

# **HTML & CSS**

## **Cours 1 :**

### **Les bases de HTML5**

*Nous allons voir les éléments fondamentaux pour créer une page Web*

## **1) Introduction**

Les langages HTML et CSS sont à la base du fonctionnement de tous les sites web.

### **1.1) HTML et CSS : deux langages pour créer un site web**

Il ne suffit pas simplement de taper le texte qu'il y aura dans son site, il faut aussi indiquer où placer ce texte, insérer des images, faire des liens entre les pages, etc... Pour expliquer au navigateur web ce que l'on veut faire, il va falloir utiliser des langages qu'il comprend.

Nous allons avoir affaire à deux langages qui se complètent car ils ont des rôles différents :

#### **1.1.1) Le rôle de HTML**

HTML (**H**yper**T**ext **M**arkup **L**anguage) 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, etc...**

### 1.1.2) Le rôle de CSS

CSS (Cascading Style Sheets, aussi appelées Feuilles de style en cascade) : Ce langage est venu compléter le HTML en 1996.

**Son rôle est de gérer l'apparence de la page web.**

**C'est donc en CSS que vous gérez le positionnement des éléments, la couleur, la taille du texte, etc...**

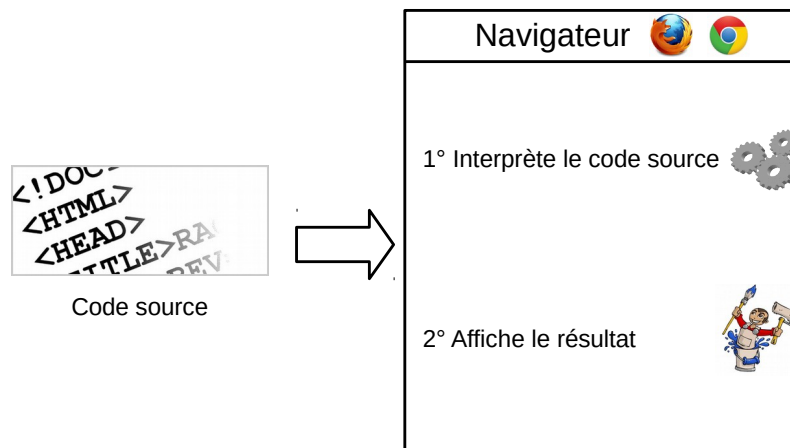
Il est tout à fait possible de créer un site web uniquement en HTML, mais celui-ci ne sera pas très beau : l'information apparaîtra « brute ». C'est pour cela que le langage CSS vient toujours le compléter.

**Le CSS a besoin d'une page HTML pour fonctionner.**

Notez que HTML5 et CSS3 ne sont pas des versions encore totalement finalisées par le W3C.

## 1.2) Les navigateurs web

### 1.2.1) Pourquoi le navigateur est important ?



**Le travail du navigateur web est d'interpréter le code HTML et CSS reçu pour afficher le résultat visuel à l'écran.**

**Si votre code CSS dit « Les titres sont en rouge », alors le navigateur affichera les titres en rouge.**

Un navigateur est un programme extrêmement complexe. Comprendre le code HTML et CSS n'est en effet pas une mince affaire.

## **Les navigateurs n'affichent pas tous les sites exactement de la même façon !**

Il faudra prendre l'habitude de vérifier régulièrement que votre site fonctionne correctement sur la plupart des navigateurs.

### **1.2.2) Les navigateurs web sur l'ordinateur**

Il existe plusieurs navigateurs web :

- Firefox : C'est le navigateur de la fondation Mozilla.
- Chrome : C'est le navigateur de l'entreprise Google.
- Internet Explorer : C'est le navigateur de l'entreprise Microsoft.
- Safari : C'est le navigateur de l'entreprise Apple.
- Opera : C'est le navigateur de l'entreprise Opera software.
- Et bien d'autres.

**Remarque :** Chaque version prend en charge de nouvelles fonctionnalités.

Des sites comme [www.caniuse.com](http://www.caniuse.com) tiennent notamment à jour une liste des fonctionnalités CSS supportées par les différentes versions de chaque navigateur.

### **1.2.3) Les navigateurs sur mobile**

Il existe des variantes de ces navigateurs que l'on retrouve sur les téléphones portables, en particulier les smartphones.

La plupart des navigateurs sur smartphones sont les mêmes que sur ordinateur, dans une version plus légère adaptée aux mobiles. Tout dépend du type de téléphone.

