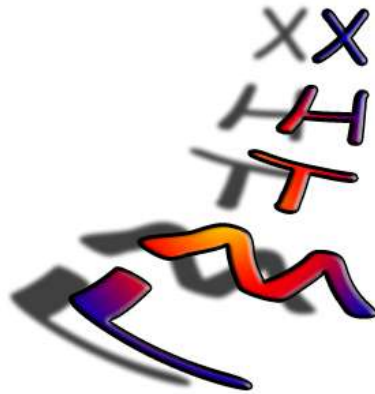


# COURS XHTML



## Chapitre 01 : Ma Première Page Web

## Disclaimer

Tous ces cours sont rédigés par un étudiant en informatique. Il est donc tout à fait possible que des erreurs se soient glissées à l'intérieur de ceux-ci. Cependant, je fais tout mon possible pour les éviter. Si vous en apercevez une, n'hésitez pas à me la signaler, que je la corrige au plus vite. Pour me contacter : [jonathanpetitcolas@yahoo.fr](mailto:jonathanpetitcolas@yahoo.fr). Remarquez que vous pouvez aussi m'envoyer un petit mot d'encouragement ! Cela fait toujours plaisir !

Cette série de cours est disponible gratuitement sur Internet. Vous trouverez la liste complète de tous les tutoriaux écrits à ce jour sur mon site personnel : [www.programmix.c.la](http://www.programmix.c.la) . Si vous désirez diffuser vous aussi mes créations, allez dans la rubrique « Diffusion » de ce même site pour prendre connaissance des quelques conditions à respecter.

# Introduction

- **Introduction :**

Suite au chapitre précédent, vous devriez avoir une idée un peu plus précise de votre site et de son contenu. Maintenant, il va falloir passer à la phase suivante, qui est sûrement l'une des plus longues : la réalisation de vos pages...

Dans ce chapitre, nous nous contenterons de créer une simple page au graphisme très dépouillé. Pour cela, nous apprendrons comment créer un fichier XHTML grâce à NotePad ++ (voir chapitre précédent), bien que la méthode soit la même avec le bloc-notes fourni par défaut avec Windows. Puis, nous étudierons la structure minimale d'une page Web avant de la tester dans un navigateur quelconque.

- **Objectifs :**

- Savoir enregistrer un fichier au format XHTML
- Utilisation de commentaires
  - \* Balise `<!-- -->`
- Connaître la syntaxe minimale d'une page WEB
  - \* Balise `<! >`
  - \* Balise `<html>`
  - \* Balise `<head>`
  - \* Balise `<body>`
  - \* Balise `<title>`

# Le format \*.html

(Une petite manipulation obligatoire)

