



TP WEB : RAPPEL

INSTITUT SUPERIEURE DES SCIENCES
APPLIQUEE ET TECHNOLOGIE DE SOUSSE



Link : <https://issatso.rnu.tn>

Mail : issatso@issatso.rnu.tn

sofiene.benahmed@issatso.rnu.tn/issats.prepa.2009@gmail.com

TP HTML :Rappel

Récupérez l'archive contenant les fichiers nécessaires aux exercices suivants et décompressez-la entièrement dans votre espace de travail.

Les fichiers créés au cours des différents exercices sont souvent réutilisés pour d'autres exercices ultérieurs. Veillez donc à toujours en avoir une version disponible.

Exercice 1 : Structure HTML minimale

1. Ouvrez dans un éditeur de texte le fichier **ex1.html** et observez-en la structure.

Il est important d'utiliser un éditeur de texte adapté à l'écriture de code informatique.

Parmi ceux à votre disposition en salle de TP, vous pouvez utiliser visual code, Notepad++...

2. Vérifiez la validité XHTML5 de ce fichier grâce à [ce validateur](#).

*Ensuite chargez le fichier à tester en sélectionnant **File Upload** dans la liste déroulante (là où est proposé **Address** par défaut) puis **Parcourir...** Vous pouvez alors utiliser le bouton **Validate** et constater le résultat.*

*Une [autre version de ce validateur](#) est proposée par le W3C. Cependant pour vérifier la validité XHTML de vos documents vos fichiers devront nécessairement avoir l'extension **xhtml**.*

3. Parvenez-vous à supprimer un élément ou une balise de ce fichier tout en le laissant valide ?

Qu'en concluez-vous ?

Exercice 2 : Informations et code source

1. Vérifiez la validité XHTML du document **ex2.html**.

2. Chargez le fichier **ex2.html** dans le navigateur.

Choisissez **Informations sur la page** dans le menu contextuel par un clic droit dans la page affichée et examinez les informations présentées.

Quel est l'encodage de caractère utilisé pour ce fichier ? Quelles *méta-informations* apparaissent ?

3. Comparez ces informations avec celles obtenues pour le document **ex1.html**.

Dans un éditeur de texte étudiez les codes contenus dans les fichiers **ex1.html** et **ex2.html** et déduisez de cette étude les parties de code de **ex2.html** qui sont à l'origine des méta-informations que vous avez pu observer pour ce document.

4. Dans la fenêtre du navigateur qui affiche le document **ex2.html** choisissez **Code source de la page** dans le menu contextuel activé par un clic droit (ou via le raccourci Ctrl+U).

Vous pouvez ainsi consulter le code source du document HTML affiché, mais sans possibilité de le modifier.

5. Dans le code source, à quoi correspondent les portions de texte placées entre les marques **<!--** et **-->** ?

Exercice 3 : Premier document HTML

1. Dans un éditeur de texte ouvrez le fichier **ex3.html**.

2. Utilisez le menu **Enregistrez sous...** de votre éditeur de texte pour enregistrer votre document en précisant un **nouveau** nom du fichier. Celui-ci devra avoir pour

extension `.html` ou `.xhtml`. Par exemple `simple.html`. Modifiez le titre et mettez à jour les méta-informations `author` et `keywords`.

3. Dans le corps du document créez un élément `<p>` dont le contenu laissé à votre imagination.
4. Visualisez le fichier enregistré en l'affichant dans le navigateur.
5. Vérifiez la validité XHTML5 de votre fichier.
6. Dans l'éditeur de texte ajoutez un second élément `<p>` à la suite du premier, à nouveau choisissez librement un contenu court.
7. Visualisez à nouveau le document dans le navigateur.

Si rien ne se passe, avez-vous pensé à sauvegarder le fichier source modifié ? Avez-vous actualisé (rechargé) le document dans le navigateur ? La touche F5 permet cette actualisation.

8. Pensez à valider à nouveau le document modifié.

Exercice 4 : Petit document HTML

1. Affichez le document `ex4.html` dans votre navigateur.
2. Lisez-le.
3. Vérifiez à l'aide du [validateur](#) que la syntaxe de cette page est correcte.
4. Consultez les informations sur le document ainsi que le code source de la page.
5. Étudiez ce code source : quelles balises y apparaissent ?

