

## Cartouche du document

Année : ING 1

Matière : Structuration de l'information : XML

Activité : Travail dirigé

## Objectifs

Cet ensemble d'exercices a pour buts :

- 1) les parcours d'arbre avec l'ordre `xsl:for-each`
- 2) les tests avec l'ordre `xsl:if`
- 3) les aiguillages avec les ordre `xsl:choose`

## Sommaire des exercices

1 - Des bulletins plus sophistiqués

2 - Des bulletins encore plus sophistiqués

## Corps des exercices

### 1 - Des bulletins plus sophistiqués

#### Énoncé :

On reprend l'arbre XML suivant :

```
<?xml version="1.0" encoding="iso-8859-1"?>
<!-- entête commune à tous les fichiers XML -->
<etudiant prenom="Maude" nom="Eme" annee="ing2">
  <ou adresse="1 rue du bo" ville="Paris" cp="75000" />
  <matiere libelle="info">
    <appreciation>tu peux mieux faire</appreciation>
    <note epreuve="ds n° 1" valeur="10.5" />
    <note epreuve="ds n° 2" valeur="11" />
    <note epreuve="ds n° 3" valeur="12" />
  </matiere>
  <matiere libelle="comptabilité générale">
    <appreciation>belle prestation</appreciation>
    <note epreuve="ds" valeur="15" />
  </matiere>
</etudiant>
```

On vous rappelle qu'il contient des informations scolaires de l'étudiante Maude Eme.

#### Question 1)

#### Énoncé de la question

Ecrire une feuille XSL qui permettra d'afficher dans l'ordre :

- le nom et le prénom de l'étudiant;
- matière par matière, l'appréciation

- matière par matière, les notes obtenues.

Chacune des rubriques ci-dessus devra être annoncée par un titre.

On utilisera l'instruction xsl:for-each.

### Solution de la question

Se conférer à la réponse globale de l'exercice

## Question 2)

### Énoncé de la question

On vous demande d'écrire une deuxième version de la feuille XSL de la question précédente en :

- affichant les matières par ordre alphabétique;
- affichant les notes des épreuves d'une matière par ordre décroissant.

### Solution de la question

Se conférer à la réponse globale de l'exercice

## Réponse globale à tout l'exercice

```
<?xml version="1.0" encoding="iso-8859-1"?>
<!-- entête commune à tous les fichiers XML -->
<!-- Début de la feuille XSL -->
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/
XSL/Transform">
<xsl:output method="html" indent="yes" encoding="utf-8" />
<!-- le patron du noeud etudiant -->
<xsl:template match="etudiant">
<html>
<head>
</head>
<body>
<h1>Bulletin de <xsl:value-of select="@prenom" /><xsl:text> </
xsl:text><xsl:value-of select="@nom" /></h1>
<h2>Appréciations</h2>
<xsl:for-each select="matiere" >
<xsl:sort select="@libelle"/>
<xsl:value-of select="@libelle"/><xsl:text> : </
xsl:text><xsl:value-of select="appreciation"/><br/>
</xsl:for-each>
<br/>
<h2>Notes</h2>
<xsl:for-each select="matiere" >
<xsl:sort select="@libelle"/>
<xsl:value-of select="@libelle"/><br/>
<table border="1">
<xsl:apply-templates select="note">
```

```

    <xsl:sort select="@valeur" order="descending" />
  </xsl:apply-templates>
</table>
<br/><br/>
</xsl:for-each>
</body>
</html>
</xsl:template>
<!-- le patron du noeud note -->
<xsl:template match="note">
<tr>
  <td><xsl:value-of select="@epreuve" /></td>
  <td><xsl:value-of select="@valeur" /></td>
</tr>
</xsl:template>
<!-- fin de la feuille xsl-->
</xsl:stylesheet>

```

## 2 - Des bulletins encore plus sophistiqués

### Énoncé :

On vous un nouvel l'arbre XML :

```

<?xml version="1.0" encoding="iso-8859-1"?>
<!-- entête commune à tous les fichiers XML -->
<etudiant prenom="Alain" nom="Térieur" annee="ing1">
  <ou adresse="3 rue du port" ville="Cergy" cp="95001" />
  <matiere libelle="info">
    <appreciation>Pas mal du tout</appreciation>
    <note epreuve="algo" valeur="13.5" />
    <note epreuve="prog C" valeur="14" />
    <note epreuve="ADO" valeur="12" />
  </matiere>
  <matiere libelle="comptabilité générale">
    <appreciation>très belle prestation</appreciation>
    <note epreuve="ds" valeur="18" />
  </matiere>
  <matiere libelle="Projet Info 1">
    <note epreuve="projet" valeur="18" />
  </matiere>
</etudiant>

```

### Question 1)

#### Énoncé de la question

On vous demande d'écrire une troisième version de la feuille XSL de l'exercice précédent en n'affichant les renseignements d'une matière que si son appréciation existe.

On utilisera l'ordre xsl:if.

