     padding: 10px;
34     text-align: center;
35     clear: both;
36 }
```



```
<div id="foot">
    <p>Footer text goes here.
</div>
```



# Les Bordures



❑ Couleur de bordure : propriété **border-color**

➤ Possibilité d'attribuer différentes couleurs pour chaque bordure (**border-top-color, border-bottom-color, border-right-color, border-left-color**)

➤ Exemples : `p{border-top-color: green;};` `img{bordercolor:#00FF00;}`

❑ Définition globale de bordure : propriété **border**

➤ Une écriture simplifiée des différentes propriétés de bordure

➤ Exemple : `h1{border: 5px double blue;}`

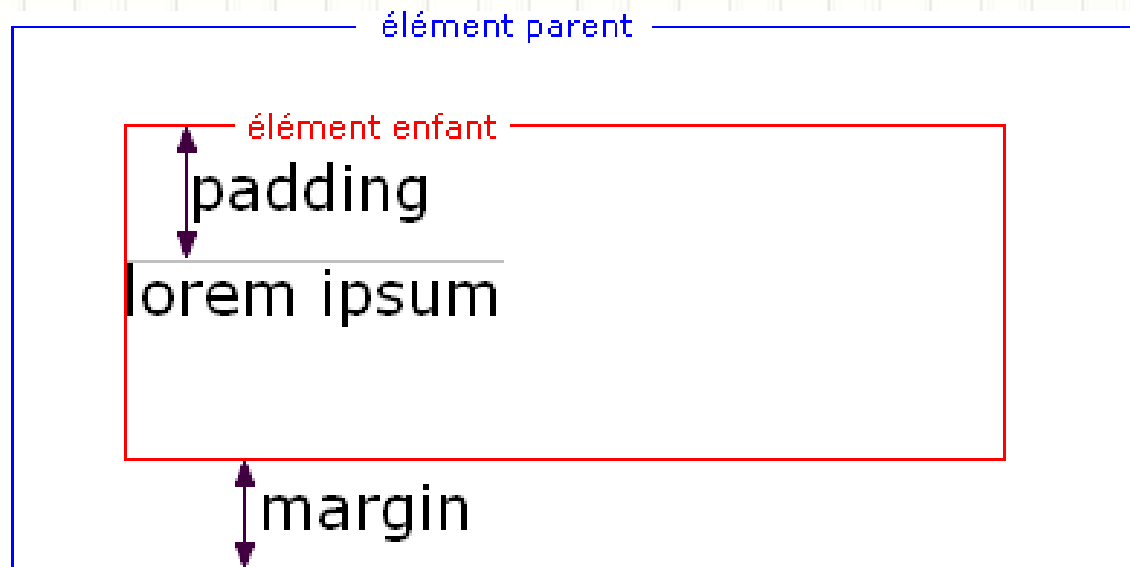
➤ Une écriture simplifiée pour chaque arête de bordure

(**border-top, border-bottom, border-right, border-left**)

➤ Exemple : `p{border-bottom: #00FFFF;}`



# Les Marges - Padding





# Les Marges

- Les marges peuvent être négatives ; dans ce cas la boîte d'un élément sort de celle de son parent
- Si la largeur de la marge est donnée en pourcentage, elle est calculée par rapport à celle du bloc parent.
- Possibilité de définir de une à quatre marges dans le sens des aiguilles d'une montre (haut, droit, bas, gauche).
- Pour réaliser un design centré quand on ne connaît pas la résolution du visiteur, on utilise **margin : auto**

- Exemples :

```
h1 { margin: 20px 10px 15px 10px; }
```

```
p { margin: 10%; }
```

```
b { margin-left: 3em; }
```

```
.block_centre { margin: auto; width:200px; }
```

# Les Paddings



- Permet comme la propriété margin d'aérer la présentation mais cette fois non pas entre deux éléments voisins mais directement autour du contenu.
- Possibilité de définir dans l'ordre les espacements haut, droit, bas et gauche de l'élément.
- Exemples:

```
h1 { padding: 20px 10px; }
```

```
h1 { padding-top: 20px;
```

```
padding-bottom: 20px;
```

```
padding-right: 10px;
```

```
padding-left: 10px;
```

```
}
```

# La création de menu horizontal



```
<ul>
<li><a href="p1.html">lien1</a>
</li>
<li><a
href="p2.html">lien2</a></li>
<li><a
href="p3.html">lien3</a></li>
</ul>
```

```
ul li{
list-style: none;
float: left;
padding:5px; }
```

```
ul li a{
text-decoration: none;
color: red;
background-color: blue; }
```

```
ul li a:hover{
color: green;
font-weight: bold;
background-color: orange; }
```



**MERCI POUR VOTRE ATTENTION**

## Partie 2 - Le langage CSS

### 1. Les façons d'écrire du CSS

#### 2. Les commentaires

#### 3. Les sélecteurs en CSS

#### 4. Le formatage du texte

#### 5. Le fond/avant-plan

#### 6. Le modèle des boîtes

#### 7. Le positionnement en CSS

#### 8. Les listes

#### 9. Les liens

#### 10. Les tableaux

#### 11. Le curseur

#### 12. Les transformations

#### 13. Les pseudo-classes

#### 14. Les Media Queries

#### 15. TP2 CSS

# Les différentes façons d'écrire du CSS

## 1- Inclure directement les instructions CSS dans les attributs des balises HTML:

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>CSS</title>
    <link rel="icon" href="favicon.ico" />
  </head>

  <body>
    <h1 style="color: red;">Le langage CSS</h1>
  </body>
</html>
```

## Partie 2 - Le langage CSS

### 1. Les façons d'écrire du CSS

2. Les commentaires
3. Les sélecteurs en CSS
4. Le formatage du texte
5. Le fond/avant-plan
6. Le modèle des boîtes
7. Le positionnement en CSS
8. Les listes
9. Les liens
10. Les tableaux
11. Le curseur
12. Les transformations
13. Les pseudo-classes
14. Les Media Queries
15. TP2 CSS

# Les différentes façons d'écrire du CSS

## 2- Inclure directement le code CSS dans le head dans la balise <style>:

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>CSS</title>
    <link rel="icon" href="favicon.ico" />

    <style>
      h1 {
        color: red;
      }
    </style>
  </head>

  <body>
    <h1>Le langage CSS</h1>
  </body>
</html>
```



## Partie 2 - Le langage CSS

### 1. Les façons d'écrire du CSS

### 2. Les commentaires

### 3. Les sélecteurs en CSS

### 4. Le formatage du texte

### 5. Le fond/avant-plan

### 6. Le modèle des boîtes

### 7. Le positionnement en CSS

### 8. Les listes

### 9. Les liens

### 10. Les tableaux

### 11. Le curseur

### 12. Les transformations

### 13. Les pseudo-classes

### 14. Les Media Queries

### 15. TP2 CSS

## Les différentes façons d'écrire du CSS

### 3- Ecrire une page indépendante CSS et l'inclure dans le head dans la balise <link> :

#### ➤ Le fichier HTML :

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>CSS</title>
    <link rel="icon" href="favicon.ico" />
    <link rel="stylesheet" href="file.css" />
  </head>
  <body>
    <h1>Le langage CSS</h1>
  </body>
</html>
```

#### ➤ Le fichier CSS « file.css » :

```
h1 {
  color: red;
}
```



Partie 2 - Le langage CSS

- 1. Les façons d'écrire du CSS
- 2. Les commentaires**
- 3. Les sélecteurs en CSS
- 4. Le formatage du texte
- 5. Le fond/avant-plan
- 6. Le modèle des boîtes
- 7. Le positionnement en CSS
- 8. Les listes
- 9. Les liens
- 10. Les tableaux
- 11. Le curseur
- 12. Les transformations
- 13. Les pseudo-classes
- 14. Les Media Queries
- 15. TP2 CSS

# Les commentaires en CSS

Ils permettent de décrire les lignes de code sans que les descriptions soient prises en compte lors de la visualisation de la page web.

