

420-B16-RO Techniques de développement d'applications hypermédia I

TP1 – HTML et CSS

Sylvain Labranche — Collège de Rosemont

Automne 2020

Remise

La date limite pour la remise est prévue **le 14 octobre 2020**. Vous aurez plusieurs heures en classe pour travailler sur votre TP.

Vous remettez tous les fichiers (un dossier par exercice, 3 exercices en tout) dans une archive **.zip** dans la boîte prévue à cet effet sur Léa.

Tout retard entraînera une diminution de 10 % de la note maximale par jour de retard.

Modalités

Ce travail est fait individuellement ou en équipes de 3 maximum. Assurez-vous que chacun des membres de l'équipe participe et comprend chaque partie du travail.

1 Site Web en HTML

1.1 Réaliser la page d'accueil *index.html*

Vous devez créer un site web comportant 3 pages: *index.html*, *etapes-programme.html* et *tableau.html*. Vous devez coder vous-mêmes l'ensemble du site en HTML en utilisant l'éditeur Notepad++. Pour cela, vous devez d'abord créer un dossier qui comportera l'ensemble des ressources du site. Ce dossier sera nommé *TP1*. À l'intérieur, il y aura le fichier *index.html* et les dossiers *contenu*, *images* et *styles*.

- Nom du fichier : *index.html*
- Localisation: la racine du site
- Titre de la page : Initiation développement d'un programme informatique

Pour éviter de saisir le texte de cette page, vous pouvez copier le texte se trouvant dans le fichier *texteIndex.txt*. Vous devez rechercher dans le Web une image libre de droits pour le lien de messagerie.

Fiez-vous à la capture d'écran sur la page suivante, vous devez reproduire exactement la page (sauf les traits noirs qui indiquent les éléments).



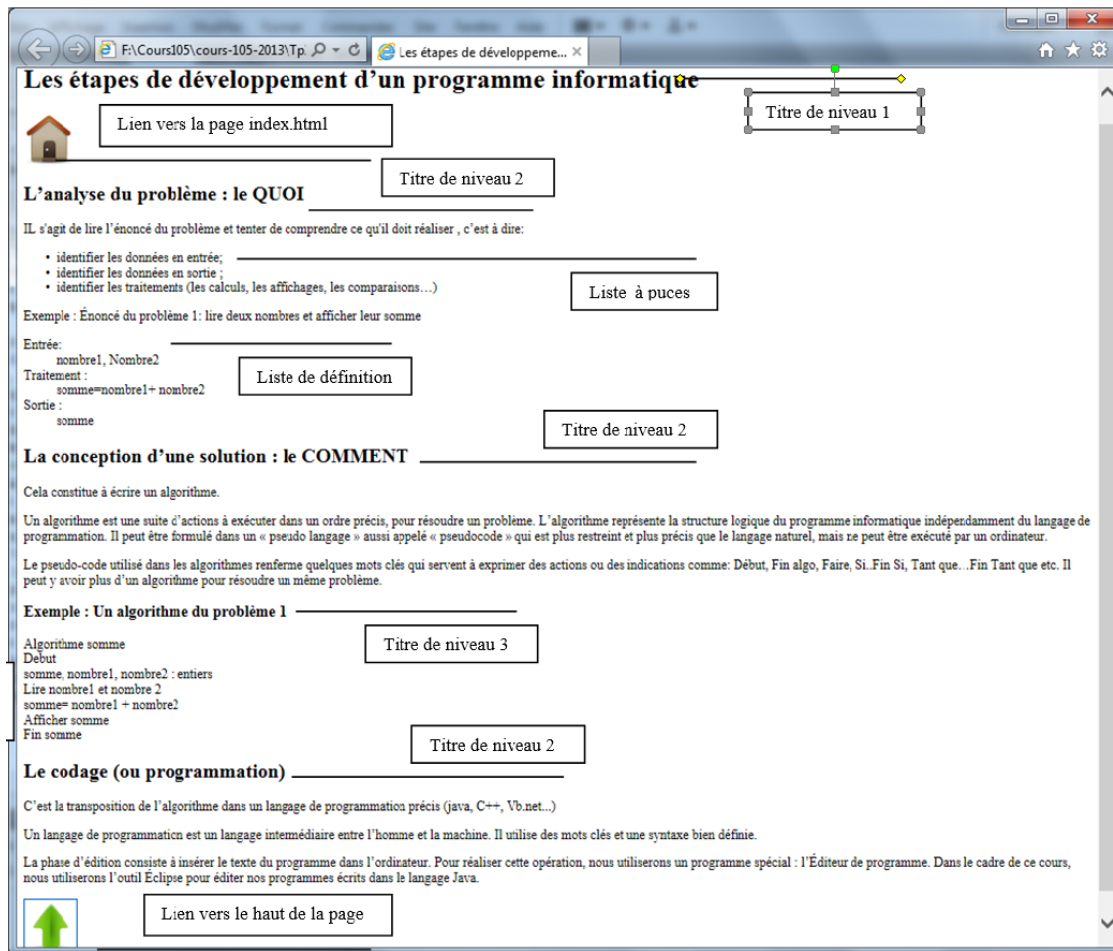
1.2 Réaliser la page *etapes-programme.html*

