

HTML 5 : A la découverte des fonctionnalités de bases

Table des matières



Objectifs	4
I - Comment fonctionne les sites web ?	5
II - Validation et compatibilité	6
III - HTML : Evolution	7
IV - HTML : le langage de balisage	8
V - L'éditeur de texte	9
VI - Comment structuré une page web ?	10
1. Structure d'une page Web	10
2. L'indentation du code	11
3. L'en-tête <HEAD>	12
4. Le corps<BODY>	12
VII - Codes des Couleurs	14
VIII - Mise en forme en HTML	15
1. Paragraphes	15
2. Titres	15
3. Texte	16
4. Style	17
5. Listes	18
IX - Les images en HTML	20
1. Les formats d'images	20
2. Insertion d'images	21
3. Les figures	21
X - Les liens hypertextes	23

1. vers un site	23
2. vers une page locale	23
3. vers une ancre	24
4. pour envoyer un e-mail	24
5. pour télécharger un fichier	25
Conclusion	26

Objectifs

- Connaître le fonctionnement des sites web
- Utiliser du code HTML
- Structurer une page web en HTML
- Insérer des images dans une page web
- Créer des liens hypertextes

Comment fonctionnent les sites web ?

I

- Les langages **HTML** et **CSS** sont à la base du fonctionnement de tous les sites web.
- L'ordinateur se base sur ce qu'on a écrit en HTML et CSS pour savoir ce qu'il doit afficher et comment.

Langages
HTML et CSS



Traduction par
l'ordinateur



Résultat visible
à l'écran



- **HTML5** sert à structurer votre contenu. Il est le fruit des travaux du W3C (World Wide Web Consortium) et du WHATWG (Web Hypertext Application Technology Working Group).
- **CSS3** (Cascading Style Sheets, aussi appelées Feuilles de style) sert à gérer l'apparence de la page web (agencement, positionnement, décoration, couleurs, taille du texte...).

Validation et compatibilité

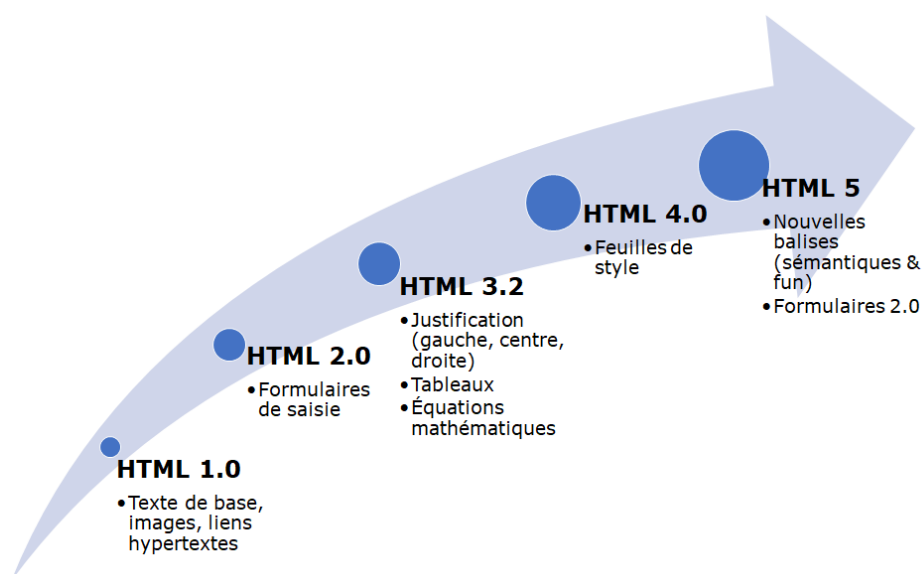


- **Différents navigateurs** peuvent produire **différents résultats** à partir d'un **même code**.
- Pourquoi ? Différents navigateurs et différentes versions de chaque navigateur coexistent + navigateurs mobiles !
- D'où la nécessité de **tester le code sous différents navigateurs** !
- Le W3C met deux outils à notre disposition
 1. **Valideur HTML** : <http://validator.w3.org>
 2. **Valideur CSS** : <http://jigsaw.w3.org/css-validator>
- **Vous devez vous efforcer d'avoir toujours un code valide.**

HTML : Evolution

III

- **HTML 1** : c'est la toute première version créée par Tim Berners-Lee en 1991.
- **HTML 2** : la deuxième version du HTML apparaît en 1994 et prend fin en 1996 avec l'apparition du HTML 3.0. C'est cette version qui posera en fait les bases des versions suivantes du HTML. Les règles et le fonctionnement de cette version sont donnés par le W3C (tandis que la première version avait été créée par un seul homme).
- **HTML 3** : apparue en 1996, cette nouvelle version du HTML rajoute de nombreuses possibilités au langage comme les tableaux, les applets, les scripts, le positionnement du texte autour des images, etc.
- **HTML 4** : cette version aura été utilisée un long moment durant les années 2000. Elle apparaît pour la première fois en 1998 et propose l'utilisation de frames (qui découpent une page web en plusieurs parties), des tableaux plus complexes, des améliorations sur les formulaires, etc. Mais surtout, cette version permet pour la première fois d'exploiter des feuilles de style, notre fameux CSS !
- **HTML 5** : c'est la dernière version. De plus en plus répandue, elle fait beaucoup parler d'elle car elle apporte de nombreuses améliorations comme la possibilité d'inclure facilement des vidéos, un meilleur agencement du contenu, de nouvelles fonctionnalités pour les formulaires, etc. C'est cette version que nous allons découvrir ensemble.