➤ Un commentaire sur une ligne :

```
/* Voici un commentaire */
```

➤ Un commentaire sur plusieurs lignes :

```
/*  
Une première ligne  
Une deuxième ligne  
Une troisième ligne  
....  
*/
```

Partie 2 - Le langage CSS

- 1. Les façons d'écrire du CSS
- 2. Les commentaires
- 3. Les sélecteurs en CSS
- 4. Le formatage du texte
- 5. Le fond/avant-plan
- 6. Le modèle des boîtes
- 7. Le positionnement en CSS
- 8. Les listes
- 9. Les liens
- 10. Les tableaux
- 11. Le curseur
- 12. Les transformations
- 13. Les pseudo-classes
- 14. Les Media Queries
- 15. TP2 CSS

# Les sélecteurs en CSS

1- Pour écrire du CSS on l'écrit sous forme:

```
balise {  
    propriété: valeur;  
}
```



2- Appliquer les même propriétés à plusieurs balises à la foi:

```
h1, h2, p {      /* On sépare les balises par des virgules */  
    propriété: valeur;  
}
```

3- Sélectionner les balises par le nom de leurs classes:

```
.nomClasse {     /* On précède le nom de la classe par un point « . » */  
    propriété: valeur;  
}
```

4- Sélectionner une balise par son identifiant « id »:

```
#nomId {         /* On précède le nom de la classe par un dièze « # » */  
    propriété: valeur;  
}
```

Partie 2 - Le langage CSS

- 1. Les façons d'écrire du CSS
- 2. Les commentaires
- 3. Les sélecteurs en CSS
- 4. Le formatage du texte
- 5. Le fond/avant-plan
- 6. Le modèle des boîtes
- 7. Le positionnement en CSS
- 8. Les listes
- 9. Les liens
- 10. Les tableaux
- 11. Le curseur
- 12. Les transformations
- 13. Les pseudo-classes
- 14. Les Media Queries
- 15. TP2 CSS

# Les sélecteurs en CSS

## 5- Le sélecteur universel:

Applique ses propriété sur tout le texte de la page quelque soit sa balise :

```
*{  
    propriété: valeur;  
}
```

## 6- Une balise contenue dans une autre:

```
h1 em{  
    propriété: valeur;  
}
```

```
<body>  
    <h1>Bonjour les <em>apprentis</em></h1>  
</body>
```

## 7- Une balise qui en suit une autre:

```
h1 + p{  
    propriété: valeur;  
}
```

```
<body>  
    <h1>Bonjour les apprentis</h1>  
    <p>Hello !</p>  
</body>
```

## 8- Une balise qui possède un attribut:

```
a[title]{  
    color: green;  
}
```

```
<body>  
    <a href="https://www.css.fr/">CSS 1</a><br/>  
    <a href="https://www.css.fr/" title="css">CSS 2</a><br/>  
    <a href="https://www.css.fr/" title="yes">CSS 3</a><br/>  
    <a href="https://www.css.fr/" title="css ici">CSS 4</a><br/>  
</body>
```

css 1

css 2

css 3

css 4

Partie 2 - Le langage CSS

- 1. Les façons d'écrire du CSS
- 2. Les commentaires
- 3. Les sélecteurs en CSS
- 4. Le formatage du texte
- 5. Le fond/avant-plan
- 6. Le modèle des boîtes
- 7. Le positionnement en CSS
- 8. Les listes
- 9. Les liens
- 10. Les tableaux
- 11. Le curseur
- 12. Les transformations
- 13. Les pseudo-classes
- 14. Les Media Queries
- 15. TP2 CSS

# Les sélecteurs en CSS

## 9- Une balise qui possède un attribut qui a une valeur exacte:

```
a[title="valeur"]{
    propriété: valeur;
}
```

```
a[title="css"]{
    color: green;
}
```

```
<body>
  <a href="https://www.css.fr/">CSS 1</a><br/>
  <a href="https://www.css.fr/" title="css">CSS 2</a><br/>
  <a href="https://www.css.fr/" title="yes">CSS 3</a><br/>
  <a href="https://www.css.fr/" title="css ici">CSS 4</a><br/>
</body>
```

CSS 1  
CSS 2  
CSS 3  
CSS 4

## 10- Une balise qui possède un attribut dont la valeur contient la valeur donnée:

```
a[title*="ici"]{
    propriété: valeur;
}
```

```
a[title*="ici"]{
    color: green;
}
```

```
<body>
  <a href="https://www.css.fr/">CSS 1</a><br/>
  <a href="https://www.css.fr/" title="css">CSS 2</a><br/>
  <a href="https://www.css.fr/" title="yes">CSS 3</a><br/>
  <a href="https://www.css.fr/" title="css ici">CSS 4</a><br/>
</body>
```

CSS 1  
CSS 2  
CSS 3  
CSS 4

Partie 2 - Le langage CSS

- 1. Les façons d'écrire du CSS
- 2. Les commentaires
- 3. Les sélecteurs en CSS
- 4. Le formatage du texte
- 5. Le fond/avant-plan
- 6. Le modèle des boîtes
- 7. Le positionnement en CSS
- 8. Les listes
- 9. Les liens
- 10. Les tableaux
- 11. Le curseur
- 12. Les transformations
- 13. Les pseudo-classes
- 14. Les Media Queries
- 15. TP2 CSS

# Le formatage du texte

## 1- Changer le style d'un texte :

|                      |                                                                         |
|----------------------|-------------------------------------------------------------------------|
| font-style: italic;  | /* Le texte sera mis en italique. */                                    |
| font-style: oblique; | /* Le texte sera aussi mis en italique en penchant plus les lettres. */ |
| font-style: normal;  | /* Le texte sera normal (valeur par défaut). */                         |
| font-style: inherit; | /* Le texte héritera le style de l'élément parent. */                   |

## 2- Changer le poids d'un texte :

|                       |                                                                        |
|-----------------------|------------------------------------------------------------------------|
| font-weight: normal;  | /* Le texte sera écrit normalement (valeur par défaut). */             |
| font-weight: bold;    | /* Le texte sera en gras.*/                                            |
| font-weight: bolder;  | /* Encore plus gras. Il n'est pas supporté par tous les navigateur. */ |
| font-weight: lighter; | /* Elle va alléger le poids du police. */                              |
| font-weight: inherit; | /* Le texte va prendre le poids de l'élément parent. */                |

## 3- Modifier la taille du texte :

### A- Taille absolue :

|                  |                                                   |
|------------------|---------------------------------------------------|
| font-size: 14px; | /* On peut utiliser pt, %, em ... comme unité. */ |
|------------------|---------------------------------------------------|

### B- Taille relative :

|                      |                         |
|----------------------|-------------------------|
| font-size: xx-small; | /* Texte minuscule */   |
| font-size: x-small;  | /* Texte très petit */  |
| font-size: small;    | /* Texte petit */       |
| font-size: medium;   | /* Texte moyen */       |
| font-size: large;    | /* Texte grand */       |
| font-size: x-large;  | /* Texte très grand */  |
| font-size: xx-large; | /* Texte gigantesque */ |