- **iPhone** : Safari Mobile.
- **Android** : Chrome Mobile Firefox Mobile.
- **Windows Phone** : Internet Explorer Mobile.
- **Blackberry** : les Blackberry font exception, car ils ont leur propre navigateur (il n'existe pas d'équivalent sur ordinateur).

Les navigateurs pour mobiles supportent la plupart des dernières fonctionnalités de HTML et CSS. De plus, le système de mise à jour automatisé des mobiles nous garantit que les utilisateurs auront le plus souvent les dernières versions.



Exemple : <titre>Mon titre</titre>

Cela signifie pour le navigateur que tout ce qui est entre ces deux balises est un titre.

**On distingue une balise ouvrante (<balise>) d'une balise fermante (</balise>) par la barre « slash » se trouvant dans la balise fermante juste après le symbole inférieur.**

Exemple :

Ceci est du texte normal <titre>Ceci est un titre</titre>

### ***b) Les balises orphelines***

Les balises orphelines servent le plus souvent à insérer un élément à un endroit précis (par exemple une image).

Il n'est pas nécessaire de délimiter le début et la fin de l'image, on veut juste dire à l'ordinateur « Insère une image ici ».

Une balise orpheline s'écrit comme ceci : <balise />

**!! Attention de ne pas se tromper sur l'emplacement du slash entre la partie fermante d'une balise par paire et la balise orpheline !!**

## **2.1.2) Les attributs**

Les attributs sont les options des balises. Ils viennent les compléter pour donner des informations supplémentaires.

**Dans le cas d'une balise par paire, on ne met les attributs que dans la balise ouvrante.**

Exemples :

<balise attribut="valeur">blablabla</balise>

<balise attribut="valeur" />

### **A quoi ça sert ?**

Prenons la balise <image />. Seule, elle ne sert pas à grand chose. On pourrait rajouter un attribut qui indique le nom de l'image à afficher :

<image nom="photo.jpg" />

Le navigateur comprend alors qu'il doit afficher l'image contenue dans le fichier photo.jpg.

Par exemple, ce code indique que la citation est de Neil Armstrong et qu'elle date du 21 Juillet 1969 :

```
<citation auteur="Neil Armstrong" date="21/07/1969">  
C'est un petit pas pour l'homme un bond de géant pour l'humanité.  
</citation>
```

**Toutes les balises que nous venons de voir sont fictives.**

## 2.2) Structure de base d'une page HTML5

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

  <body>

  </body>
</html>
```

Remarquez le décalage de certaines lignes ça permet de rendre le code source plus lisible.

On appelle ça l'indentation.

!! Lors des séances de TP ils vous sera obligatoirement demandé d'indenter votre code !!

### 2.2.1) Emboîtement des balises

Parfois, il existe une hiérarchie qui oblige certaines balises à contenir obligatoirement une balise d'un type donné.

**La balise <html> doit contenir les balises <head> et <body> et uniquement celles-ci.**

Lorsqu'une balise contient d'autres balises, il y a des règles à respecter. Une balise peut contenir aucune, une ou plusieurs autres balises.

**On appelle donc le contenant : l'élément parent et le contenu : l'élément enfant.**

Un élément peut avoir plusieurs enfants, mais ne peut avoir qu'un parent direct.

Une balise ouverte à l'intérieur d'une autre doit obligatoirement être refermée avant la fermeture de sa balise parent.

Autrement dit, les balises sont emboîtées les unes dans les autres en respectant la syntaxe suivante :

`<parent> <enfant> </enfant> </parent>`

## 2.2.2) Les balises de bases (DOCTYPE, html, head, body)

### a) Le doctype

```
<!DOCTYPE html>
```

La toute première ligne contient la balise `<!DOCTYPE html>`, elle est indispensable car c'est elle qui indique qu'il s'agit bien d'une page web HTML.

Ce n'est pas vraiment une balise comme les autres (elle commence par un point d'exclamation).

### b) La balise `<html>`

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

**Elle englobe tout le contenu de votre page.**

### c) L'en-tête `<head>` et le corps `<body>`

Une page web est constituée de 2 parties :

L'en-tête `<head>` : cette section donne quelques informations générales sur la page, comme son titre, l'encodage, etc...

Remarque : Les informations que l'en-tête contient ne sont pas affichées sur la page, ce sont des informations générales à destination du navigateur, et des moteurs de recherches.

Le corps `<body>` : c'est là que se trouve la partie principale de la page. Tout ce que nous écrirons ici sera affiché à l'écran.

## 2.2.3) L'encodage (charset)

