

62-31.3 XML

[XML-05-E01]

Révision

Solution

[Cédric Benoit]

Question 1

Quelles sont les conditions pour qu'un fichier XML soit bien formé ?

- Avoir un prologue
- Un document XML est un arbre doit avoir une balise principale
- Ouverture et fermeture des balises
- Les balises sont sensibles aux cases
- Pas d'espace dans les balises
- Pas de nombre, pas de signe de ponctuation, ... dans les noms de balise
-
- Respecter les règles syntaxiques de XML (W3cC)

Question 2

Que signifie Relax NG ? Qu'est-ce que ce langage permet de faire et qu'apporte-t-il par rapport aux schémas DTD et XSD ?

- Relax NG signifie Regular Language for XML Next Generation.
- Il est considéré comme une alternative au schéma XSD
- Langage de description de document XML simplifié et ne permet pas de gérer les valeurs à l'intérieur des éléments, mais uniquement la structure du document XML.
- Relax NG dispose de sa propre syntaxe qui plus compacte et différentes des DTD et XSD.

Question 3

Elaborer un type de données pour chaque question ci-dessous. Les bornes sont toujours incluses.

a) Un nombre entier entre -2000 et 2000.

```
<xs:simpleType name="Type3a">
  <xs:restriction base="xs:integer">
    <xs:minInclusive value="-2000"/>
    <xs:maxInclusive value="2000"/>
  </xs:restriction>
</xs:simpleType>
```

b) Un numéro de série de la forme :NUM-x-xx-xx (x = chiffre).

```
<xs:simpleType name="Type3b">
  <xs:restriction base="xs:NMTOKEN">
    <xs:pattern value="NUM\[0-9\]\-[0-9]\-[0-9]\-[0-9]" />
  </xs:restriction>
</xs:simpleType>
```

- c) Un mot dont la première lettre commence par une majuscule et qui a une longueur maximale de 32 caractères.

```
<xs:simpleType name="Type3e">  
  <xs:restriction base="xs:string">  
    <xs:maxLength value="32"/>  
    <xs:pattern value="[A-Z][A-Za-z]*/>  
  </xs:restriction>  
</xs:simpleType>
```

Question 4

Voici le fichier XML suivant :

```
<personnes>
  <personne genre="Monsieur">
    <nom>Dumont</nom>
    <prenom>Roger</prenom>
    <sexe>M</sexe>
    <datenaissance>17.05.1974</datenaissance>
    <remarques>Diabétique</remarques>
    <remarques>Végétarien</remarques>
    <adresse>
      <rue>Rue de la Gare</rue>
      <numero>36</numero>
      <npa>2000</npa>
      <ville>Neuchâtel</ville>
    </adresse>
  </personne>
  <personne genre="Madame">
    <nom>Perret</nom>
    <prenom>Julie</prenom>
    <sexe>F</sexe>
    <age>25</age>
    <adresse>
      <rue>Rue de la Tourne</rue>
      <numero>42</numero>
      <npa>2114</npa>
      <ville>Fleurier</ville>
    </adresse>
    <adresse>
      <rue>Grand Bourgeau</rue>
      <numero>25</numero>
      <npa>25300</npa>
      <ville>Les Verrières de Joux</ville>
      <pays>France</pays>
    </adresse>
  </personne>
</personnes>
```

A partir de ce fichier XML, veuillez élaborer un schéma XSD permettant de valider le fichier XML ci-dessus, en prenant en compte les remarques suivantes :

1. L'élément « remarques » peut apparaître 0 ou plusieurs fois.
2. L'élément « adresse » peut apparaître 1 ou plusieurs fois.
3. L'élément « pays » peut apparaître 0 ou 1 fois.

D'autre part, veuillez indiquer les modifications nécessaires à apporter au fichier XML, afin de pouvoir le valider avec le schéma XSD.

Schéma XSD :

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="nom" type="xs:string"/>
  <xs:element name="prenom" type="xs:string"/>
  <xs:element name="sexe" type="xs:string"/>
  <xs:element name="datenaissance" type="xs:string"/>
  <xs:element name="age" type="xs:string"/>
  <xs:element name="rue" type="xs:string"/>
  <xs:element name="numero" type="xs:string"/>
  <xs:element name="npa" type="xs:string"/>
  <xs:element name="ville" type="xs:string"/>
  <xs:element name="pays" type="xs:string"/>
  <xs:element name="remarques" type="xs:string"/>
  <xs:element name="adresse">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="rue"/>
        <xs:element ref="numero"/>
        <xs:element ref="npa"/>
        <xs:element ref="ville"/>
        <xs:element ref="pays" minOccurs="0" maxOccurs="1"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="personnes">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="personne" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="personne">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="nom"/>
        <xs:element ref="prenom"/>
        <xs:element ref="sexe"/>
        <xs:choice>
          <xs:element ref="age"/>
          <xs:element ref="datenaissance"/>
        </xs:choice>
        <xs:element ref="remarques" minOccurs="0"
maxOccurs="unbounded"/>
        <xs:element ref="adresse" minOccurs="1"
maxOccurs="unbounded"/>
      </xs:sequence>
      <xs:attribute name="genre" type="xs:string" use="required"/>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Dans le fichier XML lié au schéma XSD, il faudra modifier la balise <personnes> comme suit :

```
<personnes xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="nom_fichier.xsd">
```

Question 5

Voici la structure XML suivante :

```
<Listesocietes>
  <Societe>
    <Nom>Bernard SA</Nom>
    <Type>SA</Type>
    <Lieu>Lausanne</Lieu>
  </Societe>
  <Societe>
    <Nom>Sono Sàrl</Nom>
    <Type>Sàrl</Type>
    <Lieu>Sion</Lieu>
  </Societe>
  <Societe>
    <Nom>Arribo SA</Nom>
    <Type>SA</Type>
    <Lieu>Neuchâtel</Lieu>
  </Societe>
  <Societe>
    <Nom>MonoSport</Nom>
    <Type>individuel</Type>
    <Lieu>Neuchâtel</Lieu>
  </Societe>
</Listesocietes>
```

A partir de cette structure, réaliser un fichier XSLT permettant d'afficher sous format HTML la liste des sociétés triée par nom et affiche en gras les noms de sociétés qui sont établies à Neuchâtel, comme le montre l'exemple ci-dessous :

Liste des sociétés :

- Arribo SA
- Bernard SA
- MonoSport
- Sono Sàrl

Fichier XSLT :

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
  <xsl:output method="html"/>
  <xsl:template match="/Listesocietes">
    <html>
      <head>
        <title>Exercice révision XML</title>
      </head>
      <body>
        <xsl:text>Liste des sociétés: </xsl:text>
        <xsl:for-each select="Societe">
          <xsl:sort select="Nom"/>
          <p>
            <xsl:if test="Lieu='Neuchâtel'">
              <b>
                <xsl:value-of select="Nom"/>
              </b>
            </xsl:if>
            <xsl:if test="Lieu!='Neuchâtel'">
              <xsl:value-of select="Nom"/>
            </xsl:if>
          </p>
        </xsl:for-each>
      </body>
    </html>
  </xsl:template>
</xsl:stylesheet>
```