Exercice 5 : Validation

1. Chargez dans le navigateur le fichier `ex5-corrompu.html`. Parvenez-vous à le visualiser normalement ?
2. Testez-le avec le validateur. Combien d'erreurs sont signalées ?
3. Qu'indique le message de la première erreur signalée ?
4. A quelle ligne et sur quelle portion du code source est-elle signalée ?
5. A quoi est due cette erreur ?
6. Corrigez-la et repassez le document au validateur.

La touche F5 permet un rechargement de la page et donc de tester la validation du même document avec les mêmes options.

7. Lisez les messages d'erreur suivants en essayant d'en comprendre le message et de les localiser dans le fichier source.
8. Corrigez une à une les erreurs signalées. Testez à la validité après chaque correction (et donc vérifiez votre correction).

Pour chaque erreur décrivez concisément et précisément la cause de l'erreur.

Exercice 6 : Exploration de la structure

Nous allons voir comment les outils mis à notre disposition par le navigateur permettent d'explorer la structure HTML du document.

1. Ouvrez à nouveau le fichier `ex1.html` dans le navigateur puis dans le menu symbolisé par trois lignes horizontales en haut à droite ou dans le menu **Outils** (si besoin faites le apparaître grâce à la touche par Alt) choisissez **outils de Développement** et enfin **Inspecteur**.

La fenêtre de votre navigateur devrait se séparer en deux, avec dans la partie gauche le document et dans la partie droite l'outil *inspecteur* activé.



L'inspecteur HTML dans le navigateur.

Cet outil est également accessible par le raccourci clavier Shift+Ctrl+I.

2. Dans l'inspecteur, en agissant sur les flèches situées à gauche des éléments HTML, vous pouvez déployer ou fermer les éléments et leur contenu. Simultanément vous pouvez visualiser dans la zone document l'élément sélectionné qui y apparaît encadré. Et dans le bandeau supérieur de l'inspecteur est affichée la séquence d'emboîtement des éléments HTML jusqu'à l'élément courant qui est mis en évidence dans cette séquence.

Déployez les éléments jusqu'à sélectionner l'élément qui correspond dans la zone affichée à l'exemple de code avec la balise `<p>`.

3. Inversement, il est possible depuis la zone de document d'afficher dans l'inspecteur l'élément de structure associé. Il faut pour cela utiliser le bouton de sélection d'élément situé à la gauche du bandeau supérieur de l'inspecteur. Cliquez sur ce bouton puis déplacez votre souris dans la zone d'affichage et observez ce qu'il se passe dans l'inspecteur.

Utilisez le pour trouver la balise qui encadre de mot *FIL* dans l'hyperlien du document. Quel est l'emboîtement successif d'éléments qui y mène ?

Il existe un outil équivalent sous Chrome ou Internet Explorer, il est activable par la touche F12. Ces outils présentent globalement les mêmes fonctionnalités.

Exercice 7 :

1. Affichez le document **ex7.html** dans votre navigateur.
2. Lisez le contenu de ce document, son contenu contient des informations à savoir.
3. Vérifiez à l'aide du [validateur](#) que la syntaxe de cette page est valide XHTML5.
4. Ouvrez ce fichier dans un éditeur de code source : Parmi les balises qui y apparaissent identifier quelles sont :
 - o Les balises d'éléments de titre,
 - o Les balises d'éléments de phrasé,
 - o Les balises d'éléments de flux structurants,
 - o Les balises correspondant à d'autres éléments de flux.

Exercice 8 : Sémantique et attributs

On continue à travailler avec le document **ex7.html** de l'exercice précédent. Affichez-le dans le navigateur et ouvrez son code source dans un éditeur de texte. Dans cet exercice vous devrez faire le parallèle entre le code source html et sa version affichée par le navigateur.

1. On s'intéresse au contenu des *remarques* que l'on peut lire dans la section *Quelques remarques* du document **ex7.html** :
 1. En consultant son code source (sélectionnez-en le texte puis clic droit et *Code source de la sélection*) vérifiez l'affirmation de la première remarque.