```
<meta charset="utf-8" />
```

**L'encodage est la façon dont les caractères sont représentés.**

Cette balise indique l'encodage que le navigateur web doit utiliser pour afficher les caractères contenus dans le fichier.

Il existe d'autres techniques d'encodage utilisées en fonction des langues (ISO-8859-1,...). Un seul cependant devrait être utilisé aujourd'hui autant que possible : UTF-8. Cette méthode d'encodage permet d'afficher sans aucun problème pratiquement tous les symboles de toutes les langues de notre planète.

Il faut que le fichier du code source soit enregistré dans le même encodage que celui indiqué dans la balise `<meta charset="..." />`.



## 2.2.4) Le titre principal de la page

```
<title>Le titre de la page</title>
```

**Toute page doit avoir un titre qui décrit ce qu'elle contient.**

Le titre ne s'affiche pas dans votre page mais en haut de celle-ci (souvent dans l'onglet du navigateur).

**Remarque :** Il faut savoir que le titre apparaît aussi dans les résultats des moteurs de recherche. Autant dire que bien choisir son titre est important ! Il est conseillé que le titre soit assez court (moins de 100 caractères en général).

## 2.3) Les commentaires

**Un commentaire est un texte qui sert simplement de mémo. En HTML il n'est pas affiché par le navigateur lors du rendu de la page.**

Ça sert pour vous et les personnes qui liront le code source de votre page. Vous pouvez utiliser les commentaires pour laisser des indications sur le fonctionnement de votre page. Cela vous permettra de vous souvenir comment fonctionne votre page si vous revenez sur votre code source après un long moment d'absence.

**!! Les commentaires sont indispensables !!**

Un commentaire est une balise HTML avec une forme bien spéciale :

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

Vous pouvez le mettre où vous voulez au sein de votre code source : il n'a aucun impact sur le rendu de votre page.

```
<!DOCTYPE html>
<html>
<!-- En-tête de la page -->
<head>
  <!-- L'encodage de la page -->
  <meta charset="utf-8" />
  <title>Titre</title> <!-- Le titre de la page -->
</head>
<!-- Corps de la page -->
<body>

</body>
</html>
```

## **2.4) Le code des pages HTML est visible par tout le monde**

Une remarque importante : **tout le monde peut voir le code HTML de votre page** une fois celle-ci mise en ligne sur le Web. Il suffit de faire un clic droit sur la page et de sélectionner « Code source de la page » (l'intitulé peut changer selon le navigateur).

Vous pouvez tester sur n'importe quel site web, ça marche ! Cela s'explique assez facilement par le fait que le navigateur doit obtenir le code HTML pour savoir ce qu'il faut afficher. Le code HTML de tous les sites est donc visible.

La morale de l'histoire ? Tout le monde pourra voir votre code HTML et vous ne pouvez pas l'empêcher. Par conséquent, ne mettez pas d'informations sensibles comme des mots de passe dans les commentaires.

## **2.5) Remarque**

Dans la suite du cours je ne ferais apparaître que les portions de code utiles.

On écrit le contenu du site web entre les balises `<body></body>` qui elles même se trouvent à l'intérieur des balises `<html> </html>`.

Dans les cas où le code doit s'écrire autre part que dans le « body », je vous le signalerai.

### 3) Organisation du texte

#### 3.1) *Les paragraphes*

Le langage HTML propose la balise `<p>` pour délimiter les paragraphes.

**Exemple :**

```
<p>Bonjour et bienvenue sur mon site !</p>
```

`<p>` signifie « Début du paragraphe »

`</p>` signifie « Fin du paragraphe »

#### 3.2) *Sauter une ligne*

Dans le code source HTML si vous appuyez sur la touche « Entrée » cela ne créera pas un saut de ligne dans le rendu du navigateur.

**Si vous voulez écrire un autre paragraphe, il vous suffit d'utiliser une nouvelle balise `<p>`.**

Pour aller à la ligne dans un paragraphe il existe une balise orpheline : `<br />`.

Vous devez obligatoirement la mettre à l'intérieur d'un paragraphe.

**Exemple :**

