

# TD XSLT (1)

## Exercice 1 : XSLT

Soit le document XML suivant :

```
<B>
  <R> <a>a1</a> <b>b1</b> </R>
  <R> <a>a2</a> <b>c1</b> </R>
  <S> <a>a1</a> <c>c1</c> </S>
</B>
```

**Question 1)** Quel est le résultat obtenu par l'application des feuilles de styles suivantes sur ce document :

```
<?xml version="1.0"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0">
  <xsl:template match="/">
    CORPS_DU_TEMPLATE
  </xsl:template>
</xsl:stylesheet>
```

où CORPS\_DU\_TEMPLATE est remplacé par :

- a) <resultat><xsl:value-of select="//a"/></resultat>
- b) <resultat><xsl:copy-of select="//a"/></resultat>
- c) <resultat><xsl:apply-templates select="//a"/></resultat>

**Question 2)** Exprimer en XSL la requête qui donne tous les éléments **a** du document. Le résultat doit valider la DTD suivante :

```
<!ELEMENT resultat (a)* >
<!ELEMENT a (#PCDATA) >
```

**Question 3)** Exprimer en XSL la requête qui donne tous les éléments de B qui ont un élément **a** contenant le texte 'a1'. Le résultat de la requête est:

```
<resultat>
  <R> <a>a1</a> <b>b1</b> </R>
  <S> <a>a1</a> <c>c1</c> </S>
</resultat>
```

**Question 4)** Exprimer en XSL la requête qui donne les éléments **R** qui ont un élément **a** identique à un élément **a** de **S** (i.e. { R tel que (a de R) = (a de S) }). Le résultat de la requête est:

```
<resultat>
  <R> <a>a1</a> <b>b1</b> </R>
</resultat>
```

## Exercice 2 : Base de donnée projet et XSLT

Soit le fichier XML suivant, correspondant à un extrait partiel de la base de données du projet PHP. Certains attributs ont été volontairement omis.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE videoexpress SYSTEM "dtdProjet.dtd">
<videoexpress>
  <film couleur="Couleurs">
    <titre>Le Charme Discret de la Bourgeoisie</titre>
    <realisateur>
      <prenomr>Luis</prenomr> <nomr>Bunuel</nomr>
    </realisateur>
    <annee>1972</annee>
    <synopsis>Une étude cynique ...</synopsis>
    <acteur>
      <prenomac>Fernando</prenomac>
      <nomac>Rey</nomac>
    </acteur>
  </film>
</videoexpress>
```

```

</acteur>
<acteur>
  <prenomac>Paul</prenomac>
  <nomac>Frankeur</nomac>
</acteur>
<cassette support="DVD" statut="Empruntée">
  <empres>
    <dateempres>18/05/2005</dateempres>
    <abonne>
      <noma>Dupont</noma>
      <prenoma>Jean</prenoma>
      <adresse>
        <norue>2, avenue de l'Europe</norue>
        <codepostal>75005</codepostal>
        <ville>Paris</ville>
      </adresse>
      <telephone>01 02 03 04 05</telephone>
      <nbcassette>1</nbcassette>
    </abonne>
  </empres>
</cassette>
</film>
</videoexpress>

```

On va reconstruire en XSLT des pages HTML représentant des informations sur le film extraites de cette base XML

### Construction de la page films

Exprimer en XSL la requête qui donne tous les films du document. Le résultat sera un fichier HTML semblable à :

```

<HTML>
  <BODY>
    <H1>Titres disponibles : </H1>>
    <TABLE>
      <TR>
        <TD>Le Charme Discret de la Bourgeoisie </TD>
      </TR>
      <TR>
        <TD>Réalisateur</TD><TD>Luis Bunuel</TD>
      </TR>
      <TR>
        <TD>Année</TD><TD>1972</TD>
      </TR>
      <TR>
        <TD>Synopsis</TD>
        <TD>Une étude cynique ...</TD>
      </TR>
      <TR>
        <TD>Acteurs : </TD>
        <TD>
          <OL>
            <LI>Fernando Rey </LI>
            <LI>Paul Frankeur </LI>
          </OL>
        </TD>
      </TR>
    </TABLE>
  </BODY>
</HTML>

```

Pour cela, on construira plusieurs templates XSLT :

- Un pour le niveau global
- Un pour le niveau film
- Un pour le niveau réalisateur
- Un pour le niveau acteur