

Devoir maison n°2

Balises en HTML et manipulations vidéo

1 Contexte et objectifs

La première partie de ce devoir consiste à se familiariser avec les balises en HTML. À la fin de ce devoir, vous aurez assez d'éléments pour poursuivre sur des concepts élaborés en HTML pour mettre du contenu sur un site web. Le style sera abordé un peu plus tard avec le langage CSS.

La suite de ce devoir consiste à faire des manipulations simples sur les 3 logiciels installés précédemment : Avidemux, Aegisub et HandBrake. À l'issue de ce devoir, vous devrez être capable de faire des découpages de passages d'une vidéo, de créer des sous-titres de manière simple ainsi que de les incruster dans une vidéo.

1.1 Consignes du devoir

Vous devez créer **un fichier HTML par exercice** (exercice1.html, exercice2.html, etc). Ce travail est à rendre sous forme d'archive zip via un service de partage de fichiers (Dropbox, Skydrive, etc.) à l'adresse the-van.luong@heig-vd.ch avant le lundi 14 octobre. Tout comme le premier devoir, afin de faciliter la correction, il est important que le dossier contenant vos fichiers ainsi que l'archive zip associé soient de la forme suivante : NOM Prénom. Et non pas l'inverse. Cela facilitant le tri de vos devoirs.

Ce devoir étant assez fondamental, il est important que le rendu soit **individuel** pour se familiariser avec les concepts de balises. Il n'est pas interdit bien sûr de travailler ensemble avec d'autres camarades, mais veuillez rendre **votre propre code** et **ne pas échanger** vos fichiers.

En ce qui concerne votre note de participation finale, vous serez pénalisés 1) si votre code dans les devoirs maison contient les mêmes erreurs syntaxiques qu'un de vos camarades (facilement détectables par un programme) ; 2) si vous ralentissez la classe lors des prochaines séances de cours par rapport à des notions en HTML sensées être largement acquises.

Votre enseignant reste bien sûr disponible par mail pour toute question concernant l'HTML et surtout pour les différentes manipulations vidéo (non abordées en cours).

2 HTML

2.1 À propos de HTML

Le HTML (Hypertext Markup Language) est le **langage universel** utilisé pour communiquer sur le web. Il s'agit d'un langage constitué **de balises** conçu pour écrire des pages web.

Le langage HTML est omniprésent sur le web. Le navigateur internet **interprète les balises HTML** de chaque page web pour retransmettre les informations sous formes visuelles.

L'avantage de l'HTML est que ce langage universel s'adapte à toutes les systèmes d'exploitations que ce soit Windows, Mac ou Linux ainsi qu'à tous les types d'ordinateurs (PC, smartphone, etc.). Et il s'intègre très bien les environnements de développement de sites web.

La programmation web n'étant pas à la portée de tout le monde, le fait d'avoir des connaissances dans ce paradigme est une compétence recherchée. Le métier de développeur web figure en effet

parmi les métiers en informatique les plus demandés dans l'industrie.

2.2 Exercice 1 : Pour s'échauffer

1. Créer un document HTML minimal avec les balises de document (« html », « head » et « body »).
2. Ajouter un titre au document avec la balise « title ».
3. Ajouter une phrase dans le corps du document (contenu).

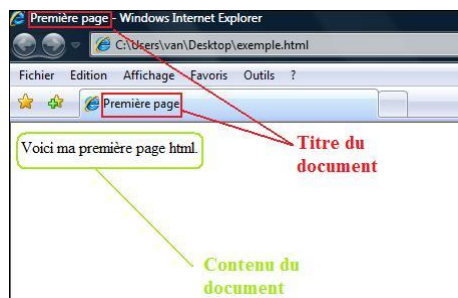


FIGURE 1 – Document HTML minimal

Note : Tous les documents HTML respectant les standards devraient avoir ce contenu minimal. Essayer de respecter **scrupuleusement l'ordre d'apparition** des ouvertures et des fermetures des balises dans toutes vos pages html.