- Nom du fichier: *etapes-programme.html*
- Localisation: dans le dossier *contenu*.
- Titre de la page : Les étapes de développement d'un programme

Vous pouvez copier le texte se trouvant dans le document *texteEtapesProgramme.txt*.

Vous devez rechercher, dans le Web, des images libres de droits.

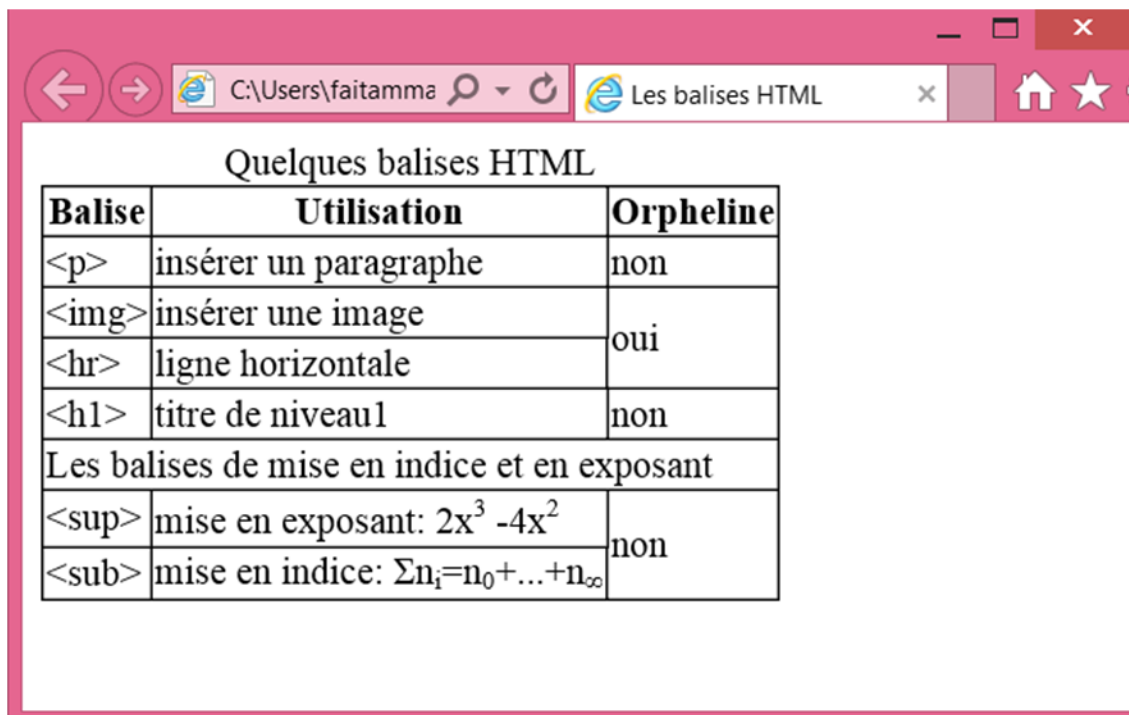
Attention: par souci de lisibilité, les trois derniers paragraphes n'ont pas été inclus dans la capture d'image. Vous devez les inclure et mettre leur titre en niveau 2. La flèche de retour vers le haut vient à la toute fin.



