

## Étude des langages de sérialisation

### Partie 2

#### Introduction

- Pour les exercices faisant appel aux fonctionnalités **HTML5**, vous aurez besoin d'utiliser alternativement les navigateurs *Firefox* et *Chromium* qui ne proposent pas le même support des éléments.
- Le test disponible à l'adresse suivante vous permettra de connaître rapidement les fonctionnalités HTML5 supportées par un navigateur donné. <https://html5test.com/results/other.html>
- Note : le sujet du mini-projet est introduit après les exercices.

#### Exercice 1 – Un peu de HTML5 : formulaire

Pour cet exercice, aucun code *JavaScript* est nécessaire. Référez-vous au site de référence *w3schools* pour trouver les fonctionnalités nécessaires.

**Question 1.1 :** Créez un document minimaliste en *HTML5* (Vérifiez avec le validateur avant de continuer qu'il est bien reconnu avec un doctype HTML5).

**Question 1.2 :** Ajoutez un formulaire avec une adresse cible quelconque, en mode *GET* (pour contrôler facilement les données envoyées), et contenant les champs *nom*, *date de naissance*, *couleur préférées*, *animal préféré*, *adresse email* et un bouton *envoyer*. On souhaite que :

- *date de naissance* offre un calendrier de sélection de date (ne fonctionne que sous *Chromium*),
- *la couleur* puisse être choisie avec un *ColorPicker* (*Chromium* seulement),
- Une adresse email ou une url invalide bloque l'envoi du formulaire.
- le champs *animal préféré* permette de saisir un texte quelconque mais offre une liste de valeurs prédéfinies {Chien, Chat, Léopard, Mr Spock} lorsque l'on appuie sur la touche *flèche vers le bas*

**Question 1.3 :** Groupez *date de naissance*, *couleur préférées* et *animal préféré* dans un panneau pliable déplié par défaut (Recherchez l'élément **details**).

**Question 1.4 :** Est-il possible d'envoyer votre formulaire avec des champs vides ? Remédiez à cela en rendant les champs *nom* et *email* obligatoires.

**Question 1.5 :** Passez votre code au validateur.

#### Exercice 2 – HTML5 – Un éditeur simple

**Question 2.1 :** Créez un document minimaliste en *HTML5*.

**Question 2.2 :** Ajoutez un paragraphe contenant le texte de votre choix et rendez le éditable en activant la correction d'orthographe.

#### Exercice 3 – HTML5 – MathML et SVG

**Question 3.1 :** Créez un document minimaliste en *HTML5*.

**Question 3.2 :** Incorporez les codes MathML et SVG de l'exercice *Extensibilité de XHTML*.

**Question 3.3 :** Cela fonctionne-t-il si vous retirez les espaces de noms ? Passez votre code au validateur.

## 1 Présentation du sujet de mini projet : SciMS

Le mini projet consiste à réaliser un petit *CMS* (*Content Management Software*) que l'on peut assimiler à un *blog* multi-utilisateur. Les articles seront à caractère scientifique c'est à dire que leur style de présentation se situera entre l'article web et l'article scientifique, et l'on souhaitera y retrouver les fonctionnalités dont l'on dispose en LaTeX, comme par exemple les références automatiques à des éléments (figures, sections, ...) ou bibliographiques, la numérotation automatique.

Ces fonctionnalités devront être réparties de manières appropriées sur les différents paradigmes (structuration, feuilles de styles, Javascript, ...)

Un article sera structuré de la manière suivante :

- un article possède un titre, un auteur (utilisateur de la plate-forme), une date de publication, une catégorie de classement ;
- un article possède un résumé (*abstract*), ainsi qu'une liste de mot-clés (*keywords*) en général affiché au début ;
- un article sera structuré en section, sous-section ;
- le contenu des (sous)section sera divisé en paragraphes, figures, formule mathématique
- Les figures pourront être des images bitmap ou du contenu vectoriel (SVG) ; une figure dispose d'un numéro pour les références et d'une légende qui seront affichés en dessous.
- Une formule mathématique sera numérotée (voir fichier d'exemple, page 3) pour y faire également référence ; Les formules seront saisies en LaTeX et automatiquement transformées en MathML. Une solution sera proposée plus tard.
- Un article se termine par une liste de références (voir fichier d'exemple, page 20).
- L'auteur pourra éventuellement décider d'incorporer un sommaire à une position de son choix qui affichera la structure des sections/sous-sections et permettra d'accéder directement à ces dernières en cliquant sur leur nom.

L'interface principale de l'application sera divisée en plusieurs parties :

- un bandeau supérieur éventuellement personnalisable ;
- la partie principale où seront affichés les articles, l'éditeur d'article ou bien les catégories et leur contenu, etc. ;
- un menu vertical à gauche permettant de naviguer dans les catégories d'articles, de se logger, d'accéder aux fonctionnalités d'un utilisateur authentifié.

Les catégories servent à classer les articles (exemple : informatique, mathématiques, physique, chimie, ...).

## 2 Réalisation

- Le sujet vous est communiqué maintenant afin que vous puissiez commencer progressivement sa réalisation au fur et à mesure des thèmes abordés. Évitez de développer des parties avant leur introduction en cours.
- Du temps pourra y être consacré en TP lorsque les exercices sont terminés.
- Le travail sera à rendre **avant** les vacances de fin d'année et pourra être réalisé en binôme **maximum**.