HTML : le langage de balisage

IV

- HTML (Hyper Text Markup Language) utilise des **balises** (tags) pour **indiquer la façon dont le document doit être affiché** et **spécifier la structure (organisation) d'un document**
- Une Balise: **<nom balise> Contenu </nom balise>**
- HTML est un langage interprété par le navigateur pour la description et le formatage de la page
- L'extension (suffixe) d'un document HTML est : **.htm** ou **.html**

1. La balise de formatage en HTML s'écrit sous cette forme : **<balise attributs> Contenu </balise>**

- **balise** (tag en anglais): un mot clé du langage associant une propriété particulière au contenu de la balise
Exemple : mettre en gras le contenu après cette balise
- **attributs** (ils ne sont pas toujours obligatoires) : Ils définissent les valeurs spécifiques à appliquer sur le contenu.
Exemple : taille 12 associée à la balise "police de caractère"
- **Contenu** : il représente le contenu à mettre en forme (texte, image, ... etc.) ou d'autres balises
- **</balise>** : c'est une étiquette précisant la fin de la balise
Exemple : arrêter de mettre en gras après cette balise

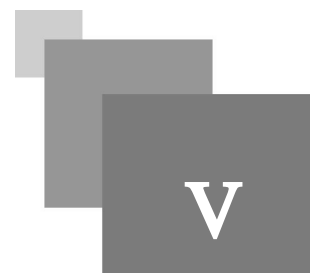
2. Les attributs d'une Balise :

- définissent les **valeurs** spécifiques à appliquer sur le contenu
- se trouvent toujours dans la **balise d'ouverture** et ils ne sont pas toujours obligatoires.
- sont séparés par des **espaces**, chaque attribut a une valeur : **attribut= valeur** et l'ordre des attributs n'a aucune importance

On distingue deux types de balises :

1. **Les balises en paires** : Elles s'ouvrent, contiennent du texte, et se ferment plus loin. Voici à quoi elles ressemblent : **<title>Ceci est un titre</title>**
2. **Les balises orphelines** : Ce sont des balises qui servent le plus souvent à insérer un élément à un endroit précis (par exemple une image). Une balise orpheline s'écrit comme ceci : **** . La balise **** Seule, elle ne sert pas à grand chose. On pourrait rajouter un attribut qui indique le nom de l'image à afficher : ****

L'éditeur de texte



- Il existe effectivement de nombreux logiciels dédiés à la création de sites web, parmi ces logiciels nous avons les **éditeurs de texte** tels que le bloc note, le logiciel d'édition de texte intégré par défaut à Windows.
- Il y a cependant des logiciels de création de site web plus puissants aujourd'hui et personne n'utilise vraiment Bloc-Notes.
- On peut classer ces logiciels de création de site web en deux catégories :

1. Les **WYSIWYG**

- (What You See Is What You Get - Ce Que Vous Voyez Est Ce Que Vous Obtenez) : ce sont des programmes qui se veulent très faciles d'emploi, ils permettent de créer des sites web sans apprendre de langage particulier.
- Parmi les plus connus d'entre eux : Mozilla Composer, Microsoft Expression Web, Dreamweaver, etc.
- Leur principal défaut est la qualité souvent assez mauvaise du code HTML et CSS qui est automatiquement généré par ces outils.
- Un bon créateur de site web doit tôt ou tard connaître HTML et CSS, c'est pourquoi l'utilisation de ces outils n'est pas recommandé.

2. Les **éditeurs de texte**

- Ce sont des programmes dédiés à l'écriture de code.
- On peut en général les utiliser pour de multiples langages, pas seulement HTML et CSS.
- Ils se révèlent être de puissants alliés pour les créateurs de sites web !
- Parmi ces éditeurs nous avons *Sublime Text*, *Notepad++*, *Brackets*, etc;

Comment structuré une page web ?

VI

1. Structure d'une page Web

- Notez bien que les balises d'une page HTML s'ouvrent et se ferment dans un ordre précis.
- Par exemple, la balise<html>est la première que l'on ouvre et c'est aussi la dernière que l'on ferme (tout à la fin du code, avec</html>).
- Les balises doivent être fermées dans le sens inverse de leur ouverture.
- Exemple :
 - <html><body></body></html>: **correct**
Une balise qui est ouverte à l'intérieur d'une autre doit aussi être fermée à l'intérieur.
 - <html><body></html></body>: **incorrect**
Les balises s'entremêlent.