1.3 Réaliser la page *tableau.html*

- Nom du fichier: *tableau.html*
- Localisation: dans le dossier *contenu*.
- Titre de la page : Un tableau HTML

Vous devez reproduire le tableau HTML contenu dans l'image. Vous mettez le CSS directement dans l'en-tête.



2 Utilisation de CSS

2.1 Ajout de code CSS aux pages *index.html* et *etapes-programme.html*

Créez un document de feuille de style CSS nommé *style.css*, à placer dans le dossier *styles*. Vous devez lier ce document aux deux pages HTML (*index.html* et *etapes-programme.html*).

1. Règles de style à insérer dans le document *style-tp4.css*.

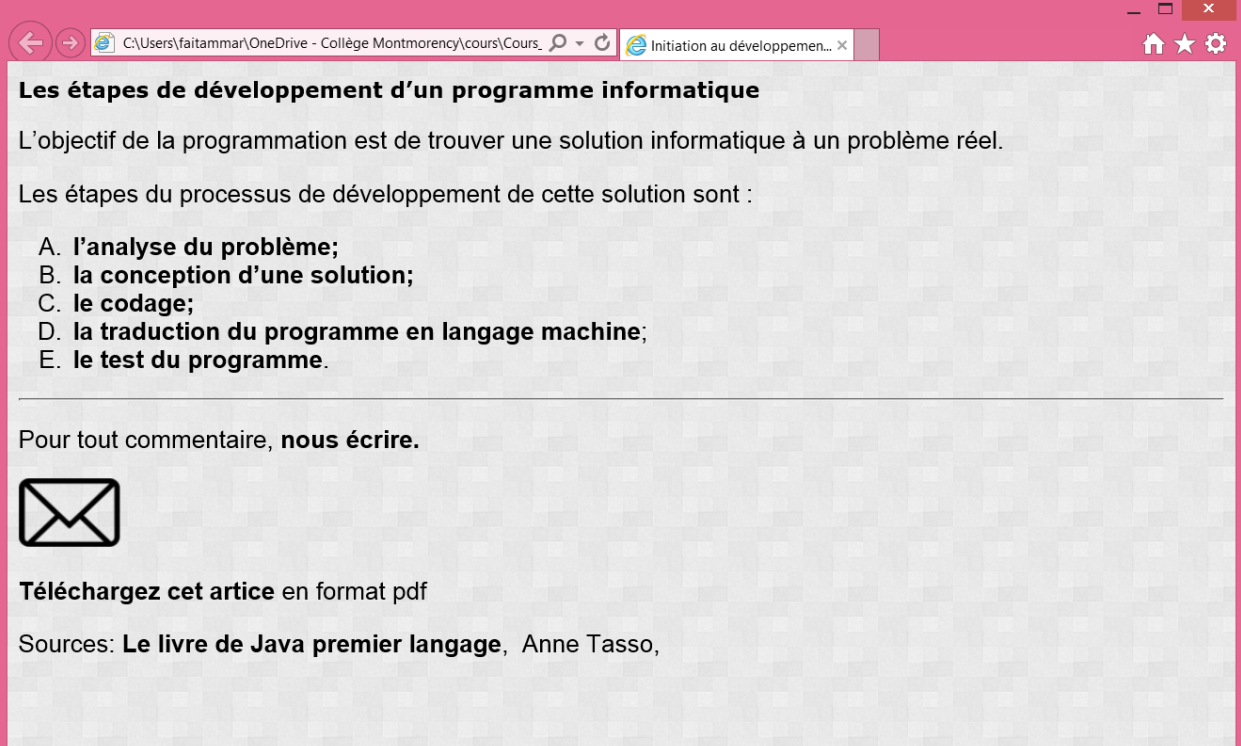
- Insérez une image d'arrière-plan sur les deux pages Web.
 - Utilisez une image de très petite taille que vous pourrez exploiter en tant que motif pour tapisser le fond de la page comme une mosaïque ou un dégradé en la répétant. Exemple un carré de 30x30 pixels peut se répéter horizontalement et verticalement ou une ligne de 2x1024 pixels peut se répéter verticalement.
 - Sauvegardez l'image dans le dossier *images*
- La police du texte, dans le corps des documents (*body*), doit être Arial, Helvetica, sans-serif, de taille 18 pixels et de couleur de votre choix, selon l'image de fond choisie.
- Les paragraphes doivent avoir un alignement justifié.
- Appliquez un style de votre choix (au moins deux règles) pour les titres de niveaux 1 et 2.
- La liste numérotée de la page *index.html* doit être de type A,B,C,D...
- la liste à puce de la page *etapes-programme.html* doit utiliser une image en guise de puce.
- Créez une classe nommée **test** qui permet de mettre le texte en italique et de couleur mauve (un mauve de votre choix). Le texte est affiché avec la police: Lucida Console, Monaco, monospace.
 - Appliquez cette classe à la liste de définition de la page *etapes-programme.html*.