Partie 2 - Le langage CSS

- 1. Les façons d'écrire du CSS
- 2. Les commentaires
- 3. Les sélecteurs en CSS
- 4. Le formatage du texte
- 5. Le fond/avant-plan
- 6. Le modèle des boîtes
- 7. Le positionnement en CSS
- 8. Les listes
- 9. Les liens
- 10. Les tableaux
- 11. Le curseur
- 12. Les transformations
- 13. Les pseudo-classes
- 14. Les Media Queries
- 15. TP2 CSS

# Le formatage du texte

## 4- Définir l'écart entre les lignes (1.5\*font-size) :

```
line-height: 21px;
```

## 5- Définir la famille de la police :

```
font-family: Arial;  
font-family: Arial Black;  
font-family: Comic Sans MS;  
font-family: Courier New;  
font-family: Georgia;  
font-family: Impact;  
font-family: Times New Roman;  
font-family: Trebuchet MS;  
font-family: Verdana;
```

➤ On peut utiliser plusieurs familles de police séparées par des virgules pour être sûr que l'une sera supportée.

```
font-family: Arial, Arial Black, Comic Sans MS;
```

## 6- Combiner les valeurs du font:

➤ Le regroupement des éléments de la police doit respecter la forme :

font: [font-style] [font-weight] [font-size]/[line-height] [font-family];

```
font: italic bold 14px/21px Georgia, Arial, serif;
```

## Partie 2 - Le langage CSS

1. Les façons d'écrire du CSS
2. Les commentaires
3. Les sélecteurs en CSS
4. **Le formatage du texte**
5. Le fond/avant-plan
6. Le modèle des boîtes
7. Le positionnement en CSS
8. Les listes
9. Les liens
10. Les tableaux
11. Le curseur
12. Les transformations
13. Les pseudo-classes
14. Les Media Queries
15. TP2 CSS

# Le formatage du texte

## 7- Définir une nouvelle police personnalisée avec « @font-face » :

### A- Télécharger une police :

➤ On peut utiliser les liens suivants qui proposent des polices gratuites à télécharger:

- <https://www.fontsquirrel.com/>
- <https://www.dafont.com/fr/>
- <https://fonts.google.com/>

➤ Il existe différents formats de fichiers de police :

- « **.ttf** » : TrueType Font. Fonctionne sur IE9 et tous les autres navigateurs.
- « **.eot** » : Embedded OpenType. Fonctionne sur Internet Explorer uniquement, toutes versions. Ce format est propriétaire de Microsoft.
- « **.otf** » : OpenType Font. Ne fonctionne pas sur Internet Explorer
- « **.svg** » : SVG Font. Le seul format reconnu sur les iPhone et iPad pour le moment.
- « **.woff** » : Web Open Font Format. Nouveau format conçu pour le Web qui fonctionne sur IE9 et tous les autres navigateurs.

Partie 2 - Le langage CSS

- 1. Les façons d'écrire du CSS
- 2. Les commentaires
- 3. Les sélecteurs en CSS
- 4. Le formatage du texte
- 5. Le fond/avant-plan
- 6. Le modèle des boîtes
- 7. Le positionnement en CSS
- 8. Les listes
- 9. Les liens
- 10. Les tableaux
- 11. Le curseur
- 12. Les transformations
- 13. Les pseudo-classes
- 14. Les Media Queries
- 15. TP2 CSS

# Le formatage du texte

## 7- Définir une nouvelle police personnalisée avec « @font-face » :

### B- Définir une nouvelle police :

➤ Pour définir une nouvelle police, on doit la déclarer avec :

```
@font-face {  
    font-family: 'MaPolice'; /* Le nom qu'on veut donner à la police */  
    src: url('MaPolice.eot'); /* Le nom du fichier de la police téléchargée */  
}
```

➤ Afin que la police soit prise en compte par la plupart des navigateurs, on doit proposer plusieurs formats de police et le navigateur téléchargera celui qu'il sait lire :

```
@font-face {  
    font-family: 'MaPolice'; /* Le nom qu'on veut donner à la police */  
    src: url('MaPolice.eot') format('eot'), /* La police téléchargée format «eot» */  
        url('MaPolice.woff') format('woff'), /* La police téléchargée format «woff» */  
        url('MaPolice.ttf') format('truetype'), /* La police téléchargée format «ttf» */  
        url('MaPolice.svg') format('svg'); /* La police téléchargée format «svg» */  
}
```

### C- Utiliser la police définie :

```
font-family: MaPolice;
```



## Partie 2 - Le langage CSS

1. Les façons d'écrire du CSS
2. Les commentaires
3. Les sélecteurs en CSS
- 4. Le formatage du texte**
5. Le fond/avant-plan
6. Le modèle des boîtes
7. Le positionnement en CSS
8. Les listes
9. Les liens
10. Les tableaux
11. Le curseur
12. Les transformations
13. Les pseudo-classes
14. Les Media Queries
15. TP2 CSS

# Le formatage du texte

## 8- Décorer le texte :

```
text-decoration: underline; /* Texte souligné.*/  
text-decoration: line-through; /* Texte barré.*/  
text-decoration: overline; /* Texte avec ligne au-dessus.*/  
text-decoration: blink; /* Texte clignotant.*/  
text-decoration: none; /* Texte normal (par défaut).*/
```

## 9- Aligner le texte :

➤ Il fonctionne que sur des balises de type block (<p>, <div>, <h1>, <h2>, ...).

```
text-align: left; /* Le texte sera aligné à gauche (c'est le réglage par défaut).*/  
text-align: center; /* Le texte sera centré.*/  
text-align: right; /* Le texte sera aligné à droite.*/  
text-align: justify; /* Le texte sera "justifié".*/  
text-align: inherit;
```

## 10- Indenter le texte :

```
text-indent: 20px; /*Décale le texte de 20 pixel vers la droite (-valeur: vers la gauche).*/
```

## 11- Changer l'aspect des caractères d'un texte :

```
text-transform: uppercase; /* Transforme le texte en majuscule.*/  
text-transform: capitalize; /* Transforme seulement la 1ère lettre en majuscule.*/  
text-transform: lowercase; /* Transforme en minuscule.*/
```

## Partie 2 - Le langage CSS

1. Les façons d'écrire du CSS
2. Les commentaires
3. Les sélecteurs en CSS
4. Le formatage du texte
5. Le fond/avant-plan
6. Le modèle des boîtes
7. Le positionnement en CSS
8. Les listes
9. Les liens
10. Les tableaux
11. Le curseur
12. Les transformations
13. Les pseudo-classes
14. Les Media Queries
15. TP2 CSS

# Le formatage du texte

## 12- Ajouter des ombres :

```
text-shadow: 3px 2px 4px rgba(56, 123, 234, 1); /*1. Le décalage horizontal de l'ombre (px)
                                                    2. Le décalage vertical de l'ombre (px)
                                                    3. L'adoucissement du dégradé ou la
                                                       propagation de l'ombre(px)
                                                    4. La couleur de l'ombre (color)*/
```

## 13- Ajuster l'espace entre les lettres et les mots :

```
letter-spacing: 5px; /* L'espace entre les lettres. En em, px... */
word-spacing: 10px; /* L'espace entre les mots */
```

## 14- Changer la couleur du texte :

```
color: red;
color: blue;
color: purple; /* Couleur violet. */
color: aqua; /* Couleur bleu très clair. */
color: green;
color: #FFFFFF; /* Valeur en hexadécimal. */
color: rgba(56, 123, 234, 1); /* Couleur en (rouge, vert, bleu, transparence entre 0 et 1) */
```

