

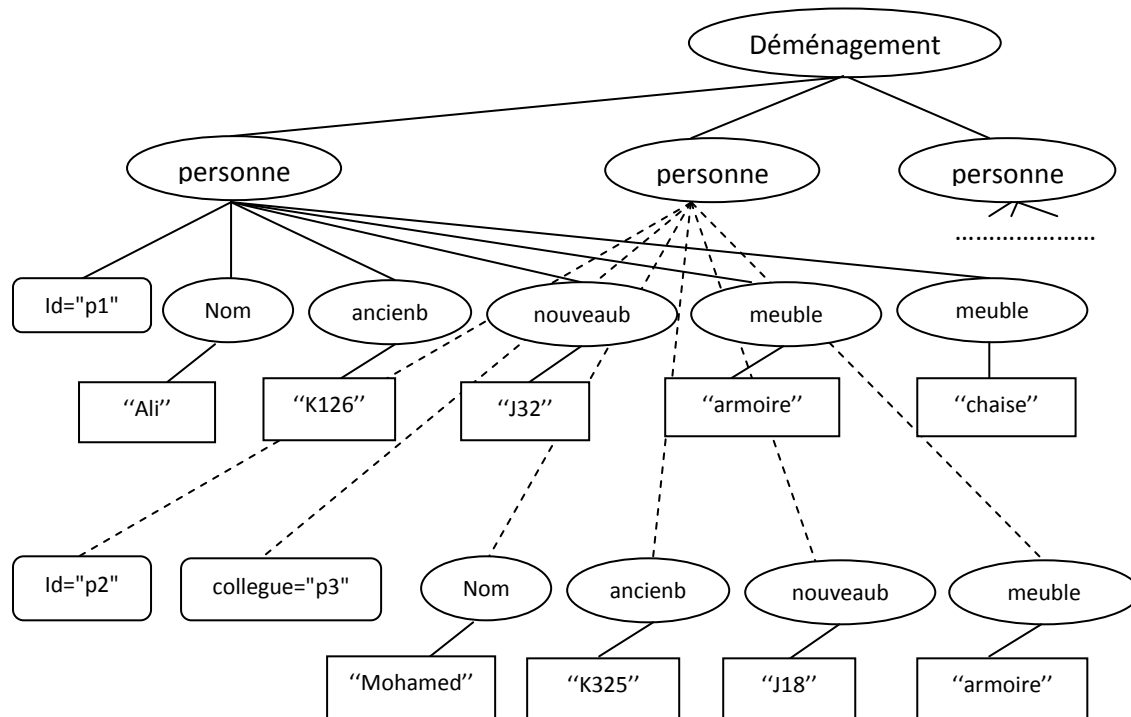
**TD3 : XML\_DTD\_XMLSchemas****Exercice 1 :**

L'entreprise « **EntX** » déménage. Chaque employé de cette entreprise a un numéro d'ancien bureau et un numéro de nouveau bureau. Par exemple, l'employé *Ali* a pour ancien bureau le numéro *K126* et pour nouveau bureau le numéro *J32*. Si deux employés sont dans le même nouveau bureau alors on dit qu'ils sont "*collègues de bureau*". Les informations du déménagement sont stockées dans un document XML. Chaque employé a un identifiant et un nom, ainsi qu'éventuellement un numéro d'ancien bureau, un numéro de nouveau bureau, ainsi que des *meubles*. Si un employé partage le bureau avec un collègue alors, pour éviter de stocker inutilement de l'information, il n'y a qu'un seul des deux employés qui possède le numéro du nouveau bureau. Par exemple, dans le document XML **EntX.xml** ci-après, l'employé *Mohamed* et l'employé *Salah* sont "*collègues de bureau*" (ils possèdent tous les deux un attribut *colleague* faisant référence à la personne avec laquelle ils partageront le nouveau bureau), mais l'employé *Salah* ne contient pas le numéro du nouveau bureau. Voici le fichier **EntX.xml** qui contient les informations du déménagement de l'entreprise **EntX**.

```
<?xml version="1.0"?>
<!DOCTYPE demenagement SYSTEM "demenagement.dtd">
<demenagement>
  <personne id='p1'>
    <nom>Ali</nom>
    <ancienb>K126</ancienb>
    <nouveaub>J32</nouveaub>
    <meuble>armoire</meuble>
    <meuble>chaise</meuble>
  </personne>
  <personne id='p2' colleague='p3'>
    <nom>Mohamed</nom>
    <ancienb>K325</ancienb>
    <nouveaub>J18</nouveaub>
    <meuble>armoire</meuble>
  </personne>
  <personne id='p3' colleague='p2'>
    <nom>Salah</nom>
    <ancienb>K229</ancienb>
    <meuble>lampe</meuble>
  </personne>
</demenagement>
```

