

## TP4 XSLT

### Tous aux fourneaux

A l'issue du TP3 vous devez disposer d'un document XML qui regroupe :

- La définition des recettes et de leurs ingrédients
- La définition des auteurs des recettes
- La composition des catégories en sous catégories comportant des recettes.

**Objectif :** A partir du document XML qui décrit les recettes de cuisine, leurs ingrédients, les auteurs, les sous catégories et les catégories de recettes de cuisine de Ôdélices, nous allons maintenant produire une page XHTML puis, plus tard, une série de pages.

### 1. Feuilles de Style XSL

#### 1.1 Première transformation

- Construisez une feuille de style XSL permettant de produire une page XHTML qui liste les noms des recettes de cuisine.
- Structurez votre feuille XSL autour d'un template nommé qui effectue le travail :

```
<xsl:stylesheet version="1.0"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
xmlns:fo="http://www.w3.org/1999/XSL/Format">

  <xsl:template match="/">
    <html>
      <body>
        <xsl:call-template name="liste-des-recettes" />
      </body></html>
    </xsl:template>

    <xsl:template name="liste-des-recettes">
      ... construire la liste des recettes de cuisine.....

    </xsl:template>
```

- Utilisez la clause xsl:output pour produire des documents codés en **iso-8859-1** et bien associés à la DTD du XHTML en version stricte. Faites en sorte que la page produite soit valide vis à vis de la norme du W3C.

## 1.2 Lister les données

### a) Données sur les auteurs

**Etape 1 :** Ajoutez une règle nommée pour produire les fiches de chaque auteur avec toutes les informations disponibles (utilisez par exemple un `<h2>` par auteur).

**Etape 2 :** Faites en sorte que chaque auteur ait un identifiant dans le fichier XML (attribut id de type ID). Utilisez cet identifiant et l'attribut id du XHTML (sur l'élément de votre choix) pour que des ancres soient définies dans votre fichier XHTML. L'URL ci-dessous doit pouvoir vous amener directement à la description de l'auteur en question :

```
votre-page.html#identifiant-auteur
```

**Etape 3 :** Dans une nouvelle règle nommée, utilisez le principe des ancres pour construire, au début de votre fichier XHTML, une liste des auteurs (avec `<ul><li>..  
</li>..  
</ul>`). Dans chaque élément de cette liste, vous placerez un lien permettant un accès direct à la fiche d'un auteur :

```
<li><a href="#identifiant-de-xyz">XYZ</a></li>
```

### b) Données sur les recettes et les ingrédients

**Etape 4 :** Faites la même chose pour les recettes et les ingrédients. Conseil : pour l'instant, ne traitez pas finement les champs de description contenant du texte enrichi. Vous pouvez simplement construire un paragraphe :

```
...
    <p><xsl:value-of select="string(resume)" /></p>
...
```

## 2. Construire un site Web statique

### 2.1 Sorties multiples

**Etape 1 :** La version 1.1 de XSLT propose l'utilisation de l'élément `<xsl:document>` pour la génération de plusieurs documents. La version 2.0 de XSLT propose l'utilisation d'un autre élément `<xsl:result-document>` pour la production de plusieurs documents. Consultez la documentation sur XSLT1.1<sup>1</sup> pour comprendre le fonctionnement de l'élément `<xsl:document>` et la documentation sur XSLT2.0<sup>2</sup> pour comprendre le fonctionnement de `<xsl:result-document>`. Utilisez l'élément qui vous semble le plus approprié.

<sup>1</sup> <http://www.w3.org/TR/xslt11/#multiple-output>

<sup>2</sup> <http://www.w3.org/TR/xslt20/#element-result-document>

**Etape 2 :** Modifiez votre feuille de style et utilisez cette directive pour produire quatre fichiers XHTML :

- index.html : page d'accueil du site,
- recettes.html : description des recettes de cuisine,
- ingrédients.html : description des ingrédients,
- auteurs.html : description des auteurs,

**Attention :** Les liens entre les recettes et auteurs ainsi que les liens entre les recettes et ingrédients doivent toujours fonctionner.

**Etape 3 :** En utilisant l'élément `<xsl:for-each ..>` de XSL, faites en sorte de produire autant de fichiers XHTML que de recettes.

