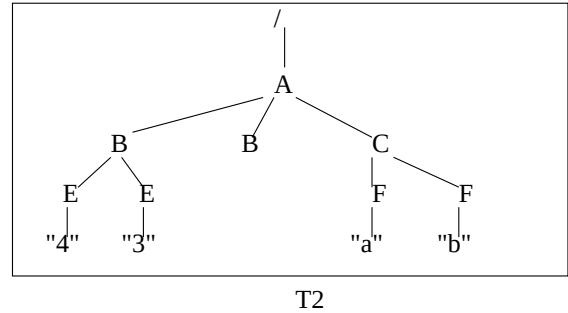
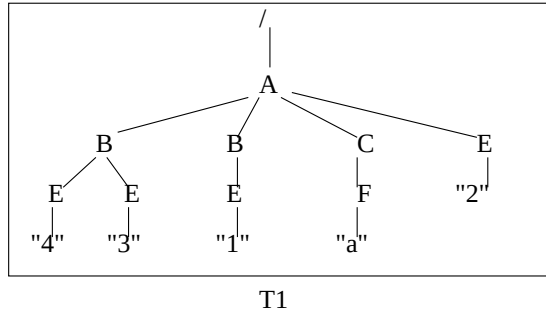


Bases de Données - Valeur C - Examen XML

24 Avril 2001 - 2 Heures

1 XML

La figure suivante montre deux arbres XML :



- Traduisez ces arbres en documents XML X1.xml et X2.xml.

Solution : X1.xml:

```

<?xml version="1.0" encoding="UTF-8"?>
<A>
  <B>
    <E>4</E>
    <E>3</E>
  </B>
  <B>
    <E>1</E>
  </B>
  <C>
    <F>a</F>
  </C>
  <E>2</E>
</A>
  
```

X2.xml:

```

<?xml version="1.0" encoding="ISO-8859-1"?>
<A>
  <B>
    <E>4</E>
    <E>3</E>
  </B>
  
```

```

    <B></B>
    <C>
      <F>a</F>
      <F>b</F>
    </C>
  </A>

```

2. Est-ce que la DTD suivante accepte les deux documents XML? Justifiez votre réponse.

```

<!ELEMENT A ((B)*,C) >
<!ELEMENT B (E)* >
<!ELEMENT C (F)+ >
<!ELEMENT E (#PCDATA) >
<!ELEMENT F (#PCDATA) >

```

Solution : Cette DTD accepte le document X2 mais pas le document X1 qui contient, par exemple, un élément de type A avec un fils de type E, ce qui n'est pas permis par la DTD.

3. Définissez une DTD qui accepte les deux documents (arbres) XML.

Solution :

```

<!ELEMENT A ((B)*,C,(E)?)>
<!ELEMENT B (E)*>
<!ELEMENT C (F)+>
<!ELEMENT E (#PCDATA)>
<!ELEMENT F (#PCDATA)>

```

4. Définissez une DTD qui accepte le premier document XML, mais pas le deuxième.

Solution :

```

<!ELEMENT A ((B)*,C,E)>
<!ELEMENT B (E)+>
<!ELEMENT C F>
<!ELEMENT E (#PCDATA)>
<!ELEMENT F (#PCDATA)>

```

2 XPath

1. Donnez le résultat des expressions XPath suivantes (vous pouvez numéroté les noeuds et donner le résultat sous forme d'une liste de numéros) :

(a) X1.xml/child::A/child::B

Solution : les deux premiers fils B du noeud A.

```

<output>
<B>
  <E>4</E>

```

```

        <E>3</E>
    </B>
    <B>
        <E>1</E>
    </B>
</output>

```

- (b) X1.xml/child::A/child::*[1]

Solution : le premier fils du noeud A

```

<output>
<B>
    <E>4</E>
    <E>3</E>
</B>
</output>

```

- (c) X2.xml/descendant::C/child::*

Solution : les deux fils F du seul noeud C

```

<output>
    <F>a</F><F>b</F>
</output>

```

- (d) X2.xml/descendant::B[child::E]

Solution : le premier fils B du noeud A

```

<output>
<B>
    <E>4</E>
    <E>3</E>
</B>
</output>

```

2. Donnez les expressions XPath pour trouver dans le document X1 :

- (a) tous les éléments de type E.

Solution : X1/descendant::E

- (b) tous les enfants d'éléments de type B.

Solution : X1/descendant::B/child::*

- (c) tous les noeuds de type texte (feuilles).

Solution : X1/descendant::text()

3 XSLT

1. Donnez le résultat de l'application de la feuille de style suivante au document (arbres) X1.xml :

```

<?xml version="1.0" encoding="ISO-8859-1"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform">

```

```

<xsl:template match='/'>
  <xsl:apply-templates select='A' />
</xsl:template>

<xsl:template match='A'>
  <A>
    <xsl:apply-templates select='B[1]' />
    <B></B>
    <xsl:apply-templates select='C' />
  </A>
</xsl:template>

<xsl:template match='B'>
  <B>
    <E> <xsl:value-of select='E' /> </E>
  </B>
</xsl:template>

<xsl:template match='C'>
  <C>
    <F> <xsl:value-of select='F' /> </F>
  </C>
  <C>
    <F>b</F>
  </C>
</xsl:template>
</xsl:stylesheet>

```

Solution :

```

<?xml version="1.0" encoding="utf-8"?>
<A>
  <B>
    <E>4</E>
  </B>
  <B/>
  <C>
    <F>a</F>
  </C>
  <C>
    <F>b</F>
  </C>
</A>

```

2. Modifiez la feuille de style pour

- (a) enlever tous les éléments (et ses sous-éléments) de type C
- (b) remplacer tous les chemins A/B/E par le chemin A/E.

Solution :

```

<?xml version="1.0" encoding="ISO-8859-1"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
  <xsl:template match="/">
    <xsl:apply-templates select='A' />
  </xsl:template>

  <xsl:template match='A'>
    <A>
      <xsl:apply-templates select='B[1]' />
      <B></B>
      <xsl:apply-templates select='C' />
    </A>
  </xsl:template>

  <xsl:template match='B'>
    <E> <xsl:value-of select='E' /> </E>
  </xsl:template>

  <xsl:template match='C' />

</xsl:stylesheet>

```

4 Langage de requêtes

Donnez le résultat de la requête XML-QL suivante :

```

<RESULTAT>
  where <$L><E> $E </></> in document('X1.xml')
  construct <$L> La valeur du fils E est $E </>
</RESULTAT>

```

Solution :

```

<RESULTAT>
  <B> La valeur du fils E est 4 </B>
  <B> La valeur du fils E est 3 </B>
  <B> La valeur du fils E est 1 </B>
  <A> La valeur du fils E est 2 </B>
</RESULTAT>

```