

ENC/ XSLT/ TD XML vers XML (séance 5)

Jean-Damien G      

2025

Environnement de travail

- Document XML : `class_5_journal_lefevre.xml`.
- Vous travaillez avec un document XML-TEI contenant une   dition de plusieurs chapitres du *Journal du chancelier Le F  vre*. Cet exercice a pour objectif de manipuler les donn  es de ce document et de le transformer en plusieurs fichiers XML.

1. Cr  er des documents de sortie individuels

- Chaque `<div>` du `<body>` doit   tre enregistr  e dans un fichier s  par  . Le nom du fichier sera bas   sur l'attribut `@n` de la balise `<div>` (par exemple, `chapitre1.xml`, `chapitre2.xml`, etc.).
- Pour cela, vous pouvez utiliser:
 - `<xsl:for-each/>`
 - `<xsl:result-document/>`

2. Modifier et enrichir le `<teiHeader>`

Pour chaque document de sortie :

- **2.1 Dans `<titleStmt>` :**
 - Ajouter un attribut `@level="m"`    l'  l  ment `<title>` existant.
 - Ajouter un nouvel   l  ment `<title>` contenant le texte de `<head>` de la balise `<div>` du chapitre courant, avec un attribut `@level="a"`.
- **2.2 Trier les r  f  rences bibliographiques `<biblStruct>` par ordre croissant des dates dans les balises `<date>`.**
- Pour cela, vous pouvez utiliser:
 - `<xsl:for-each/>`.

- `<xsl:sort/>`.
- **2.3 Modifier le `<revisionDesc>` :**
 - Ajouter une balise `<change>` avec un attribut `@when` contenant la date du jour (utilisez la fonction `current-date()`).
- **2.4 Conserver le reste du `<teiHeader>` sans modification.**

3. Modifier le `<text>`

- Numéroter chaque paragraphe `<p>` en utilisant l'attribut `@n`. Les numéros doivent correspondre à l'ordre d'apparition dans chaque `<div>`.
- Ajouter une nouvelle `<div>` à la fin de chaque fichier contenant un index des noms de personnes extraits des balises `<persName>` de ce chapitre. Chaque nom ne doit apparaître qu'une seule fois.
- Pour cela, vous pouvez utiliser:
 - `<xsl:for-each-group/>`.
 - `current-grouping-key()`.