

SAE203 – Intégration Web – S2 – TD1

Objectifs et contraintes : développer le site Web côté client en HTML/CSS/JS.

- Le style est donné (fichier **sae203.css** dans l'archive) et n'est pas modifiable. Vous pourrez modifier le style lorsque toutes les fonctionnalités côté client seront réalisées (si il vous reste du temps), dans ce cas, les modifications seront définies dans un ou plusieurs fichiers CSS supplémentaires à celui fourni.
- Le code JS général au site doit être écrit dans le fichier **sae203.js**, du code spécifique à un document sera écrit en interne.
- La structure des documents, le choix des balises ainsi que le identifiants et les classes, doivent être respectés pour que le style puisse être appliqué et pour que le JS puisse s'exécuter correctement.

Ce 1^{er} TD permet de prendre en main l'architecture du site, la structure de ses pages, l'application du style avec la feuille de style fournie, la programmation du comportement du menu (et sous-menus), et la génération de contenus (téléphone, email, nom, date).

1) Architecture, structure et style

L'architecture du site est simple, celle de l'archive, elle contient :

- À la racine : index.html et les répertoires suivants
- css : fichiers CSS
- js : fichiers JS
- affichage : pages Web pour la rubrique Affichage
- insertion : pages Web pour la rubrique Insertion
- documents : contient des documents (énoncés, ...)

Cette architecture doit être respectée et devra être complétée avec vos scripts Php. Le nom des répertoires et des fichiers doit être respecté. Toutes les pages demandées ne sont pas encore créées, vous le ferez au fur et à mesure des séances de TD/TP/PT, à la fois dans la partie intégration et dans celle dédiée au développement côté serveur.

Conseil : sauvegardez une version de votre site complet (avec la date dans le nom du répertoire principal) à chaque séance.

Afin d'illustrer la structure et le style fixés, nous considérons les pages index.html et affichage/films.html. Toutes les pages doivent avoir la même structure globale :

- header contenant le menu nav
- main contenant la partie principale de la page
 - un titre h1 (la classe visible permet de l'afficher et la classe invisible permet de ne pas le rendre si nécessaire).
 - une ou plusieurs sections
- footer contenant la liste des pages, les contacts, et le copyright.

Le style dépend de cette architecture, il faut donc la respecter. Remarquez que le script est inclus dans le header.

Prenez quelques minutes pour lire les fichiers HTML index.html et affichage/films.html, et le fichier de style css/sae203.css. Observez les différences entre les fichiers, dans le

menu (la classe), dans le footer (la classe). Nous utiliserons souvent les classes pour décrire le comportement visuel en CSS et changer la classe d'éléments d'une page à l'autre, ou dans une même page en fonction des événements produits par l'utilisateur. Dans ce TD nous ne modifions pas le CSS ni le HTML, nous nous concentrons sur le JS.

2) Première utilisation du JS – Ajout du nom de l'auteur

Considérons le fichier index.html. Le nom de l'auteur est précisé dans la balise meta (dans le head). Remplacez le contenu Monom par votre nom, et de même pour le copyright. Cela ne change rien si vous rechargez la page, pour rappel cette information est destinée au référencement, robots.

Remarquez que le texte monom est aussi affiché dans le contenu du main (en bas à droite) et dans le copyright du footer, avec des balise de classe monom. L'objectif est de récupérer, en JS, le nom renseigné dans la balise meta auteur et de l'utiliser pour remplacer le contenu des balises de classe monom. Programmez cela dans le document js/sae203.js dans la fonction monom(). Remarquez que cette fonction est déjà appelée dans l'écouteur sur l'événement DOMContentLoaded. Rechargez la page avec la console Web ouverte (en cas d'erreur) pour observez le résultat.