**Etape 4 :** Même principe pour les auteurs (une page XHTML par auteur) et pour les ingrédients.

## 2.2 Traiter les catégories et les sous-catégories

Ajoutez à votre feuille XSL la production d'une page XHTML par catégorie. Ces pages doivent présenter la catégorie, sa description, les sous-catégories et mettre en place les liens entre sous-catégories et recettes de cuisine.

**Conseil :** N'hésitez pas à construire une nouvelle feuille que vous allez ensuite inclure dans la feuille principale (clause `<xsl:include ...>`).

## 2.3 Ajouter les liaisons inverses

**Etape 1:** Ajoutez à la fiche de chaque auteur, la liste des recettes de cuisine qu'il a proposées. Faites en sorte que cette liste soit constituée de liens vers les descriptions de ces recettes de cuisine.

**Conseil :** Utilisez une variable XSL associée à une expression XPATH qui va, pour chaque auteur, extraire les recettes.

**Etape 2:** Ajoutez à la description de chaque recette, la liste des sous-catégories dans lesquelles elle apparaît. Faites en sorte que cette liste soit constituée de liens vers les descriptions de ces sous-catégories.

**Etape 3:** Ajoutez à la description de chaque ingrédient, la liste des recettes dans lesquelles il est utilisé. Faites en sorte que cette liste soit constituée de liens vers les descriptions des recettes.

## 3. Améliorons nos pages

### 3.1 Ajouter un menu

Ajoutez dans chaque page XHTML un menu statique (voir le TP1 pour la création d'un menu) qui permet d'accéder

- à la page d'accueil
- à la liste des auteurs
- à la liste des recettes de cuisine
- à la liste des ingrédients
- à la liste des catégories et sous catégories

### 3.2 Des requêtes XPath plus compliquées

**Etape 1:** Commencez par construire une règle nommée avec paramètres pour construire une liste de liens à partir d'un ensemble de nœuds.

```
<xsl:template name="faire-une-liste">
  <xsl:param name="objets"/>
  <xsl:param name="nom"/>

  <!-- produire une liste de liens vers ces objets-->
</xsl:template>
```

**Etape 2:** Utilisez cette règle pour construire la liste des recettes de 5 ingrédients.

**Etape 3:** Construire la liste des auteurs qui n'ont proposé que des recettes de difficulté « facile ».

### 3.3 Un traitement plus rapide

Pour améliorer l'efficacité de vos transformations XSL, utilisez la clause `xsl:key`<sup>3</sup> afin de pouvoir retrouver facilement les recettes à partir des identifiants des auteurs. Profitez-en pour lister (avec la règle nommée précédente), les auteurs qui ont publié des recettes faciles.

## 4. Transformation de XML vers XML

Récupérez le fichier XML décrivant les catégories, les recettes, les ingrédients et les auteurs des recettes (fichier `donnees-recettes.zip`).

A la fin de ce TP, vous avez dû concevoir une DTD et un document XML décrivant les recettes. Vous pouvez maintenant, à l'aide d'une feuille de style XSL, enrichir votre description en exploitant les informations qui se trouvent dans le document XML récupéré. Attention, ce document n'a pas de structure particulière. Vous devez donc le transformer pour respecter votre DTD.

Voilà un petit exemple (à titre d'illustration) de feuille de style XSL permettant de reconstruire une unité à partir des informations du fichier XML donné.

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

  <xsl:output method="xml" indent="yes" />

  <xsl:template match="/">
    <recettes>
      <xsl:for-each select="//objet[@type = 'recette']">
        <recette>
          <id><xsl:value-of select="@id"/></id>
          <nom><xsl:value-of select="info[@nom='nom']/@value"/></nom>
```

<sup>3</sup> <http://www.w3.org/TR/xslt20/#xsl-key>

```
        <nbre_personnes><xsl:value-of
select="info[@nom='nbre_personnes']/@value"/></nbre_personnes>
        <Preparation><xsl:copy-of
select="info[@nom='Préparation']/*"/></Preparation>
    </recette>
</xsl:for-each>
</recettes>
</xsl:template>
</xsl:stylesheet>
```