Le code suivant présente un exemple de code HTML d'une page web

```
1 <html>
2   <head>
3     <meta name="author" content="Aimad QAZDAR">
4     <title>Titre de ma page </title>
5   </head>
6   <body>
7     <!-- Ceci est un commentaire -->
8     Contenu de ma page
9   </body>
10 </html>
```



Résultat de l'interprétation du code HTML par le navigateur

Pour tester ce code vous devez :

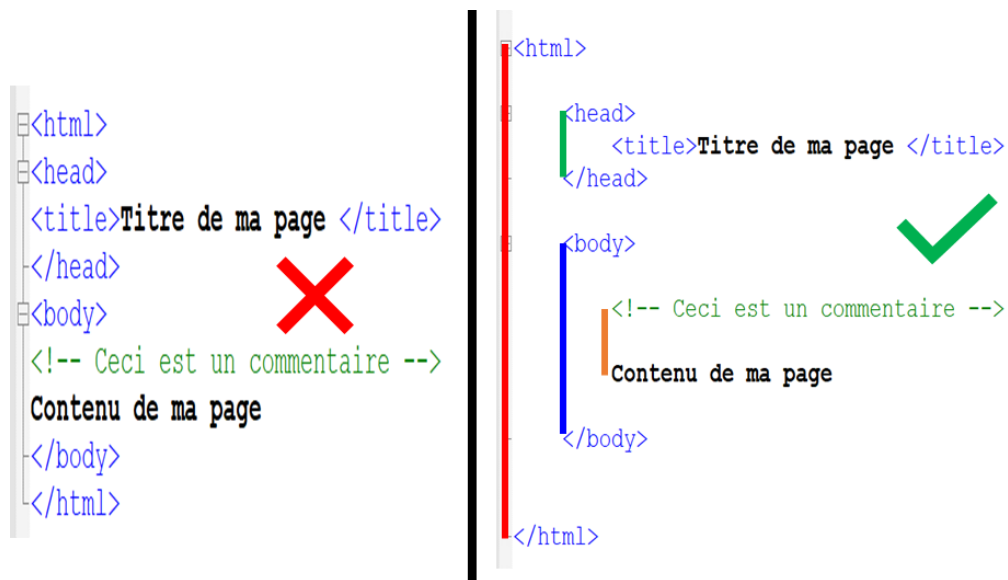
1. Écrire les mêmes phrases de code dans votre éditeur de texte
2. Enregistrez ce fichier. Comme dans tous les programmes, vous avez un menu Fichier>Enregistrer (ou File >Save en anglais) Une boîte de dialogue vous demande où enregistrer le fichier et sous quel nom. Enregistrez-le où vous voulez. Donnez au fichier le nom que vous voulez, en terminant par l'extension . **html**, par exemple **test.html**.
3. Ouvrir maintenant l'explorateur de fichiers dans le dossier où vous avez enregistré votre page (l'Explorateur sous Windows ou le Finder sous Mac). Vous y verrez le fichier que vous venez de créer.
4. L'icône qui représente le fichier dépend de votre navigateur web par défaut en général. Vous pouvez y voir une icône de Chrome , de Firefox, etc. Faites un double-clic sur ce fichier et votre navigateur s'ouvre et affiche le texte que vous avez écrit

Ci-dessous l'explication des lignes du code HTML

- **<HTML> ... </HTML>** : Marqueurs de début et de fin du document HTML
- **<HEAD> ... </HEAD>** : Ensemble des méta informations relatives au document
- **<TITLE> Mon titre </TITLE>** : Titre du document qui apparaît sur la barre de titre du navigateur
- **<META>** : Paramètres utiles notamment pour l'indexation du document par les moteurs de recherche
- **<BODY> ... </BODY>** : Corps du document
- **<!-- Mon commentaire -->** : pour insérer des commentaires dans le document

2. L'indentation du code

- En informatique, l'indentation consiste en l'ajout de **tabulations** ou d'**espaces** dans un fichier texte.
- Le **style d'indentation** décrit les différentes manières que les programmeurs utilisent pour faire **ressorti un bloc de code**.
- L'indentation se définit par la manière d'**arranger les blocs de code**, mais surtout par le nombre d'espace utilisés à chaque niveau.
- **L'indentation peut ne pas être qu'esthétique**. En effet, elle s'avère obligatoire pour certains langages afin de former des blocs ou des lignes valides (par exemple Occam, Fortran ou Python).



Comparaison entre deux versions de même code, la version à gauche n'est pas indentée et la version de droite est indentée

Assurez- vous toujours de l'indentation de votre code

3. L'en-tête <HEAD>

- Il contient des **informations relatives au document**, notamment sur les scripts, les feuilles de style, les méta-balises, des scripts, ... etc.
- Ces informations ne sont pas affichées par les navigateurs.
- Quelques balises qui peuvent y être intégrées :
 - **<SCRIPT>** : Contient le code du programme qui sera appelé dans une balise de <BODY>
 - **<STYLE>** : Définit les styles de présentation associée à des balises (règles et feuilles de styles)
 - **<TITLE>** : C'est le titre du document affiché dans la barre de titre du navigateur ainsi que dans les signets et les favoris. Il est nécessaire au référencement par les robots d'indexation.

```

1 <!DOCTYPE html>
2 <html>
3   <head>
4     <!-- En-tête de la page -->
5     <meta charset="utf-8" />
6     <title>Titre</title>
7   </head>
8
9   <body>
10    <!-- Corps de la page -->
11  </body>
12 </html>