## 15- Changer la transparence du texte :

```
opacity: [0,1]; /* La transparence entre 0 et 1 */
```

## Partie 2 - Le langage CSS

1. Les façons d'écrire du CSS
2. Les commentaires
3. Les sélecteurs en CSS
4. Le formatage du texte
- 5. Le fond/avant-plan**
6. Le modèle des boîtes
7. Le positionnement en CSS
8. Les listes
9. Les liens
10. Les tableaux
11. Le curseur
12. Les transformations
13. Les pseudo-classes
14. Les Media Queries
15. TP2 CSS

# Le fond/avant-plan

## 1- Donner une couleur de fond :

```
background-color: red; /* Un fond rouge */
```

## 2- Donner une image de fond :

```
background-image: url("img.png"); /* Si l'image ne couvre pas le block, elle sera répétée */
```

## 3- Activer/Désactiver la répétition de l'image du fond :

```
background-repeat: no-repeat; /* Permet d'éviter la répétition de l'image du fond */  
background-repeat: repeat; /* Répétition suivant l'axe X et Y (par défaut) */  
background-repeat: repeat-x; /* Répétition que suivant l'axe X */  
background-repeat: repeat-y; /* Répétition que suivant l'axe Y */
```

## 4- Adapter l'image à la taille du block :

```
background-size: cover; /* L'image s'adapte et couvre toute la partie visible du block */
```

## 5- Fixer l'image du fond :

```
background-attachment: fixed; /* Permet de faire glisser le texte sur l'image du fond sans  
la faire bouger */  
background-attachment: scroll; /* Permet de défiler le fond avec la page (par défaut) */
```

## 6- Positionner l'image du fond :

```
background-position: left top; /* Par exemple « 10px 20px » positionnera l'image à 10px par  
rapport au bord gauche et à 20px par rapport au bord haut */
```

## Partie 2 - Le langage CSS

1. Les façons d'écrire du CSS
2. Les commentaires
3. Les sélecteurs en CSS
4. Le formatage du texte
- 5. Le fond/avant-plan**
6. Le modèle des boîtes
7. Le positionnement en CSS
8. Les listes
9. Les liens
10. Les tableaux
11. Le curseur
12. Les transformations
13. Les pseudo-classes
14. Les Media Queries
15. TP2 CSS

# Le fond/avant-plan

## 7- Regrouper les commande de mise en forme de l'image du fond :

➤ Le regroupement des éléments de l'image de fond doit respecter la forme :  
**background: [background-image] [background-attachment] [background-repeat] [background-position];**

```
background: url("img.png") fixed no-repeat left top;
```

## 8- Ajouter plusieurs images de fond :

```
background: url("img0.png") fixed no-repeat left top,  
            url("img1.png") fixed no-repeat left top,  
            url("img2.png") fixed no-repeat left top;
```

## 9- Appliquer un dégradé de couleurs :

➤ Pour faire un dégradé il faut utiliser les préfixes vendeurs:

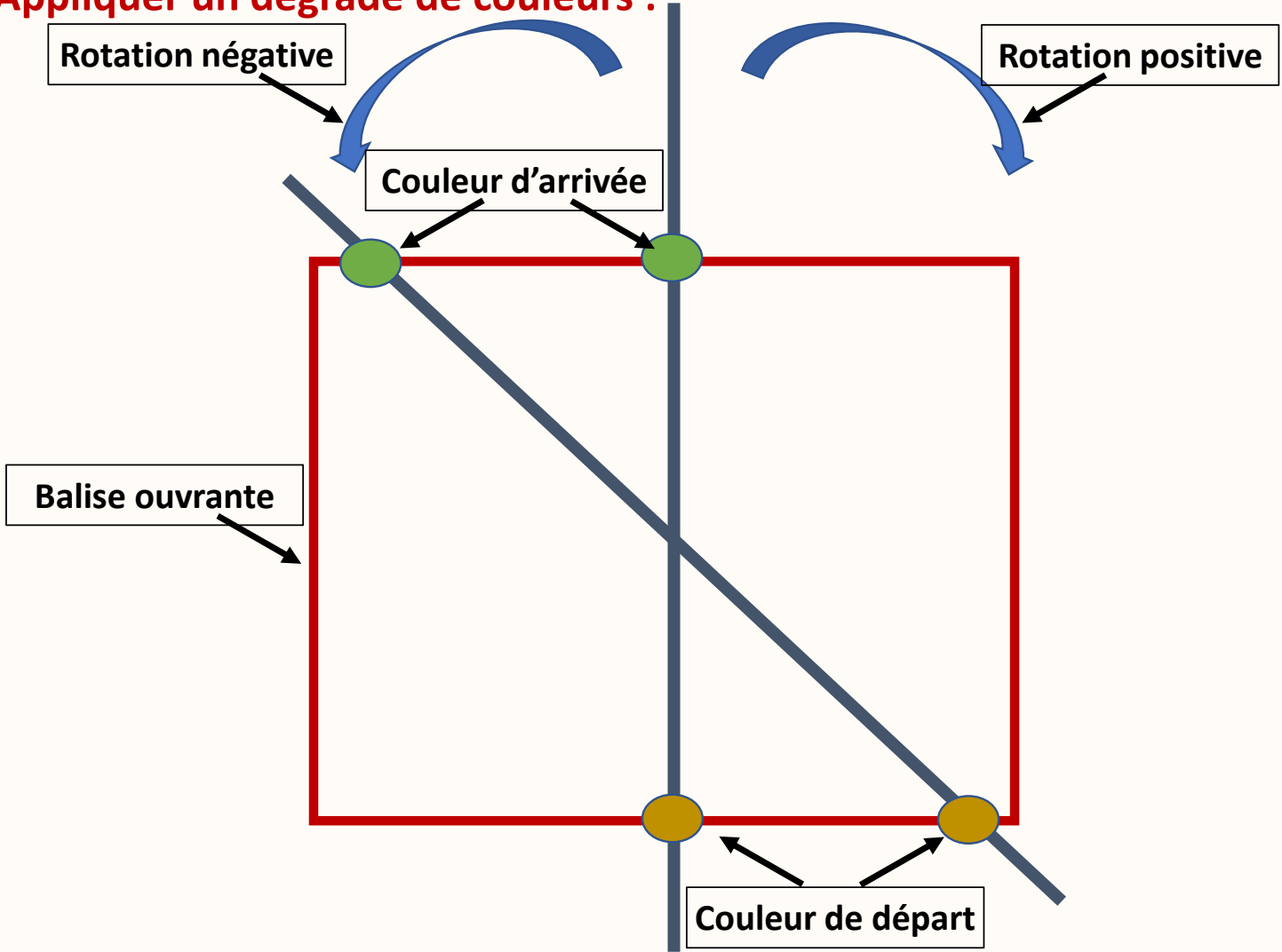
- Chrome et Safari: **-webkit-**
- Mozilla Firefox: **-moz-**
- Internet Explorer: **-ms-**
- Opera: **-o-**

Partie 2 - Le langage CSS

- 1. Les façons d'écrire du CSS
- 2. Les commentaires
- 3. Les sélecteurs en CSS
- 4. Le formatage du texte
- 5. Le fond/avant-plan
- 6. Le modèle des boîtes
- 7. Le positionnement en CSS
- 8. Les listes
- 9. Les liens
- 10. Les tableaux
- 11. Le curseur
- 12. Les transformations
- 13. Les pseudo-classes
- 14. Les Media Queries
- 15. TP2 CSS

Le fond/avant-plan

9- Appliquer un dégradé de couleurs :