2.3 Exercice 2 : Révisions

1. Créer un document HTML minimal avec les balises de document.
2. Montrer un exemple d'utilisation d'une phrase avec du **gras**.
3. Montrer un exemple d'utilisation d'une phrase avec de *l'italique*.
4. Montrer un exemple d'utilisation d'une phrase avec du souligné.
5. Montrer un exemple d'utilisation d'une phrase contenant à la fois du **gras**, de *l'italique* et du souligné (c'est-à-dire il faut que la phrase soit à la fois en gras, en italique et en souligné).
6. Montrer un exemple d'utilisation pour aller à la ligne.
7. Montrer un exemple d'utilisation pour sauter une ligne.

Note : On rappelle que certaines balises contiennent du texte entre leur ouverture (<balise1>) et leur fermeture (</balise1>), et d'autres n'en contiennent pas (<balise2 />).

2.4 Exercice 3 : Titres dans le document

1. Créer un document HTML minimal avec les balises de document.
2. Montrer un exemple **détailé et pertinent** d'utilisation des balises de titres similaire à l'exemple ci-dessous.

2.5 Exercice 4 : Balises supplémentaires de forme

Créer un document HTML minimal avec les balises de document. On voudrait essayer de nouvelles balises supplémentaires. Donner un exemple d'utilisation **pertinent** de chacune des balises ci-dessous. Il ne se suffit pas juste « d'écrire les balises telles quelles » mais de trouver des bons exemples qui illustrent vraiment visuellement chaque balise.

Activities

Year 2007

Distributed metaheuristics for the Quadratic Assignment Problem (in progress)

ProActive

ProActive provides programming models such as Master/Worker to simplify the programming of distributed applications. Many features such as load-balancing, fault-tolerance, security and mobility are managed by ProActive.

Distributed methods

Thanks to parallelism and distribution, we could reduce computational time, solve bigger instances, and improve the quality of the solutions.

Ant Colony Optimization for the Traveling Salesman Problem

This work is a study of Ant Colony Optimization, and has allowed me to discover original and powerful optimization methods.

Project

One goal of this internship is also to study the advantages obtained by using distribution models on multi CPUs, instead of pure sequential algorithms. For doing this, we will use more than 60 computing experiments with the ProActive Library.

Year 2006

A Fast Hybrid Genetic Algorithm for the Quadratic Assignment Problem

This work has allowed me to improve my knowledge about metaheuristics (Robust Tabu Search from Taillard), and I have also learned to use an Evolutionary Computation Framework (ParadisEO).

FIGURE 2 – Un exemple d'utilisation de balises de titres

<code><big>...</big></code>	Taille du texte plus grande
<code><small>...</small></code>	Taille du texte plus petite
<code><strike>...</strike></code>	Texte rayé
<code><tt>...</tt></code>	Texte formaté genre courrier
<code>...</code>	Texte plus gras
<code><sup>...</sup></code>	Place le texte en exposant
<code><sub>...</sub></code>	Place le texte en indice
<code><center>...</center></code>	Centre le texte dans la page

En plus des balises précédentes, trouver un exemple d'utilisation **d'une balise de votre choix qui n'est pas dans le cours**. Chercher sur internet une telle balise et indiquer sur quels navigateurs web (au moins 2) elle a fonctionné.

2.6 Exercice 5 : Balises avec attributs

Les balises avec attributs représentent un premier concept évolué au niveau du langage HTML. On rappelle que certaines balises peuvent contenir des attributs facultatifs ou obligatoires dans leur ouverture (`<balise attribut1=".." attribut2=".." etc.> ... </balise>`).

1. Ajouter une image de votre choix dans le document.
2. Soit la balise « font » avec l'attribut « color ». Montrer par quelques exemples comment on peut changer la couleur de certaines phrases.

2.7 Couleurs RGB

En utilisant le code **hexadécimal** des couleurs,

- Changer la couleur du texte par défaut de la page HTML.
- Changer la couleur de l'arrière-plan par défaut de la page HTML.
- Donner quelques exemples de traits horizontaux en couleurs de différentes épaisseurs et largeurs.

