

GLSE01 - TP séance 2 : Page Web (HTML + CSS)

Objectifs

Construction d'une page web :

- comprendre et utiliser un fichier CSS ;
- définir la structure d'une page web ;

Consignes et Pré-requis

Lire attentivement cet énoncé de TP en suivant les instructions. En cas de question, faire appel à l'enseignant qui est là pour vous répondre, et surtout ne pas rester bloqué sur une des tâches à réaliser. Essayer dans tous les cas de chercher sur internet la réponse à vos questions.

CSS

Pour commencer voici quelques exercices pour prendre (ou reprendre) la main avec le fichier CSS et leur syntaxe.

Une feuille de style consiste en une suite de déclarations de la forme :

```
sélecteur_1, sélecteur_2, ..., sélecteur_m {  
    propriété_1: valeur_1;  
    propriété_2: valeur_2;  
    ...  
    propriété_n: valeur_n;  
}
```

où sélecteur est souvent un simple élément HTML comme body ou h1, où propriété est une propriété CSS permise pour cet élément et où valeur est une valeur permise pour la propriété concernée.

Ces définitions vont expliciter l'affichage de(s) élément(s) sélectionné(s). Noter les deux points après le nom de la propriété et le point virgule entre chaque déclaration.

Chaque élément possède ses propres propriétés (par exemple, la propriété `list-style` pour l'affichage des listes) mais quelques propriétés peuvent être utilisées avec tous les éléments. Voyons celles qui vont nous servir dans ce TP :

- **background-color** : permet de spécifier une couleur pour l'arrière plan. Les valeurs peuvent être soit un nom de couleur (mais il n'en existe qu'une dizaine de prédéfinies) soit une valeur hexadécimale de la forme #xxxxxx où x est un chiffre hexadécimal (le raccourci #xyz désigne la couleur #xyyzz). Plusieurs moyens s'offrent à vous pour connaître la valeur hexadécimale d'une couleur. Vous pouvez trouver une ressource sur le web (comme la page www.w3schools.com/css/css_colornames.asp).
- **background-image** : Permet de choisir une image pour l'arrière plan. Les valeurs permises sont url(x) ou url('y'), avec x une url, et y un chemin local.
- **background-position** : Permet de définir la position de l'image dans l'arrière plan

de l'élément sélectionné. Les valeurs permises sont top, center, right, top center, center center, etc.

- **color** : Permet de définir une couleur pour l'élément sélectionné. Les valeurs permises sont les mêmes que pour la propriété background-color
- **font-style** : Pour spécifier le style de police. Les valeurs possibles sont normal, italic ou oblique.
- **text-align** : Permet de définir l'alignement horizontal d'un texte. Les valeurs possibles sont left, right, center ou justify (pour que les marges gauches et droites soient régulières).
- **margin-left, margin-right, margin-top, margin-bottom** : Permet de définir un espace entre l'élément sélectionné et l'élément qui l'encadre. Les valeurs sont des nombres suivis d'une unité (par exemple cm pour le centimètre, px pour le pixel, em pour l'unité liée à la taille de la police de l'élément, . . .).
- **font-family** : Permet de changer de famille de polices, par exemple times, arial, courier

A titre d'exemple télécharger le fichier style.css du site web du cours (<http://www.lirmm.fr/~cherubini/GLSE301/style.css>). Si vous avez des doutes sur les propriétés, cherchez vous même plus d'informations sur internet.

Exercice

Récupérer (sauvegarder) le fichier **tp.html** (déjà utilisé dans la première séance du TP) à l'adresse <http://www.lirmm.fr/~cherubini/GLSE301/TP/TP1/tp.html> et le sauvegarder dans le sous-répertoire *public_html*.

Nous allons ensuite créer le fichier CSS qui sera associé à cette page. Vous pouvez éditer le fichier CSS avec n'importe quel éditeur texte (kate, kwrite,)

Exercice 1 (Couleur de fond)

Créer un fichier style.css et définir une nouvelle couleur de fond (par exemple la couleur #C8C8C8) pour la page que vous avez récupérée (tp.html).

Exercice 2 (Intégration du fichier CSS dans une page HTML)

Rajouter la ligne suivante dans l'entête de la page : `<link href="style.css" rel="stylesheet" type="text/css" />`. Télécharger le fichier <http://www.lirmm.fr/~cherubini/GLSE301/index.html> et regarder comment intégrer le fichier css dans la page html. Vérifier que la couleur de fond est bien appliquée et ensuite essayer d'autres couleurs de fond.