*En plus de la possibilité déjà évoquée d'afficher le code source de la page, il est possible de n'afficher le code source que d'une partie de la page en sélectionnant à la souris la zone du document souhaitée et en choisissant **Code source de la sélection** dans le menu contextuel de la page. Vous pouvez le tester sur le texte de cette première remarque.*

2. Examinez la balise `img` du document pour vérifier la seconde remarque.
 3. Repérez quelques entités dans le code source. A quel caractère correspond l'entité `&` ?
 4. Cherchez une zone en commentaire dans le code source et vérifiez qu'elle est bien ignorée lors de l'affichage.
2. Pour chacun des éléments de phrases suivants, cherchez un exemple d'élément dans le code source et retrouvez comment celui-ci est affiché dans le navigateur. Vous pouvez également utiliser l'outil inspecteur.
1. ``
 2. `<code>`
 3. ``

A chaque élément html est associée une sémantique qui détermine à quel usage il est destiné. Celle-ci est **indépendante de l'aspect visuel**.

Décrivez en une ou deux phrases pour chacun des éléments ci-dessus son contexte d'usage et vérifiez votre réponse à partir des informations sur ces éléments accessibles depuis [cette page de référence](#) sur le *Mozilla Developer Network* (MDN par la suite) (Ctrl+F permet de faire une recherche dans la page).

3. A propos des listes
1. Quel est le rôle (ou la sémantique) des éléments `` et `` ? Dans quels cas faut-il utiliser l'un plutôt que l'autre ?

Explorez en remplaçant un élément `` par un élément `` (ou réciproquement) dans le document et constatez les changements.

N'oubliez pas de modifier à la fois les balises ouvrantes et fermantes.

2. Quelles différences existent-ils entre ces deux éléments ?

Consolidez votre réponse

4. Les balises peuvent être complétées par des attributs, parfois de manière obligatoire.
1. Quels attributs sont définis pour l'élément `` présent dans la page ?
 2. A l'aide du validateur vérifiez après l'avoir supprimé que l'attribut `src` est obligatoire.
 3. Devinez le sens de ces attributs et vérifiez avec les informations sur la page du MDN.
 4. L'attribut de l'élément `<abbr>` présent dans la page est-il obligatoire ? Comment pouvez-vous le vérifier ? Faites le.
 5. Quel attribut, autre que `href`, est présent dans l'une des balises d'hyperlien. En testant les différents hyperliens déterminez l'effet de cet attribut.

Vous allez dans cet exercice créer votre premier document html. Vous devez vous inspirer des codes sources des documents utilisés dans les exercices précédents pour les différentes balises à utiliser ainsi que pour leur syntaxe.

1. Créez nouveau document html dont vous définirez les méta-informations titre et auteur.

*Comme déjà indiqué dans un exercice précédent, le plus simple pour garantir une structure valide XHTML5 consiste à partir d'un document modèle pour ne pas avoir à retenir ni à recopier à chaque fois tous les éléments obligatoires. Le document **ex4.html** est une très bonne base. Il vous suffit donc d'ouvrir ce fichier dans votre éditeur de texte puis de l'enregistrer sous un nouveau nom puis de modifier les informations dans l'entête.*

2. Dans le corps du document :

- Ajoutez un titre citant le nom du parcours d'enseignement suivi ce semestre
- Créez un paragraphe présentant en quelques phrases courtes la formation que vous suivez ce semestre,
- Créez une liste énumérant les UE que vous suivez ce semestre,
- Ajoutez un paragraphe indiquant votre projet de poursuite d'études à l'issue de cette année d'études.

Dans les éléments de texte précédents, s'ils apparaissent les mots informatiques, programmation et web sont jugés comme **important** et doivent donc être marqués comme tel et mis en évidence à l'aide de la balise que vous jugez la mieux adaptée. Les termes tels que UE, SESI, S3H, etc. pourront être marqués comme des **abréviations**.

3. Vérifiez la validité XHTML5 du document que vous avez créé puis visualisez le dans un navigateur.

Exercice 10 : *img* et figure

*L'élément **** est destiné à recevoir une image stockée dans un fichier externe au document. Ce fichier est identifié par l'attribut **src** et sa valeur est l'URL de l'image. Comme pour les hyperliens sa valeur peut donc être une adresse relative ou absolue.*

Le plus souvent on trouvera des images aux formats PNG, JPEG ou GIF.