```
<body>
  <p>
    Bonjour et bienvenue sur mon site !<br />
    Ceci est mon premier test, alors soyez indulgents s'il vous plaît,
    j'apprends petit à petit comment ça marche.
  </p>
  <p>
    Pour l'instant c'est un peu vide, mais revenez dans quelques jours quand
    j'aurai appris un peu plus de choses.
  </p>
</body>
```

**Remarque :** Vous pouvez théoriquement mettre plusieurs balises `<br />` d'affilée pour faire plusieurs sauts de lignes, mais on considère que c'est une mauvaise pratique qui rend le code délicat à maintenir. Pour décaler un texte avec plus de précision, on utilisera le CSS, ce langage qui vient compléter le HTML.

### 3.3) Les titres

Lorsque le contenu de votre page va s'étoffer avec de nombreux paragraphes, cela va devenir difficile pour vos visiteurs de se repérer. C'est là que les titres deviennent utiles.

En HTML il existe 6 niveaux de titres différents, donc 6 balises :

- `<h1> </h1>` : Signifie « titre très important ». En général, on s'en sert pour afficher le titre de la page au début de celle-ci.
- `<h2> </h2>` : Signifie « titre important ».
- `<h3> </h3>` : Pareil, c'est un titre un peu moins important (on peut dire un « sous-titre »).
- `<h4> </h4>` : Titre encore moins important.
- `<h5> </h5>` : Titre pas important.
- `<h6> </h6>` : Titre vraiment, mais alors là vraiment pas important du tout.

**Attention : ne confondez pas avec la balise `<title>` !** La balise `<title>` affiche le titre de la page dans la barre de titre du navigateur. Les titres `<hX>` servent à créer des titres qui seront affichés dans la page web.

Dans la pratique, on n'utilise que les balises `<h1>`, `<h2>` et `<h3>`, et très rarement les autres. Votre navigateur affiche le titre très important en très gros, le titre un peu moins important en un peu moins gros, etc.

**Ne choisissez pas la balise de titre en fonction de la taille qu'elle procure au texte !**

**Il faut impérativement bien structurer sa page en commençant par un titre de niveau 1 (`<h1>`), puis un titre de niveau 2 (`<h2>`), etc.**

Si vous voulez modifier la taille du texte, cela se fera en CSS.

**Exemple :**

```
<body>
  <h1>Titre très important</h1>
  <h2>Titre important</h2>
  <h3>Titre un peu moins important (sous-titre)</h3>
  <h4>Titre pas trop important</h4>
</body>
```

### 3.4) La mise en valeur

Au sein de vos paragraphes, certains mots sont parfois plus importants que d'autres et vous aimeriez les faire ressortir. HTML vous propose différents moyens de mettre en valeur le texte de votre page.

#### 3.4.1) Mettre bien en valeur

**Pour mettre un texte bien en valeur, on utilise la balise `<strong>...</strong>` qui signifie « important ».**

**Exemple :**

```
<body>
  <p>
    Bonjour et bienvenue sur mon site !<br />

    Ceci est mon premier test, alors <strong>soyez indulgents</strong> s'il
vous plaît, j'apprends petit à petit comment ça marche.
  </p>
</body>
```

**Remarque :** Par défaut le texte s'affiche en gras, c'est le navigateur qui choisit comment afficher les mots. On lui dit que les mots sont assez importants et pour faire ressortir cette information, il change l'apparence du texte en choisissant d'afficher les mots importants en gras.

#### 3.4.2) Mettre un peu en valeur

**Pour mettre un peu en valeur du texte, vous devez utiliser la balise `<em>...</em>`.**

**Exemple :**