```

4. Le corps<BODY>

- Comme nous avons vu dans l'exemple *de la structure d'une page HTML (cf. p.10)*, le contenu du site web est écrit entre les balises<body></body>
- La balise <body> accepte un ensemble d'attribut tels que :
 - **bgcolor** : la couleur d'arrière-plan pour le document.
 - **background** : l'image à utiliser en tant qu'arrière-plan.
 - **text** : la couleur à utiliser pour la police du texte. Cette méthode n'est pas conforme, il faut utiliser la propriété CSS color sur l'élément à la place.
- Cette page contient la liste de tous les attributs de la balise body ainsi leurs compatibilités avec les navigateurs web

```

1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>Exemple des attributs de la balise Body</title>
5   </head>
6   <body
7     bgcolor="#0000CC"
8     text="#FF0000"
9   >
10    Je suis un texte avec la couleur rouge dont le code est #FF0000 <br>
11    et je suis dans une page HTML dont la couleur de l'arriere plan est Bleu ;)
12  </body>
13 </html>

```

L'image ci-dessous représente le résultat du code HTML



Notez bien que :

- HTML utilise des **valeurs hexadécimales** pour **exprimer les couleurs fondamentales** que sont le Rouge, le Vert et le Bleu (RVB)
- La balise **
** dans la ligne 9 du code permet d'insérer un **retour à la ligne** à l'intérieure du texte.
- Le fait de taper sur la touche ENTRE du clavier permet de faire un retour à la ligne dans l'éditeur de texte mais ne permet pas de l'insérer réellement dans la page web.

Codes des Couleurs

VII

- Comme nous avons déjà vu, HTML utilise des valeurs hexadécimales pour exprimer les couleurs fondamentales que sont le Rouge, le Vert et le Bleu (RVB)
- Le tableau suivant représente les codes des 16 couleurs de base

Nom	Code	Résultat
Black	#000000	
Maroon	#800000	
Green	#008000	
Olive	#808000	
Navy	#000080	
Purple	#800080	
Teal	#008080	
Gray	#808080	

Nom	Code	Résultat
White	#FFFFFF	
Red	#FF0000	
Lime	#00FF00	
Yellow	#FFFF00	
Blue	#0000FF	
Fuchsia	#FF00FF	
Aqua	#00FFFF	
Silver	#C0C0C0	

Mise en forme en HTML

VIII

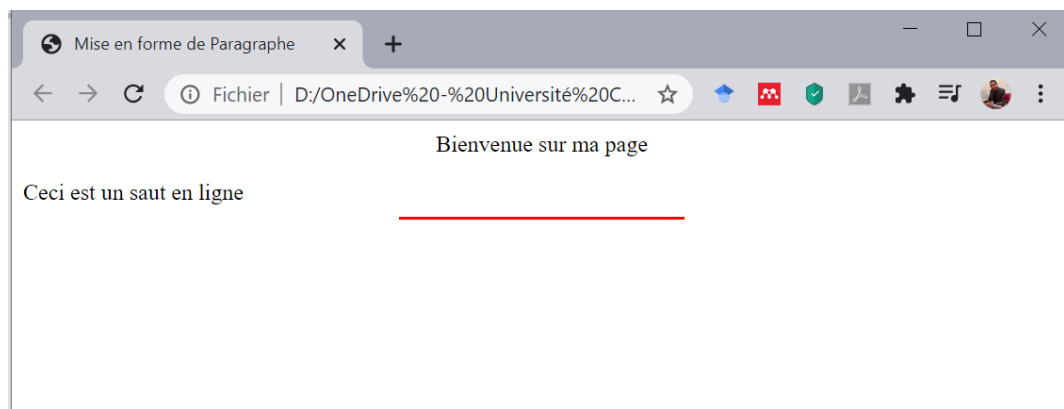
1. Paragraphes

- La plupart du temps, lorsqu'on écrit du texte dans une page web, on le fait à l'intérieur de paragraphes. Le langage HTML propose justement la balise **<p>** pour délimiter les paragraphes.

```
1 <p>Bienvenue sur ma page</p>
```

- **<p>** signifie « Début du paragraphe » ; **</p>** signifie « Fin du paragraphe ».

```
1 <html>
2   <head>
3     <title>Mise en forme de Paragraphe</title>
4   </head>
5   <body >
6     <p align="center"> Bienvenue sur ma page</p>
7     Ceci est un saut en ligne <br>
8     <hr width="200" color="#FF0000" align="center" size="2">
9   </body>
10 </html>
11
```



Remarque : la balise **<hr>** permet d'insérer une ligne horizontale

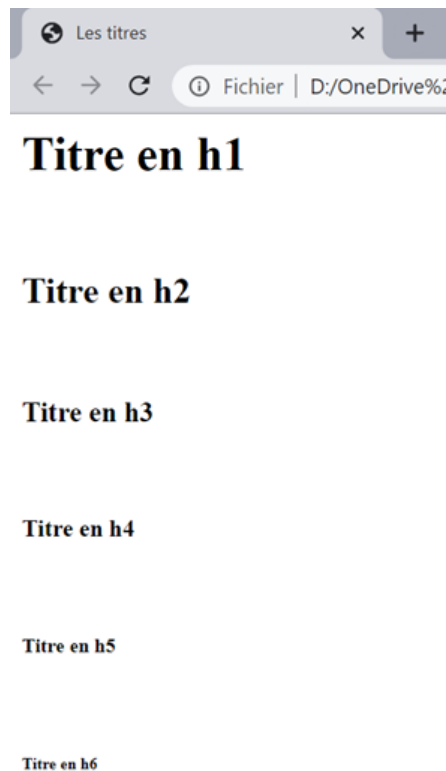
2. Titres

En HTML, il est possible d'utiliser six niveaux de titres différents :

1. **<h1> </h1>** : signifie « titre très important ». En général, on s'en sert pour afficher le titre de la page au début de celle-ci.
2. **<h2> </h2>** : signifie « titre important ».