## 3 Tâches réalisables dans le cadre du présent TP

Les propositions suivantes consistent en l'écriture de résultats statiques en HTML (aucune programmation ni mise en forme).

### Exercice 4 – Un premier article

**Question 4.1 :** Récupérez l'exemple d'article (`Exemple d'article.pdf`) dans le dossier `mini-projet` de l'archive et utilisez son contenu pour créer la structure d'un premier article en HTML. (Au préalable, choisissez la version du langage que vous comptez utiliser dans votre projet).

**Question 4.2 :** Ne reproduisez pas toutes les formules mathématiques. Quelques-une suffiront pour donner une idée (on pourra même reprendre les formules MathML du TP précédent).

**Question 4.3 :** Vous pourrez extraire les figures en utilisant l'outil *Inkscape* et en les sauvegardant en SVG.

**Question 4.4 :** Pensez à passer le résultat au validateur.

### **Exercice 5 – Quelques menus**

Dans des fichiers séparés (le code pourra être intégré plus tard)

**Question 5.1 :** Créez une interface de login.

**Question 5.2 :** Créez une interface pour ouvrir un compte (Nom, prénom, email, mot-de-passe, date de naissance, etc.)

**Question 5.3 :** Créez un squelette votre menu latéral gauche.

**Question 5.4 :** Créez une page d'accueil affichant la liste des catégories et sous chacune d'elle le titre des derniers articles publiés dans la catégorie en question (5 maximum).

— L'esthétique importe peu pour l'instant, évitez d'utiliser des styles. Si toutefois vous vouliez effectuer un peu de mise en forme, assurez-vous que vos feuilles de styles seront ré-utilisables.

### **Exercice 6 – Pistes à explorer/étudier**

Quelques pistes, si vous choisissez HTML5 :

**Question 6.1 :** Pensez aux éléments structurants dédiés aux blogs évoqués en cours.

**Question 6.2 :** Envisager les *web-component* avec les templates et balises personnalisées.

---

## EXERCICES BONUS

---

### Exercice 7 – Création d'un tableau

**Question 7.1 :** Réalisez le tableau ci-dessous, sachant que

- la totalité du texte montré sur la figure fait parti du tableau,
- exceptionnellement, nous activerons les bordures pour nous aider (`border="1"`).

Mes livres			
ISBN	Description		
	Prix	Pages	Titre
978-0596157609	25,81 euros	352	HTML & CSS: The Good Parts
978-0470523964	29,48 euros	792	HTML, XHTML, and CSS Bible
Total	35 euros		

**Question 7.2 :** Passez votre code au validateur et corrigez toutes les erreurs.

**Question 7.3 :** Incorporez le fichier de style `tableau.css` et vérifiez que vous obtenez bien le résultat ci-dessous :

Mes livres			
ISBN	Description		
	Prix	Pages	Titre
978-0596157609	25,81 euros	352	HTML & CSS: The Good Parts
978-0470523964	29,48 euros	792	HTML, XHTML, and CSS Bible
Total	35 euros		

**Question 7.4 :** Pour finir, dupliquez les lignes de données du tableau un grand nombre de fois (votre code source doit faire environ 350 lignes) puis demandez à Firefox un *aperçu avant impression*. Que constatez vous sur la deuxième page ?

### Exercice 8 – Pages multiples : les cadres

**Question 8.1 :** Réalisez le jeu de cadres ci-contre qui contiendra 3 cadres (frames) sans bordure :

- un cadre horizontal en haut qui affichera de manière permanente le bandeau du document précédent (pas de barre de défilement),
- un cadre à droite affichant les sections sélectionnées depuis le menu (barres de défilement si nécessaire seulement)
- un cadre vertical à gauche qui affichera un menu (pas de barre de défilement) dans lequel vous mettrez des entrées pour changer :
  - le contenu du cadre droit,
  - le contenu du bandeau,
  - le contenu intégral de la fenêtreavec le contenu de votre choix.

Bandeau	
Menu	Cadre Principal

La page affichée par défaut dans le cadre principale sera la page de la première section.

**Note :** le schéma de définition des ensembles de cadres est le suivant :

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Frameset//EN"
"http://www.w3.org/TR/html4/frameset.dtd">
<html>
<head>
<title>Texte du titre</title>
</head>
<frameset ...> <!-- définition d'un jeu de cadres -->
  <frame ...> <!-- définition d'un cadre -->
  <noframes>
    Est affiché quand le navigateur ne peut pas afficher de cadres
  </noframes>
</frameset>
</html>

```

## **Exercice 9 – Création de formulaires – Une page google personnalisée**

**Question 9.1 :** Rendez vous sur la page de google et effectuez une recherche simple. Relevez et analysez l'url correspondant à la requête. (revenez sur la page d'accueil et cherchez une explication dans le code source de son formulaire)

**Question 9.2 :** Pouvez-vous écrire une url minimale correspondant à cette requete ?

**Question 9.3 :** En partant de ce constant, écrivez votre propre page d'accueil google

**Question 9.4 :** En reprenant l'url générée initialement par google, modifiez votre page d'accueil pour permettre à l'utilisateur de choisir la valeur des paramètres cachés. Vous veillerez à ce que les étiquettes de vos champs soient cliquables et donnent le focus au champs associé. (Suggestion : une liste pour la langue, un bouton radio pour la sécurité.)