```
<body>
  <p>
    Bonjour et bienvenue sur mon site !<br />

    Ceci est mon premier test, alors <em>soyez indulgents</em> s'il vous plaît,
j'apprends petit à petit comment ça marche.
  </p>
</body>
```

**Remarque :** Par défaut la balise `<em>` a pour conséquence de mettre le texte en italique. Là encore, l'italique n'est qu'une conséquence. Le navigateur a choisi d'afficher en italique les mots assez importants pour faire ressortir cette information.

### 3.4.3) HTML pour le fond, CSS pour la forme (Résumé)

Il est très important de faire un bilan.

Les débutants font souvent la même grosse erreur à ce stade. Ils ont vu les balises `<em>`, `<strong>`... et ils se disent : « Je sais comment mettre en italique, en gras du texte en HTML ! ».

Et pourtant... ce n'est pas à ça que servent ces balises ! Vous allez dire « Mais quand j'utilise `<strong>` le texte apparaît en gras, donc c'est pour mettre en gras », et pourtant, c'est une erreur de croire que cette balise sert à ça.

**Le rôle des balises est d'indiquer le sens du texte.**

**Ainsi, `<strong>` indique au navigateur « Ce texte est important ». C'est tout.**

Et pour montrer que le texte est important, le navigateur décide de le mettre en gras (mais il pourrait aussi bien l'écrire en rouge !). La plupart des navigateurs affichent les textes importants en gras, mais rien ne les y oblige.

À quoi ça sert que l'ordinateur sache qu'un texte est important ?

De nombreux programmes analysent les codes source des pages web, à commencer par les robots de moteurs de recherche. Ces robots parcourent le web en lisant le code HTML de tous les sites. C'est le cas des robots de Google et de Bing par exemple. Les mots-clés « importants » ont tendance à avoir plus de valeur à leurs yeux, donc si quelqu'un fait une recherche sur ces mots il a plus de chances de tomber sur votre site.

Bien entendu c'est une explication grossière, et il ne faut pas croire qu'utiliser la balise `<strong>` à tout-va améliorera votre référencement.

Mais alors, comment on fait pour mettre spécifiquement en gras, pour écrire en rouge, etc... ?

Tout cela se fait en CSS.

**Le HTML définit le fond (contenu, logique des éléments).**

**Le CSS définit la forme (apparence)**

Nous verrons le CSS plus loin, pour l'instant nous nous concentrons sur le HTML et ses balises, qui ont chacune un sens particulier.

### 3.5) Les listes à puces

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

Nous allons découvrir ici deux types de listes à puces :

- Les listes non ordonnées
- Les listes ordonnées

#### 3.5.1) Liste non ordonnée

Une liste non ordonnée ressemble à ceci :

- Fraises
- Framboises
- Cerises

C'est un système qui nous permet de faire une liste d'éléments, sans notion d'ordre (il n'y a pas de « premier » ni de « dernier »).

Créer une liste à puces non ordonnée est très simple. Il suffit d'utiliser la balise `<ul>` `</ul>`.

Puis l'on va écrire chacun des éléments de la liste entre la balise `<li>` `</li>`. Toutes ces balises doivent se trouver entre `<ul>` et `</ul>`.

**Exemple :**

```
<ul>
  <li>Fraises</li>
  <li>Framboises</li>
  <li>Cerises</li>
</ul>
```

`<ul>` `</ul>` délimite toute la liste, (Unordered List).

`<li>` `</li>` délimite un élément de la liste (une puce), (List Item).

Vous pouvez mettre autant d'éléments que vous voulez dans la liste à puces.

**Remarque :**

Il est possible d'imbriquer des listes à puces (créer une liste à puces dans une liste à puces). Si vous voulez faire ça, ouvrez une seconde balise `<ul>` à l'intérieur d'un élément `<li>` `</li>`.

### 3.5.2) Liste ordonnée

**Une liste ordonnée fonctionne de la même façon, seule une balise change : il faut remplacer `<ul></ul>` par `<ol></ol>`, (Ordered List).**

À l'intérieur de la liste, on utilise toujours des balises `<li></li>` pour délimiter les éléments.

L'ordre dans lequel vous mettez les éléments de la liste est important. Le premier `<li></li>` sera l'élément n°1, le second sera le n°2 etc...

**Exemple :**

```
<h1>Ma journée</h1>
<ol>
  <li>Je me lève</li>
  <li>Je déjeune</li>
  <li>Je vais au lycée</li>
</ol>
```




## 4) Créer des liens

Un site web est composé de plusieurs pages. Pour aller d'une page vers une autre on utilise les liens.

Qu'est ce qu'un lien ?

Il s'agit d'un élément sur lequel on peut cliquer pour se rendre sur une autre page.

Bonjour. Souhaitez-vous visiter le [Site du lycée Jean Rostand](#) ?  
C'est un joli site ! 

On peut faire un lien d'une page vers une autre page du même site web, mais on peut aussi faire un lien vers un autre site. Dans les deux cas le fonctionnement est le même.

Pour faire un lien, il faut :

Utiliser la balise `<a>`.

Et lui ajouter l'un attribut « href » pour indiquer vers quelle page le lien doit conduire.

Un lien doit toujours être placé dans un paragraphe.

### 4.1) Un lien vers un autre site

Par exemple, le code ci-dessous est un lien qui amène vers le site du lycée, situé à l'adresse `http://www.lycee-jrostand.edu`

Exemple :