*L'élément **** impose un attribut **alt** donnant une description de l'image. Sa valeur est destinée à être affichée à la place de l'image si celle-ci ne peut être chargée par le navigateur. Si l'image est importante pour le document (elle n'est pas uniquement décorative) il convient donc que l'attribut **alt** soit suffisamment précis. Il est souvent proposé comme première bonne approche pour en créer la valeur de répondre à la question : Que diriez-vous à la place de l'image à quelqu'un au téléphone à qui vous devriez lire le document HTML..*

*Pour plus d'information sur la bonne manière de créer l'attribut **alt** d'un élément ****, vous pouvez consulter [cette page](#) du W3C. ou encore ce [document](#) qui présentent des bonnes pratiques de définition de l'attribut **alt**. Ce sera l'occasion de vous rendre compte que la création de document HTML est beaucoup plus complexe et subtile qu'il n'y paraît des fois...*

*Parmi les autres attributs de l'élément ****, on trouve **width** et **height** qui définissent la taille en pixels de la zone d'affichage de l'image (la valeur est donnée sans unité). Si leurs valeurs ne correspondent pas aux dimensions réelles de l'image, celle-ci est redimensionnée en conséquence, et éventuellement déformée. Si une seule des deux valeurs est fournie, l'autre est adaptée pour conserver les proportions de l'image. Nous verrons ultérieurement que les dimensions d'affichage d'une image peuvent être également déterminées par les feuilles de style.*

Il est parfois recommandé de fixer les largeur et hauteur des images pour des raisons visuelles mais aussi de performances du navigateur. Celui-ci n'a pas à attendre le chargement de l'image pour connaître, et donc allouer, l'espace à lui réserver dans la page.

L'élément `<figure>` peut accueillir des éléments de flux (on rencontre souvent des éléments `` comme contenu d'un élément `<figure>` mais ce n'est pas une obligation) qui pourront être accompagnés d'une légende placée avant ou après de contenu et définie par l'élément `<figcaption>`. Il n'y a aucune obligation à placer une image dans un élément `<figure>`.

1. Créez une page contenant un élément `` dont la source est le fichier **pont-du-gard.jpg** fourni.
2. N'oubliez pas de vérifier la validité de la page créée.
3. La taille originale de cette image est de **800x600**. Faites différents essais sur la page avec différentes valeurs pour les attributs `width` et `height`, y compris en n'en fixant qu'un seul des deux. Que constatez-vous ?
4. Modifiez la page pour placer cette image dans un élément `<figure>` pour lequel vous définirez une légende.

Exercice 11 : Hyperliens (1)

*Les hyperliens se construisent à l'aide de l'élément `<a>`. La cible de l'hyperlien est identifiée par l'attribut `href`. Elle contient l'adresse du document cible. Cette adresse est appelée **URL**. Le document cible peut-être un document HTML ou non.*

L'adresse du lien peut être absolue. Elle précise alors une adresse complète du document, y compris le protocole utilisé, par exemple `http://www.issatso.rnu.tn`. Elle peut-être aussi relative. Elle se base alors sur l'adresse du document courant et se positionne relativement à celui-ci, par exemple `../document/timoleon.html`.

Le contenu d'un élément `<a>` peut être constitué d'éléments de phrasé ou de flux.

1. A l'aide du document **ex4.html** créez dans un même dossier deux nouveaux documents HTML. Lorsque vous créez vos fichiers n'oubliez pas de leur donner une extension `.html`.

Les corps de ces documents (partie `<body>`) sont pour l'instant vides, nous allons les remplir à l'aide du document **ex7.html**.

Dans le premier copiez-y le contenu du corps de **ex7.html** allant de la balise `<body>` à la seconde balise `<h2>` et dans le corps du second copiez le reste du contenu du corps de **ex7.html** (jusque `</body>` donc).

Évidemment les deux fichiers créés doivent être des fichiers HTML valides. Vérifiez-le.

2. En bas du premier document ajoutez le code html représentant un lien dont le texte est suivant et dont la cible est le chemin **relatif** vers le second document. Faites de même dans le second document en ajoutant un lien précédent vers le premier document.

