

## SOLUTION TP2 : XML

### Exercice 1:

Requête 1 :

=====

Sélectionne les livres publiés par Addison-Wisley après 1991. Pour chacun d'entre eux, retourne un élément book contenant le titre du livre et possédant un attribut year contenant l'année de publication. Notez l'utilisation du where pour ajouter une contrainte sur les éléments sélectionnés par la boucle for.

L'ensemble de ces éléments book est placé dans un élément bib.

Requête 2 :

=====

Construit à partir des données bibliographiques une liste d'éléments result contenant les associations titre de livre - auteur (jointure). Ces éléments sont placés au sein d'une balise results. Attention, l'enchaînement de variables avec des virgules dans un for correspond en fait à plusieurs boucles for imbriquées.

Requête 3 :

=====

Construit pour chaque auteur la liste des livres dont il est un des auteurs. Pour chaque auteur, ces informations sont placées dans une balise result. L'ensemble des ces balises est placé dans un élément results. Une double boucle est nécessaire pour 1/ parcourir la liste des auteurs et 2/ pour chaque auteur, parcourir la liste des livres pour recenser les livres dont il est auteur.

### Exercice 2 :

1.

```
for $serie in doc("collection.xml")//serie
return
```

```
  <serie
    titre="{ $serie/@nom}"
    editeur="{ $serie/editeur/@nom}"
  />
```

2.

```
for $tome in doc("collection.xml")//tome
let $serie := doc("collection.xml")//serie[tome=$tome]
return
  <tome numero="{ $tome/@numero}" serie="{ $serie/@nom}" titre="{ $tome/titre/text()}" />
```

3.

```
for $tome in doc("collection.xml")//tome
let $serie := doc("collection.xml")//serie[tome=$tome]
let $titre := $tome/titre/text()
order by $titre
return
  <tome numero="{ $tome/@numero}" serie="{ $serie/@nom}" titre="{ $titre}" />
```

### Exercise 3:

```
declare function local:sortie_cat($cat as element(categorie)) as element(div) {
  <div>{
    <h3>{ (: element h3 avec le nom de la categorie :)
      $cat/@nom/data()
    }</h3>,
    (: paragraphe avec la description si elle existe :)
    if ($cat/description) then
      <p>{
        $cat/description/text()
      }</p>
    else (),
    (: puis on fait la liste des bookmarks - avec hiérarchie des sous-categories :)
    <ul>{
      for $elt in $cat/element()
      (: $elt represente un element fils de l'element $cat :)
      return
        if ($elt instance of element(categorie))
        then
          (: $elt est une sous categorie, donc appel récursif :)
          <li>{ local:sortie_cat($elt) }</li>
        else (if ($elt instance of element(bookmark))
          then
            (: $elt est un bookmark, on appelle la fonction de la question 1 :)
            <li>{ local:sortie_bookmark($elt) }</li>
          else ()) (: on ne traite pas les autres types d'elements, s'il y en a :)
    }</ul>
  }</div>
};
```

```
(: pour tester la question :  
for $b in //categorie  
return local: sortie_cat($b)  
:)
```