**Questions :**

1. Donner l'arbre XML correspondant



2. Rédiger la DTD demenagement.dtd pour qu'elle respecte les contraintes données dans l'énoncé et qu'elle valide le document "EntX.xml".

```
<!ELEMENT demenagement (personne)* >
<!ELEMENT personne (nom,ancienb?,nouveau?,meuble*) >
<!--ATTLIST personne id ID #REQUIRED colleague IDREF #IMPLIED-->
<!ELEMENT nom (#PCDATA) >
<!ELEMENT ancienb (#PCDATA) >
<!ELEMENT nouveau (#PCDATA) >
<!ELEMENT meuble (#PCDATA) >
```

## Exercice 2 :

On souhaite écrire un livre en utilisant le formalisme XML. Le livre est structuré en sections (au moins 2), qui contient chacune au moins 2 chapitres et chaque chapitre contient au moins 2 paragraphes ; le livre doit contenir la liste des auteurs (avec nom et prénom).

Tous les éléments doivent posséder un titre sauf le paragraphe qui contient du texte.

### Questions:

1. Proposer une structuration XML du document « livre.xml », avec 2 auteurs, 2 sections, 2 chapitres par section et 2 paragraphes par chapitre.

```

<?xml version="1.0" encoding="iso-8859-1"?>
<livre titre="Mon livre">
  <auteurs>
    <auteur nom="Brillant" prenom="Alexandre"/>
    <auteur nom="Briand" prenom="Aristide"/>
  </auteurs>
  <sections>
    <section titre="Section 1">
      <chapitre titre="Chapitre 1">
        <paragraphe>Premier paragraphe</paragraphe>
        <paragraphe>Deuxième paragraphe</paragraphe>
      </chapitre>
      <chapitre titre="Chapitre 2">
        <paragraphe>Premier paragraphe</paragraphe>
        <paragraphe>Deuxième paragraphe</paragraphe>
      </chapitre>
    </section>
    <section titre="Section 2">
      <chapitre titre="Chapitre 1">
        <paragraphe>Premier paragraphe</paragraphe>
        <paragraphe>Deuxième paragraphe</paragraphe>
      </chapitre>
      <chapitre titre="Chapitre 2">
        <paragraphe>Premier paragraphe</paragraphe>
        <paragraphe>Deuxième paragraphe</paragraphe>
      </chapitre>
    </section>
  </sections>
</livre>

```

## 2. Créer la DTD « livre.dtd » correspondante

```

<!ELEMENT livre (auteurs,sections) >
<!ATTLIST livre titre CDATA #REQUIRED>
<!ELEMENT auteurs (auteur)+ >
<!ELEMENT auteurs EMPTY >
<!ATTLIST auteur nom CDATA #REQUIRED
              prenom CDATA #REQUIRED>
<!ELEMENT sections (section, section+)>
<!ELEMENT section (chapitre, chapitre+) >
<!ATTLIST section titre CDATA #REQUIRED>
<!ELEMENT chapitre (paragraphe, paragraphe+) >
<!ATTLIST chapitre titre CDATA #REQUIRED>
<!ELEMENT paragraphe (#PCDATA) >