2.8 Exercice 6 : Balises emboîtées

2.8.1 Liste numérotée et caractères spéciaux

Dans le cours, on a pu voir un exemple de balises emboîtées qui est la notion de liste à puce « ul » dans laquelle chaque élément est représenté par la balise « li ». On voudrait maintenant utiliser la notion de liste numérotée « ol ». Créer une liste numérotée (« ol ») avec 5 éléments qui respecte les contraintes sur la numérotation :

1. La numérotation de la liste doit contenir des lettres minuscules.
2. La numérotation de la liste doit commencer à partir de « c ».
3. Chaque élément devra contenir un exemple de phrase dans lequel vous devrez insérer un caractère spécial représenté **en ANSI, unicode ou UTF-8**. (5 phrases = au moins 5 caractères différents).

2.8.2 Listes emboîtées

On voudrait utiliser simultanément les notions de listes à puce et de listes numérotées. Voici un exemple particulier de listes emboîtées en latex/pdf :

1. Algorithmes à population de solutions pour l'optimisation combinatoire
 - Algorithmes génétiques
 - Colonies de fourmis
 - Essaim particulaire
2. Algorithmes à solution unique pour l'optimisation combinatoire
 - Recuit simulé
 - Recherche tabou
3. Méthodes exactes

Donner votre propre exemple de listes emboîtées en HTML (« ol » et « ul »).

3 Manipulations vidéo : découpage vidéo, sous-titrage, conversion et incrustation

Dans l'exercice 7, on se propose de faire les manipulations vidéo de base demandées par l'équipe pédagogique du Master 2 TSD. Pour cela, nous allons travailler sur les 3 logiciels libres que vous précédemment installés. Toutes les vidéos peuvent normalement s'ouvrir dans chacun des logiciels avec un simple « drag-and-drop ».

L'objectif de l'exercice est de travailler à partir d'une vidéo brute, de faire des découpages vidéo pour en tirer un extrait, de créer des sous-titres d'un passage donné et de les incruster directement dans une vidéo.

Pour cela, vous devrez expliquer vos manipulations avec des mini tutoriels en HTML à chaque étape :

- Courte description des manipulations effectuées : juste avec du texte.
- Faire une seule capture d'écran à chaque manipulation pour donner une bonne vue d'ensemble (4 captures au total).
- Afin ne pas alourdir votre archive zip, il n'est pas demandé de joindre les vidéos dans votre travail.
- Tous les tutoriels devront être faits en HTML.

3.1 Découpage vidéo : Avidemux

Avidemux est un logiciel libre de montage vidéo permettant d'éditer, d'encoder et de convertir une vidéo dans différents formats populaires : AVI, MP4, MKV, etc.

On se propose de travailler avec la vidéo *Film.mkv* téléchargeable depuis la page web du cours. L'objectif souhaité est de découper la vidéo à différents endroits afin de sélectionner un passage qui nous intéressera pour un futur sous-titrage.

Indiquer les étapes nécessaires pour créer une nouvelle vidéo *Extrait.mkv* qui sera un nouvel extrait (par exemple entre 01 : 13 et 03 : 24) de la vidéo fournie ci-dessus.

Indications :

- Il faudra se focaliser à placer les marqueurs A et B ainsi que sur le menu éditer.
- On ne cherche pas ici à faire de la compression de données ou du redimensionnement. Ainsi, les sorties vidéo et audio devront être laissés en mode copy. Il en est de même pour le format de sortie qu'on laissera également au format MKV.

