

LABD

mars 2017

Programmer avec XQuery

Téléchargez l'archive `labd-7.tgz`, disponible sur le portail, et décompressez-la dans votre espace de travail. Vous devez obtenir un fichier de nom `saxon9he.jar` ainsi que trois répertoires de noms respectifs `exercice-1`, `exercice-2` et `exercice-3`. Au cours de ce TP, si vous voulez tester vos requêtes en ligne de commande (vous pouvez aussi tout faire dans `editix` mais les messages d'erreur seront très peu explicites), utilisez le jar exécutable `saxon9he.jar` de la façon suivante :

```
java -cp saxon9he.jar net.sf.saxon.Query nom-du-programme.xq
```

Exercice 1 : Exemples de requêtes XQuery

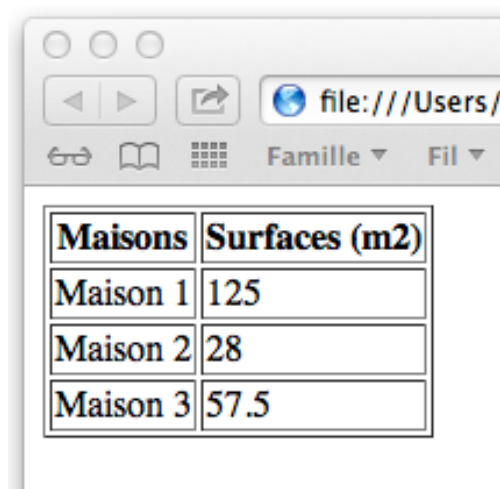
Le répertoire `exercice-1` contient sept requêtes XQuery (les fichiers `query_bibi.xq`) qui travaillent toutes sur le fichier `biblio.xml` disponible dans le même répertoire.

Question 1 : Pour chacune de ces requêtes XQuery, expliquez ce qu'elle fait et anticipez le résultat de son exécution. Vérifier ensuite à l'aide d'`editix` ou de la librairie `saxonica`.

Exercice 2 :

Vous trouverez dans le répertoire `exercice-2` le fichier de description de maisons que vous connaissez-bien de nom `maisons.xml`.

Question 1 : Écrire un programme XQuery qui, à partir de ce fichier XML, calcule, pour chaque maison, sa superficie totale. La sortie du programme sera un fichier HTML dont la visualisation correspond à la capture écran suivante :



| Maisons | Surfaces (m2) |
|----------|---------------|
| Maison 1 | 125 |
| Maison 2 | 28 |
| Maison 3 | 57.5 |

Exercice 3 : Pas d’orchidée pour XQuery¹

On considère les trois fichiers `xml` suivants, disponibles dans le répertoire `exercice-3`

`plant_catalog.xml` est un catalogue de plantes;

`plant_families.xml` qui indique à quelle famille appartiennent certaines plantes;

`plant_order.xml` est une commande de plantes.

Question 1 : Donnez un programme XQuery qui produit à partir des fichiers `plant_catalog.xml` et `plant_families.xml` un document XML en ajoutant dans chaque élément `PLANT` apparaissant dans `plant_catalog.xml` un élément `FAMILY` qui donne le nom de la famille à laquelle appartient la plante comme dans l’exemple ci-dessous :

```
<PLANT>
  <COMMON>Bloodroot</COMMON>
  <BOTANICAL>Sanguinaria canadensis</BOTANICAL>
  <ZONE>4</ZONE>
  <LIGHT>Mostly Shady</LIGHT>
  <PRICE>$2.44</PRICE>
  <AVAILABILITY>031599</AVAILABILITY>
  <FAMILY>Papaveraceae</FAMILY>
</PLANT>
```

Question 2 : Donnez un programme XQuery qui classe et regroupe les éléments `PLANT` du fichier `plant_catalog.xml` en fonction du contenu de leur élément `LIGHT`. Vous devez obtenir un document identique au fichier `exposure.xml` disponible dans le répertoire `exercice-3`.

Question 3 : Donnez un programme XQuery qui réalise les 2 opérations des questions ?? et ?? en classant en outre les éléments `LIGHT` par ordre alphabétique du contenu des éléments `EXPOSURE` et en classant les éléments `PLANT` par ordre alphabétique du contenu des éléments `COMMON`.

Question 4 : Donnez un programme XQuery qui calcule le montant total de la commande décrite dans `plant_order.xml` en donnant le résultat dans un élément `PRICE`, on doit obtenir :

```
<PRICE>663.2</PRICE>
```

1. Ces questions avaient été réalisées en cours en XSLT.