

---

---

## Travaux Pratiques N°4

---

---

### NB:

Pour la manipulation d'un document source.xml par le modèle model.xsl vous pouvez utiliser la commande :

**xsltproc model.xsl source.xml > sortie.html (.xml ou .txt)**

**Exercice 01:** Soit le fichier xml suivant :

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<biblio>
<livre>
<!-- Élément enfant titre -->
<titre>Les Misérables</titre>
<auteur>Victor Hugo</auteur>
<nb_tomes>3</nb_tomes>
</livre>
<livre>
<titre>L'Assomoir</titre>
<auteur>Émile Zola</auteur>
</livre>
<livre lang="en">
<titre>David Copperfield</titre>
<auteur>Charles Dickens</auteur>
<nb_tomes>3</nb_tomes>
</livre>
</biblio>
```

1. Définissez une feuille de style biblio.xsl afin de produire l'affichage sous la forme ci-dessous : titre, noms des livres en gras, éléments livres séparés par des sauts de ligne, éléments livres séparés par des marques de paragraphe) :

<p><b>Ma Bibliothèque</b></p> <p><b>Titre :</b> Les Misérables <b>Auteur :</b> Victor Hugo <b>Nombre de tomes :</b> 3</p> <p><b>Titre :</b> L'Assomoir <b>Auteur :</b> Émile Zola <b>Nombre de tomes :</b></p> <p><b>Titre :</b> David Copperfield <b>Auteur :</b> Charles Dickens <b>Nombre de tomes :</b> 3</p>
-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

2. Définissez une deuxième feuille de style biblio2.xsl afin de ranger les livres par ordre alphabétique, sans afficher le nombre de tomes lorsqu'il n'est pas renseigné et indiquer lorsqu'un livre est en anglais. Produire l'affichage sous la forme d'une table html comme ci-dessous. Tester.

## Ma Bibliothèque

**Titre:** David Copperfield

**Auteur:** Charles Dickens

**Nombre de tomes:** 3

*Ce livre est en anglais.*

**Titre:** L'Assomoir

**Auteur:** Émile Zola

**Titre:** Les Misérables

**Auteur:** Victor Hugo

**Nombre de tomes:** 3

3. Définissez une troisième feuille de style biblio3.xsl afin de produire l’affichage sous la forme d’une table html comme ci-dessous.

## Ma bibliotheque

Titre	Auteur	Nombre de tomes	Langue
David Copperfield	Charles Dickens	Nombre de tomes :3	<i>Ce livre est en anglais.</i>
L'Assomoir	Émile Zola	-	-
Les Misérables	Victor Hugo	Nombre de tomes :3	-

### Exercice 02:

On considère des fichiers XML qui contiennent une liste de documents sur internet. Chaque document est associé à son type (page web ou image), son url et à une description succincte du document. Ces fichiers XML suivent la DTD docs.dtd suivante :

```
<!ELEMENT docs (doc*) >
<!ELEMENT doc (type, url, description) >
<!ATTLIST doc numero ID #REQUIRED>
<!ELEMENT type (#PCDATA) >
<!ELEMENT url (#PCDATA) >
<!ELEMENT description (#PCDATA) >
```

Par exemple, le fichier exemple.xml ci-dessous contient deux documents (un de type “page web” et un autre de type “image”) :

```
<docs>
<doc numero='1'>
<type>page web</type>
<url>http://www.upmc.fr</url>
<description>Site web de l'UPMC</description>
</doc>
<doc numero='2'>
<type>image</type>
<url>http://www.upmc.fr/logo.jpg</url>
<description>Nouveau logo de l'UPMC</description>
```

```
</doc>
</docs>
```

Remarque : L'exemple ci-dessus contient seulement deux documents, mais vous répondrez aux questions suivantes afin que vos feuilles XSL puissent traiter tous fichiers XML au format docs.dtd.

1. Ecrire une feuille XSL qui transforme un document XML (qui suit la DTD docs.dtd) en un autre document XML contenant seulement une liste d'URL. Ce nouveau document suit la DTD :

```
<!ELEMENT adresses (ad*) >
<!ELEMENT ad (#PCDATA)>
```

Par exemple, le résultat obtenu pour le fichier exemple.xml est :

```
<adresses>
<ad>http://www.upmc.fr</ad>
<ad>http://www.upmc.fr/logo.jpg</ad>
</adresses>
```

2. Ecrire une feuille XSL qui génère un nouveau document XML (qui respecte la DTD docs.dtd) contenant seulement les documents de type « image ».

**Exercice 03:** Soit le fichier xml suivant :

```
<?xml version="1.0" encoding="UTF-8"?>
<papier type="scientifique">
  <titre>Réinterroger les structures documentaires</titre>
  <sousTitre>De la numérisation à l'informatisation</sousTitre>
  <auteur>Stéphane Crozat</auteur>
  <auteur>Bruno Bachimont</auteur>
  <resume>Nous proposons dans cet article d'aborder ...</resume>
  <abstract>In this paper we define...</abstract>
  <motsCles>
    <terme>Ingénierie des connaissances</terme>
    <terme>XML</terme>
    <terme>Document</terme>
  </motsCles>
  <keywords>
    <word>Knowledge engineering</word>
    <word>XML</word>
    <word>Document</word>
  </keywords>
  <publication date="2004-07-05"/>
  <version maj="1" min="0"/>
  <ressource
uriSrc="http://archivesic.ccsd.cnrs.fr/docs/00/06/23/97/PDF/sic_00001015.pdf"/>
</papier>
```

Compléter le fichier XSLT transf.xsl afin de générer, pour chaque élément terme, une instruction SQL d'insertion dans une table relationnelle de schéma : tMotsCles (terme, titre, url) (où terme est le terme sélectionné, titre est le titre du document et url est l'adresse de la ressource associée).

### **Rappel :**

La syntaxe d'insertion de données dans une table relationnelle en SQL : INSERT INTO <Nom de la table> (<Liste ordonnée des propriétés à valoriser>) VALUES (<Liste ordonnée des valeurs à affecter>).

INSERT INTO tMotsCles (terme, titre, url) VALUES ( 'Ingénierie des connaissances', 'Réinterroger les structures documentaires', 'http://archivesic.ccsd.cnrs.fr/docs/00/06/23/97/PDF/sic\_00001015.pdf' );

```
INSERT INTO tMotsCles (terme, titre, url) VALUES ('XML','Réinterroger les structures documentaires',  
'http://archivesic.ccsd.cnrs.fr/docs/00/06/23/97/PDF/sic_00001015.pdf');
```

```
INSERT INTO tMotsCles (terme, titre, url) VALUES ('Document', 'Réinterroger les structures  
documentaires',  
'http://archivesic.ccsd.cnrs.fr/docs/00/06/23/97/PDF/sic_00001015.pdf');
```