## Partie 2 - Le langage CSS

1. Les façons d'écrire du CSS
2. Les commentaires
3. Les sélecteurs en CSS
4. Le formatage du texte
- 5. Le fond/avant-plan**
6. Le modèle des boîtes
7. Le positionnement en CSS
8. Les listes
9. Les liens
10. Les tableaux
11. Le curseur
12. Les transformations
13. Les pseudo-classes
14. Les Media Queries
15. TP2 CSS

# Le fond/avant-plan

## 9- Appliquer un dégradé de couleurs (suite) :

`background: linear-gradient(red, blue);` /\* Un dégradé linéaire. On peut mettre en tant de couleur qu'on veut. De haut -> bas \*/

`background: linear-gradient(blue, red, yellow);` /\* Un dégradé de plusieurs couleurs. on peut mettre des valeurs « rgb » et « hexa » \*/

`background: -webkit-linear-gradient(left, red, blue);` /\* Un dégradé avec préfixe vendeur \*/

`background: linear-gradient(to right, red, blue);` /\* Un dégradé linéaire de gauche -> droite \*/

`background: linear-gradient(to left, red, blue);` /\* Un dégradé linéaire de droite -> gauche \*/

`background: linear-gradient(to top, red, blue);` /\* Un dégradé linéaire de bas -> haut \*/

`background: linear-gradient(to bottom, red, blue);` /\* Un dégradé linéaire de haut -> bas \*/

`background: linear-gradient(to bottom right, red, blue);` /\* Un dégradé diagonal de haut à gauche vers en bas à droite. idem pour les autres dégradés diagonaux \*/

/\* La méthode précise avec des valeurs en degré. \*/

`background: linear-gradient(90deg, red, blue);` /\* Un dégradé linéaire de gauche -> droite \*/

/\* Dégradé radial: \*/

`background: radial-gradient(blue, red, yellow);` /\* Un dégradé à partir d'un point central \*/

`background: radial-gradient(0px 0px, blue, red, yellow);` /\* Permet de faire un dégradé à partir d'un point central. La 1ère valeur (largeur hauteur) précise les dimensions du dégradé. \*/

Partie 2 - Le langage CSS

- 1. Les façons d'écrire du CSS
- 2. Les commentaires
- 3. Les sélecteurs en CSS
- 4. Le formatage du texte
- 5. Le fond/avant-plan
- 6. Le modèle des boîtes
- 7. Le positionnement en CSS
- 8. Les listes
- 9. Les liens
- 10. Les tableaux
- 11. Le curseur
- 12. Les transformations
- 13. Les pseudo-classes
- 14. Les Media Queries
- 15. TP2 CSS

# Le modèle des boîtes

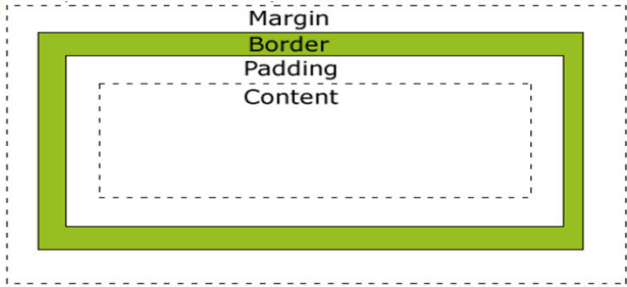
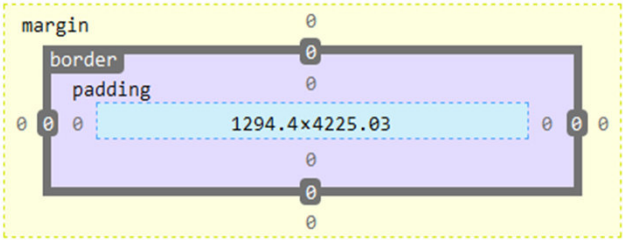
- La largeur par défaut d'un élément est définit par son type. La hauteur par défaut d'un élément est définit par son contenu.
- En HTML, la plupart des balises peuvent se ranger dans deux catégories :
  - Les balises block : crée automatiquement un retour à la ligne avant et après (<p></p>).
  - Les balises inline : se trouve obligatoirement à l'intérieur d'une balise "block". Elle ne crée pas de retour à la ligne à la fin. Le texte qui se trouve à l'intérieur s'écrit donc à la suite du texte précédent, sur la même ligne (<a></a>).
- Les éléments « inline » ne prennent que leurs places nécessaires alors que les éléments « block » prennent toute la largeur.

```
<h1>Titre (block)</h1>

<p>Paragraphe blablabla blablabla
blablabla <a>Lien inline</a> blabla
blablabla et toujours blabla
</p>
```

| Balises blocks | Balises inline |
|----------------|----------------|
| <p>            | <em>           |
| <footer>       | <strong>       |
| <h1>           | <mark>         |
| <h2>           | <a>            |
| <article>      | <img />        |
| ...            | ...            |

➤ Les marges :



Partie 2 - Le langage CSS

- 1. Les façons d'écrire du CSS
- 2. Les commentaires
- 3. Les sélecteurs en CSS
- 4. Le formatage du texte
- 5. Le fond/avant-plan
- 6. Le modèle des boîtes
- 7. Le positionnement en CSS
- 8. Les listes
- 9. Les liens
- 10. Les tableaux
- 11. Le curseur
- 12. Les transformations
- 13. Les pseudo-classes
- 14. Les Media Queries
- 15. TP2 CSS

# Le modèle des boîtes

## 1- Modifier les dimensions d'un block :

```
width: 200px;      /* Changer la largeur */
min-width: 20%;    /* Pour indiquer une largeur minimal */
max-width: 20%;    /* Pour indiquer une largeur max */
height: 100px;     /* Changer la hauteur */
height: auto;      /* Laisser le navigateur adapter la hauteur */
```

## 2- Éviter que le texte dépasse le bloc :

```
overflow: hidden; /* Permet de cacher le texte qui dépasse le bloc */
overflow: auto;    /* Permet de rajouter des barres de défilement quand c'est nécessaire */
word-wrap: break-word; /* Pour couper les textes trop larges au lieu d'utiliser une barre de défilement horizontale */
overflow: scroll;   /* Affiche les barres de défilement qui permet de voir le texte */
```

## 3- Modifier la marge intérieure :

➤ C'est la distance entre le contenu d'un élément HTML et ses bords.

```
padding: 5px; /* La distance entre le contenu d'un élément HTML et ses bords. Ici « 5px » */
padding: 5px 10px; /* Si on donne deux valeurs, alors la première valeur correspond à « top » et « bottom » et la second à « left » et « right » */
padding: 5px 10px 5px 10px; /* Si on donne 4 valeurs, alors elles correspondent successivement à « top », « right », « bottom » et « left » */
```



Partie 2 - Le langage CSS

- 1. Les façons d'écrire du CSS
- 2. Les commentaires
- 3. Les sélecteurs en CSS
- 4. Le formatage du texte
- 5. Le fond/avant-plan
- 6. Le modèle des boîtes
- 7. Le positionnement en CSS
- 8. Les listes
- 9. Les liens
- 10. Les tableaux
- 11. Le curseur
- 12. Les transformations
- 13. Les pseudo-classes
- 14. Les Media Queries
- 15. TP2 CSS

# Le modèle des boîtes

## 4- Modifier la marge extérieure :

➤ C'est la distance entre un élément HTML et ses voisins.

```
margin: 5px; /* La distance entre un élément HTML et ses voisins. Ici « 5px » */
margin: 5px 10px; /* Si on donne deux valeurs, alors la première valeur correspond à
                  « top » et « bottom » et la second à « left » et « right »*/
margin: 5px 10px 5px 10px; /* Si on donne 4 valeurs, alors elles correspondent
                             successivement à « top », « right », « bottom » et « left » */
```

