

## Série « XML Basic Notions » : Corrigé

### Exercice 1

Soient les deux instances de documents XML suivantes :

<pre>&lt;element name=" exm:examen"&gt;   &lt;complexType&gt;     &lt;sequence&gt;       &lt;element ref="exm:exercice"/&gt;     &lt;/sequence&gt;   &lt;/complexType&gt; &lt;/element&gt;</pre> <p style="text-align: center;"><b>Examen1.xml</b></p>	<pre>&lt;xsd:element name="examen"&gt;   &lt;xsd:complexType&gt;     &lt;xsd:sequence&gt;       &lt;xsd:element ref="exercice" /&gt;     &lt;/xsd:sequence&gt;   &lt;/xsd:complexType&gt; &lt;/xsd:element&gt;</pre> <p style="text-align: center;"><b>Examen2.xml</b></p>
--	--

Compléter les en-têtes des schémas **Examen 1.xsd** et **Examen 2.xsd** relativement à l'utilisation des espaces de noms.

#### Examen1.xsd

```
<?xml version="1.0"?>
<schema
  xmlns="http://www.w3.org/2001/XMLSchema"
  targetNamespace=" http://www.exam.org "
  xmlns : exm ="http://www.exam.org"
  elementFormDefault="qualified">
```

#### Examen2.xsd

```
<?xml version="1.0"?>
<xsd:schema
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  targetNamespace=" http://www.exam.org "
  xmlns="http://www.exam.org"
  elementFormDefault=" qualified ">
```

## Exercice 2

Soit la DTD suivante qui indique que l'ordre d'apparition des éléments *identification*, *titre* et *salaire* n'est pas important, donner une solution à l'aide d'un schéma XML qui évite l'énumération des combinaisons autorisées.

```
<!ELEMENT RegistreEmploye (Employe)+>
<!ELEMENT Employe (nom, position, salaire) | (nom, salaire,
position) | (position, salaire, nom) | (position, nom,
salaire) | (salaire, nom, position) | (salaire, position, nom) >
<!ELEMENT nom (#PCDATA)>
<!ELEMENT position (#PCDATA)>
<!ELEMENT salaire (#PCDATA)>
```

Réponse : Utilisation du constructeur « all »

## Exercice 3

Soit le document *company.xml* suivant

```
<Company>
  <Address xsi:type="US-Address">
    <Name>Main office</Name>
    <Street>Grosvenor Blvd.</Street>
    <City>Los Angeles</City>
    <State>California</State>
    <Zip>3141</Zip>
  </Address>
  <Division>
    <Division-Name>Sales</Division-Name>
    <Location>Washington</Location>
    <Person Manager="true" Degree="MA">
      <First>Allison</First>
      <Last>Andersen</Last>
      <PhoneExt>111</PhoneExt>
      <EMail>Andersen@work.com</EMail>
    </Person>
    <Person Manager="false" Degree="BA">
      <First>Bruce</First>
      <Last>Burrwinkle</Last>
      <PhoneExt>222</PhoneExt>
      <EMail>Burrwinkle@work.com</EMail>
    </Person>
  </Division>
  <Division>
    .....
  </Division>
</Company>
```

Dans le cas où c'est possible, indiquez comment les contraintes suivantes peuvent être représentées via une DTD et/ou un schéma XML

- La valeur de *degree* est soit MA soit BA

Uniquement avec un schéma en utilisant la facet énumération

- Une adresse e-mail doit être de la forme « string.string@string »

*Uniquement avec un schéma en utilisant la facet pattern pour les strings*

- US-Adress est une extension de Address à laquelle on ajoute les éléments state et zip

*Uniquement avec un schéma en utilisant l'extension des types complex*

- L'élément zip a une valeur comprise entre 1000 et 4000

*Uniquement avec un schéma en utilisant la facet mininclusive et maxinclusive*

- Les personnes, dans les instances xml, doivent être ordonnées par ordre alphabétique

*Pas possible ni avec DTD ni schéma*

## Exercice 4

On considère le document XML suivant:

```
<university>
  <student matrNo="325422">
    <name>
      <firstName>Mike</firstName>
      <lastName>Down</lastName>
    </name>
    <attendsLecture lectureName="Introduction to Databases"/>
    <attendsLecture lectureName="Introduction to Artificial
Intelligence"/>
  </student>
  <researchAssistant employeeNo="134234">
    <name>
      <FN>Dominik</FN>
      <LN>Luebbers</LN>
    </name>
    <organizingLecture lectureName="Introduction to Databases"/>
    <hiwis>
      <hiwi matrNo="325422"/>
    </hiwis>
    <email>luebbers@i5.informatik.rwth-aachen.de</email>
    <telephone>8021512</telephone>
  </researchAssistant>
</university>
```

On suppose que:

- matrNr and employeeNo sont uniques.  
(utiliser le constructeur Unique de XML schéma)
- matrNr and employeeNo sont des entiers de longueurs 6 ne dépassant pas 999999.  
(utiliser la facet mininclusive et maxinclusive d'entier [100000, 999999])
- "Research assistants" peuvent organiser au maximum une "lecture".  
(utiliser la cardinalité maxoccurrence = 1)
- "Research assistants" peuvent avoir une ou plusieurs "email" adresses.  
(utiliser la cardinalité minoccurrence= 1 et maxoccurrence = unbounded)

- "firstName" et "lastName" sont des chaînes de caractères de longueur ne dépassant pas les trois caractères.

(utiliser la facet length pour les strings)

- "firstName" et "lastName" ainsi que "FN" et "LN" sont utilisés de manière équivalente.

(Utiliser le mécanisme de substitution Group)

Ecrire un schéma XML décrivant la structure du document XML précédent et respectant les hypothèses définies.