

TD-Schemas

Exercice 1 : Déclarations d'éléments

Solution : schema1.xsd

Nous allons reprendre le texte du TD sur les DTD, mais cette fois-ci avec un schéma. On ne déclarera, pour le moment, que des types de chaînes de caractères.

Rédiger un Schema XML pour une bibliographie. Cette bibliographie :

- contient des livres et des articles ;
- les informations nécessaires pour un livre (élément `livre`) sont :
 - son titre général (élément `titre`) ;
 - les noms des auteurs (éléments `auteur`) ;
 - ses tomes (élément `tomes`) et pour chaque tome (éléments `tome`), leur nombre de pages (élément `pages`) ;
 - des informations générales sur son édition (élément `infosEdition`) comme par exemple le nom de l'éditeur (élément `editeur`), le lieu d'édition (élément `lieuEdition`), le lieu d'impression (élément `lieuImpression`), son numéro ISBN (élément `ISBN`) ;
- les informations nécessaires pour un article (élément `article`) sont :
 - son titre (élément `titre`) ;
 - les noms des auteurs (éléments `auteur`) ;
 - ses références de publication (élément `infosPublication`) : nom du journal (élément `nomJournal`), numéro des pages (élément `pages`), année de publication (élément `anneePublication`) et numéro du journal (élément `numeroJournal`)
- on réservera aussi un champ optionnel, pour chaque livre et ahcque article, pour un avis (élément `avis`) personnel.

Tester ce Schema XML avec un fichier XML que l'on écrira *ex-nihilo* et validera.

Exercice 2 : Déclaration d'attributs

Solution : schema2.xsd

Modifier le Schéma précédent... On ne déclarera, pour le moment, que des types de chaînes de caractères.

- ... en ajoutant un attribut optionnel `soustitre` à l'élément `titre` ;
- ... en faisant de l'élément `tome` un élément vide et en lui ajoutant un attribut requis `nbPages` et un attribut optionnel `sousTitre` ;
- ... en faisant de l'élément `nomJournal` un attribut de l'élément `infosPublication` et en lui donnant comme valeur par défaut `Feuille de Chou` ;

Utiliser ce Schéma pour créer un fichier XML valide.

Exercice 3 : Déclarations de types

Solution : schema3.xsd

Nous allons modifier le Schema précédent pour tirer parti des fonctionnalités additionnelles de ce format, relatives aux types de données.

- Utilisation des types intégrés
 1. L'élément `pages` doit être un entier positif, ainsi que l'élément `numeroJournal`.
 2. L'élément `anneePublication` doit être déclaré comme... une année.
- Déclaration de types

Nous allons maintenant définir nos propres types de données, en les dérivant par rapport aux types intégrés.

1. Définir un type simple nommé `typeISBN`, basé sur une *restriction* du type `xsd:string`, se limitant à une chaîne de caractères composée de 10 chiffres. L'utiliser dans la déclaration de l'élément `ISBN`.
2. En s'inspirant de l'exemple donné en cours du formatage d'une adresse électronique, déclarer un type `typePages`, basé sur une *restriction* du type `xsd:string`, se limitant à un nombre, puis la chaîne de caractères " à ", puis un autre nombre.

On vérifiera le fonctionnement à l'aide d'un fichier XML.