Entrée:
 nombre1, Nombre2
Traitement :
 somme=nombre1+ nombre2
sortie :
 Somme

- Appliquez également cette classe à l'exemple de la page *etapes-programme.html*.
- Créez une classe nommée **droite** appliquée à la balise *img* qui permet d'aligner une image à droite. Appliquez cette classe aux deux images de la page *etapes-programme.html*). Pensez à la propriété *float*.

- (i) La couleur des liens doit être la même que celle du texte du corps du document, mais en gras pour les mettre en évidence (sélecteur *a*). Les liens ne devraient pas être soulignés, sauf au passage de la souris dessus (sélecteur *a:hover*) et quand le lien est cliqué (sélecteur *a:active*).

Aperçus des pages une fois les styles appliqués: Comme vous avez le choix pour plusieurs règles, vos pages pourront différer largement de ces exemples.




Les étapes de développement d'un programme informatique

L'objectif de la programmation est de trouver une solution informatique à un problème réel.

Les étapes du processus de développement de cette solution sont :

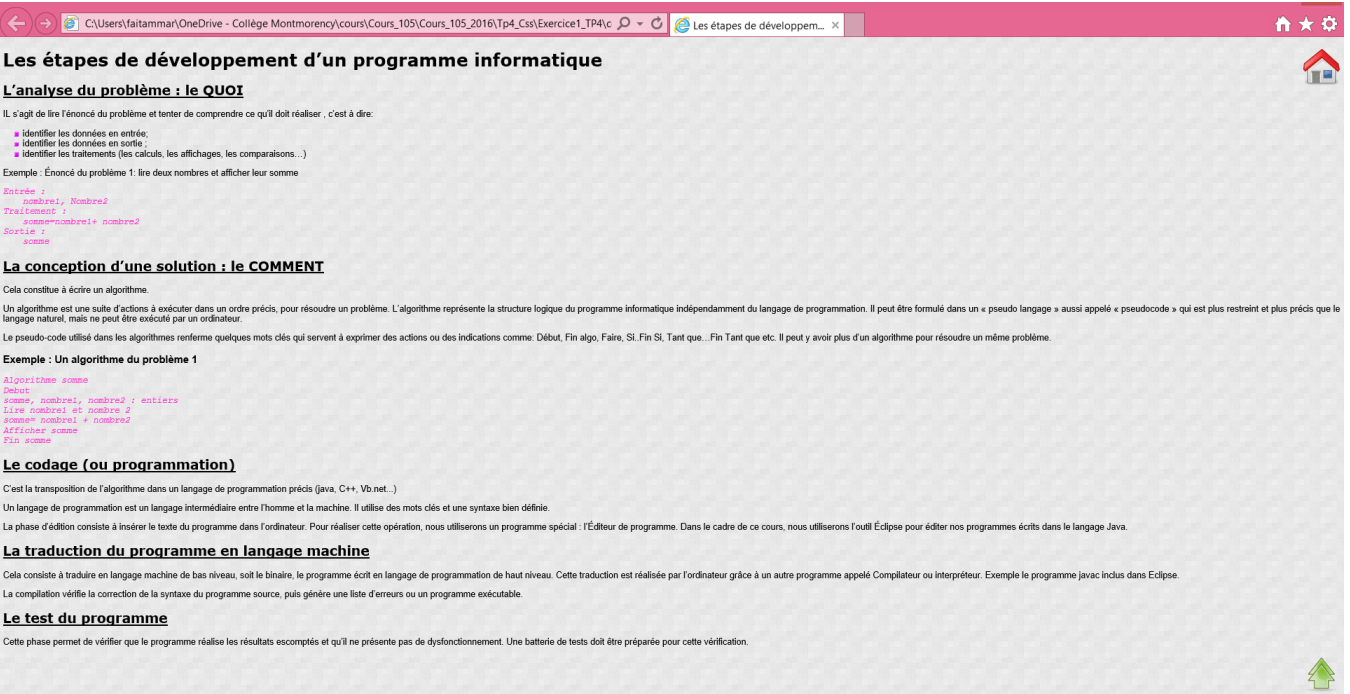
- A. l'analyse du problème;
- B. la conception d'une solution;
- C. le codage;
- D. la traduction du programme en langage machine;
- E. le test du programme.