```

## 3. Modifier la DTD tel que la définition de l'attribut *titre* soit unique à l'aide d'une entité paramètre.

```

<!ENTITY %Titre "titre CDATA #REQUIRED" >
<!ELEMENT livre (auteurs,sections) >
<!ATTLIST livre %Titre; >
<!ELEMENT auteurs (auteur)+ >
<!ELEMENT auteurs EMPTY >
<!ATTLIST auteur nom CDATA #REQUIRED
                prenom CDATA #REQUIRED>
<!ELEMENT sections (section, section+)>
<!ELEMENT section (chapitre, chapitre+) >
<!ATTLIST section %Titre; >
<!ELEMENT chapitre (paragraphe, paragraphe+) >
<!ATTLIST chapitre %Titre; >
<!ELEMENT paragraphe (#PCDATA) >

```

### Exercice 3 :

Soit un document XML contenant un nombre indéterminé d'éléments (description de commandes) sous la forme :

```

<commande ref="111" date="2002-01-15">
<facturation provisoire="true">
<societe soc="soc1">CREA</societe>
<nom>Service technique</nom>
<rue num="10" place="rue">de la Mairie</rue>
<ville>Rennes</ville>
<code.postal>35000</code.postal>
<tel direct="true">0299123456</tel>
<tel direct="false">0299789101</tel>
<email href="finance@crea.com"
  direct="true"/>
</facturation>
<livraison>
<societe refsoc="soc1">CREA</societe>
<nom>Service technique</nom>
<rue num="10" place="rue">de la Mairie</rue>
<ville>Rennes</ville>
<code.postal>35000</code.postal>
<tel direct="true">0299789101</tel>
<email href="info@crea.com" direct="false"/>
<email href="dupont@crea.com"
  direct="true"/>
</livraison>

```

1

```

<!-- section items -->
<item>
<quantite>5</quantite>
<code.produit>ABS-21</code.produit>
<description>ramettes bristol</description>
<prix.unitaire>6</prix.unitaire>
</item>
<item>
<quantite>1</quantite>
<code.produit>ABC-11</code.produit>
<description>reliures spirales</description>
<prix.unitaire>0.3</prix.unitaire>
</item>
</commande>

```

2

- une ou plusieurs commandes portant un identifiant unique (*ref*), une seule adresse de facturation et une seule adresse de livraison par commande, éventuellement un ou plusieurs *items* par commande, *email* facultatif, plusieurs email et plusieurs téléphones

possibles, chaque société portant un identifiant unique (*soc*) ou une référence vers la société portant cet identifiant (*refsoc*)

### Questions :

1. A partir de ce document XML créer la DTD "commandes.dtd" :
  - déclarer une entité pour décrire les adresses (de facturation ou de livraison)
  - déclarer une entité pour définir les valeurs booléennes de certains attributs

```
<!ELEMENT commandes (commande)+ >
<!ELEMENT commande (facturation, livraison, item+) >
<!ATTLIST commande ref ID #REQUIRED
                    date CDATA #REQUIRED >
<!ENTITY %adr "(societe, nom, rue, ville, code.postal, email*,
                    tel+)" >
<!ENTITY %Boolean "(true|false) true" >
<!ELEMENT facturation %adr; >
<!ATTLIST facturation provisoire %Boolean; #REQUIRED >
<!ELEMENT livraison %adr; >
<!ELEMENT societe #PCDATA >
<!ATTLIST societe soc ID #IMPLIED
                    refsoc IDREF #IMPLIED>
<!ELEMENT nom #PCDATA >
<!ELEMENT rue #PCDATA >
<!ATTLIST rue num CDATA #REQUIRED
                    place CDATA #REQUIRED>
<!ELEMENT ville #PCDATA >
<!ELEMENT code.postal #PCDATA >
<!ELEMENT email EMPTY >
<!ATTLIST email href CDATA #REQUIRED
                    direct %Boolean; #REQUIRED>
<!ELEMENT tel #PCDATA >
<!ATTLIST tel direct %Boolean; #REQUIRED>
<!ELEMENT item(quantite,code.produit,description,
                    prix.unitaire)>
<!ELEMENT quantite #PCDATA >
<!ELEMENT code.produit #PCDATA >
<!ELEMENT description #PCDATA >
<!ELEMENT prix.unitaire #PCDATA >
```