Sauvegardez puis vérifiez le bon fonctionnement de vos liens dans le navigateur.

3. Fermez les documents ouverts dans le navigateur puis créez dans votre dossier deux sous-dossiers. Appelez-les par exemples *dossier1* et *dossier2*.

Placez dans *dossier1* le premier des deux documents HTML créés précédemment et dans *dossier2* le second. Vous pouvez vérifier dans le navigateur qu'évidemment vos liens ne fonctionnent plus.

Corrigez ensuite les chemins relatifs des cibles des liens dans les deux documents pour qu'ils fonctionnent correctement. Vérifiez votre travail dans le navigateur.

Exercice 12 : Hyperliens(2)

1. Les liens peuvent cibler des documents autres que des documents HTML. Reprenez le document **ex7.html** et ajoutez en début de document un lien qui cible (vers) le fichier **tim_berniers-lee.pdf** fourni dans l'archive. Testez votre lien.
2. Dans ce même document sur le texte *En XHTML5* dans la remarque numéro 3 ajoutez un lien qui cible l'image **W3C_Icon.png** (fourni). Faites en sorte que ce lien s'ouvre dans un nouvel onglet.
3. L'attribut **title** permet d'ajouter une information sur un élément. En particulier pour les éléments `<a>` sa valeur permet d'informer sur la cible du lien grâce à une infobulle affichée par le navigateur lorsqu'on laisse le pointeur de la souris sur le lien.

Ajoutez un attribut **title** ayant pour valeur *Le logo du W3C* sur le lien créé à la question précédente et vérifiez-en l'effet.

4. Le contenu d'un élément `<a>`, c'est-à-dire la zone cliquable correspondant au lien, peut-être autre chose qu'une portion de texte.

Toujours dans le document **second.html**, ajoutez un hyperlien dont le **contenu** sera l'élément `` qui apparaît à la fin du document et dont la cible sera le site <https://developer.mozilla.org>. Vous pouvez ajouter un attribut de titre à ce lien indiquant qu'il pointe vers le site du Mozilla Developer Network.

5. Le contenu peut également porter sur une portion complète du document, regroupant plusieurs éléments.

Vérifiez-le en créant un élément `<a>` dont le contenu sera l'ensemble du code couvrant le texte dans la troisième remarque allant de `<p>Voici quelques autres exemples...` (placez la balise `<a>` avant la balise `<p>`) à `etc.` (placez la balise `` après la balise ``).

Faites pointer ce lien sur <https://developer.mozilla.org/fr/docs/Glossaire/Entity> par exemple.

Exercice 13 : Création d'un document HTML

L'objet de cet exercice est de construire un document HTML reprenant globalement la structure du document **tim_berniers-lee.pdf** de l'archive. Le texte brut est fourni dans **tim_berniers-lee.txt**.

Inspirez-vous du code html des exemples fournis avec les exercices précédents pour construire votre document.

1. Construisez le document HTML en respectant les règles suivantes :
 - Créez un entête approprié (titre, encodage, auteur, mots-clés,...).
 - Aux titres de section du PDF correspondront des sections de titre de même niveau.
 - Les paragraphes de texte seront placés dans des éléments `<p>`.
 - Utilisez les éléments de phrasé suivants
 - Pour les citations du texte, l'élément `<q>`
 - Pour les textes mis en évidence en gras, l'élément ``,
 - Pour le terme *World Wide Web* qui apparaît en légende sous la figure en police *sans serif* l'élément ``.

L'usage de ces balises dans chacun de ces cas vous paraît-elle justifiée (vérifiez à l'aide du site du MDN) ?

- Vous placerez les images avec leur légende au sein d'un élément `<figure>`. Deux ont pour source un fichier local l'autre une cible distante.

