

Exemplier XSLT : Fonctions vues dans les sessions 1 et 2

xsl:apply-templates

Permet d'appliquer un motif à des éléments du document source.

```
<xsl:template match="/racine">
  <xsl:apply-templates select="élément_a_traiter"/>
</xsl:template>
```

xsl:copy-of

Permet de copier le contenu d'un nœud sélectionné sans le modifier.

```
<xsl:template match="mon_element">
  <xsl:copy-of select="sous_element"/>
</xsl:template>
```

xsl:value-of

Permet d'extraire le texte d'un nœud

```
<xsl:template match="mon_element">
  <xsl:value-of select="mon_sous_element"/>
</xsl:template>
```

xsl:number

Permet de produire automatiquement une numérotation

```
<xsl:template match="mon_element_xml">
<xsl:number count="element_a_compter" level="type_numerotation"
format="format_numerotation"/>
</xsl:template>
```

xsl:if

Contient un motif qui est exécuté si et seulement si l'expression XPath dans l'attribut obligatoire @test est vraie.

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

xsl:choose, xsl:when, xsl:otherwise

Permet de mettre en place des conditions pour l'exécution des motifs

```
<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>
```

xsl:for-each

Itère sur les nœuds sélectionnés par @select et applique le motif du contenu à chaque élément.

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

xsl:sort

Trie les nœuds contextuels. À utiliser dans un xsl:for-each ou xsl:apply-templates.

```
<xsl:template match="mon_element">
  <xsl:for-each select="sous_elements">
    <xsl:sort select="cle_tri" order="ascending"/>
    <xsl:copy-of select="."/>
  </xsl:for-each>
</xsl:template>
```

Principaux attributs

- @select : clé de tri
- @data-type : text (défaut) ou number
- @order : ascending (défaut) ou descending
- @case-order : upper-first ou lower-first

xsl:for-each-group

Itère sur des groupes de nœuds regroupés selon un critère par @group-by

```
<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>
```

Rappel XPath:

- . : élément courant
- .. : parent
- *[position()=1] : premier enfant
- contains(@wit, 'Z') : contient la lettre Z dans l'attribut wit
- ends-with(., '.xml') : la chaîne de caractères se termine par la séquence .xml
- count(l) : compte le nombre de vers