Pour tout commentaire, **nous écrire**.



Téléchargez cet article en format pdf

Sources: **Le livre de Java premier langage**, Anne Tasso,



Les étapes de développement d'un programme informatique

L'analyse du problème : le QUOI

IL s'agit de lire l'énoncé du problème et tenter de comprendre ce qu'il doit réaliser, c'est à dire:

- identifier les données en entrée;
- identifier les données en sortie;
- identifier les traitements (les calculs, les affichages, les comparaisons...)

Exemple : Énoncé du problème 1: lire deux nombres et afficher leur somme

```
Entrée :  
nombre1, nombre2  
Traitement :  
somme=nombre1+ nombre2  
Sortie :  
somme
```

La conception d'une solution : le COMMENT

Cela constitue à écrire un algorithme.

Un algorithme est une suite d'actions à exécuter dans un ordre précis, pour résoudre un problème. L'algorithme représente la structure logique du programme informatique indépendamment du langage de programmation. Il peut être formulé dans un « pseudo langage » aussi appelé « pseudocode » qui est plus restreint et plus précis que le langage naturel, mais ne peut être exécuté par un ordinateur.

Le pseudo-code utilisé dans les algorithmes renferme quelques mots clés qui servent à exprimer des actions ou des indications comme: Début, Fin algo, Faire, Si...Fin Si, Tant que...Fin Tant que etc. Il peut y avoir plus d'un algorithme pour résoudre un même problème.

Exemple : Un algorithme du problème 1

```
Algorithme somme  
Début  
somme, nombre1, nombre2 : entiers  
Lire nombre1 et nombre 2  
somme<- nombre1 + nombre2  
Afficher somme  
Fin somme
```

Le codage (ou programmation)

C'est la transposition de l'algorithme dans un langage de programmation précis (Java, C++, Vb.net...)

Un langage de programmation est un langage intermédiaire entre l'homme et la machine. Il utilise des mots clés et une syntaxe bien définie.

La phase d'édition consiste à insérer le texte du programme dans l'ordinateur. Pour réaliser cette opération, nous utiliserons un programme spécial : l'Éditeur de programme. Dans le cadre de ce cours, nous utiliserons l'outil Éclipse pour éditer nos programmes écrits dans le langage Java.

La traduction du programme en langage machine

Cela consiste à traduire en langage machine de bas niveau, soit le binaire, le programme écrit en langage de programmation de haut niveau. Cette traduction est réalisée par l'ordinateur grâce à un autre programme appelé Compilateur ou interpréteur. Exemple le programme javac inclus dans Eclipse.

La compilation vérifie la correction de la syntaxe du programme source, puis génère une liste d'erreurs ou un programme exécutable.

Le test du programme

Cette phase permet de vérifier que le programme réalise les résultats escomptés et qu'il ne présente pas de dysfonctionnement. Une batterie de tests doit être préparée pour cette vérification.

2.2 Présentation du tableau 3 à l'aide des feuilles de styles CSS

Vous devez reprendre le tableau html de l'exercice 1.3 et modifier ses règles de style. Vous devez mettre à jour la feuille de style interne déjà existante (**dans la partie head du document**).