3. **<h3> </h3>** : pareil, c'est un titre un peu moins important (on peut dire un « sous-titre » si vous voulez).
4. **<h4> </h4>** : titre encore moins important.
5. **<h5> </h5>** : titre pas important.
6. **<h6> </h6>** : titre vraiment, mais alors là vraiment pas important du tout

```
1 <html>
2   <head>
3     <title>Les titres</title>
4   </head>
5   <body >
6     <h1> Titre en h1 </h1>
7     <h2> Titre en h2 </h2>
8     <h3> Titre en h3 </h3>
9     <h4> Titre en h4 </h4>
10    <h5> Titre en h5 </h5>
11    <h6> Titre en h6 </h6>
12  </body>
13 </html>
```



3. Texte

- La balise **** joue un rôle très important pour rendre le texte du site Web plus **convivial** et **augmenter** la lisibilité du contenu.
- Cette balise possède trois attributs appelés :
 - **size** : qui permet de préciser la taille du texte,
 - **color** : pour modifier la couleur

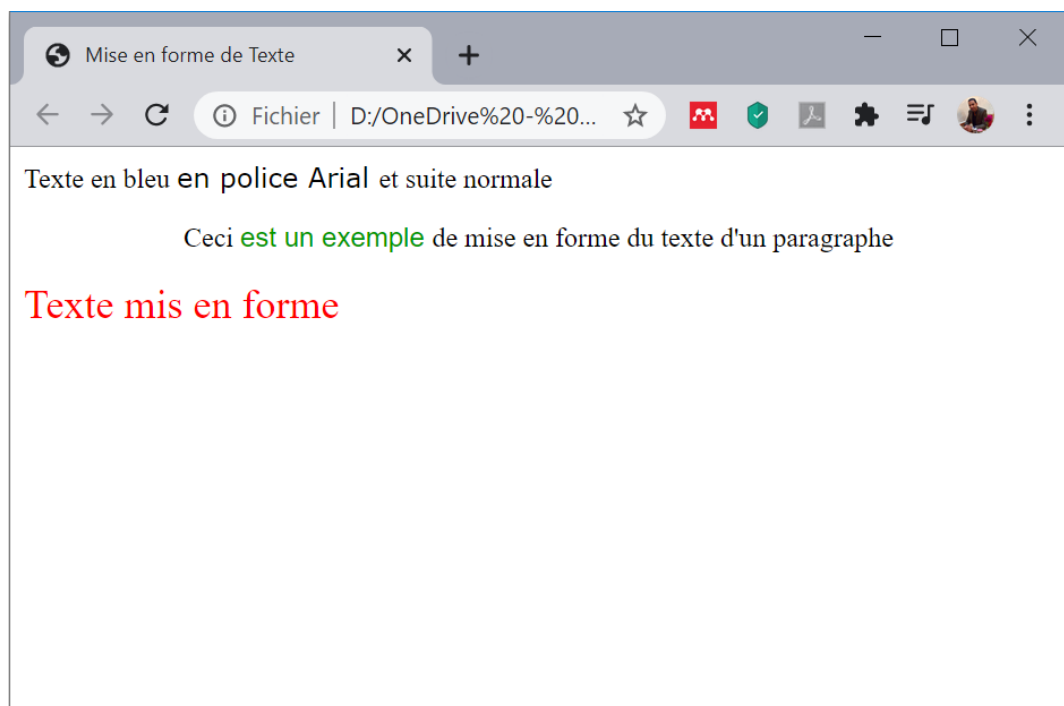
- **face** : pour personnaliser la police.

```

1 <html>
2   <head>
3     <title>Mise en forme de Texte</title>
4   </head>
5   <body>
6     texte ="#000099"
7   >
8     Texte en bleu <font face="Verdana, Arial, Helvetica, sans-serif"> en police
    Arial </font> et suite normale<br>
9     <p align="center"> Ceci <font color="#009900" face="Geneva, Arial, Helvetica,
    sans-serif"> est un exemple </font> de mise en forme du texte d'un paragraphe </p>
10    <font size="+2" color="#FF000000" face="Times New Roman, Times, serif"> Texte
    mis en forme </font> <br>
11  </body>
12 </html>

```

Le résultat du code est représenté dans l'image ci-dessous :