```
<a href="http://www.lycee-jrostand.edu">Site du lycée Jean Rostand</a>
```

Ce type de lien est appelé « **liens absolus** » car on indique l'adresse complète.

Par défaut, le lien s'affiche en bleu souligné, si vous avez déjà ouvert la page le lien s'affiche en violet.

Nous verrons comment changer cette apparence lorsque nous étudierons le CSS.

Si vous faites un lien vers un site qui comporte une adresse contenant le symbole « & », vous devrez remplacer tous les « & » par « &amp; » dans le lien.

Exemple :

`http://www.site.com/?data=15&name=toto` (mal écrit)

`http://www.site.com/?data=15&name=toto` (bien écrit)

Cela est nécessaire pour avoir une page web correctement construite en HTML5.

## 4.2) Un lien vers une autre page de son site

### 4.2.1) Deux pages situées dans un même dossier

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 :

```
<a href="destination.html">Cliquez ici</a>
```

On dit que c'est un **lien relatif** car l'adresse à laquelle on fait référence est écrite

en fonction de la situation du fichier de départ.

Exercice :

Écrire la partie de code des deux fichiers, page1.html et page2.html.

Aller à la [page 2](#)

## Bienvenue sur la page 2

Page1.html

```
<p>
```

```
    Aller à la <a href="http://www.lycee-jrostand.edu">page  
2</a>
```

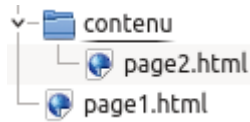
```
</p>
```

Page2.html

```
<h1>Bienvenue sur la page 2</h1>
```

### 4.2.2) Deux pages situées dans des dossiers différents

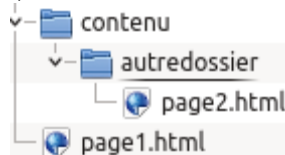
Imaginons que page2.html se trouve dans un sous-dossier appelé « contenu » comme sur la figure suivante :



Le lien doit être écrit comme ceci :

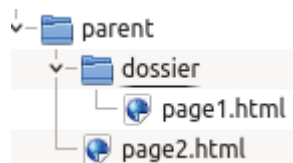
```
<a href="contenu/page2.html">
```

S'il y avait plusieurs sous-dossiers, on écrirait ceci :



```
<a href="contenu/autredossier/page2.html">
```

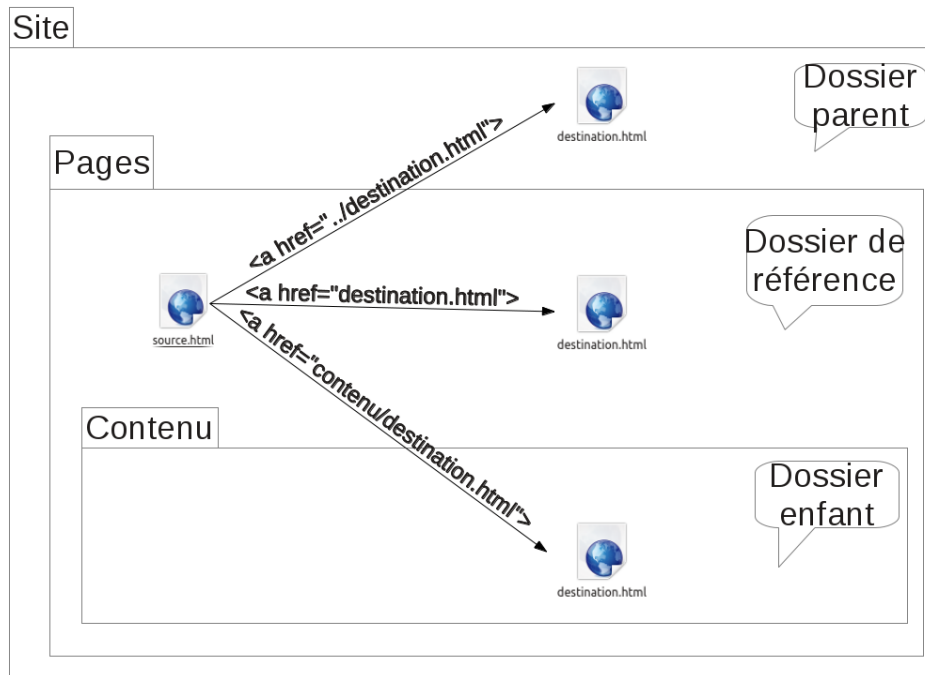
Si le fichier cible est placé dans un dossier qui se trouve « plus haut » dans l'arborescence :



```
<a href="../page2.html">. Les deux points signifient un dossier plus haut dans l'arborescence.
```

### 4.2.3) Résumé en images

Pour l'utilisation des liens relatifs, il suffit de regarder dans quel « niveau de dossier » se trouve le fichier cible pour savoir comment écrire le lien.



### 4.3) Un lien qui affiche une infobulle au survol

L'attribut « title » affiche une bulle d'aide lorsque l'on passe la souris sur le lien. Cet attribut est facultatif.

La bulle d'aide peut être utile pour informer le visiteur avant même qu'il n'ait cliqué sur le lien.