Exercice 3 (Un peu d'air)

Pour aérer un peu la lecture d'un document HTML, on met en général un espace entre les bords de la fenêtre et le corps du document. Définir une marge gauche, haute et droite pour le document de façon à ce qu'elle soit appliquée à tout le contenu de la page.

Exercice 4 (Liste de sélecteurs)

À l'aide d'une seule définition, spécifier une même couleur pour les éléments h1 et h2. Le fait de définir des propriétés pour plusieurs éléments en même temps n'empêche pas de rajouter plus loin dans la feuille de style des définitions pour un élément particulier de la liste. C'est toujours la dernière définition qui compte. Rajouter après une définition pour que l'élément h1 soit de plus centré sur la page et modifier sa couleur.

Exercice 5 (Mise en retrait des titres)

Faire en sorte que les titres de type h2 soient un peu en retrait (vers la gauche) vis à vis des autres éléments (Les valeurs de type numérique peuvent être négatives).

Exercice 6 (Police)

Changer la police des éléments internes du tableau sans modifier celle des titres.

Les sélecteurs de classe

Il est parfois utile de pouvoir définir des comportements différents pour un même élément. Pour cela, on utilise des **sélecteurs de classe** qui sont de la forme `élément.nom_de_classe` (le corps de la définition suit les mêmes règles qu'avant). Pour que cette définition s'applique dans le document HTML, il faut alors rajouter à l'élément l'attribut `class="nom_de_classe"`.

Exercice 7 (Un paragraphe spécial)

Définir deux comportements différents pour l'élément p : l'un général qui justifie les lignes d'un paragraphe et un autre spécial (nommé `special`) qui affiche le texte en rouge. Editer le document HTML pour que cette dernière propriété soit appliquée à l'avant dernier paragraphe.

Exercice 8 (Le tableau)

On va changer l'affichage du tableau :

1. Définir une nouvelle couleur de fond pour le tableau et le décaler vers la droite.
2. Définir une autre couleur de fond pour l'affichage des noms des colonnes.
3. En général, l'affichage des titres de livres se fait à l'aide d'une police italique. Réaliser cet affichage (nouvelle classe à définir).

Exercice 9 (Pour aller plus loin : Classe générique)

Une classe générique n'est pas liée à un élément de la page HTML, mais peut être utilisé par tous les éléments. La définition dans le fichier css est `.nom_de_classe` (l'élément n'est pas spécifié). Regarder l'exemple relatif au css du site du cours.

Ecrire une classe générique qui va vous permettre de décaler le texte de 10 cm à droite. L'utiliser sur le dernier paragraphe (`<p>`) ainsi que pour son titre (`<h2>`).

Les sélecteurs de pseudo-classe

La dernière forme de sélecteur que nous verrons dans ce TP concerne celle des pseudo-classes, de la forme `élément : pseudo_classe`. Elle est surtout employée pour changer l'affichage des liens dans une page web : `a:link` pour les liens non visités,

`a:visited` si l'internaute a déjà cliqué sur ce lien, `a:hover` lorsque la souris passe au dessus du lien et `a:active` lorsque l'internaute est en train de cliquer sur le lien.

Exercice 10 (Pour aller plus loin : Les liens)

Définir une couleur différente pour chacun des événements ci-dessus et tester le résultat.

Travailler avec des images

Voici certaines propriétés de l'élément `img`, utilisable dans le fichier CSS, important pour le TP :

- **height** : Permet de choisir la taille de l'image. Les valeurs peuvent se donner, comme habituellement, en px, cm ou %.
- **vertical-align** : Permet d'aligner l'image verticalement. Les valeurs permises sont `baseline`, `sub`, `super`, `top`, `text-top`, `middle`, `bottom`, `text-bottom`, numérique (positive ou négative suivie de px ou % ou ex ou em).
- **border x y z** : Permet de créer un cadre pour l'image. x est la taille du cadre (`border-width`), y, le style du cadre (`border-style`, les valeurs possibles sont `none`, `hidden`, `dotted`, `dashed`, `solid`, `double`, `groove`, `ridge`, `inset`, `outset`, ...). Et, en n, z est la couleur du cadre.

