

III - Créer une page web

Le langage d'une page Web est le **HTML : HyperText Markup Language** qui signifie en français « langage de balisage d'hypertexte ». C'est un langage qui permet de mettre en forme du texte à l'aide de balises. On peut également y placer des éléments interactifs comme des liens, des images ou bien encore des animations.

Il est possible de créer un site Web sans apprendre le HTML, en utilisant un système de gestion de contenu (ou CMS) qui offre des fonctionnalités permettant de faire rapidement un site avec un bel aspect. Cela reste cependant limité (un peu moins en prenant un abonnement) et si on souhaite faire évoluer le site et y intégrer des programmes rendant les pages dynamiques, il est nécessaire d'apprendre ce langage.

Ainsi pour avoir la main sur tous les éléments constituant nos pages, il est judicieux de savoir construire un fichier HTML « à la main ». La mise en forme du site sera gérée par le **CSS (Cascade Style Sheets) ou feuille de style en cascade**.

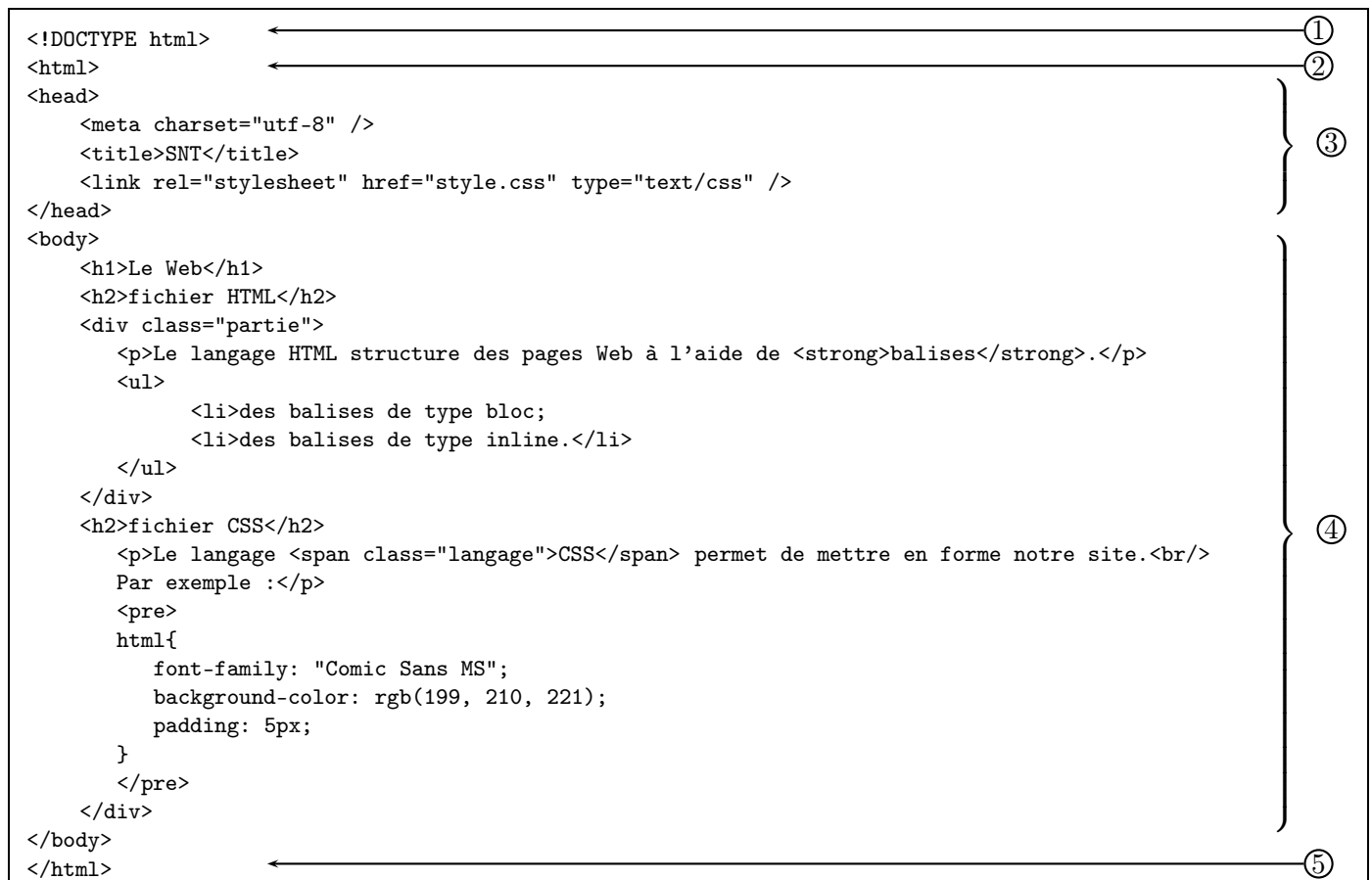
Un même éditeur permet de concevoir ces types de fichiers (mais également du javascript ou php dans le cas de sites dynamiques). Il existe plusieurs, comme [Notepad ++](#), [Sublimetext](#) ou [Visual Studio Code](#), ce dernier étant selon moi le plus pratique.

Des interfaces en ligne permettent également de s'entraîner à la conception de pages web simples [jsfiddle](#), [repl.it](#) ou [glitch](#).

a) Le langage HTML

Vidéo de présentation du langage HTML

Une page HTML doit être construite sous une forme qu'il faut respecter pour que le navigateur l'affiche correctement :



- ① Indique au navigateur que la page est en HTML ;
- ② Balise ouvrante (il y en a un peu partout dans le document) ;
- ③ En-tête (head) : informations générales sur le document ;
- ④ Corps (body) : c'est el contenu de la page organisé à l'aide des balises ;
- ⑤ Balise fermante (toute balise ouvrante doit avori sa balise fermante)

On peut retrouver la liste de tous les éléments HTML existant :

<https://developer.mozilla.org/fr/docs/Web/HTML/Element>.