```
<p>BTS      <a      href="http://www.lycee-jrostand.edu"
title="Systèmes Numériques option Informatique et Réseaux">SN-IR</a></p>
```

## 5) Les images

### 5.1) *Les différents formats d'images*

Selon le format de l'image, le poids (en Ko, voire en Mo) est plus ou moins élevé et sa qualité varie.

Certains formats sont plus adaptés que d'autres selon l'image (photo, dessin, image animée...).

#### 5.1.1) Le JPEG

Les images au format JPEG (Joint Photographic Expert Group) sont très répandues sur le Web. Ce format est conçu pour réduire le poids des photos (c'est-à-dire la taille du fichier associé), qui peuvent comporter plus de 16 millions de couleurs différentes. Les images JPEG sont enregistrées avec l'extension .jpg ou .jpeg.

**Notez que le JPEG détériore un peu la qualité de l'image, d'une façon généralement imperceptible. C'est ce qui le rend si efficace pour réduire le poids des photos.**

Quand il s'agit d'une photo, on ne peut généralement pas détecter la perte de qualité. Par contre, si ce n'est pas une photo, vous risquez de voir l'image un peu « baver ». Dans ce cas, il vaut mieux utiliser le format PNG.

#### 5.1.2) Le PNG

Le format PNG (Portable Network Graphics) est le plus récent de tous. Ce format est adapté à la plupart des graphiques (on pourrait dire « à tout ce qui n'est pas une photo »).

Le PNG a deux gros avantages :

- il peut être rendu transparent
- il n'altère pas la qualité de l'image.

Le PNG a été inventé pour concurrencer un autre format, le GIF, à l'époque où il fallait payer des royalties pour pouvoir utiliser des GIF. Depuis, le PNG a bien évolué et c'est devenu le format le plus puissant pour enregistrer la plupart des images.

Le PNG existe en deux versions, en fonction du nombre de couleurs que doit comporter l'image :

- PNG 8 bits : 256 couleurs
- PNG 24 bits : 16 millions de couleurs (autant qu'une image JPEG)

Au fait, si le PNG 24 bits peut afficher autant de couleurs qu'une image JPEG, et qu'en plus il peut être rendu transparent sans modifier la qualité de l'image... quel est l'intérêt du JPEG ?

**La compression du JPEG est plus puissante sur les photos. Une photo enregistrée en JPEG se chargera toujours beaucoup plus vite que si elle était enregistrée en PNG. Je vous conseille donc toujours de réserver le format JPEG aux photos.**

### 5.1.3) Le GIF

C'est un format assez vieux, qui a été néanmoins très utilisé (et qui reste très utilisé par habitude). Aujourd'hui, le PNG est globalement bien meilleur que le GIF (les images sont généralement plus légères et la transparence est de meilleure qualité). Je vous recommande donc d'utiliser le PNG autant que possible.

**Le format GIF est limité à 256 couleurs (alors que le PNG peut aller jusqu'à plusieurs millions de couleurs).**

**Néanmoins, le GIF conserve un certain avantage que le PNG n'a pas : il peut être animé.**

### 5.1.4) Il existe un format adapté à chaque image

Si on résume, voici quel format adopter en fonction de l'image que vous avez :

- **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.

### 5.1.5) A faire et à ne pas faire

#### a) *Choisissez bien le nom de votre image*

Si vous voulez éviter des problèmes, prenez l'habitude d'enregistrer vos fichiers avec :

des noms en minuscules,

sans espace ni accent, par exemple : « mon\_image.png ».

Vous pouvez remplacer les espaces par le caractère underscore (tiret-bas).

#### b) *Bannissez les autres formats*

Les autres formats non cités ici, comme le format BITMAP (\*.bmp) sont à bannir car bien souvent ils ne sont pas compressés, donc trop gros. Ils ne sont pas du tout adaptés au Web. On peut en mettre sur son site mais le chargement sera vraiment extrêmement long !

## 5.2) Insérer une image

### 5.2.1) Insertion d'une image : <img />

La balise <img /> permet d'insérer une image.

C'est une balise de type orpheline. En effet, nous n'avons pas besoin de délimiter une portion de texte, nous voulons juste insérer une image à un endroit précis.

Les images doivent se trouver obligatoirement à l'intérieur d'un paragraphe (<p></p>).

**La balise doit être accompagnée de deux attributs obligatoires :**

#### a) *L'emplacement de l'image : attribut « src »*

L'attribut src : permet d'indiquer l'emplacement de l'image que l'on veut insérer.

Vous pouvez soit mettre **un chemin absolu**

src="http://www.site.com/images/fleur.png"

soit mettre le **chemin en relatif** (ce qu'on fait le plus souvent). Ainsi, si votre image est dans un sous-dossier images, vous devrez taper :

src="images/fleur.png"

#### b) *Texte alternatif : attribut « alt »*

L'attribut « alt » : signifie « texte alternatif »

On doit toujours indiquer un texte alternatif à l'image, c'est-à-dire un court texte qui décrit ce que contient l'image.

Ce texte sera affiché à la place de l'image si celle-ci ne peut pas être téléchargée (cela arrive), ou dans les navigateurs de personnes handicapées (non-voyants) qui ne peuvent malheureusement pas « voir » l'image. Cela aide aussi les robots des moteurs de recherche pour les recherches d'images.

Pour une fleur, on mettrait par exemple : alt="Rose rouge".

### 5.2.2) Exemple d'insertion d'image