4. Style

Le code suivante représente l'utilisation de quelque balise de **style de texte** tels que **** , **<I>**, **<U>** , etc.

```

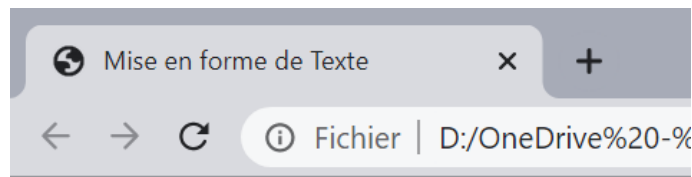
1 <html>
2   <head>
3     <title>Mise en forme de style</title>
4   </head>
5   <body texte ="#000099">
6     <B>texte gras</B> <br>
7     <I>texte italique</I><br>
8     <U>texte souligné</U> <br>
9     <S>texte barré</S> <br>
10    texte en<SUB>indice</SUB><br>
11    texte en<SUP>exposant</SUP><br>

```

```

12 </body>
13 </html>
14
15
16
17

```



texte gras
texte italique
texte souligné
~~texte barré~~
 texte en_{indice}
 texte en^{exposant}

5. Listes

Les listes permettent d'ordonner nos informations. Il existe deux types de listes :

1. **les listes non ordonnées** ou **listes à puces** ;
2. **les listes ordonnées** ou **listes numérotées** ou encore **énumérations**

- Liste non ordonnée

Une liste non ordonnée ressemble à ceci :

- HTML
- CSS
- PHP

Pour créer une liste non ordonnée, il suffit d'utiliser deux balises :

- `` délimite toute la liste ;
- `` délimite un élément de la liste (une puce).

- Liste ordonnée

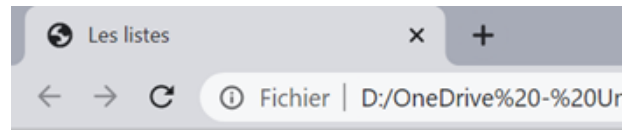
Une liste ordonnée fonctionne de la même façon, seule une balise change : il faut remplacer `` par ``.

```

1 <h2> Exemple de liste non ordonnée </h2>
2   <ul>
3     <li>HTML</li>
4     <li>CSS</li>
5     <li>PHP</li>
6   </ul>
7
8 <h2> Exemple de liste ordonnée </h2>
9   <ol>

```

```
10     <li>HTML</li>
11     <li>CSS</li>
12     <li>PHP</li>
13 </ol>
```



Exemple de liste non ordonnée

- HTML
- CSS
- PHP

Exemple de liste ordonnée

1. HTML
2. CSS
3. PHP

Les images en HTML

IX

1. Les formats d'images

- Certains formats sont plus adaptés que d'autres selon l'image (photo, dessin, image animée...).
- Toutes les images diffusées sur Internet ont un point commun : elles sont compressées. Cela veut dire que l'ordinateur fait des calculs pour qu'elles soient moins lourdes et donc plus rapides à charger.
- Voici quel format à adopter en fonction de l'image :
 - Une photo : utilisez un JPEG.
 - N'importe quel graphique avec peu de couleurs (moins de 256) : utilisez un PNG 8 bits ou éventuellement un GIF.
 - N'importe quel graphique avec beaucoup de couleurs : utilisez un PNG 24 bits.
 - Une image animée : utilisez un GIF animé.

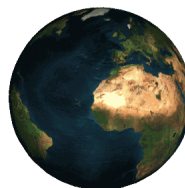
Une photo en JPEG



Une image en PNG



Un GIF animé



2. Insertion d'images

- La balise **** permet l'insertion d'une image dans une page web.
- Cette balise doit être accompagnée d'au moins deux attributs obligatoires:
 1. **src** : le chemin de l'image à insérer. Vous pouvez soit mettre un chemin absolu ou relatif (ce qu'on fait le plus souvent). (**Obligatoire**)
 2. **alt** : cela signifie « texte alternatif ». Ce texte sera affiché à la place de l'image si celle-ci ne peut pas être téléchargée. (**Obligatoire**)
 3. **width** : la largeur de l'image
 4. **height** : la longueur de l'image
 5. **border** : pour définir une bordure (cadre) de l'image
 6. etc.

```

1 <html>
2   <head>
3     <title>Balise d'insertion d'images</title>
4   </head>
5   <body texte = "#000099">
6     Scolarisation et education des enfants
7     <img src = "ecole_coranique2.jpg" width = "259" height = "194" align = "middle" alt =
8       "Ecole coranique" border = "3">
9       au sein de l'école coranique.
10  </body>
11 </html>

```

Scolarisation et education des enfants



au sein de l'école coranique.

Exemple d'école coranique qui ressemblait beaucoup à celle d'Ichach

3. Les figures

- Ce sont des éléments qui viennent enrichir le texte pour compléter les informations de la page. Les figures ne sont pas forcément des images : un code source aussi illustre le texte.
- Les figures peuvent être de différents types
 - images ;
 - citations ;
 - etc.
- En HTML5, on dispose de la balise **<figure>**. Voici comment on pourrait l'utiliser :

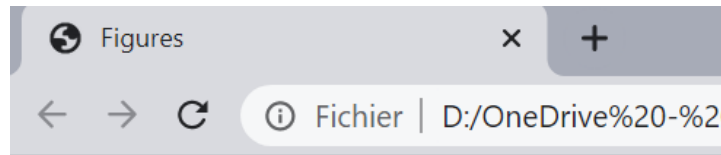
```

1 <figure>
2   <img src = "images/firefox.png" alt = "Logo Mozilla Firefox" />
3 </figure>