## 2. Donner le schéma XML équivalent.

```

<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">

  <xsd:element name="commandes" type="commandesType" />

  <xsd:complexType name="commandesType">
    <xsd:sequence>
      <xsd:element name="commande" type="commType" minOccurs="1"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="commType">
    <xsd:sequence>
      <xsd:element name="facturation" type="factType"/>
      <xsd:element name="livraison" type="livType"/>
      <xsd:element name="item" type="itemType" minOccurs="1"/>
    </xsd:sequence>
    <xsd:attribute name="ref" type="xsd:ID" use="required"/>
    <xsd:attribute name="date" type="xsd:Date" use="required"/>
  </xsd:complexType>

  <xsd:group name="socFGpe">
    <xsd:element name="societe">
      <xsd:complexType>
        <xsd:SimpleContent>
          <xsd:extension base="xsd:String">
            <xsd:attribute name="soc" type="xsd:ID" use="required"/>
          </xsd:extension>
        </xsd:SimpleContent>
      </xsd:complexType>
    </xsd:element>
  </xsd:group>

  <xsd:group name="socLGpe">
    <xsd:element name="societe">
      <xsd:complexType>
        <xsd:SimpleContent>
          <xsd:extension base="xsd:String">
            <xsd:attribute name="refsoc" type="xsd:IDRef" use="required"/>
          </xsd:extension>
        </xsd:SimpleContent>
      </xsd:complexType>
    </xsd:element>
  </xsd:group>

  <xsd:complexType name="factType">
    <xsd:sequence>
      <xsd:group ref="socFGpe"/>
      <xsd:group ref="adrGpe"/>
    </xsd:sequence>
    <xsd:attribute name="provisoire" type="xsd:Boolean" use="required"/>
  </xsd:complexType>

```

```

<xsd:group name="adrGpe">
  <xsd:sequence>
    <xsd:element name="nom" type="xsd:String" />
    <xsd:element name="rue" >
      <xsd:complexType >
        <xsd:SimpleContent >
          <xsd:extension base="xsd:String" >
            <xsd:attribute name="num" type="xsd:Integer" use="required"/>
            <xsd:attribute name="place" type="xsd:String" use="required"/>
          </xsd:extension >
        </xsd:SimpleContent >
      </xsd:complexType >
    </xsd:element >
    <xsd:element name="ville" type="xsd:String" />
    <xsd:element name="code.postal" type="xsd:Integer" />
    <xsd:element name="tel" minOccurs="1">
      <xsd:complexType >
        <xsd:SimpleContent >
          <xsd:extension base="xsd:Integer" >
            <xsd:attribute name="direct" type="xsd:Boolean" use="required"/>
          </xsd:extension >
        </xsd:SimpleContent >
      </xsd:complexType >
    </xsd:element >
    <xsd:element name="email" minOccurs="0">
      <xsd:complexType >
        <xsd:attribute name="href" type="xsd:String" use="required"/>
        <xsd:attribute name="direct" type="xsd:Boolean" use="required"/>
      </xsd:complexType >
    </xsd:element >
  </xsd:sequence>
</xsd:group >
<xsd:complexType name = "livType">
  <xsd:sequence>
    <xsd:group ref="socLGpe"/>
    <xsd:group ref="adrGpe"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name = " itemType ">
  <xsd:sequence>
    <xsd:element name="quantite" type="xsd:Integer"/>
    <xsd:element name="code.produit" type="xsd:String" />
    <xsd:element name="description" type="xsd:String"/>
    <xsd:element name="prix.unitaire" type="xsd:Float"/>
  </xsd:sequence>
</xsd:complexType>

</xsd:schema>