Nouvelles règles de style à appliquer.

- (a) La bordure du tableau est de 3 pixels, de type plein (solide), et de couleur #900.
- (b) La police d'écriture est de type "Tahoma, Geneva, sans-serif" et de taille 16 pixels.
- (c) La largeur du tableau est 550 pixels.
- (d) La hauteur du tableau est 350 pixels.
- (e) La ligne d'en-tête a une couleur de remplissage qui est #99ccff.
- (f) La ligne d'en-tête est en majuscule, gras et souligné.
- (g) La dernière colonne (Orpheline) est centrée horizontalement et verticalement (propriété **vertical-align**). Vous pouvez créer une classe nommée **milieu** comportant les 2 propriétés et l'appliquer aux balises **td** des cellules de cette colonne.

Aperçu du tableau une fois les styles appliqués:

Quelques balises HTML

<u>BALISE</u>	<u>UTILISATION</u>	<u>ORPHELINE</u>
<p>	insérer un paragraphe	non
	insérer une image	oui
<hr>	ligne horizontale	
<h1>	titre de niveau 1	non
Les balises de mise en indice et en exposant		
<sup>	mise en exposant: $2x^3 - 4x^2$	non
<sub>	mise en indice: $\sum n_i = n_0 + \dots + n_\infty$	

3 Utilisation d'un gabarit HTML et CSS simple

Dans cette partie, vous allez apprendre à utiliser un gabarit HTML-CSS pour la mise en page de vos pages Web. Je vous fournis les gabarits **base.css** et **modele01.css** que vous pouvez utiliser, mais vous êtes libres de prendre n'importe quel gabarit que vous trouvez sur le Web. Des explications sur l'utilisation de ces gabarits vous seront fournies pendant les périodes de travail sur le TP.

Vous devez réaliser 3 pages HTML : *index.html*, *netiquette.html*, *apropos.html*.

Le site doit être bien organisé : les fichiers image, css et html doivent être dans des sous-dossiers, groupés par type, à l'exception de la page *index.html* qui doit être dans la racine du site (même organisation que le site du TP3).

Dans vos trois pages, vous devez lier les feuilles de style **base.css** et **modele01.css** (dans cet ordre). Vos pages doivent être exactement comme celle en exemple sur la page suivante.

1. Description de la page d'accueil *index.html*.

- Le texte de cette page est fourni dans le fichier *textIndex.txt*

- Le titre de la page est : *Le courrier électronique*.
- En cliquant sur le lien Principe et fonctionnement, la page index.html s'affiche.
- En cliquant sur le lien Nétiquette, la page netiquette.html s'affiche.
- En cliquant sur le lien À propos de nous, la page apropos.html s'affiche.
- En cliquant sur le lien URL du paragraphe "Que peut-on envoyer ? ", le site www.commentcamarche.net/contents/542-url s'affiche dans un nouvel onglet.
- Vous devez utiliser les balises **strong** pour mettre en évidence le texte en gras et *em* pour celui écrit en italique.
- Dans le pied de page "Commentcamarche" est un lien vers www.commentcamarche.net.

