

**Exercice 1.**

entraîneurs.dtd

```
<ELEMENT entraîneurs (entraîneur+)>
<ELEMENT entraîneur (nom, (club | nationale)*)>
<!ATTLIST entraîneur id ID #REQUIRED>
<ELEMENT club (debut, fin?)>
<!ATTLIST club nom CDATA #REQUIRED>
<ELEMENT nationale (debut, fin?)>
<!ATTLIST nationale nom CDATA #REQUIRED>
<ELEMENT nom (#PCDATA)>
<ELEMENT debut (#PCDATA)>
<ELEMENT fin (#PCDATA)>
```

**Exercice 2.**

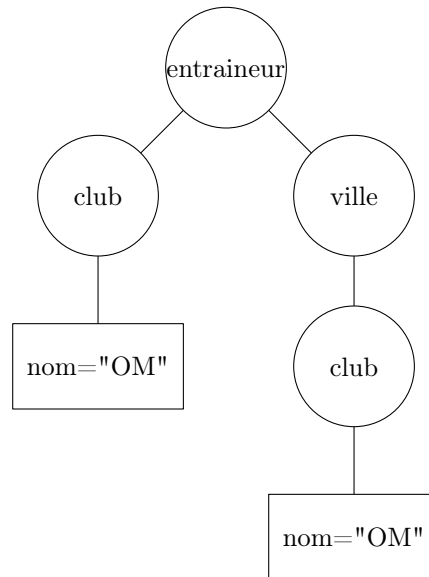
squelette.xsd

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:simpleType name="typeAnnee">
    <xs:restriction base="xs:string">
      <xs:pattern value="[0-9]{4}" />
    </xs:restriction>
  </xs:simpleType>
  <xs:complexType name="typeMandat">
    <xs:sequence>
      <xs:element name="debut" type="typeAnnee"/>
      <xs:element name="fin" type="typeAnnee" minOccurs="0"/>
    </xs:sequence>
    <xs:attribute name="nom" type="xs:string" use="required"/>
  </xs:complexType>
  <xs:complexType name="typeEntraîneur">
    <xs:sequence>
      <xs:element name="nom" type="xs:string"/>
      <xs:choice maxOccurs="unbounded">
        <xs:element name="club" type="typeMandat"/>
        <xs:element name="nationale" type="typeMandat"/>
      </xs:choice>
    </xs:sequence>
    <xs:attribute name="id" type="xs:string"/>
  </xs:complexType>
  <xs:element name="entraîneurs">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="entraîneur" type="typeEntraîneur" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

**Exercice 3.**

*Question 1.*

- (a) Les deux requêtes retournent 3.
- (b) Les deux requêtes ne donneront pas toujours le même résultat. Par exemple, l'arbre XML suivant produira des résultats différents.



Question 2.

- (a) `//entraîneur[nationale and not(nationale/fin)]/nom/text()`
- (b) `//entraîneur[club[@nom="LOSC" and not(fin)]]/nom/text()`

Exercice 4.

Question 1.

<b>entraîneur</b>	<b>club</b>	<b>debut</b>	<b>fin</b>
Rudi Garcia	LOSC	2008	2013
Rudi Garcia	OM	2016	
Marcelo Bielsa	OM	2014	2015
Marcelo Bielsa	LOSC	2017	
Didier Deschamps	OM	2009	2012

Question 2.

stylesheet.xsl

```

<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
  <xsl:output method="html"/>

  <xsl:template match="/">
    <html>
      <body>
        <table border="1">
          <tr>
            <th>entraîneur</th>
            <th>club</th>

```

```

                <th>debut</th>
                <th>fin</th>
            </tr>
            <xsl:apply-templates select="//entraîneur"/>
        </table>
    </body>
</html>
</xsl:template>
<xsl:template match="entraîneur">
    <xsl:apply-templates select="./club[fin]"/>
</xsl:template>

<xsl:template match="club">
    <tr>
        <td><xsl:value-of select="../nom"/></td>
        <td><xsl:value-of select="./@nom"/></td>
        <td><xsl:value-of select="debut"/></td>
        <td><xsl:value-of select="fin"/></td>
    </tr>
</xsl:template>
</xsl:stylesheet>

```

## Exercice 5.

### Question 1.

python-q1.py

```

1 #!/usr/bin/env python
2 # -*- coding: utf-8 -*-
3
4 # Importation de l'API.
5 import xml.etree.ElementTree as ET
6
7 # Chemin vers le fichier xml.
8 FILE = 'entraîneurs.xml'
9
10 # Création du parser et récupération de l'arbre XML du document.
11 root = ET.parse(FILE).getroot()
12
13 for club in root.findall('entraîneur/club'):
14     print(club.get('nom'))

```

### Question 2.

python-q2.py

```

1 #!/usr/bin/env python
2 # -*- coding: utf-8 -*-
3
4 # Importation de l'API.
5 import xml.etree.ElementTree as ET
6
7 # Chemin vers le fichier xml.
8 FILE = 'entraîneurs.xml'
9
10 # Création du parser et récupération de l'arbre XML du document.
11 root = ET.parse(FILE).getroot()
12
13 # Retourne la liste des entraîneurs associés à un club donné.
14 def getTrainer(nom):
15     l = []
16     for trainer in root.findall('entraîneur'):
17         for club in trainer.findall('club'):
18             clubName = club.get('nom')
19
20             if nom == clubName:
21                 l.append(trainer.get('id'))

```

```
22
23     return l
24
25 clubDict = {}
26
27 for club in root.findall('entraîneur/club'):
28     clubName = club.get('nom')
29     listTrainer = getTrainer(clubName)
30     if len(listTrainer) > 1:
31         print clubName
```

---