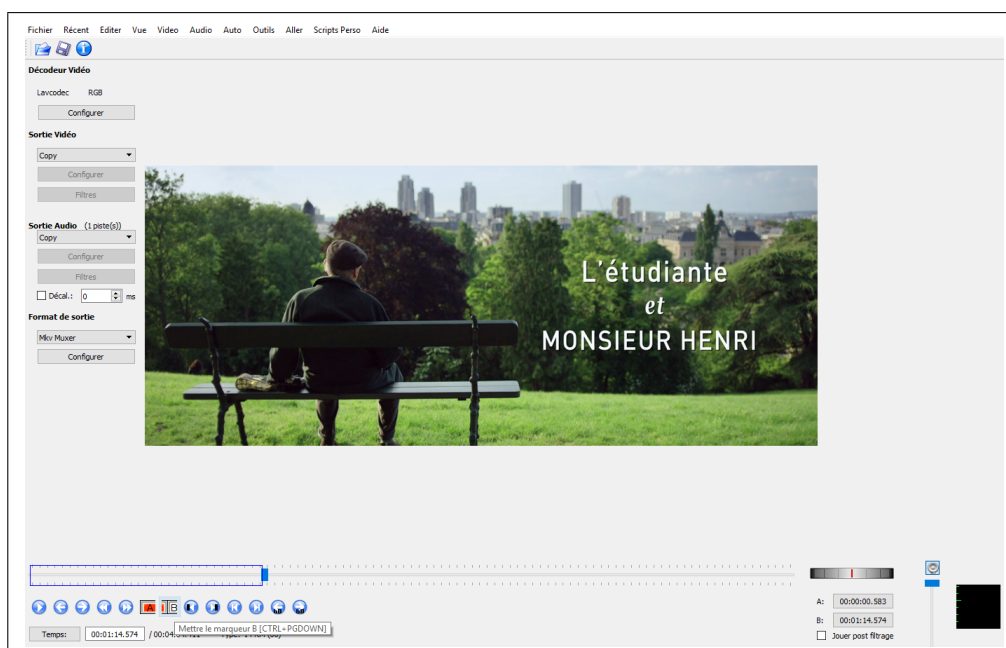


FIGURE 3 – Dans Avidemux, on se concentrera sur le placement des marqueurs A et marqueurs B pour effacer/couper les parties à supprimer.

3.2 Création de sous-titres : Aegisub

Aegisub est un logiciel libre permettant la création et l'édition des sous-titres. Le format de fichier produit est le ASS (Advanced SubStation Alpha) qui permet de formater et styliser le texte utilisé dans les sous-titres.

On se propose de travailler sur l'extrait vidéo *Extrait.mkv* que vous avez précédemment créé avec Avidemux. On voudrait créer une ou deux lignes de sous-titres dans une des répliques de Monsieur Henri.

Indiquer les étapes nécessaires pour réaliser cela. Votre fichier créé devra s'appeler *Extrait.ass*.

Lorsque tout est terminé, pour tester votre vidéo avec vos sous-titres, vous devez avoir un lecteur qui prenne en charge le format ASS (ce n'est pas le cas de Windows Media Player). Par exemple, VLC sous Windows prend en charge ce format (la version Mac OS nécessitant l'installation de bibliothèques supplémentaires).

Indications :

- L'interface d'Aegisub se décompose en trois cadres : la prévisualisation vidéo (partie gauche), la piste audio et l'édition (partie droite) ainsi que la grille de lignes de sous-titres (partie en bas).
- La première étape consiste à cliquer sur la piste audio pour trouver le passage qui nous intéresse (apparition de délimiteurs rouge et bleu). On pourra ensuite passer à la création/édition de lignes de sous-titres.
- On peut utiliser les zooms verticaux et horizontaux (avec par exemple le mode analyse de spectre) pour délimiter plus facilement les passages voulus.
- Lorsque vous écrivez du texte, utiliser `\N` pour passer à la ligne si besoin.