- Vous marquerez comme des abréviations les termes : *CERN*, *MIT*, *W3C*, *URL*, *HTTP* et *HTML* et préciserez leur attribut `title`.
 - Les notes de bas de page seront remplacées par des hyperliens qui porteront sur les mots suivis du numéro de la note de bas de page, la cible de cet hyperlien est le contenu de la note. Les numéros des notes ne doivent donc plus apparaître dans le document html final.
 - Utilisez un moteur de recherche pour trouver une balise permettant de placer une portion de texte en exposant, comme dans le texte *XXe siècle* de la fin du texte.
 - Créez des listes pour les deux parties du texte avec des énumérations, justifiez dans chaque cas votre choix d'élément `` ou ``.
 - Les dates du document seront identifiées par un élément `<time>`. Quelle information faut-il préciser pour que le document reste valide ?
2. Vérifiez la validité XHTML5 de votre document, et si besoin corrigez-le jusqu'à ce qu'il soit valide.

Exercice 14 : Ancres

*L'élément `<a>` permet également de définir des liens internes au document. C'est-à-dire des liens dont la cible est un autre élément du document. On parle généralement d'**ancres** pour ces cibles.*

*Il est nécessaire que l'élément ciblé soit identifié au sein du document. En HTML5 cela se réalise en lui ajoutant un attribut `id` avec une valeur. Cet attribut permet de nommer un élément de manière **unique** au sein du document. Il est conseillé de ne pas mettre d'espace ni de caractère accentué ou de ponctuation dans la valeur donnée à l'attribut `id`.*

Un lien vers un élément au sein d'un document se fait en faisant suivre l'adresse, relative ou absolue, du document d'un dièse `#` suivi de la valeur de l'`id` de l'élément ciblée.

1. On appelle lien interne un lien vers une autre partie du même document. Si un document contient un élément dont l'attribut `id` a pour valeur "`value`", quelle est la valeur de l'attribut `href` d'un lien **interne** vers cette ancre au sein de ce document ?
2. Reprenez le document **ex7.html**. Faites-en une copie puis ajoutez des `id` à chacun des titres de section de ce document.
3. Ajoutez en début de document, après la phrase d'introduction, un menu (balise `<nav>`) qui prend la forme d'une liste ordonnée dont les items reprennent les textes de chacun des titres (les éléments `h?`) de section.
A l'aide des ancres, faites en sorte que ces textes correspondent à des liens internes vers ces sections.
4. Créez maintenant un nouveau document HTML, que vous pouvez appeler **sommaire.html** par exemple. Dans ce document placez-y un menu identique à la question précédente. Il vous faut cependant modifier les cibles des liens en conséquence. Faites-le et testez.
5. Dans la section *Les balises* sur le texte *contenu de section* du premier élément de la liste ajoutez un lien vers la zone d'`id` `Sectionnement_du_contenu` de la page du MDN : <https://developer.mozilla.org/fr/docs/Web/HTML/Element>.

Exercice 15 : Tables

1. Étudiez les informations contenues dans le document **tableHTML.pdf** sur les éléments `<table>`, `<caption>`, `<tr>`, `<th>` et `<td>`.
2. Utilisez ensuite ces éléments pour compléter le document **table-UE-S4.html** en lui ajoutant une table présentant les informations sur les volumes horaires de quelques UE du S4 Informatique. Les données se trouvent dans ce lien :

<https://issatso.rnu.tn/formation/type/ingenieurs> et la table à créer doit reprendre cette structure :

les unités d'étude

UE	nombre d'heures cours	nombre d'heures td	nombre d'heures tp	crédits
IHM				
SGBD				
DevWeb				
MIA				
Anglais				

Table à reproduire.

Les éléments de la première ligne et de la première colonne joueront le rôle d'entête et la table doit disposer d'une légende (*caption*) pertinente.

- Les noms des UE sont des abréviations pour lesquelles vous utiliserez des éléments `<abbr>`.

Sur ces noms vous ajouterez des liens vers leur page du portail pédagogique du FIL.

Vous trouverez ces pages (par exemple <http://portail.fil.univ-lille1.fr/ls4/poo>) à partir de [la page du S4](#) sur le portail.

- Vérifiez la validité XHTML5 de votre document.
- Etudiez les informations contenues dans le document **tableHTML.pdf** sur les éléments `<thead>`, `<tbody>` et `<tfoot>`.
- Ajoutez une ligne qui fait un total de chaque colonne .

Placez ensuite la première ligne dans un élément `<thead>` pour définir l'entête, la dernière (celle ajoutée) dans un élément `<tfoot>` pour le pied du tableau et les autres qui constituent le corps du tableau dans un élément `<tbody>`.