- **But de la conversion :**

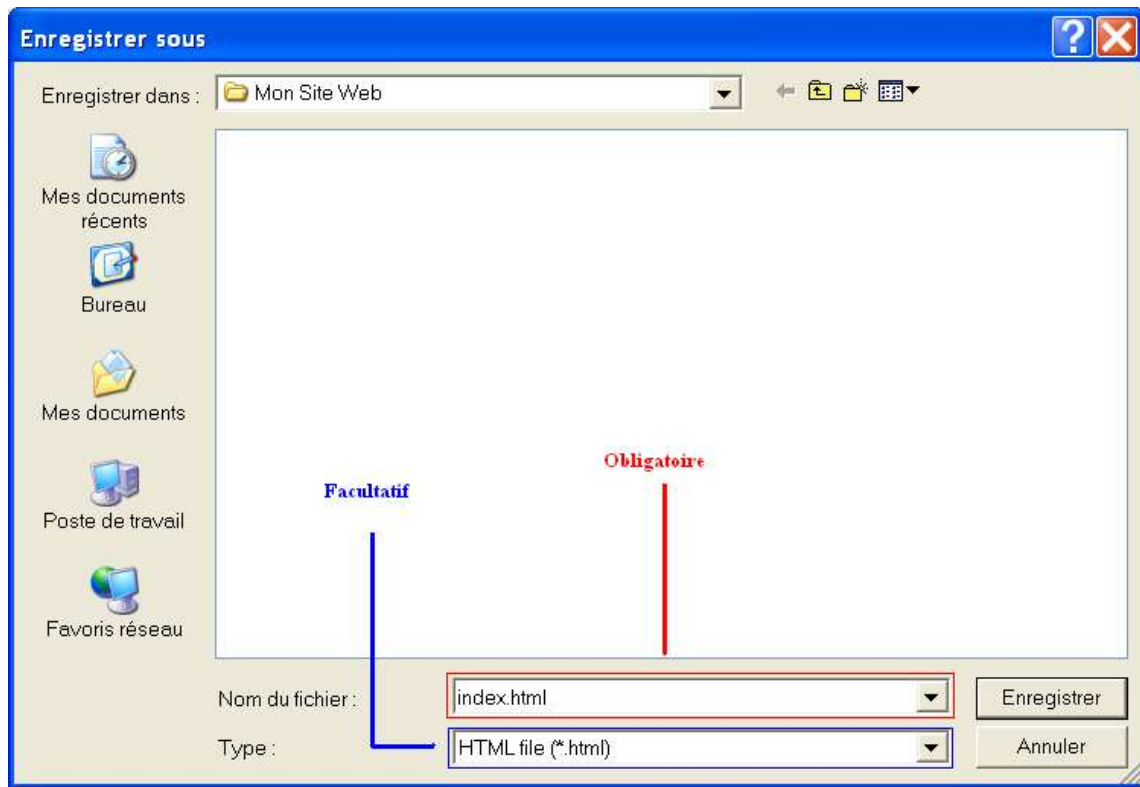
Lors de la création de vos pages, vous écrivez dans un fichier texte (extension \*.txt). Or, pour qu'un navigateur puisse interpréter vos lignes de code, il faut lui indiquer qu'il s'agit de code XHTML. C'est le but de l'extension \*.html.

Il existe aussi le format \*.htm. Quelle différence ? Le / final... A part ça, sachez qu'il n'y a aucune. Donc, libre à vous d'enregistrer vos pages en \*.html ou en \*.htm. Mais, alors, pourquoi deux formats pour une seule et même chose ? La raison vient du passé, et de l'ancien Windows 3.1 ! En effet, jadis, Windows n'acceptait que les extensions de trois caractères. Maintenant, peu importe. Par contre, la raison de l'apparition de ce / reste assez obscure...

Une fois votre fichier enregistré au bon format (la marche à suivre arrive dans quelques lignes), vous pourrez alors l'ouvrir avec votre navigateur. Pour cela, il vous suffira de double-cliquer sur votre fichier, et magie ! Votre page apparaîtra sous vos yeux ébahis ! Notez qu'il vous sera toujours possible de rouvrir votre page avec le bloc-notes pour modifier deux ou trois petites choses. Pour cela, il vous suffira de faire un clic-droit sur le fichier, et de choisir *Ouvrir avec le bloc-notes...* ou *Ouvrir avec...* > *Bloc-Notes* dans le menu déroulant qui s'affiche.

A présent, voyons comment enregistrer en XHTML...

- **Convertir un \*.txt en \*.html :**



*La fenêtre d'enregistrement...*

Cela n'est pas très difficile, que vous utilisiez NotePad ++ ou le bloc-notes. Dans les deux cas, il vous suffit de créer un nouveau fichier (ou d'en ouvrir un déjà existant), et d'aller dans le menu *Fichier > Enregistrer Sous...* Une fenêtre devrait s'ouvrir. Choisissez alors un emplacement sur votre disque dur dans lequel toutes vos pages seront enregistrées. Je vous conseille de créer un dossier spécialement prévu à cet effet. Dans l'exemple ci-dessus, il s'agit du dossier "Mon site Web".

Donnez un nom à votre fichier. Peu importe celui-ci. Dans l'exemple ci-dessus, je l'ai baptisé *index*. C'est un nom obligatoire à donner impérativement à l'une de vos pages. En effet, par convention, c'est cette page que les navigateurs recherchent en premier, lorsqu'ils se connectent à un serveur. Mais, nous reverrons cela plus tard...

Maintenant, passons à la conversion \*.txt / \*.html. A la fin de la case *Nom*, écrivez *\*.html*, sans l'étoile, qui remplace le nom de votre fichier (voir ci-dessus). Cela aura pour effet d'enregistrer votre page au format qui nous intéresse. Il existe aussi la case *Type*, mais elle n'est pas obligatoire, pourvu que vous n'oubliez pas d'ajouter l'extension comme décrit ici.