```
<p>
  Voici une photo de vacances à la montagne :<br />
  
</p>
```

### 5.2.3) Ajouter une infobulle

L'attribut permettant d'afficher une bulle d'aide est le même que pour les liens : il s'agit de title. Cet attribut est facultatif (contrairement à alt).

Voici ce que cela peut donner :

```
<p>
  Voici une photo de vacances à la montagne :<br />
  
</p>
```

### 5.2.4) Miniature cliquable

**Si l'image est très grosse, il est conseillé d'en afficher la miniature sur le site.**

Ajoutez ensuite un lien sur cette miniature pour que les visiteurs puissent afficher l'image en taille originale.



## Table des matières

1)Introduction.....	1
1.1)HTML et CSS : deux langages pour créer un site web.....	1
1.1.1)Le rôle de HTML.....	1
1.1.2)Le rôle de CSS.....	2
1.2)Les navigateurs web.....	2
1.2.1)Pourquoi le navigateur est important ?.....	2
1.2.2)Les navigateurs web sur l'ordinateur.....	3
1.2.3)Les navigateurs sur mobile.....	3
2)Structure d'un document HTML.....	4
2.1)Les balises et leurs attributs.....	4
2.1.1)Les balises.....	4
2.1.2)Les attributs.....	5
2.2)Structure de base d'une page HTML5.....	7
2.2.1)Emboîtement des balises.....	7
2.2.2)Les balises de bases (DOCTYPE, html, head, body).....	7
2.2.3)L'encodage (charset).....	8
2.2.4)Le titre principal de la page.....	8
2.3)Les commentaires.....	9
2.4)Le code des pages HTML est visible par tout le monde.....	9
2.5)Remarque.....	10
3)Organisation du texte.....	11
3.1)Les paragraphes.....	11
3.2)Sauter une ligne.....	11
3.3)Les titres.....	12
3.4)La mise en valeur.....	13
3.4.1)Mettre bien en valeur.....	13
3.4.2)Mettre un peu en valeur.....	13
3.4.3)HTML pour le fond, CSS pour la forme (Résumé).....	14
3.5)Les listes à puces.....	15
3.5.1)Liste non ordonnée.....	15
3.5.2)Liste ordonnée.....	15
4)Créer des liens.....	17
4.1)Un lien vers un autre site.....	17
4.2)Un lien vers une autre page de son site.....	18
4.2.1)Deux pages situées dans un même dossier.....	18
4.2.2)Deux pages situées dans des dossiers différents.....	18
4.2.3)Résumé en images.....	20
4.3)Un lien qui affiche une infobulle au survol.....	20
5)Les images.....	21

5.1)Les différents formats d'images.....	21
5.1.1)Le JPEG.....	21
5.1.2)Le PNG.....	21
5.1.3)Le GIF.....	21
5.1.4)Il existe un format adapté à chaque image.....	22
5.1.5)A faire et à ne pas faire.....	23
5.2)Insérer une image.....	23
5.2.1)Insertion d'une image : <img />.....	23
5.2.2)Exemple d'insertion d'image.....	24
5.2.3)Ajouter une infobulle.....	24
5.2.4)Miniature cliquable.....	24