## 5- Center un bloc horizontalement :

➤ Pour center un bloc horizontalement, on lui donne une largeur et « margin: auto »

```
width: 200px; /* Changer la largeur */
margin: auto; /* Marge calculée automatiquement */
```

## 6- Donner une bordure :

```
border-width: 2px; /* La taille de la bordure */
border-color: rgb(56, 123, 234, 1); /* La couleur de la bordure */
/* Type de la bordure : */
border-style: none; /* Pas de bordure (par défaut) */
border-style: solid; /* Un trait simple. */
border-style: dotted; /* Pointillés. */
border-style: dashed; /* Tirets. */
border-style: double; /* Bordure double. */
border-style: groove; /* En relief. */
border-style: ridge; /* Autre effet relief. */
border-style: inset; /* Effet 3D enfoncé. */
border-style: outset; /* Effet 3D surélevé. */
```

## Partie 2 - Le langage CSS

1. Les façons d'écrire du CSS
2. Les commentaires
3. Les sélecteurs en CSS
4. Le formatage du texte
5. Le fond/avant-plan
- 6. Le modèle des boîtes**
7. Le positionnement en CSS
8. Les listes
9. Les liens
10. Les tableaux
11. Le curseur
12. Les transformations
13. Les pseudo-classes
14. Les Media Queries
15. TP2 CSS

# Le modèle des boîtes

## 6- Donner une bordure :

```
/* L'arrondissement de la bordure: */  
border-radius: 10px;           /* Donne la même valeur aux 4 coins de la bordure */  
border-top-left-radius: 10px;  /* coin en haut à gauche seulement */  
border-top-right-radius: 10px; /* coin en haut à droite seulement */  
border-bottom-left-radius: 10px; /* coin en bas à gauche seulement */  
border-bottom-right-radius: 10px; /* coin en bas à droite seulement */  
border-radius: 3px 3px 3px 3px; /* Arrondir tous les coins en un seul coup. Ça correspond  
à (top left, top right, bottom right, bottom left) */
```

## 7- Combiner les valeurs de la bordure :

➤ Le regroupement des éléments de la bordure doit respecter la forme :

**border: [border-width] [border-color] [border-style];**

```
border: 5px rgba(0,0,0,1) solid; /* Pour combiner taille, couleur et style. */  
border-top: 5px rgba(0,0,0,1) solid; /* Bordure en haut. */  
border-bottom: 5px rgba(0,0,0,1) solid; /* Bordure en bas. */  
border-left: 5px rgba(0,0,0,1) solid; /* Bordure à gauche. */  
border-right: 5px rgba(0,0,0,1) solid; /* Bordure à droite. */
```

## Partie 2 - Le langage CSS

1. Les façons d'écrire du CSS
2. Les commentaires
3. Les sélecteurs en CSS
4. Le formatage du texte
5. Le fond/avant-plan
- 6. Le modèle des boîtes**
7. Le positionnement en CSS
8. Les listes
9. Les liens
10. Les tableaux
11. Le curseur
12. Les transformations
13. Les pseudo-classes
14. Les Media Queries
15. TP2 CSS

# Le modèle des boîtes

## 8- Ajouter une ombre à la bordure :

```
box-shadow: 1px 1.5px 3px rgba(0,0,0,0.6); /*Crée des ombres autour de la boîte*/  
/*  
▪ La valeur 1: Le décalage horizontal de l'ombre (px).  
▪ La valeur 2: Le décalage vertical de l'ombre (px).  
▪ La valeur 3: L'adoucissement du dégradé ou la propagation de l'ombre(px).  
▪ La valeur 4: La couleur de l'ombre (color).  
▪ La valeur 5: 'inset' si on l'ajoute ça inverse le l'ombre et le met du côté intérieur.  
*/
```

## 9- Rendre un élément invisible :

```
display: none;
```

## Partie 2 - Le langage CSS

1. Les façons d'écrire du CSS
2. Les commentaires
3. Les sélecteurs en CSS
4. Le formatage du texte
5. Le fond/avant-plan
6. Le modèle des boîtes
7. Le positionnement en CSS
8. Les listes
9. Les liens
10. Les tableaux
11. Le curseur
12. Les transformations
13. Les pseudo-classes
14. Les Media Queries
15. TP2 CSS

# Le positionnement en CSS

- Les éléments de type « **inline** » ( <a>, <img>, <span>...) ne prennent que leurs places nécessaires alors que les éléments de type « **block** » ( <h1>, <p>, <div>, ...) prennent toute la largeur.
- On peut définir un nouveau type en CSS, « inline-block » qui permet de récupérer les meilleurs propriétés des éléments « inline » et « block ». Ce sont des éléments dont on peut changer la largeur (width) et la hauteur (height) et en plus ils s'alignent en permettant à d'autres éléments de se mettre à côté s'il y a suffisamment de place.

## 1- Faire flotter un élément autour d'un autre :

```
float: left; /*L'élément flottera à gauche en se positionnant sur le bord de l'élément parent*/  
float: right; /*L'élément flottera à droite*/
```

- Si l'élément flottant est de type « block » ou « inline-block », on peut lui donner une largeur et une marge extérieure :

```
width: 140px;  
margin-right: 20px; /* Si l'élément flotte à droite */  
margin-left: 20px; /* Si l'élément flotte à gauche */
```

- Pour arrêter le flottement des éléments, on applique sur le premier élément où on veut arrêter le flottement :

```
clear: left; /* Le texte se poursuit en-dessous après un float: left; */  
clear: right; /* Le texte se poursuit en-dessous après un float: right; */  
clear: both; /* Le texte se poursuit en-dessous, que ce soit après un float: left ou right; */
```

Partie 2 - Le langage CSS

- 1. Les façons d'écrire du CSS
- 2. Les commentaires
- 3. Les sélecteurs en CSS
- 4. Le formatage du texte
- 5. Le fond/avant-plan
- 6. Le modèle des boîtes
- 7. Le positionnement en CSS
- 8. Les listes
- 9. Les liens
- 10. Les tableaux
- 11. Le curseur
- 12. Les transformations
- 13. Les pseudo-classes
- 14. Les Media Queries
- 15. TP2 CSS

# Le positionnement en CSS

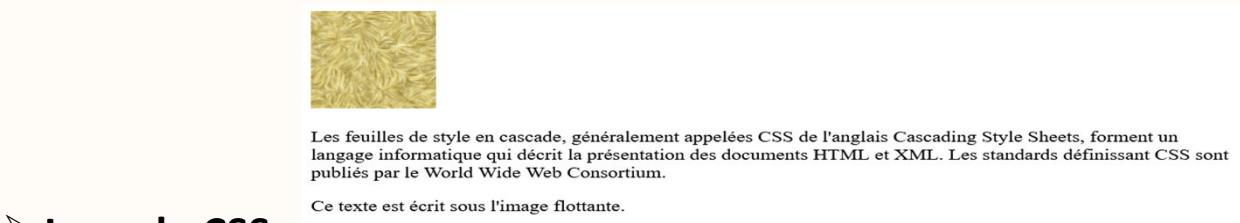
## 1- Faire flotter un élément autour d'un autre (suite) :

➤ Le code HTML :

```

<p>Les feuilles de style en cascade, généralement appelées CSS de l'anglais Cascading Style
Sheets, forment un langage informatique qui décrit la présentation des documents HTML et
XML. Les standards définissant CSS sont publiés par le World Wide Web Consortium.</p>
<p class="dessous">Ce texte est écrit sous l'image flottante.</p>
```