```

- Une figure est le plus souvent accompagnée d'une légende.
- Pour ajouter une légende, utilisez la balise **<figcaption>** à l'intérieur de la balise **<figure>**, comme ceci :

```
1 <figure>
2   
3   
4   
6   <figcaption>Logos des différents navigateurs</figcaption>
7 </figure>
```



Logos des différents navigateurs

Les liens hypertextes

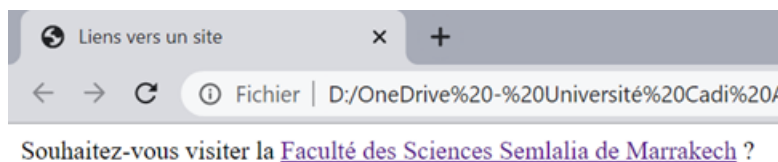
X

1. vers un site

- Il est facile de reconnaître les liens sur une page : ils sont mis en forme d'une façon différente (par défaut en bleu et soulignés) et un curseur en forme de main apparaît lorsqu'on pointe dessus.
- Pour faire un lien, il faut utiliser la balise `<a>` et lui ajouter un attribut, **href**, pour indiquer vers quelle page le lien doit conduire.

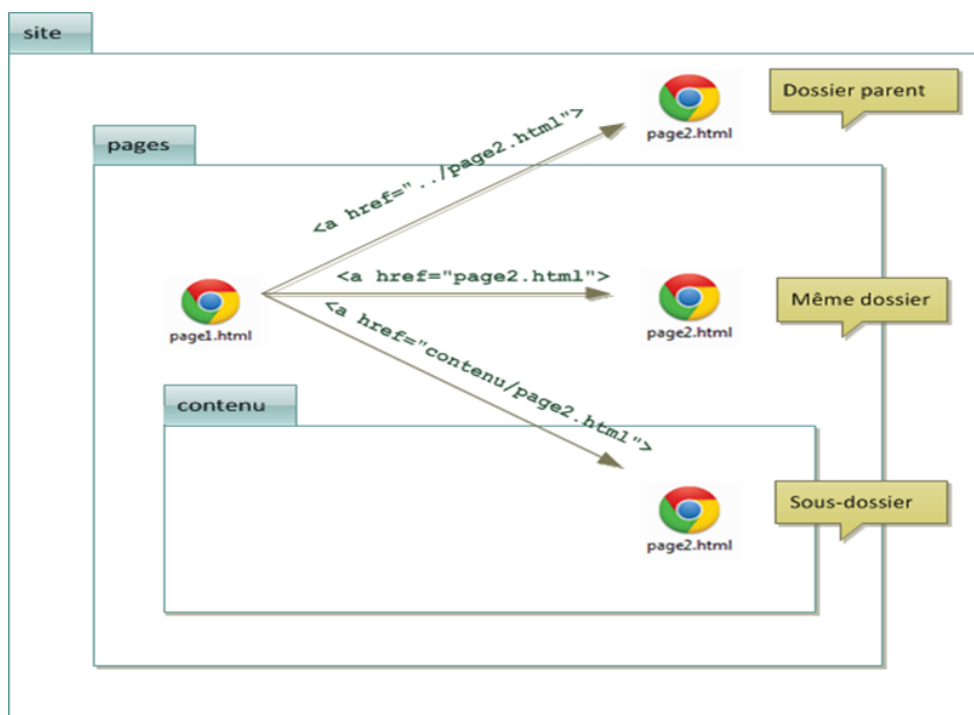
Exemple :

```
1 <p>Souhaitez-vous visiter le <a href="http://fssm.uca.ma/" title="FSSM">La
  Faculté des Sciences Semlalia de Marrakech</a> ?</p>
```



2. vers une page locale

La figure suivante représente les différents cas de liens vers une page local



1. Si les deux fichiers sont situés dans le même dossier, il suffit d'écrire comme cible du lien le nom du fichier vers lequel on veut amener. Par exemple : ``
2. Si les pages sont situées dans des dossiers différents, le lien doit être rédigé en indiquant le chemin relatif : ``
3. S'il y avait plusieurs sous-dossiers, on écrirait ceci : ``
4. Si votre fichier cible est placé dans un dossier qui se trouve « plus haut » dans l'arborescence, il faut écrire deux points comme ceci : ``

3. vers une ancre

- Une **ancre** est une sorte de point de repère que vous pouvez mettre dans vos pages HTML lorsqu'elles sont très longues. En effet, il peut alors être utile de faire un lien amenant plus bas dans la même page pour que le visiteur puisse sauter directement à la partie qui l'intéresse.
- Pour créer une ancre, il suffit de rajouter l'attribut **id** à une balise qui va alors servir de repère. Ce peut être n'importe quelle balise, un titre par exemple. Utilisez l'attribut **id** pour donner un nom à l'ancre. Cela nous servira ensuite pour faire un lien vers cette ancre.
- Par exemple : `<h2 id="mon_ancre">Titre</h2>`
- Ensuite, il suffit de créer un lien comme d'habitude, mais cette fois l'attribut **href** contiendra un dièse (#) suivi du nom de l'ancre.
- Exemple :

```
1 <a href="#mon_ancre">Aller vers l'ancre</a>
```

