

Corrigé type partie 2 : activité « Créez un schéma XML »

Afin que vous puissiez noter correctement les exercices que vous devez corriger, je vous propose de revenir ensemble sur l'ensemble des critères de correction. Cela nous permettra de parcourir le corrigé type par la même occasion.

Lisibilité du Schéma XML

La première chose à noter est la lisibilité du Schéma XML. Par lisibilité on parle de la “forme” du document. Typiquement, si l'on jette un coup d'œil au corrigé type, même si on l'ouvre dans un bloc-note “primaire” sans coloration syntaxique, on arrive quand même à le déchiffrer. En effet, le contenu est espacé, clair et l'indentation permet de se rendre facilement compte de l'architecture du document. Les commentaires, facilement repérables grâce à leur syntaxe spécifique aident d'autant plus à se repérer dans le document.

Quand vous ouvrez le Schéma XML que vous devez corriger avec un éditeur de texte ou un logiciel spécialisé comme EditiX, êtes-vous en face d'un document parfaitement lisible, lisible ou peu lisible ? Est-ce que l'indentation permet de se rendre compte de l'architecture du document ? Le document est-il commenté ?

Si ce critère peut sembler subjectif, il est pourtant très important car le jour où vous travaillerez à plusieurs sur un document, il convient que chacun puisse le prendre en main le plus rapidement possible sans perdre du temps à le déchiffrer. N'oubliez jamais qu'un document bien présenté permet de faciliter sa prise en main.

Le Schéma XML est-il bien formé ?

Pour fonctionner, un Schéma XML doit être bien formé, c'est-à-dire qu'il ne comporte aucune erreur de syntaxe.

Pour vérifier que le Schéma XML que vous devez corriger est bien formé, le plus simple est encore d'utiliser un outil spécifique comme par exemple EditiX. Si vous ne savez pas comment vérifier que le Schéma XML est bien formé à l'aide d'EditiX, vous pouvez consulter [cette partie du cours](#).

Le Document XML est-il valide ?

Le but du Schéma XML est de décrire très précisément ce qu'un document XML peut contenir ou pas. Le meilleur moyen de vérifier que le Schéma XML que vous devez corriger est correct à première vue est de valider le document XML donné dans l'énoncé une fois associé au Schéma XML.

Pour vérifier que le Document XML est bien valide, le plus simple est encore d'utiliser un outil spécifique comme par exemple EditiX. Si vous ne savez pas comment vérifier que le Document XML est valide à l'aide d'EditiX, vous pouvez consulter [cette partie du cours](#).

L'attribut "type"

L'une des difficultés de cet exercice est de correctement définir les attributs du document XML dans le Schéma XML.

Prenons l'attribut "type" porté par la balise <livre />. D'après l'énoncé, il est indiqué qu'« **un livre peut avoir un "type" mais ce n'est pas obligatoire** » et que « **le seul type possible pour un livre est le type "poche"** ».

Dans la définition de l'attribut, il faut donc indiquer qu'il était optionnel à l'aide des mots-clefs "**use="optional"**". Il est également indispensable de préciser que la seule valeur possible pour cet attribut est la valeur "poche". Dans la déclaration de l'attribut, il convient donc de "**fixer**" la valeur à l'aide des mots-clefs "**fixed="poche"**".

Pour obtenir tous les points, ces deux mots-clefs doivent impérativement apparaître. Finalement, voici ce à quoi doit ressembler la définition de l'attribut dans le schéma XML :

```
<xsd:attribute name="type" fixed="poche" use="optional"/>
```

L'attribut "style"

Continuons notre étude des attributs avec l'attribut "style" porté par la balise <livre />. D'après l'énoncé, il est indiqué qu'« **un livre possède obligatoirement un style** » et que « **les styles d'un livre sont prédéfinis via la liste suivante** » :

- Policier
- Théâtre
- Fantastique
- Bande Dessinée
- Nouvelle
- Biographie
- Auto-biographie
- Documentaire

Dans la définition de l'attribut, vous devez donc indiquer qu'il était obligatoire à l'aide des mots-clefs "**use="required"**".