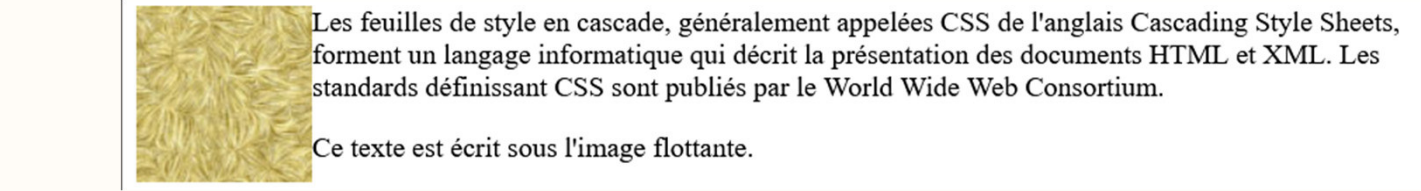
➤ Résultat visuel :



➤ Le code CSS :

```
.imageFlottante
{
    float: left;
}
```

➤ Résultat visuel :



## Partie 2 - Le langage CSS

1. Les façons d'écrire du CSS
2. Les commentaires
3. Les sélecteurs en CSS
4. Le formatage du texte
5. Le fond/avant-plan
6. Le modèle des boîtes
7. Le positionnement en CSS
8. Les listes
9. Les liens
10. Les tableaux
11. Le curseur
12. Les transformations
13. Les pseudo-classes
14. Les Media Queries
15. TP2 CSS

# Le positionnement en CSS

## 1- Faire flotter un élément autour d'un autre (suite) :

➤ Arrêter le flottement à partir de l'élément de class « dessous » :

```
.dessous  
{  
  clear: left;  
}
```

➤ Résultat visuel :



Ce texte est écrit sous l'image flottante.

Les feuilles de style en cascade, généralement appelées CSS de l'anglais Cascading Style Sheets, forment un langage informatique qui décrit la présentation des documents HTML et XML. Les standards définissant CSS sont publiés par le World Wide Web Consortium.

## 2- Changer le type d'un élément HTML :

|                                     |                                                                                                                                            |
|-------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------|
| <code>display: inline;</code>       | <code>/* Changer les balises « bloc » en balises « inline » afin qu'ils s'écrivent les uns à la suite des autres */</code>                 |
| <code>display: block;</code>        | <code>/* Changer les balises « inline » en balise « bloc ». */</code>                                                                      |
| <code>display: inline-block;</code> | <code>/* Positionner les bloc les un à coté des autres, et en plus donner leurs une largeur « width » et une hauteur « height ». */</code> |

## Partie 2 - Le langage CSS

1. Les façons d'écrire du CSS
2. Les commentaires
3. Les sélecteurs en CSS
4. Le formatage du texte
5. Le fond/avant-plan
6. Le modèle des boîtes
7. Le positionnement en CSS
8. Les listes
9. Les liens
10. Les tableaux
11. Le curseur
12. Les transformations
13. Les pseudo-classes
14. Les Media Queries
15. TP2 CSS

# Le positionnement en CSS

## 3- Aligner les éléments « inline-block » qui se trouvent l'un à côté de l'autre :

|                                        |                                                                                                   |
|----------------------------------------|---------------------------------------------------------------------------------------------------|
| <code>vertical-align: top;</code>      | <code>/* Aligne les éléments par rapport au bord haut */</code>                                   |
| <code>vertical-align: baseline;</code> | <code>/* Aligne la base de l'élément avec celle de l'élément parent (valeur par défaut) */</code> |
| <code>vertical-align: middle;</code>   | <code>/* Aligne les éléments par rapport au milieu */</code>                                      |
| <code>vertical-align: bottom;</code>   | <code>/* Aligne les éléments par rapport au bord bas */</code>                                    |
| <code>vertical-align: 10px;</code>     | <code>/* Aligne à une distance donnée par rapport à la ligne de base */</code>                    |

## 4- Centrer un élément « block » ou « inline-block » :

➤ Pour centrer un bloc horizontalement, on lui donne une largeur et « `margin: auto` »

|                            |                                                   |
|----------------------------|---------------------------------------------------|
| <code>width: 200px;</code> | <code>/* Changer la largeur */</code>             |
| <code>margin: auto;</code> | <code>/* Marge calculée automatiquement */</code> |

## 5- Mettre ou non le contenu d'un élément sur la même ligne :

➤ Pour décider si on veut que le contenu, d'un élément conteneur, qui peut être un texte ou des éléments en « inline-block » reste sur la même ligne même si ça dépasse la largeur du conteneur, on utilise la propriété « `white-space` » :

|                                   |                                                                                                                  |
|-----------------------------------|------------------------------------------------------------------------------------------------------------------|
| <code>white-space: nowrap;</code> | <code>/* Pour garder tous les éléments « inline-block » ou texte sur la même ligne même si ça dépasse. */</code> |
|                                   | <code>/* Des barres de défilement vont se rajouter en cas de dépassement. */</code>                              |
| <code>white-space: normal;</code> | <code>/* L'état normal qui permet un retour à la ligne si ça dépasse la ligne */</code>                          |

## Partie 2 - Le langage CSS

1. Les façons d'écrire du CSS
2. Les commentaires
3. Les sélecteurs en CSS
4. Le formatage du texte
5. Le fond/avant-plan
6. Le modèle des boîtes
- 7. Le positionnement en CSS**
8. Les listes
9. Les liens
10. Les tableaux
11. Le curseur
12. Les transformations
13. Les pseudo-classes
14. Les Media Queries
15. TP2 CSS

# Le positionnement en CSS

## 6- Forcer les éléments d'un block à se mettre dans la même ligne :

- Pour décider si on veut que le contenu, d'un élément conteneur, qui peut être un texte ou des éléments en « inline-block » reste sur la même ligne même si ça dépasse la largeur du conteneur, on utilise la propriété « white-space » :

```
#conteneur {  
  display: flex; /* Garder les blocs dans le contenu sur la même ligne même si ça dépasse la  
                 largeur. Ça adapte la largeur des éléments afin que ça rentre dans la ligne. */  
  
  justify-content: center; /* Éléments groupés au centre */  
  justify-content: left;   /* Éléments groupés à gauche */  
  justify-content: right;  /* Éléments groupés à droite */  
  
  justify-content: normal; /* Alignement normal */  
  
  /* Alignement distribué : */  
  justify-content: space-between; /* Les éléments sont répartis équitablement. Pas d'espace  
                                   entre le premier et dernier éléments et les bords du  
                                   conteneur. */  
  justify-content: space-around; /* Marge équitable autour des éléments même les  
                                   éléments aux extrémités. */  
  justify-content: space-evenly; /* Espace équitable entre les éléments même entre les  
                                   éléments aux extrémités les bords du conteneur. */  
}
```



## Partie 2 - Le langage CSS

1. Les façons d'écrire du CSS
2. Les commentaires
3. Les sélecteurs en CSS
4. Le formatage du texte
5. Le fond/avant-plan
6. Le modèle des boîtes
- 7. Le positionnement en CSS**
8. Les listes
9. Les liens
10. Les tableaux
11. Le curseur
12. Les transformations
13. Les pseudo-classes
14. Les Media Queries
15. TP2 CSS