- Visualisez puis vérifiez la validité XHTML5 de votre document.

Exercice 17 : Tables (un autre)

- Le fichier **evol-pop.html** contient une table HTML. Ouvrez-le dans un éditeur de texte et étudiez en le contenu.
- Visualisez le dans votre navigateur et vérifiez en la validité XHTML5.
- Le texte dans le premier bloc en commentaire doit constituer la légende de la table. Un élément `<caption>` peut contenir des éléments de flux. Ajoutez à la table sa légende dont vous structurerez le contenu en 2 titres, le deuxième étant moins important que le premier (par exemple `<h2>` et `<h4>`). Visualisez le document et vérifiez sa validité syntaxique.
- Retirez les marqueurs de commentaire du second bloc en début de document. Visualisez le document et testez sa validité. Qu'indique les messages d'erreur ?
- Modifiez les deux lignes ajoutées pour faire en sorte d'obtenir la présentation ci-dessous (et un document valide !) :

Année	Population au 1er janvier						Répartition par tranche d'âges (en %)					
	de 0 à 19 ans	dont 0-14 ans	de 20 à 59 ans	de 60 à 64 ans	65 ans ou plus	dont 75 ans ou plus	de 0 à 19 ans	dont 0-14 ans	de 20 à 59 ans	de 60 à 64 ans	65 ans ou plus	dont 75 ans ou plus
1946	11 838 569	8 592 117	21 848 568	1 998 090	4 440 003	1 379 698	29,5	21,4	54,4	5,0	11,1	3,4
...

Le résultat à obtenir

- Définissez l'entête du tableau avec ces deux lignes et le corps avec les autres. Modifiez ensuite l'entête pour obtenir la structure suivante :

Année	Population au 1er janvier					Répartition par tranche d'âges (en %)				
	de 0 à 19 ans	de 20 à 59 ans	de 60 à 64 ans	65 ans ou plus		de 0 à 19 ans	de 20 à 59 ans	de 60 à 64 ans	65 ans ou plus	
	dont 0-14 ans			dont 75 ans ou plus		dont 0-14 ans	de 20 à 59 ans	de 60 à 64 ans	dont 75 ans ou plus	
1946	11 838 569	8 592 117	21 848 568	1 998 090	4 440 003	1 379 698	29,5	21,4	54,4	5,0
...

Le résultat à obtenir

7. Vérifiez la validité XHTML5 de votre document.

Inspiré par Bruno Bogaert.

Exercice 18 : Tables *colspan* et *rowspan*

1. Complétez le fichier **exo-col-rowspan.html** avec le code XHTML permettant de définir les deux tableaux suivants, chacun dans un élément `<table>`. Dans votre document les deux tables peuvent s'afficher l'une en dessous de l'autre (ce sera d'ailleurs très certainement le cas).

Table question 1

1	2	
	3	4

Table question 2

7			
6	3		4
	2	1	
	5		

Les tables à reproduire.

2. Vérifiez la validité XHTML5 de votre code.

Exercice 19 : Structure arborescente

Exercice à faire sur feuille.

1. Dessinez le sous-arbre de racine `<body>` correspondant au document décrit dans le fichier **premier.html**. Vous ne représenterez pas les nœuds texte.
2. Vérifiez votre structure à l'aide de l'outil inspecteur.
3. Combien de nœuds comporte cet arbre ?

Exercice 20 : réaliser un formulaire d'authentification.

Créez un formulaire contenant un champ de texte pour saisir un login un champ password pour le mot de passe de huit caractères au maximum, chacun étant précédé d'un label explicatif. Ajoutez un bouton d'envoi et un bouton de réinitialisation.

Exercice 21 : réaliser un formulaire d'inscription.

Créer un formulaire (voir l'exemple ci-dessous) qui contient les éléments suivants :

1. Champs de saisie de nom, prénom et e-mail.
2. Deux boutons radio pour le choix du genre.
3. Un champ destiné au transfert de fichier photo dont le format est restreint aux types JPEG ou PNG.
4. Une liste de choix pour les pays.
5. Une liste de choix multiple pour les langages préférés.

Des cases à cocher pour choisir les domaines d'activités.