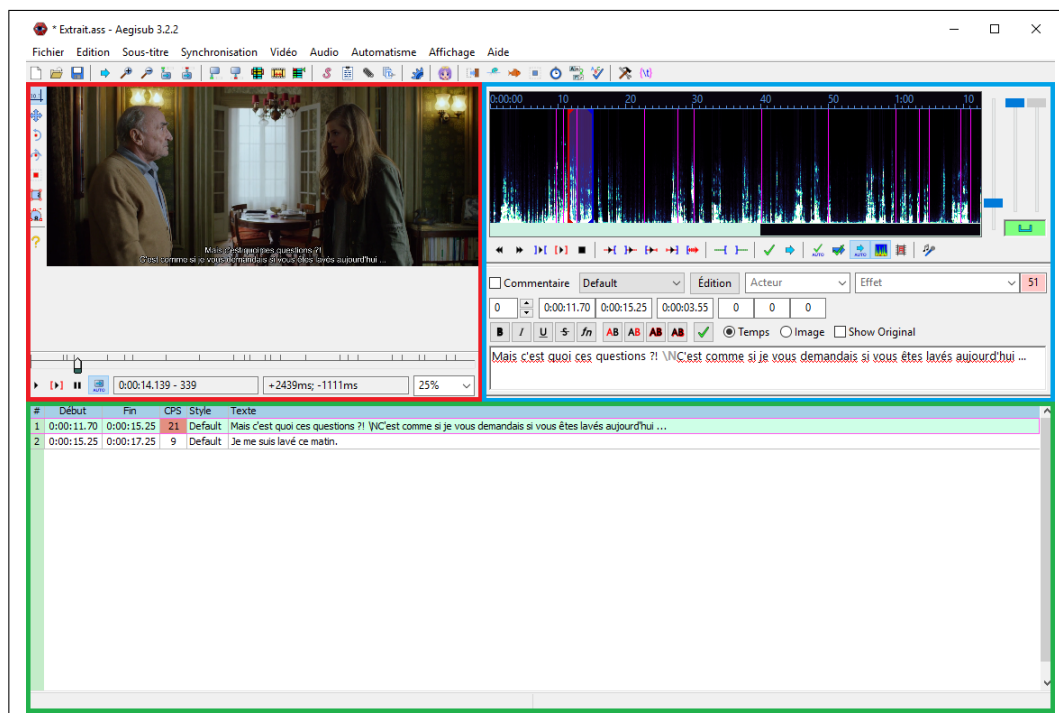


FIGURE 4 – Dans Aegisub, on se concentrera tout d'abord sur la partie audio et édition (cadre à droite).

3.3 Conversion format ASS vers SRT

Le format SRT (SubRip text) est le format le plus simple et le plus répandu pour les sous-titres. Il est largement utilisé dans la majorité des vidéos. Ce format n'inclut cependant aucune donnée spécifique au format de la vidéo. Il s'agit juste d'un fichier de texte qui peut se créer ou s'éditer à partir de n'importe quel éditeur de texte comme Notepad.

Par défaut, la sauvegarde de base d'Aegisub est au format ASS. Pour utiliser un logiciel comme HandBrake ou un lecteur comme Windows Media Player (ou logiciels intégrés à la télévision), il faudra ainsi penser à exporter et choisir le format SRT. Une autre solution consiste à convertir vos fichiers `.ass` en `.srt` grâce à un convertisseur en ligne (google -> convert ass to srt).

Montrer comment vous avez réussi pour obtenir au format SRT.

Ensuite, vous pouvez ouvrir votre fichier `.srt` produit avec un éditeur de texte comme Notepad pour vérifier que les caractères accentués et spéciaux soient pris en compte. Faites une capture d'écran de cette étape.

Vérifiez que vos nouveaux sous-titres du `Extrait.srt` fonctionnent avec n'importe quel autre

lecteur vidéo lorsque vous ouvrez la vidéo *Extrait.mkv*. Sur certains lecteurs comme Windows Media Player, il est nécessaire d'aller dans les menus pour sélectionner manuellement le fichier de sous-titres SRT.

Indication : si vous rencontrez des problèmes au niveau des caractères accentués ou spéciaux, assurez-vous avec votre éditeur de texte d'enregistrer votre fichier pour que l'encodage soit au format UTF-8.

3.4 Incrustation de sous-titres : HandBrake

HandBrake est un logiciel libre utilisé pour transcoder une source vidéo. Il peut servir notamment à extraire un film à partir d'un Blu-Ray vers du MP4 ou MKV. Tout comme Avidemux, un nombre de presets est installé pour compresser la vidéo. Une autre fonctionnalité que le logiciel propose est sur l'incrustation de sous-titres dans une vidéo (c'est-à-dire que les sous-titres sont directement attachés aux images de la vidéo).