# Le positionnement en CSS

## 7- Positionnement absolue :

- Pour positionner un élément en position absolue par rapport à la page indépendamment des autres éléments de la page on utilise :

```
position: absolute; /* On précise qu'on positionne l'élément en absolu */
/* Ensuite on précise les coordonnées de l'élément par rapport au bord de la fenêtre : */
top: 100px; /* Position par rapport au bord en haut */
bottom: 50px; /* Position par rapport au bord en bas */
left: 50px; /* Position par rapport au bord à gauche */
right: 50px; /* Position par rapport au bord à droite */
```

- Si le bloc conteneur est positionné en absolu, fixe ou relatif, alors le bloc se positionnera par rapport au bords du bloc conteneur.
- Donc pour positionner un élément par rapport à son parent, il suffit d'ajouter :

```
#parent { position: relative; } /* Ne changera rien pour l'élément parent */
#enfant {
    position: absolute; /* Permettra de positionner l'élément par rapport à son parent */
    /* La position par rapport à l'élément parent */
}
```

- Les éléments positionnés en absolu sont placés par-dessus du reste des éléments de la page. Pour indiquer quel élément doit apparaître par-dessus des autres, on spécifie le « z-index » :

```
z-index: 1; /* L'élément ayant « z-index » le plus élevé sera placé par-dessus des autres. */
```

## Partie 2 - Le langage CSS

1. Les façons d'écrire du CSS
2. Les commentaires
3. Les sélecteurs en CSS
4. Le formatage du texte
5. Le fond/avant-plan
6. Le modèle des boîtes
- 7. Le positionnement en CSS**
8. Les listes
9. Les liens
10. Les tableaux
11. Le curseur
12. Les transformations
13. Les pseudo-classes
14. Les Media Queries
15. TP2 CSS

# Le positionnement en CSS

## 8- Positionnement fixe :

- Positionner un élément en position fixe par rapport à la partie visible de la page indépendamment des autres éléments :

```
position: fixed;      /* On précise qu'on positionne l'élément en fixe */  
/* Ensuite on précise les coordonnées de l'élément par rapport au bord de la fenêtre : */  
top: 100px;          /* Position par rapport au bord en haut */  
bottom: 50px;        /* Position par rapport au bord en bas */  
left: 50px;           /* Position par rapport au bord à gauche */  
right: 50px;         /* Position par rapport au bord à droite */
```

## 9- Positionnement relatif :

- Positionner un élément en position relative par rapport à sa position initiale (le décaler par rapport à sa position) :

```
position: relative;   /* On précise qu'on positionne l'élément en relatif */  
/* Ensuite on précise les coordonnées de l'élément par rapport à sa position initiale : */  
top: 100px;           /* Position par rapport au bord en haut */  
bottom: 50px;         /* Position par rapport au bord en bas */  
left: 50px;            /* Position par rapport au bord à gauche */  
right: 50px;          /* Position par rapport au bord à droite */
```

## Partie 2 - Le langage CSS

1. Les façons d'écrire du CSS
2. Les commentaires
3. Les sélecteurs en CSS
4. Le formatage du texte
5. Le fond/avant-plan
6. Le modèle des boîtes
- 7. Le positionnement en CSS**
8. Les listes
9. Les liens
10. Les tableaux
11. Le curseur
12. Les transformations
13. Les pseudo-classes
14. Les Media Queries
15. TP2 CSS

# Le positionnement en CSS

## 10- Centrer un élément « block » ou « inline-block » verticalement :

➤ Pour centrer un bloc verticalement, on doit ajouter à son élément parent la propriété « `display: flex;` » :

```
/* Le conteneur (l'élément parent) : */
#conteneur
{
    display: flex;
    width: 100%; /* La largeur du conteneur */
    height: 100%; /* La hauteur du conteneur */
}

/* Le bloc qu'on veut centrer dans l'élément parent « #contenuer » : */
#bloc
{
    width: 980px; /* Obligatoire si on veut centrer le bloc horizontalement */
    height: 610px; /* Obligatoire si on veut centrer le bloc verticalement pour Internet
                    explorer mais pas pour les autres. */
    margin: auto;
}
```

## Partie 2 - Le langage CSS

1. Les façons d'écrire du CSS
2. Les commentaires
3. Les sélecteurs en CSS
4. Le formatage du texte
5. Le fond/avant-plan
6. Le modèle des boîtes
7. Le positionnement en CSS
8. Les listes
9. Les liens
10. Les tableaux
11. Le curseur
12. Les transformations
13. Les pseudo-classes
14. Les Media Queries
15. TP2 CSS

# Les listes

## 1- Listes non ordonnées :

```
ul
{
    /* Modifier le style des puces : */
    list-style-type: none;      /* Pas de puce. */
    list-style-type: disc;      /* Disque plein (par défaut). */
    list-style-type: circle;    /* Cercle (vide). */
    list-style-type: square;    /* Carré. */

    /* Utiliser une image comme puce : */
    list-style-image: url("img.png"); /* Utiliser une image personnalisée pour la puce. */
}
```

## 2- Listes ordonnées :

```
ol li
{
    /* Modifier le style de numérotation : */
    list-style-type : upper-roman; /*Numéro en chiffres romains, en utilisant des majuscules.*/
    list-style-type : lower-roman; /*Numéro en chiffres romains, en utilisant des minuscules.*/
    list-style-type : upper-latin; /*Lettre de l'alphabet latin, en utilisant des majuscules.*/
    list-style-type : lower-latin; /*Lettre de l'alphabet latin, en utilisant des minuscules.*/
}
```

Partie 2 - Le langage CSS

- 1. Les façons d'écrire du CSS
- 2. Les commentaires
- 3. Les sélecteurs en CSS
- 4. Le formatage du texte
- 5. Le fond/avant-plan
- 6. Le modèle des boîtes
- 7. Le positionnement en CSS
- 8. Les listes
- 9. Les liens
- 10. Les tableaux
- 11. Le curseur
- 12. Les transformations
- 13. Les pseudo-classes
- 14. Les Media Queries
- 15. TP2 CSS

# Les liens

## 1- Changer la mise en forme des liens :

```
a{  
  color: red;  
  text-decoration: none;  
}
```

## 2- Changer la mise en forme des liens ayant le focus :

```
a:focus  
{ background-color: red; }
```

## 3- Changer la mise en forme des liens au survole :

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

## 4- Changer la mise en forme des liens actifs :

```
a:active  
{ background-color: red; }
```

## 5- Changer la mise en forme des liens visités :

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

## 6- Changer la mise en forme des liens non visités :

```
a:link  
{ background-color: red; }
```

Partie 2 - Le langage CSS

- 1. Les façons d'écrire du CSS
- 2. Les commentaires
- 3. Les sélecteurs en CSS
- 4. Le formatage du texte
- 5. Le fond/avant-plan
- 6. Le modèle des boîtes
- 7. Le positionnement en CSS
- 8. Les listes
- 9. Les liens
- 10. Les tableaux
- 11. Le curseur
- 12. Les transformations
- 13. Les pseudo-classes
- 14. Les Media Queries
- 15. TP2 CSS

# Les tableaux

## 1- Donner une bordure au tableau :

```
td
{
  border: 1px solid black;
}
```

## 2- Fusionner les bordures des cellules :

```
table
{
  border-collapse: collapse; /* Les bordures seront collées entre elles. */
  border-collapse: separate; /* Les bordures seront dissociées (valeur par défaut). */
}
```

## 3- Positionner le titre du tableau :

```
table
{
  caption-side: top; /* Le titre sera placé en haut du tableau (par défaut). */
  caption-side: bottom; /* Le titre sera placé en bas du tableau. */
  caption-side: left; /* Le titre sera placé à gauche du tableau. */
  caption-side: right; /* Le titre sera placé à droite du tableau. */
}
```