

SÉANCE 4 : CONDITIONS ET BOUCLES

I-xsl:if

```
<xsl:template match="mon_element">
  <xsl:if test="chemin_Xpath_a_verifier">
    <xsl:copy-of select="."/>
  </xsl:if>
</xsl:template>
```

II-xsl:choose, xsl:when, xsl:otherwise

```
<xsl:template match="mon_element">
  <xsl:choose>
    <xsl:when test="chemin_xpath_a_verifier">
      [motifs à appliquer]
    </xsl:when>
    <xsl:otherwise/>
  </xsl:choose>
</xsl:template>
```

III- xsl:for-each, xsl:sort et xsl:for-each-group

A-for-each

L'instruction *xsl:for-each* itère sur les nœuds sélectionnés par son attribut *@select* et applique le modèle de son contenu à chacun des éléments du nœud.

```
<xsl:template match="mon_element">
  <xsl:for-each select="sous_elements_a_traiter">
    [motifs à appliquer]
  </xsl:for-each>
</xsl:template>
```

B-sort

```
<xsl:template match="mon_element">
  <xsl:for-each select="sous-éléments">
    <xsl:sort select="cle_tri" order='ascending|descending' />
    <xsl:copy-of select="."/>
  </xsl:for-each>
</xsl:template>
```

A) For-each-group

L'instruction *xsl:for-each-group* itère sur les groupes de nœuds sélectionnés par son attribut *@select* applique le modèle de son contenu à chacun d'entre eux, tandis que l'attribut *@group-by* sert de clé. Dans le motif, la fonction *xpath current-grouping-key()* permet de retourner la valeur de la clé de regroupement de la boucle en cours.

```
<xsl:template match="mon_element">
  <xsl:for-each-group select="sous-elements"
    group-by="cle_regroupement">
    [motifs à appliquer]
  </xsl:for-each-group>
</xsl:template>
```