3) Footer

a) Ajout de la date

Dans le footer, la date est précisée dans la propriété datetime de la balise time. Selon le même principe qu'à l'exercice 2, dans la fonction footer(), remplacez le contenu de time (initialement le texte date) par la date précisée par la propriété datetime. Pour cela vousinstancierez un objet [Date](#) et utiliserez sa méthode [toLocaleString\(\)](#) pour afficher la date automatiquement au format du client (pour nous en français jj/mm/aaaa).

b) Génération du numéro de téléphone

Observez que le numéro de téléphone n'est pour l'instant pas affiché ni contenu en dur dans le footer (balise a avec tel dans l'URL) pour éviter les spams (on peut faire mieux en ajoutant un captcha par exemple, nous verrons cela plus tard). Nous considérons le code suivant :

```
function footer() {  
  
    // pour le numéro de téléphone  
    const telNum = [2,3,0,5,0,0,3,0]; // le vrai tel est +33 2 33 33 50 00  
    const telId = [1,6,6,1,3,5,7,2]; // positions dans telNum pour obtenir le numéro  
  
    // récupérer le nœud a du téléphone  
    let telNode = document.querySelector('footer>address>a:first-of-type');  
    let txt = "+332";  
  
    // reconstruire le numéro à partir des tableaux  
    for (let i = 0; i < telId.length; i++) txt += telNum[telId[i]];  
  
    // écraser la valeur de l'attribut et le contenu de la balise a  
    telNode.setAttribute('href',"tel:"+txt);  
    telNode.innerText = "Phone: "+txt;  
}
```

Il utilise deux tableaux pour sauvegarder le numéro, le tableau telNum contient les chiffres dans le désordre et le tableau telId contient les indices des chiffres dans le tableau telNum dans le bon ordre. La boucle permet de reconstruire le numéro en parcourant le tableau telNum dans l'ordre défini par le tableau telId. Le reste du code permet de mettre à jour les données dans le document HTML. Recopiez ce code dans la fonction footer() et observez le résultat en rechargeant la page dans le navigateur.

c) Génération de l'email

Selon le même principe qu'à la question b, complétez la fonction footer() pour générer l'email (propriété et contenu de la balise a).

4) Menu

Pour l'instant le menu n'est pas fonctionnel (excepté pour l'accueil), remarquez que les sous-menu ne sont pas affichés bien qu'ils soient présents dans le document HTML. Le document CSS css/sae203.css précise display: none pour les balises ul des sous-menus. Nous souhaitons que lorsque l'utilisateur clique sur une des deux rubriques (une balise a de classe rubrique), le sous-menu s'affiche si il ne n'est pas déjà (propriété display: block) ou disparaît si il est déjà affiché. L'affichage est contrôlé par la classe visible dont le code CSS associé est le suivant dans sae203.css :

```
.visible { display: block; }
```

Il suffit donc d'ajouter un écouteur sur l'événement click des balises de classe rubrique. Cet écouteur ajoute la classe visible à l'élément cliqué si n'est pas déjà de classe visible ou supprime la classe si il l'est déjà. La méthode [toggleClass\(\)](#), de la liste des classes associée à un élément, permet de réaliser ce comportement. Programmez ce comportement dans la fonction menu(). Vous ne devez pas oublier de :

- inhiber le comportement par défaut du clic avec [preventDefault\(\)](#).
- Refermer un sous-menu déjà ouvert si l'utilisateur demande à en ouvrir un autre. Utilisez la méthode [contains\(\)](#) de la liste de classes.
- Refermer un sous-menu ouvert si la souris de l'utilisateur quitte la boîte associée au sous-menu incluant le titre de la rubrique (donc l'élément li de la rubrique), et donc sur l'événement [mouseleave](#).

5) URL de base

Ouvrez maintenant le fichier affichage/films.html. Remarquez la balise base dans le head. Elle permet de définir l'URL de base pour tous les liens relatifs de la page. Ceci permet d'éviter d'avoir à modifier les URL pour les liens dans les différentes pages, en particulier le menu ou encore la liste (fil d'Ariane) dans le footer, puisque tous les fichiers ne sont pas dans les mêmes répertoires. Prenez quelques minutes pour lire la documentation de la balise [base](#).

6) Autres éléments du menu et pages

Complétez le menu et créez les pages associées en modifiant, dans chaque nouvelle page les classes dans le menu et le footer.