2. Description de la page *netiquette.html*

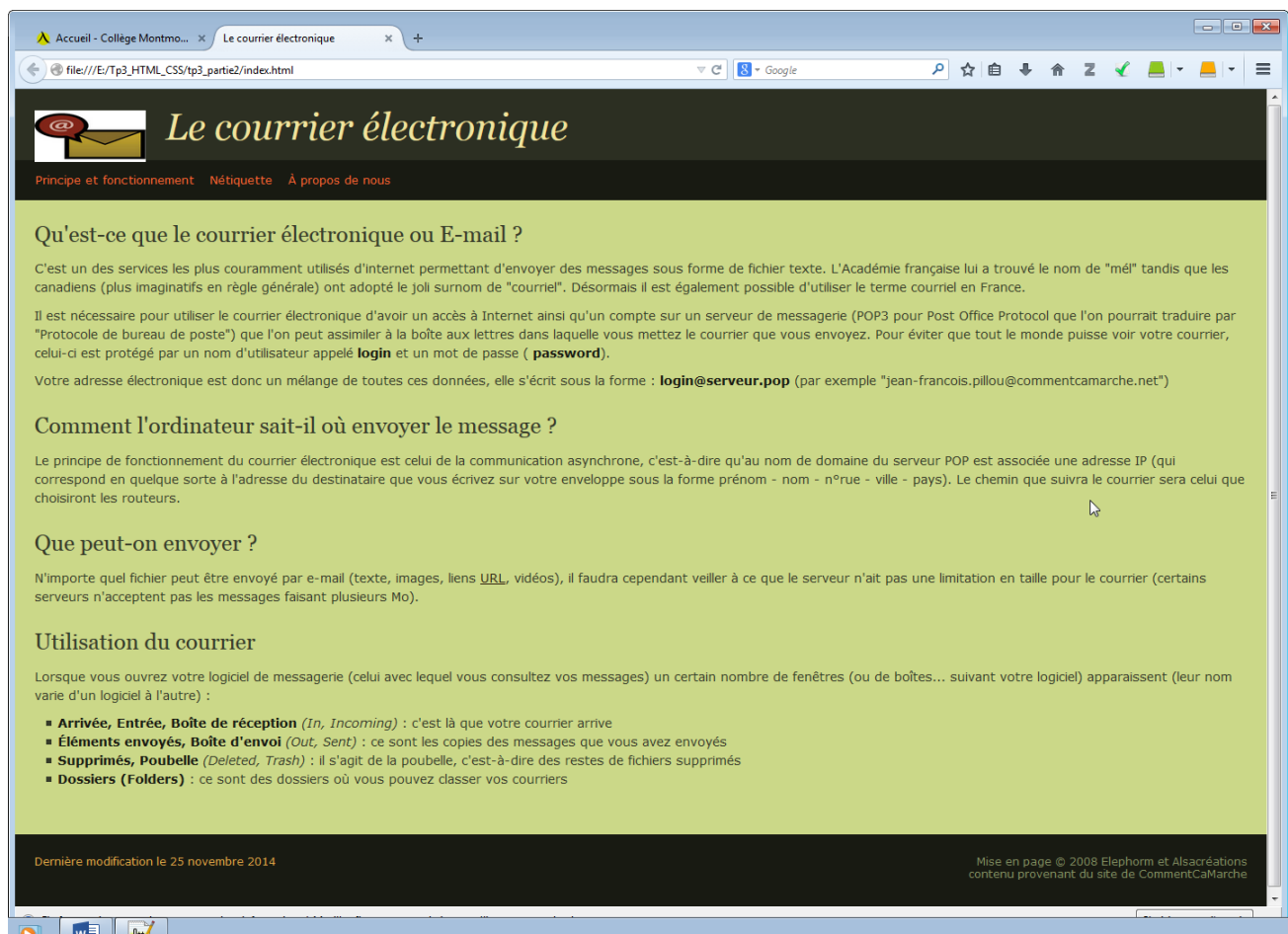
Cette page doit avoir le même format que la page d'accueil (les même en-tête, pied de page et barre de navigation que la page index.html). Vous devez composer le contenu vous-même (dans vos propres mots) et choisir un titre approprié.

Exigences particulières :

- Vous devez insérer une image en rapport avec la nétiquette qui sera alignée à droite et avoir au moins un paragraphe à gauche de cette image.
- Vous devez indiquer vos sources dans le pied de page (maximum 2 sources).
- Vous devez avoir deux titres :
 - (a) Définition de la Nétiquette : donnez une définition de la Nétiquette en général (une à 3 lignes)
 - (b) Nétiquette dans les courriels : énoncez les règles de la nétiquette propre au courrier électronique (5 à 7 règles les plus significatives)

3. Description de la page *apropos.html*

Vous devez vous présenter dans cette page : nom, prénom, collègue, loisirs, objectif de carrière, etc. Vous devez insérer une photo de vous. La conception de cette page vous appartient. Cette page doit aussi avoir le même formatage que *index.html*, notamment le menu de navigation.



Évaluation

Ce travail compte pour 15 % de votre session. Il y a 3 exercices qui comptent pour 5% chacun. Une directive non-respectée fait perdre des points, mais la qualité du code est aussi évaluée, notamment le respect des normes XHTML.