Exercice 11 (Pour aller plus loin : Fond d'écran)

Télécharger l'image http://www.lirmm.fr/~cherubini/GLSE301/TP/TP2/fond_ecran.jpg. Rajouter l'image en fond d'écran du formulaire. Centrer l'image. Voici un exemple pour vous aider <http://css.mammouthland.net/image-de-fond-background-css.php>.

Exercice 12 (Pour aller plus loin : Inclure une image)

Télécharger l'image <http://www.lirmm.fr/~cherubini/GLSE301/TP/TP2/cafe.gif>
Rajouter l'image après l'avant dernier paragraphe. A l'aide de la feuille de style, réduire la taille de l'image et rajouter un cadre

Structure d'une page web

La structure d'une page web correspond à la disposition de différents blocs à l'intérieur de la page elle-même. La Figure 1 présente la structure de la page à réaliser dans ce TP. La page html sera divisée en 4 blocs, chaque bloc correspondra au tag **div** dans le fichier html.

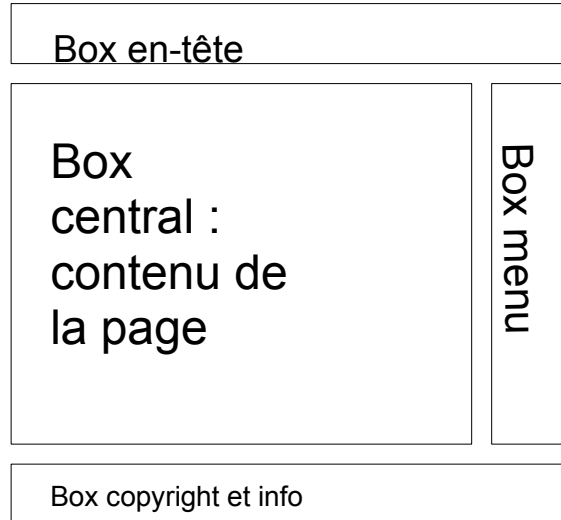


Figure 1

Il faut réaliser 2 fichiers :

- index.html : utilisation du tag div pour créer le 4 blocs;
- style.css : nécessaire pour placer correctement le 4 blocs.

Attention:

1. On utilisera des valeurs en pourcentage pour spécifier les mesures des différents blocs afin d'obtenir une page web adaptable à toutes les résolutions video.
2. Pour obtenir la division en blocs on utilisera que le tag div (pas de tableau pour structurer la page)
3. Le fichier de style ne contient pas les spécifications pour le texte (police) et la couleur.

Exercice 1

A vous de créer le fichier CSS qui aura une déclaration pour chaque bloc de la page comme décrit par la suite.

```
#entete {  
    margin: 0;  
    padding: 10px 10px 10px 10px;  
    border-bottom: 1px solid #cccccc;  
}
```

Le bloc en-tête aura une longueur correspondant à toute la page sans marge (margin 0). Le contenu de ce bloc aura un espacement de 10 px. Enfin, le bloc aura une bordure en bas.

```
#contenu {  
    float:left;  
    width: 75%;  
    margin: 0;  
    padding: 0 3% 0 10px;  
}
```

Le bloc contenu correspondra au 75% de la page (en largeur) et il sera placé à gauche.

```
#menu {  
    margin: 0 0 0 79%;  
    padding: 0px;  
    background-color: #eeeeee;  
    border-left: 1px solid #ccc;  
    border-bottom: 1px solid #ccc;  
}
```

Le bloc menu aura une couleur différente et une bordure à gauche et en bas. Le paramètre margin spécifie que ce bloc sera décalé à droite de 79% (ou si vous voulez la marge gauche est de 79%). La valeur 79% dépend du fait que le bloc *contenu* correspond au 75% de la page plus le padding de 3% (total de 78%, donc il faut que le bloc menu soit décalé d'au moins 79%).

Enfin la description du dernier bloc.

```
#copyright {  
    clear: both;  
    border: 1px solid #cccccc;  
    padding: 10px 10px 10px 10px;  
    margin-top: 0px;  
}
```

Maintenant, il ne vous reste plus qu'à créer le fichier html pour tester votre page web. Le fichier html contiendra un tag div pour chaque bloc. Il vous faudra spécifier le type de chaque tag div pour appliquer le formatage décrit dans le fichier CSS (paramètre **id** à utiliser à l'intérieur du tag div).

Exercice 2

Maintenant à vous de définir une structure d'une page web et ensuite de créer les fichiers html et CSS associés.