On se propose ainsi d'incruster les sous-titres du fichier *Extrait.srt* à la vidéo *Extrait.mkv* de telle façon à produire une nouvelle vidéo (par exemple *Extrait2.mkv*). Indiquer ce qu'il faut faire pour cela. Libre à vous d'utiliser un preset pour compresser la vidéo (par exemple General Fast 1080p30 ou bien Legacy High Profile pour peu de compression).

Indications :

- On a besoin d'indiquer l'emplacement et le nom du nouveau fichier (destination) à générer (Save As).
- On souhaite dans cet exercice conserver du MKV et non du MP4.
- L'onglet subtitles est celui que vous devez utiliser pour utiliser votre fichier SRT.
- N'oubliez pas de mettre l'encodage UTF-8 et de cocher l'option Burn in.

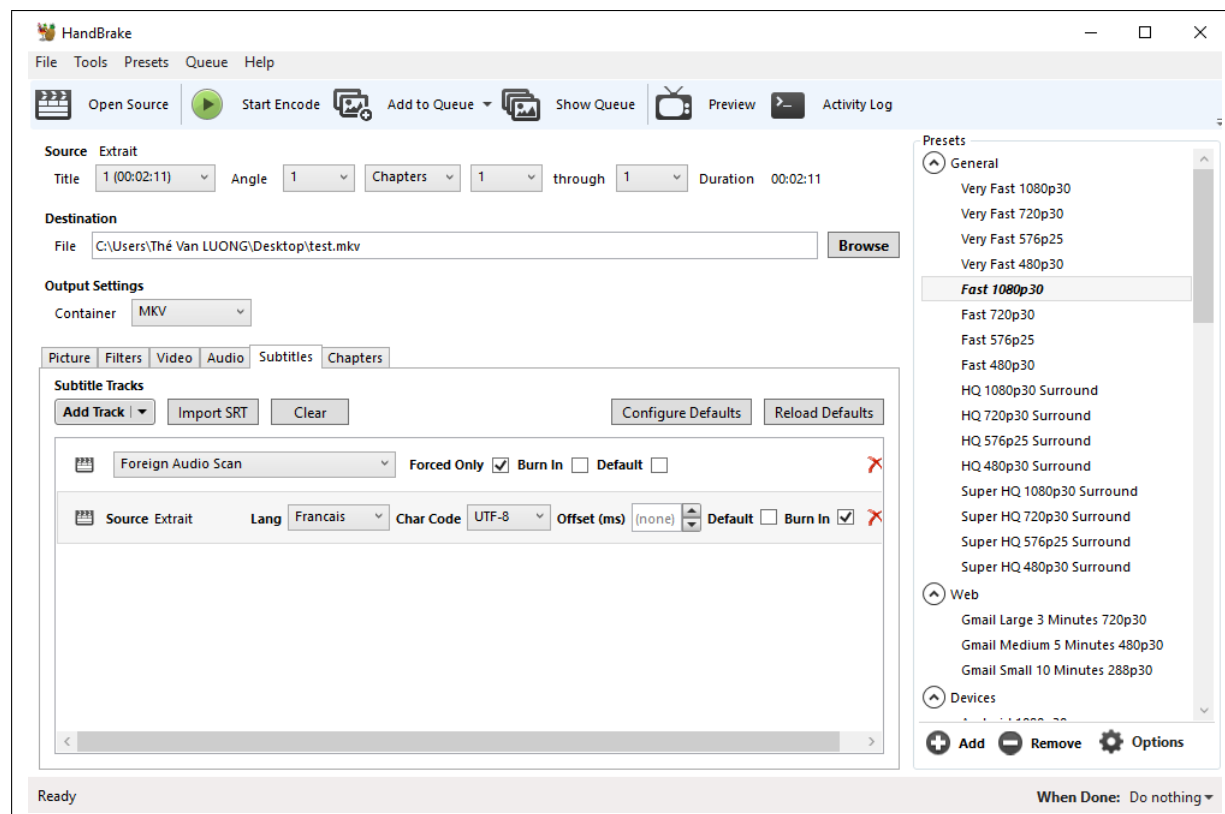


FIGURE 5 – Dans HandBrake, on se concentrera sur l'onglet subtitles pour travailler sur l'incrustation de sous-titres dans la vidéo.