Il convient également de mettre en place une énumération qui définit les 8 valeurs possibles pour l'attribut. À noter que ces 8 valeurs sont des chaînes de caractères. [En se référant au cours sur la façon dont on met en place une énumération dans un Schéma XML](#), voici qui doit figurer :

```
<xsd:simpleType>  
  
  <xsd:restriction base="xsd:string">
```

```
<xsd:enumeration value="Policier"/>  
  
<xsd:enumeration value="Théâtre"/>  
  
<xsd:enumeration value="Fantastique"/>  
  
<xsd:enumeration value="Bande Dessinée"/>  
  
<xsd:enumeration value="Nouvelle"/>  
  
<xsd:enumeration value="Biographie"/>  
  
<xsd:enumeration value="Auto-biographie"/>  
  
<xsd:enumeration value="Documentaire"/>
```

```
</xsd:restriction>
```

```
</xsd:simpleType>
```

Pour obtenir tous les points, voici ce que l'apprenant doit avoir noté :

```
<xsd:attribute name="style" use="required">
```

```
<xsd:simpleType>
```

```
<xsd:restriction base="xsd:string">
```

```
<xsd:enumeration value="Policier"/>
```

```
<xsd:enumeration value="Théâtre"/>
```

```
<xsd:enumeration value="Fantastique"/>
```

```
<xsd:enumeration value="Bande Dessinée"/>
```

```
<xsd:enumeration value="Nouvelle"/>
```

```
<xsd:enumeration value="Biographie"/>
```

```
<xsd:enumeration value="Auto-biographie"/>
```

```
<xsd:enumeration value="Documentaire"/>
```

```
</xsd:restriction>
```

```
</xsd:simpleType>
```

```
</xsd:attribute>
```

L'élément "annee"

La difficulté avec la balise `<annee />`, c'est qu'il convient de correctement la typer pour être le plus précis possible. Techniquement, on peut la décrire comme une simple chaîne de caractères :

```
<xsd:element name="annee" type="xsd:string"/>
```

Le document XML et le Schéma XML sont tous les deux bien formés et le document XML est valide.

Mais, comme je le disais un peu plus haut, le but d'un Schéma XML est d'être le plus précis possible. Aussi, définir l'année comme une simple chaîne de caractère est une erreur, car cela permet d'écrire ce que l'on veut dans le document XML, comme par exemple :

```
<annee>Pouet Pouet</annee>
```

Pour éviter ces dérives, il convient donc de correctement renseigner le type de la balise `<annee />`. D'après l'énoncé, il s'agit d'une année de parution. Si vous vous attardez sur [cette partie du tutoriel](#), vous allez voir qu'il existe des types spéciaux pour les dates. Celui qui représente une année est le type **"gYear"**.

Il convient donc de définir l'élément "annee" de la façon suivante :

```
<xsd:element name="annee" type="xsd:gYear"/>
```

L'attribut "lang"

Continuons notre étude des attributs avec l'attribut "lang" porté par la balise `<titre />`. D'après l'énoncé, il est indiqué qu'**« un titre doit forcément s'accompagner de sa langue (français, anglais, etc.) »**.

Dans la définition de l'attribut, l'apprenant doit donc indiquer qu'il était obligatoire à l'aide des mots-clefs **"use="required"**.

Techniquement, la langue peut être définie comme une simple chaîne de caractères :

```
<xsd:attribute name="lang" type="xsd:string" use="required"/>
```

Le document XML et le Schéma XML auraient été tous les deux bien formés et le document XML aurait été valide.

Mais, une nouvelle fois, le but d'un Schéma XML est d'être le plus précis possible. Aussi, définir la langue comme une simple chaîne de caractère est une erreur car elle permet d'écrire ce que l'on veut dans le document XML, comme par exemple :

```
<titre lang="Pouet">The Templar Legacy</titre>
```

Pour éviter ces dérives, il convient donc de correctement renseigner le type de l'attribut "lang". Si vous vous attardez sur [cette partie du cours](#), vous allez voir qu'il existe un type spécial pour les langues à savoir le type **"language"**.

Il convient donc de définir l'attribut "lang" de la façon suivante :

```
<xsd:attribute name="lang" type="xsd:language" use="required"/>
```