#### Solution de la question

On reprend le template étudiant :

```

<!-- le patron du noeud etudiant -->
<xsl:template match="etudiant">
<html>
  <head>
</head>
  <body>
    <h1>Bulletin de <xsl:value-of select="@prenom" /><xsl:text> </
xsl:text><xsl:value-of select="@nom" /></h1>
    <!-- le titre est mis s'il y a au moins une appreciation -->
    <xsl:if test="matiere/appreciation">
      <h2>Appréciations</h2>
    </xsl:if>
    <xsl:for-each select="matiere" >
      <xsl:sort select="@libelle"/>
      <xsl:if test="appreciation">
        <!-- les renseignements sur la matière ne sont mis que si
l'appreciation existe -->
        <xsl:value-of select="@libelle"/><xsl:text> : </
xsl:text><xsl:value-of select="appreciation"/><br/>
      </xsl:if>
    </xsl:for-each>
    <br/>
    <h2>Notes</h2>
    <xsl:for-each select="matiere" >
      <xsl:sort select="@libelle"/>
      <!-- les renseignements sur la matière ne sont mis que si
l'appreciation existe -->
      <xsl:if test="appreciation">
        <xsl:value-of select="@libelle"/><br/>
        <table border="1">
          <xsl:apply-templates select="note">
            <xsl:sort select="@valeur" order="descending"/>
          </xsl:apply-templates>
        </table>
        <br/><br/>
      </xsl:if>
    </xsl:for-each>
  </body>
</html>
</xsl:template>

```

## Question 2)

## Énoncé de la question

On vous demande de rectifier la feuille XSL précédente, en affichant toutes les matières. Mais pour les matières dont l'appréciation n'existe pas, on affiche "Pas d'appréciation".

On utilisera l'ordre xsl:choose.

## Solution de la question

On reprend le template étudiant :

```
<!-- le patron du noeud etudiant -->
<xsl:template match="etudiant">
<html>
  <head>
  </head>
  <body>
  <h1>Bulletin de <xsl:value-of select="@prenom" /><xsl:text> </
xsl:text><xsl:value-of select="@nom" /></h1>
  <!-- le titre est mis s'il y a au moins une appreciation -->
  <xsl:if test="matiere/appreciation">
    <h2>Appréciations</h2>
  </xsl:if>
  <xsl:for-each select="matiere" >
    <xsl:sort select="@libelle"/>
    <xsl:choose>
      <xsl:when test="appreciation">
        <xsl:value-of select="@libelle"/><xsl:text> : </
xsl:text><xsl:value-of select="appreciation"/><br/>
      </xsl:when>
      <xsl:otherwise>
        <xsl:value-of select="@libelle"/><xsl:text> : pas
        d'appréciation</xsl:text><br/>
      </xsl:otherwise>
    </xsl:for-each>
  <br/>
  <h2>Notes</h2>
  <xsl:for-each select="matiere" >
    <xsl:sort select="@libelle"/>
    <xsl:value-of select="@libelle"/><br/>
    <table border="1">
      <xsl:apply-templates select="note">
        <xsl:sort select="@valeur" order="descending"/>
      </xsl:apply-templates>
    </table>
  <br/><br/>
</xsl:for-each>
```

```

</body>
</html>
</xsl:template>

```

### Question 3)

#### Énoncé de la question

On change encore la feuille XSL. Pour chaque matière, on affiche les notes sous forme d'une liste sans le nom des épreuves. Pour notre exemple, dans la matière info, on doit afficher **15.5, 14, 18**.

On utilisera les fonctions last() et position() dans un test xsl:if en remarquant que la seule note qui n'est pas suivie d'un virgule est la dernière note.

#### Solution de la question

On reprend le template étudiant :

```

<!-- le patron du noeud etudiant -->
<xsl:template match="etudiant">
<html>
<head>
</head>
<body>
<h1>Bulletin de <xsl:value-of select="@prenom" /><xsl:text> </
xsl:text><xsl:value-of select="@nom" /></h1>
<!-- le titre est mis s'il y a au moins une appreciation -->
<xsl:if test="matiere/appreciation">
<h2>Appréciations</h2>
</xsl:if>
<xsl:for-each select="matiere" >
<xsl:sort select="@libelle"/>
<xsl:if test="appreciation">
<!-- les renseignements sur la matière ne sont mis que si
l'appreciation existe -->
<xsl:value-of select="@libelle"/><xsl:text> : </
xsl:text><xsl:value-of select="appreciation"/><br/>
</xsl:if>
</xsl:for-each>
<br/>
<h2>Notes</h2>
<xsl:for-each select="matiere" >
<xsl:sort select="@libelle"/>
<!-- les renseignements sur la matière ne sont mis que si
l'appreciation existe -->
<xsl:if test="appreciation">
<xsl:value-of select="@libelle"/><br/>
<table border="1">

```

```
<xsl:for-each select="note">
  <xsl:value-of select="@valeur"/>
  <xsl:if test="not(last() = position())">
    <xsl:text>,</xsl:text>
  </xsl:if>
</xsl:for-each>
</table>
<br/><br/>
</xsl:if>
</xsl:for-each>
</body>
</html>
</xsl:template>
```