# Les Balises

(Découverte de l'élément de base du langage)

Le XHTML est, tout comme son aîné le HTML, un langage de balises. Comprendre ce qu'est une balise ainsi que sa syntaxe est nécessaire pour la suite de ce cours. Donc, essayez de retenir les idées générales, ainsi que le vocabulaire (ce sera le même que celui utilisé dans la suite de ces tutoriaux).

Une balise, dans ce contexte bien évidemment, est un mot-clé entouré de chevrons < et > . Ce n'est pas bien compliqué, n'est-ce pas ? Prenons comme exemple une des balises que vous utiliserez de nombreuses fois, la balise *HTML*. En voici une (ou une demie... ça dépend comment on voit les choses...) :

```
<html>
```

Remarquez que j'utiliserais les mêmes couleurs que celles par défaut de NotePad++, et que le code sera toujours écrit comme ci-dessus. Enfin, contrairement au HTML qui ne vérifiait pas la casse, le XHTML nécessite que toutes vos balises soit écrites en minuscules.

Pas très joli, n'est-ce pas ? Ne vous inquiétez pas ! Vos visiteurs ne verront pas les balises apparaître. Sauf si vous avez commis des erreurs dans votre source...

Les balises fonctionnent toujours par paires. Une balise *ouvrante* et une balise *fermante*. Je ne suis pas sûr que ces termes soient normalisés, mais ce sont que j'utiliserai, et ils résument plutôt bien leurs rôles. La balise ci-dessus est une balise ouvrante, tandis que la suivante est une fermante :

```
</html>
```

La différence ? Le slash / avant le mot-clef. Les deux balises transformeront tout ce qu'il y a entre elles. Vous comprendrez un peu mieux à la fin de ce chapitre, car cela doit être assez abstrait pour l'instant, non ?

Sachez que toute balise ouverte doit être fermée ! Une bonne habitude consiste donc à refermer la balise tout de suite après l'avoir ouverte. Ainsi, pas d'oubli ! Et, grâce à la magie des traitements de texte, vous pourrez écrire entre elles ensuite !

Certaines balises prennent ce que l'on appelle des *arguments* ou *attributs*. Ceux-ci sont en fait des paramètres supplémentaires qui permettent un peu plus de variétés dans le choix des couleurs, la position sur la page, etc... Ils se placent toujours avant le chevron fermant de la balise ouvrante... Un exemple :

```
<body bgcolor="#000000">
```

Ici, il s'agit de la balise indiquant le début du corps de votre page. Remarquez l'attribut *bgcolor*. Il est écrit à la suite, avant le chevron fermant. Quant au signe égal et à la valeur derrière, nous n'y toucherons pas dans ce chapitre. Sachez simplement que cette balise met un fond noir sur votre page.

Les noms d'attributs aussi doivent être écrits en minuscules. Mais, pour les valeurs de ceux-ci, vous pouvez mettre ce que vous voulez : majuscules ou minuscules ! N'oubliez quand même pas de les écrire entre guillemets, et ceux, même pour les valeurs numériques.

Une dernière remarque concernant les attributs : ils ne se mettent que dans la balise ouvrante. Pour la balise fermante, il vous suffit juste de récrire le mot-clé, et rien de plus. Exemple d'un couple de balises :

```
<body bgcolor="#000000">
</body>
```

J'ai dit plus haut que les balises fonctionnaient toujours par paires... Et, comme toute règle, celle-ci a ses exceptions... Il en existe en effet certaines qui préfèrent la solitude... On les reconnaît au / juste avant le chevron fermant. Un exemple avec la balise suivante, qui permet un retour à la ligne :

```
<br/>
```

Beaucoup d'informations, n'est-il pas ? Essayez de retenir l'essentiel : une balise, lorsqu'elle est double, est en deux parties : une ouverture et une fermeture. Tout s'écrit en minuscules, sauf éventuellement les valeurs des arguments, à mettre impérativement entre guillemets.

Ces quelques généralités très théoriques sur les balises nous permettent d'aborder tranquillement mais sûrement la création d'une page Web. Mais avant, un petit détour par les commentaires (le dernier, promis !)...