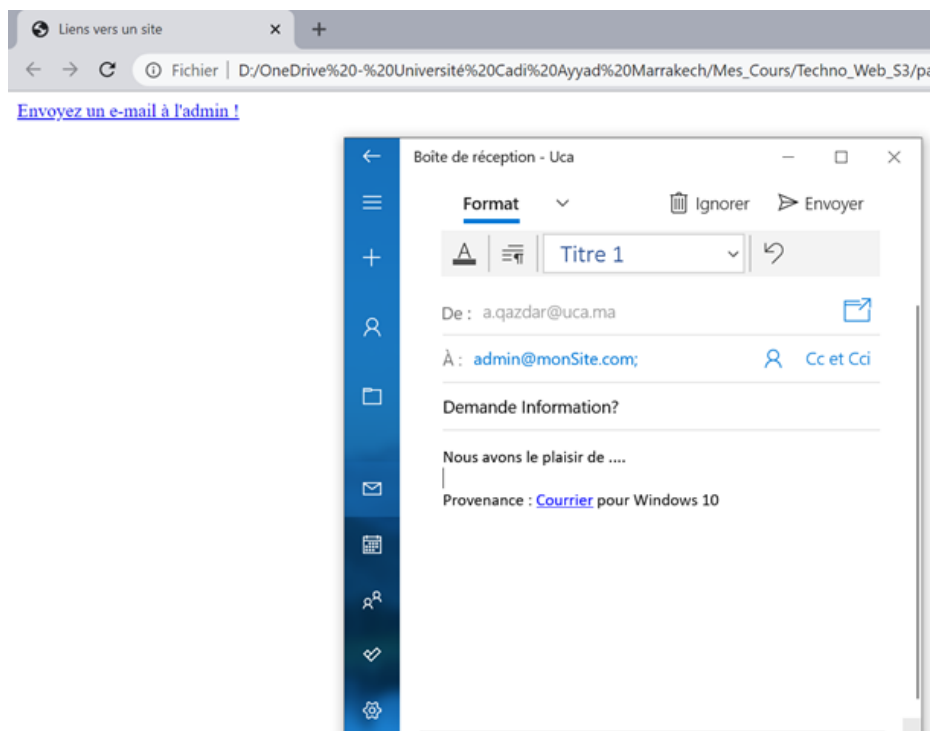
- Pour créer un lien qui ouvre une autre page ET qui amène directement à une ancre située plus bas sur cette page, il suffit de taper le nom de la page, suivi d'un dièse (#), suivi du nom de l'ancre.
- Par exemple :

```
1 <a href="page2.html#BasDePage">
```

4. pour envoyer un e-mail

Si vous voulez que vos visiteurs puissent vous envoyer un e-mail, vous pouvez utiliser des liens de type **mailto** en modifiant la valeur de l'attribut **href** comme ceci :

```
1 <p><a href="mailto:admin@monSite.com">Envoyez un e-mail à l'admin !</a></p>
```

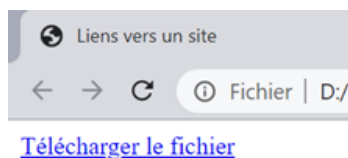



5. pour télécharger un fichier

- Pour insérer un lien de téléchargement vous devez procéder exactement comme si vous faisiez un lien vers une page web, mais en indiquant cette fois le nom du fichier à télécharger.
- Le navigateur, voyant qu'il ne s'agit pas d'une page web à afficher, va lancer la procédure de téléchargement lorsqu'on cliquera sur le lien.

Exemple :

```
<p><a href="monfichier.pdf">Télécharger le fichier</a></p>
```



Conclusion

- Dans ce chapitre nous avons vu les bases du fonctionnement des sites web et comment créer notre premier site web.
- On utilise l'éditeur de texte (Sublime Text, Notepad++, etc) pour créer un fichier ayant l'extension .html (par exemple : test.html). Ce sera notre page web.
- À l'intérieur du fichier, nous écrivons le contenu de notre page, accompagné de balises HTML.
- Les balises peuvent avoir plusieurs formes : `<balise> </balise>` ou `<balise />`
- Les balises sont parfois accompagnées d'attributs pour donner des indications supplémentaires (exemple : `<image nom="photo.jpg" />`).
- Une page web est constituée de deux sections principales : un en-tête (`<head>`) et un corps (`<body>`).
- Les paragraphes sont définis par la balise `<p> </p>` et les sauts de ligne par la balise `
`.
- Il existe six niveaux de titre, de `<h1> </h1>` à `<h6> </h6>`, à utiliser selon l'importance du titre.
- Pour créer des listes, on doit utiliser la balise `` (liste à puces, non ordonnée) ou `` (liste ordonnée). À l'intérieur, on insère les éléments avec une balise `` pour chaque item.
- On insère une image avec la balise ``. Elle doit obligatoirement comporter au moins ces deux attributs : `src` (nom de l'image) et `alt` (courte description de l'image).
- Pour insérer un lien, on utilise la balise `<a>` avec l'attribut `href` pour indiquer l'adresse de la page cible.
- On peut insérer un lien vers une autre page, un lien vers un endroit sur la même page ou sur une autre page, un lien pour télécharger un fichier et un lien pour l'envoi de l'email