

Editions scientifiques en TEI: concepts et outils

Laurent Pugin
laurent.pugin@rism-ch.org

École d'automne 2016
Modules d'approfondissement

Les fichiers utilisés pour ces exercices ainsi que ce document sont accessibles à:

<https://github.com/lpugin/heg2016-tei>

Un zip peut être téléchargé directement à:

<https://github.com/lpugin/heg2016-tei/archive/master.zip>

Exercice 1 – Analyse

Comparez le matériel éditorial des projets suivants:

- <http://www.janeausten.ac.uk/index.html>
- <http://www.chopinonline.ac.uk/ocve/>
- http://research.cch.kcl.ac.uk/proust_prototype/
- <https://www.briefedition.alfred-escher.ch/>
- <http://godwindiary.bodleian.ox.ac.uk/>
- <http://www.sd-editions.com/bayeux/index.html>
- <http://codexsinaiticus.org/en/>
- <http://dme.mozarteum.at/DME/main/cms.php?tid=132&sec=libkat&l=2>

Exercice 2 – XML

Lequel des éléments suivants est bien formé et lequel ne l'est pas ?

1. `<seg>some text</seg>`
2. `<seg><foo>some</foo> <bar>text</bar></seg>`
3. `<seg><foo>some <bar></foo> text</bar></seg>`
4. `<seg type="text">some text</seg>`
5. `<seg type='text'>some text</seg>`
6. `<seg type=text>some text</seg>`
7. `<seg type = "text">some text</seg>`
8. `<seg type="text">some text<seg/>`
9. `<seg type="text">some text<gap/></seg>`
10. `<seg type="text">some text< /seg>`
11. `<seg type="text">some text</Seg>`

Exercice 3 – Validation TEI

Pour cet exercice nous pouvons utiliser le service de validation en ligne du site “TEI By Example” (TBE) <http://tei.byexample.org/xquery/TBEvalidator.xq> . Vous pouvez aussi utiliser le logiciel Oxygen ou tout autre éditeur XML intégrant un outil de validation.

Essayer de valider le fichier “proust.xml”. Que se passe-t-il? Le document est-il bien formé? Et valide? Essayer de corriger les erreurs.

Exercice 4 – Création de données

Pour cet exercice nous allons utiliser l’outil de conversion “OxGarage” <http://www.tei-c.org/oxgarage/>

Essayer de convertir le fichier “La_Mare_au_Diable.docx” en TEI P5. Qu’obtenez-vous? Le fichier est-il valide? Utiliser les outils de l’exercice précédent pour vérifier.

Exercice 5 – Transformation via XSLT

Pour cet exercice nous appliquer une transformation XSLT au fichier “proust.xml”. Pour cela, il faut ajouter la référence `<?xml-stylesheet type="text/xsl" href="proust.xsl"?>` avant l’élément racine `<tei>`. Voyez le résultat en ouvrant le fichier “proust.xml” sous Firefox.

Si vous n’avez pas Firefox vous pouvez utiliser le service <http://www.w3schools.com/xsl/tryxslt.asp?xmlfile=catalog&xsltfile=catalog> et y copier le contenu des fichiers “proust.xml” et proust.xsl”.

Essayez maintenant de modifier le fichier “proust.xsl” pour transformer les `<lb/>` en `
` ou encore le `<hi>` (et son contenu) en `<sup>`.

Essayez d’ajouter des balise `` dans le fichier “proust.xml” et de le rendre en barré dans le résultat.

Exercice 6 – Utilisation d’un outil d’annotation

Nous utiliserons pour cet exercice l’outil “TILE” <http://mith.umd.edu/tile/sandbox/> . Il permet de faire correspondre un texte à une image soit manuellement soit de manière semi-automatique. Y parvenez-vous?

Essayer aussi d’exporter le résultat sous le format original proposé. Le fichier est-il valide? Sinon, pourquoi?