```

**Exercice 4** : Soit le document XML suivant:

<pre> &lt;?xml version="1.0" encoding="ISO-8859-1" standalone="yes"?&gt; &lt;Enseignement&gt; &lt;Professeur idProf="p1"/&gt; &lt;Professeur idProf="p2"/&gt; &lt;Professeur idProf="p3"/&gt; &lt;Module idMod="m1" profRespModule="p1" nbEleves="122"&gt; &lt;Intitule&gt;Base de données&lt;/Intitule&gt; &lt;Cours idCours="c1" profRespCours="p2"&gt; &lt;Intitule&gt;BD Web&lt;/Intitule&gt; &lt;Seance type="cours magistral" prof="p2"&gt; &lt;Salle&gt;A12&lt;/Salle&gt; &lt;Horaire&gt;Lundi 15h30&lt;/Horaire&gt; &lt;/Seance&gt; &lt;Seance type="td" groupe="1" prof="p1"&gt; &lt;Salle&gt;C831&lt;/Salle&gt; &lt;Horaire&gt;Jeudi 8h30&lt;/Horaire&gt; &lt;/Seance&gt; &lt;Seance type="td" groupe="2" prof="p1"&gt; &lt;Salle&gt;C833&lt;/Salle&gt; &lt;Horaire&gt;Vendredi 10h30&lt;/Horaire&gt; &lt;/Seance&gt; &lt;Remarque&gt;Pas de cours la semaine du 12/03&lt;/Remarque&gt; </pre>	<pre> &lt;/Cours&gt; &lt;Cours idCours="c2" profRespCours="p1"&gt; &lt;Intitule&gt;BD Reparties&lt;/Intitule&gt; &lt;Seance type="cours magistral" prof="p2"&gt; &lt;Salle&gt;A21&lt;/Salle&gt; &lt;Horaire&gt;Lundi 8h30&lt;/Horaire&gt; &lt;/Seance&gt; &lt;/Cours&gt; &lt;/Module&gt; &lt;Module idMod="m2" profRespModule="p3" nbEleves="81"&gt; &lt;Intitule&gt;Programmation orientée objet&lt;/Intitule&gt; &lt;Cours idCours="c3" profRespCours="p1"&gt; &lt;Intitule&gt;La programmation en Java&lt;/Intitule&gt; &lt;Seance type="cours magistral" prof="p1"&gt; &lt;Salle&gt;A15&lt;/Salle&gt; &lt;Horaire&gt;Mardi 13h30&lt;/Horaire&gt; &lt;/Seance&gt; &lt;Seance type="td" groupe="1" prof="p3"&gt; &lt;Salle&gt;C829&lt;/Salle&gt; &lt;Horaire&gt;Mercredi 10h30&lt;/Horaire&gt; &lt;/Seance&gt; &lt;/Cours&gt; &lt;/Module&gt; &lt;/Enseignement&gt; </pre>
1	2

**Question** : Donner le schéma XML correspondant**Correction**

```

<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">

<xsd:element name="Enseignement" type="ensType" />

<xsd:complexType name="ensType">
<xsd:sequence>
<xsd:element name="Professeur" type="ProfType" minOccurs="1"/>
<xsd:element name="Module" type="ModuleType" minOccurs="1"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="ProfType">
<xsd:attribute name="idProf" type="xsd:ID" use="required"/>
</xsd:complexType>
<!-- groupe IntGpe-->
<xsd:group name="IntGpe">
<xsd:element name="intitule" type="xsd:string"/>
</xsd:group>
<!--ModuleType-->
<xsd:complexType name="ModuleType">
<xsd:sequence>
<xsd:group ref="IntGpe">
<xsd:element name="cours" type="CoursType" minOccurs="1"/>
</xsd:sequence>

```



```

    <xsd:attribute name="idMod" type="xsd:ID" use="required"/>
    <xsd:attribute name="profRespModule" type="xsd:IDRef" use="required"/>
    <xsd:attribute name="nbEleves" type="xsd:integer" use="required"/>
  </xsd:complexType>

  <!--coursType-->
  <xsd:complexType name="coursType">
    <xsd:sequence>
      <xsd:group ref="IntGpe">
        <xsd:element name="seance" type="seanceType" minOccurs="1" />
        <xsd:element name="Remarque" type="xsd:String" minOccurs="0"
          maxOccurs="1" />
      </xsd:sequence>
      <xsd:attribute name="idCours" type="xsd:ID" use="required"/>
      <xsd:attribute name="profRespCours" type="xsd:IDRef" use="required"/>
    </xsd:complexType>

    <!-- seanceType-->
    <xsd:complexType name="seanceType">
      <xsd:sequence>
        <xsd:element name="Salle" type="xsd:String"/>
        <xsd:element name="Horaire" type="xsd:String"/>
      </xsd:sequence>
      <xsd:attribute name="type" type="xsd:String" use="required"/>
      <xsd:attribute name="prof" type="xsd:IDRef" use="required"/>
      <xsd:attribute name="groupe" type="xsd:String" use="optional"/>
    </xsd:complexType>

  </xsd:schema>