# Commentaires

Les commentaires sont des morceaux de codes qui ne seront pas interprétés par le navigateur. A quoi cela sert-il donc alors ? Eh bien, imaginons que vous soyez plusieurs à faire un site. Il est fortement probable que chacun programmera un morceau de page dans son coin, et que vous réunirez les parties par la suite. Mais, si vous avez un problème et que vous demandez à votre ami de vous aider, il lui faudra comprendre votre code ! Donc, mettre quelques indications ne pourront que lui être utile.

Vous programmez seul ? Peu importe ! Quelques commentaires sont toujours les bienvenus. En effet, si vous ne touchez plus à votre site pendant plusieurs semaines et que vous voulez le remettre à jour, il va vous falloir replonger dans le bain ! Et, voir des centaines de lignes de code sans aucun souvenirs de leurs significations est assez déplaisant. Ainsi, si vous vous facilitez le travail dès maintenant en vous mettant quelques indices, il y aura moins de difficultés plus tard à le modifier !

Ce sont les deux seules raisons d'être des commentaires ! Ne les sous-estimez pas ! Ils sont la clé d'un programme (et donc de pages) réussi ! Même si il est complètement inutile de commenter des choses évidentes... Du moins, une fois un langage connu ! Cependant, sur le WEB, il y en a assez peu... Pourquoi ? Nul ne le sait...

Pour écrire des commentaires en XHTML, il y a une balise spécialement conçu pour l'occasion. Elle est assez particulière, car en plus d'être une solitaire, elle s'écrit à sa façon, à savoir :

```
<!-- Écrivez votre commentaire ici. -->
```

Observez et retenez les symboles suivants et devançants les chevrons de cette balise. Ce sont eux qui indiquent qu'il s'agit d'un commentaire.

Maintenant que vous connaissez votre première balise, passons aux choses sérieuses !



# Structure minimale

(Enfin, on commence !)

Après de nombreuses, très nombreuses (trop nombreuses ?) tergiversations, nous voici enfin au point crucial du chapitre : la création d'une page Web ! Depuis le temps que nous en parlons, il serait peut-être temps de commencer.

Ouvrez donc un nouveau fichier et enregistrez-le sous le nom que vous voulez. Au hasard, *index.htm*. Puis, échauffez vos doigts, il va falloir écrire !

Tout d'abord, écrivez une ligne plutôt méchante... Elle devrait normalement tenir sur une seule ligne... Mais, pour des raisons de lisibilités, je l'étale sur deux.

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

Cette ligne permet de faire savoir que vous utilisez le langage XHTML, version 1.1. Il s'agit de la version la plus récente à ce jour ( 1er juin 2005 ). Inutile de l'apprendre par coeur : il vous suffira de la copier/coller si vous en avez besoin par la suite... Cela dit, vous pouvez toujours vous amuser !

Quelle est donc son utilité ? En déclarant ce *DocType*, vous indiquez au navigateur Internet que vous connaissez les standards du Web, et que vous allez les utiliser. Ainsi, il interprétera de façon très rigoureuse tout votre code. De cette manière, votre page respectera les différentes normes en vigueur, et l'affichage de celle-ci sera alors la même partout. Si vous n'en mettez pas, en effet, chaque navigateur essaiera de comprendre votre code à sa manière, et des variations pourront se voir selon le browser utilisé. D'où l'importance de cette ligne !

A présent, commençons notre page ! Insérez la ligne suivante :

```
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="fr" lang="fr" >
```

Assez compliqué aussi, non ? Ne vous en faites pas, elles ne sont pas toutes comme cela. Ici, on va chercher tout ce dont nous aurons besoin pour programmer en XHTML sur le site du W3C (<http://www.w3c.org>). Il s'agit de l'organisme international qui fixe les règles et normes en vigueur sur le Web.

Le premier argument indique *l'espace de nom* à utiliser. C'est un lien vers une page du W3C, qui définit les différents mots-clef du langage. Essayez de retenir l'adresse... Elle n'est pas bien difficile à apprendre.

Quant au deuxième argument, il s'agit de la langue de vos pages. Dans ce cas, ce sera une page française (valeur *fr*). Pour une page anglaise, il vous faudra mettre *en*, pour une italienne *it*, etc... Cela est utile pour augmenter l'accessibilité de vos pages : les moteurs de recherche sauront quelle est la langue de votre site, ce qui permettra un meilleur référencement.