Dans la construction d'une page HTML, on n'utilise finalement qu'une partie infime de toutes ces balises.

Remarque : ATTENTION à l'organisation des balises : on peut en insérer les unes dans les autres mais il faut faire attention à l'ordre dans lequel les fermer :

`<balise1>... <balise2> ... </balise2> ... </balise1>` où ... peut être du texte.

Par contre `<balise1>... <balise2> ... </balise1> ... </balise2>` ne fonctionne pas car on doit fermer en premier lieu la dernière balise ouverte et ce n'est pas le cas ici.

b) Le langage CSS

Vous avez dû repérer la balise **link** de la partie `<head>` du fichier html qui permet d'utiliser un fichier de style nommé CSS (Cascading Style Sheets ou feuille de styles en cascade). Dans notre exemple, le fichier est appelé style.css.

Dans un fichier CSS, on trouve principalement trois éléments différents :

- les noms de balises (ou des classes et id) : celles dont on veut modifier l'apparence par défaut ;
- des propriétés CSS. Nous en verrons certaines, mais il n'est pas envisageable de tout explorer.

Voici trois liens pour des compléments :

- des propriétés rangées par catégories :

<https://openclassrooms.com/fr/courses/1603881-apprenez-a-creer-votre-site-web-avec-html5-etcss3/1608902-memento-des-proprietes-css>

- des propriétés dans l'ordre alphabétique :

<http://www.css-faciles.com/proprietes-css-liste-alphabetique.php>

- un site qui permet de créer ses propres styles :

<https://html-css-js.com/css/generator/box-shadow/>

- les valeurs : pour chaque propriété, on doit indiquer une valeur.

ATTENTION à ne pas oublier le point-virgule à la fin de chaque ligne...

Exemple :

```
html {
    font-family : "Comic Sans MS", Verdana, sans-serif;
    background-color : rgb(199, 210, 221);
    padding : 5px;
}
h1 {
    background : border-box rgba(200, 0, 30, 0.7);
    border-radius : 7px;
    padding-left : 10px;
    font-style : italic;
    font-weight : bold;
    text-transform : uppercase;
}
.langage {
    color : blue;
}
```

Trois attributs ont été donnés à la balise html, les valeurs sont pour tout le document.

Pour les titres h1, on impose un style particulier pour le fond (background), un bord arrondi, puis un décalage à gauche et enfin trois attributs pour la police de caractère.

Enfin, chaque élément de la **classe** langage (**identifiée dans le CSS par un point**) sera en bleu. On peut aussi donner à un id des propriétés particulières, dans ce cas, l'id est précédé dans le fichier CSS par un #.

c) Exploration du DOM

J'ai créé un site web statique :

<https://icnlauregatet.github.io/SNT/documents/site/pageInit.html>

contenant une page html et une feuille de style.

Il est possible de voir dans le navigateur la structure HTML de cette page : sous Firefox faire Ctrl+Maj+I, sous Chrome appuyer sur la touche F12.

Il est alors possible de déplier pour voir toute la structure de la page (y compris l'interprétation du fichier en cas d'erreur avec les balises).

Il est alors possible de modifier certains textes dans cette partie du navigateur. Sélectionner un texte donne aussi accès à ses éléments de style (également modifiables).

Attention, en aucun cas vous ne modifiez ainsi la page Web déposée sur le serveur, on s'en rend compte en actualisant la page (F5).

d) Travail à faire

Je vous demande de télécharger le fichier HTML et le fichier CSS du site dont il est question ci-dessus et l'ouvrir dans un environnement. Voici une vidéo expliquant la méthode suivant la façon dont vous souhaitez travailler :

- avec [NotePad++](#) ;
- avec [Visual Studio Code](#) :
- sur le site [repl.it](#) :

Cette page html contient 3 parties dont les titres sont « fichier HTML », « fichier CSS » et « fichiers Javascript et Php ». L'objectif est de comprendre comment sont construits certains éléments pour en ajouter de nouveaux.

1. Chercher une image du logo CSS et du logo Javascript ou Php. Dans les parties 2 et 3, ajouter l'image représentant le CSS pour la partie 2 et le Javascript ou Php pour la 3ème partie. Intégrer ces images comme pour celle sur le HTML.

Pour cela, identifier la balise `` présente dans la partie HTML (là où vous trouverez le logo) et regarder comment elle a été utilisée (devinez pourquoi je l'ai mise dans une balise `<p>...`).

Cette balise `` possède deux attributs. Déterminer leur utilité pour la réutiliser avec les logos du CSS et du Javascript.

2. Les titres h1 et h2 sont dans un certain style (voir fichier CSS), modifier ces styles de sorte d'avoir quelque chose à votre goût (s'aider des liens mis dans la partie sur le CSS).
3. Le paragraphe relatif au HTML a un fond différent. En modifiant uniquement le fichier HTML, faire en sorte que les deux autres paragraphes soient avec le même fond (bien regarder ce qui fait qu'il y a ce fond dans le paragraphe 1 sachant que c'est la définition du fond est dans le css).
4. Certains mots sont en bleu. Modifier le fichier css pour qu'ils soient sous une autre forme.
5. Ajouter une quatrième partie avec la même structure que les précédentes. Elle présentera les difficultés que vous avez éprouvées dans ce travail ou ce qui vous a le plus intéressé. Y ajouter un lien vers un site dont le design vous fait envie.

