

Examen XML -un corrigé possible

Exercice 1 :

- I. N'est pas bien formé car les noms des balises sont sensibles à la casse (le parser considère que la balise <chapitre> n'a pas été fermée)
- II. N'est pas bien formé, pour deux raisons :

D'abord le non-respect de la casse (balises <page> et <Page>)

Ensuite parce que les noms des balises xml ne sont pas valides (ne doivent pas contenir d'espaces)

Exercice 2 :

- I. N'est pas valide car n'est pas bien formé, en effet :

La casse des attributs des éléments b et e n'est pas respectée (x et X, y et Y)

La référence à l'élément indicatif n'est pas correcte (il manque un ; après &indicatif)

Les éléments b doivent se suivre dans leur ordre d'apparition, ce qui n'est pas le cas (un élément c est imbriqué entre deux apparitions de l'élément b)
- II. Non valide car n'est pas bien formé, en effet :

Les noms des éléments ne sont pas des identificateurs valides.

Il n'y a pas d'entité ensias.ma déclarée dans la DTD interne.

Exercice 3 :

1.

```
<?xml version="1.0" encoding="UTF-8"?>
<etudiants>
  <etudiant code="1">
    <nom>etd1</nom>
    <prenom>prenom1</prenom>
    <age>22</age>
    <niveau>2A</niveau>
    <groupe>G11</groupe>
    <telephone>0021233333</telephone>
    <mail>etd1@ensias.ma</mail>
```

```

<modules>
  <module nomModule="M4.1">16</module>
  <module nomModule="M4.2">17</module>
  <module nomModule="M4.3">18</module>
  <module nomModule="M4.4">19</module>
</modules>
</etudiant>
<etudiant code="2">
  <nom>etd2</nom>
  <prenom>prenom2</prenom>
  <age>22</age>
  <niveau>2A</niveau>
  <groupe>G11</groupe>
  <telephone>0021233334</telephone>
  <mail>etd2@ensias.ma</mail>
<modules>
  <module nomModule="M4.1">15</module>
  <module nomModule="M4.2">16</module>
  <module nomModule="M4.3">18</module>
  <module nomModule="M4.4">17</module>
</modules>
</etudiant>
</etudiants>

```

2. Etd.dtd

```

<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT ENTER_NAME_OF_ROOT_ELEMENT_HERE EMPTY>
<!ELEMENT etudiants (etudiants+)>
<!ELEMENT etudiant (nom, prenom, age,niveau, groupe, telephone, mail, modules)>
<!ELEMENT nom (#PCDATA)>
<!ELEMENT prenom (#PCDATA)>
<!ELEMENT age (#PCDATA)>
<!ELEMENT niveau (#PCDATA)>
<!ELEMENT groupe (#PCDATA)>
<!ELEMENT telephone (#PCDATA)>
<!ELEMENT mail (#PCDATA)>
<!ELEMENT modules (module+)>
<!ELEMENT module (#PCDATA)>
<!ATTLIST etudiant code ID #REQUIRED>
<!ATTLIST module nomModule CDATA #REQUIRED>

```

3. Etd.xsd

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:element name="etudiants">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="nom" type="xs:string"/>
        <xs:element name="prenom" type="xs:string"/>
        <xs:element name="age">
          <xs:simpleType>
            <xs:restriction base="xs:integer">
              <xs:minInclusive value="15"/>
              <xs:maxInclusive value="60"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:element>

```

```

</xs:simpleType>
</xs:element>

<xs:element name="niveau">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="1ere"/>
      <xs:enumeration value="2eme"/>
      <xs:enumeration value="3eme"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="groupe">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="G1"/>
      <xs:enumeration value="G2"/>
      <xs:enumeration value="G3"/>
      <xs:enumeration value="G4"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="telephone">
  <xs:simpleType>
    <xs:pattern value="+[0-9]{7}"/>
  </xs:simpleType>
</xs:element>
<xs:element name="mail">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:pattern value="[a-z]+@[a-z]+\.[a-z]{2,3}"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="modules">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="module" minOccurs="4" maxOccurs="4"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:schema>

```

4. 1. Sélectionner tous les nœuds title ayant pour parent un nœud Y ayant un parent X et ayant un fils Z de valeur supérieure à 35
2. tous les éléments D appartenant à l'espace de noms my et vérifiant ...(je ne me rappelle plus...)
5. je ne me rappelle plus encore une fois..
6. principales différences entre html et xml :

Xml est extensible (balisage dynamique et évolutif) par contre le balisage html est figé

Xml permet de séparer la présentation du contenu, chose que html ne permet pas

Xml permet de définir des modèles de documents (une notion proche de l'orienté objet pour les documents)

Xml permet d'exploiter le contenu des documents et d'effectuer une interrogation de ces document pour en extraire l'information voulue

.....etc