Le troisième argument n'est normalement pas obligatoire, mais il est conseillé de le mettre afin de rendre vos pages compatibles avec les anciens navigateurs, qui ne sont pas encore à jour au niveau des normes Internet (n'est-ce pas IE ?).

A présent, nous allons retourner à des choses beaucoup plus simples. Mais avant, n'oubliez pas de refermer la balise `<html>`. Vous l'aviez oublié, pas vrai ? Car, même s'il y a quelques attributs, il s'agit quand même d'une balise ouvrante !

Nous allons définir la tête et le corps de votre page. Pour cela, complétez votre page jusqu'à obtenir ceci (les commentaires sont facultatifs, mais pour votre première page, je vous conseille de les mettre).

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="fr" lang="fr">
  <head>
    <!-- Tête de votre page -->
  </head>
  <body>
    <!-- Corps de votre page -->
  </body>
</html>
```

Comme vous pouvez le constater, les balises ajoutées sont beaucoup plus simples que les précédentes. Elles définissent en fait les deux zones de votre page.

La zone de tête (*head* en anglais) concerne tout ce qui n'est pas sur la page à proprement parler. Je m'explique : elle concerne tout ce qu'il y a autour : dans la barre de titre du navigateur, dans la barre d'état, etc... Elle comporte aussi ce qui ne doit pas s'afficher, comme les balises nécessaires au référencement de votre site. Mais, patience ! Vous avez encore du chemin à parcourir, jeune programmeur !

Le corps de votre page (*body* en anglais) comportera tout ce qui devra s'afficher sur la page, dans la zone principale de votre navigateur.

Nous allons modifier un tout petit peu le contenu de ces deux parties... Tout d'abord, nous allons écrire un petit message dans la barre de titre de votre navigateur. Vous savez, la barre bleue qui est tout en haut et sur laquelle est dessiné les boutons pour fermer, agrandir ou réduire la fenêtre. C'est la balise `<title>` qui nous permet cela. Voici la ligne à ajouter (entre les deux balises `<head>`) :

```
<title> Ma première page web ! </title>
```

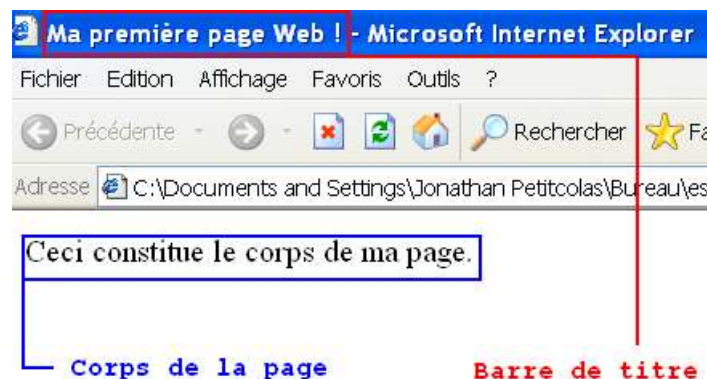
Avant d'essayer votre page, nous allons ajouter un petit quelque chose au corps de votre page. Par exemple, nous pouvons écrire un petit mot. Modifiez donc votre code pour obtenir ceci :

```
<body>
Ceci constitue le corps de ma page !
</body>
```

Pour ceux qui n'auraient pas tout suivi, voici le code que vous devriez avoir obtenu :

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="fr" lang="fr">
  <head>
    <title> Ma première page web ! </title>
  </head>
  <body>
    Ceci constitue le corps de ma page !
  </body>
</html>
```

Essayez votre page maintenant ! Miracle ! Ça marche ! Alors, content ? Sachez juste que ce que vous venez de faire est la syntaxe minimale d'une page XHTML. C'est elle que vous retrouverez partout ! Il va donc sans dire qu'il faut la connaître sur le bout des doigts !



*Le résultat que vous devriez obtenir...*

Pour clore ce chapitre, une dernière petite remarque concernant l'organisation du code. Je vous conseille de mettre des tabulations (ou des alinéas, c'est la même chose) ainsi que de souvent aller à la ligne. Cela augmente la lisibilité de votre code. Et, vu qu'il va vous falloir passer plusieurs heures dessus, autant les rendre le plus agréable possible.

**La suite dans le prochain chapitre !**