```

### Exercice 5 :

Etant donné un document planning.xml validé par la DTD suivante :

```

<!ELEMENT planning ((etudiant|enseignant)+, cours+, inscription+) >
<!ELEMENT etudiant (nom, coord) >
<!ATTLIST etudiant no ID >
<!ELEMENT inscription (date) >
<!ATTLIST inscription etudiant IDREF
                  cours IDREF >
<!ELEMENT enseignant (nom, email) >
<!ATTLIST enseignant no ID >
<!ELEMENT cours (intitule,salle) >
<!ATTLIST cours no ID
                  responsable IDREF>
<!ELEMENT coord (adresse, email) >

```

<!ELEMENT nom #PCDATA >  
 <!ELEMENT salle #PCDATA >  
 <!ELEMENT adresse #PCDATA >  
 <!ELEMENT email #PCDATA >  
 <!ELEMENT intitule #PCDATA >  
 <!ELEMENT date #PCDATA >

### Questions :

1. Créer un document " planning.xml" qui soit valide par rapport à cette DTD
2. Donner le schéma XML équivalent.

```

<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">

  <xsd:element name="planning" type="planType" />
  <xsd:complexType name = "planType">
    <xsd:sequence>
      <xsd:group minOccurs="1" maxOccurs="unbounded">
        <xsd:choice>
          <xsd:element name="etudiant" type="etdType"/>
          <xsd:element name="enseignant" type="ensType"/>
        </xsd:choice>
      </xsd:group>
      <xsd:element name="cours" type="coursType" minOccurs="1"/>
      <xsd:element name="inscription" type="inscriType" minOccurs="1"/>
    </xsd:sequence>
  </xsd:complexType>
  <xsd:group name="nomGpe">
    <xsd:element name="nom" type="xsd:String"/>
  </xsd:group>
  <xsd:attributeGroup name="noAtGpe">
    <xsd:attribute name="no" type="xsd:ID" />
  </xsd:attributeGroup >
  <xsd:complexType name = "etdType">
    <xsd:sequence>
      <xsd:group ref="nomGpe"/>
      <xsd:element name="coord" type="coordType"/>
    </xsd:sequence>
    <xsd:attributeGroup ref="noAtGpe"/>
  </xsd:complexType>
  <xsd:group name="mailGpe">
    <xsd:element name="email" type="xsd:String"/>
  </xsd:group>
  <xsd:complexType name = "coordType">
    <xsd:sequence>
      <xsd:element name="adresse" type="xsd:String"/>
      <xsd:group ref="mailGpe"/>
    </xsd:sequence>
  </xsd:complexType>

```

```
</xsd:complexType>

<xsd:complexType name = "ensType">
  <xsd:sequence>
    <xsd:group ref="nomGpe"/>
    <xsd:group ref="mailGpe"/>
  </xsd:sequence>
  <xsd:attributeGroup ref=" noAtGpe"/>
</xsd:complexType>
<xsd:complexType name = "coursType">
  <xsd:sequence>
    <xsd:element name="intitule" type="xsd:String"/>
    <xsd:element name="salle" type="xsd:String"/>
  </xsd:sequence>
  <xsd:attributeGroup ref=" noAtGpe"/>
  <xsd:attribute name="responsable" type="xsd:IDRef" />
</xsd:complexType>
<xsd:complexType name = "inscriType">
  <xsd:element name="date" type="xsd:Date"/>
  <xsd:attribute name="etudiant" type="xsd:IDRef" use="required"/>
  <xsd:attribute name="cours" type="xsd:IDRef" use="required" />
</xsd:complexType>
```