

# TD XSLT (2)

## Exercice 1 : Examen Juin 2007

Soit le document XML suivant :

```
<A>
  <B> <X> x1 </X> </B>
  <B> <X> x2 </X> </B>
  <C>
    <B> <X> x3 </X> </B>
    <Y> y1 </Y>
  </C>
</A>
```

**Question 1.** Ecrire un fichier XSL qui sélectionne l'élément B dont l'élément X vaut « x2 ». Le résultat doit respecter la DTD suivante :

```
<!ELEMENT resultat (B *) />
<!ELEMENT B (X) />
<!ELEMENT X (#PCDATA) />
```

**Question 2.** Ecrire un fichier XSL qui sélectionne les éléments X et Y contenus dans des éléments C. Le résultat doit respecter la DTD suivante :

```
<!ELEMENT resultat (X*, Y*) />
<!ELEMENT X (#PCDATA) />
<!ELEMENT Y (#PCDATA) />
```

**Question 3.** Ecrire un fichier XSL qui renvoie les éléments B non contenus dans un élément C. Le résultat devra respecter la DTD suivante :

```
<!ELEMENT resultat (B *) />
<!ELEMENT B (X) />
<!ELEMENT X (#PCDATA) />
```

**Question 4.** Ecrire un fichier XSL qui renvoie (en passant par une sélection sur le document d'origine) :

```
<resultat>
  <Y> y1 </Y>
  <X> x3 </X>
</resultat>
```

## Exercice 2 : Examen Juin 2008 Session 1

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.

**Question 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>

```

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

**Question 3 :** Ecrire une feuille XSL qui transforme le document XML en une page HTML indiquant le nombre de documents de type « image » et le nombre de documents de type « page web » dans le fichier XML. Par exemple, pour le fichier exemple.xml, le code HTML obtenu est :

```

<HTML>
  <p>Nombre d'images : 1</p>
  <p>Nombre de pages web : 1 </p>
</HTML>

```

**Question 4.** Ecrire une feuille XSL qui transforme un document XML (suivant la DTD docs.dtd) en un page HTML qui affiche un tableau HTML contenant une ligne par document. Pour chaque document, on affiche seulement son numéro (première colonne du tableau) et son url (deuxième colonne du tableau) : les urls des documents de type « page web » sont affichées comme des liens cliquables ayant la description du document comme valeur du lien, et les urls des documents de type « image » sont remplacées par l'image avec un attribut alt contenant la description du document.

Par exemple, pour le fichier exemple.xml, le code HTML obtenu est :

```

<HTML>
  <TABLE BORDER="1">
    <TR><TH>Numero</TH><TH>URL</TH></TR>
    <TR>
      <TD>1</TD>
      <TD><A href="http://www.upmc.fr">Site web de l'UPMC</A></TD>
    </TR>
    <TR>
      <TD>2</TD>
      <TD><IMG width="200" src="http://www.upmc.fr/logo.jpg"
          alt="Nouveau logo de l'UPMC"></TD>
    </TR>
  </TABLE>
</HTML>

```