

# Examen Structuration de l'information XML

Durée 2H

Ordinateur interdit

Aucun document autorisé sauf une feuille A4 recto-verso

## Introduction

On étudie une nomenclature de produits vendus par un magasin de bureautique. La description des différents produits vous est donnée à travers le schéma XSD nomenclature.xsd en annexe. Tous les produits ont des caractéristiques communes (exemple : la référence) et d'autres propres à chaque type de produit (exemple : un crayon a une dureté). Un lot est un produit qui est constitué d'autres produits du catalogue. Les prix sont stockés en centimes d'euros avec des entiers.

## Questions

**Q1 :** Il manque des cardinalités dans ce schéma. Signaler les changements à effectuer dans le schéma pour indiquer qu'un catalogue est constitué d'un nombre quelconque de produits et qu'un lot est constitué d'au moins une référence de produit.

**Q2 :** Donner un exemple XML de produit validé par ce schéma. On ne vous demande qu'un seul produit et on vous impose de prendre un cahier.

**Q3 :** Donner une DTD correspondante pour décrire un produit. Décrire uniquement le nœud produit et ses attributs.

**Q4 :** Donner une expression XPATH partant d'un produit et permettant de :

- a) savoir si c'est une gomme ;
- b) désigner tous les produits du catalogue de même marque que ce produit ;
- c) désigner toutes les caractéristiques données sous forme d'attribut (à n'importe quelle profondeur) de ce produit.

**Q5 :** Que calcule l'expression XPATH suivante, partant de l'élément catalogue ?

`max(produit[descendant-or-self::*/@couleur='rouge']/@prix)`

**Q6 :** Que calcule l'expression XPATH suivante, partant d'un produit constituant un lot ?

`sum(//produit[@reference = current()]/lot/refProduit/@value)/@prix - @prix`

**Q7 :** Dans le cadre d'une feuille XSL en mode texte, on considère déjà écrits les templates associés à chaque type de produit (stylo, crayon, etc.). Ces derniers affichent les caractéristiques propres à chaque type de produit. Ecrire un template pour un produit affichant toutes ses caractéristiques simples : celles communes à tous les produits (référence, marque, intitulé et prix) puis celles dépendant du type de produit ; on ajoutera à la fin la description du produit.

**Q8 :** Ecrire un template XSL permettant de produire les données suivantes :

- a) afficher la liste des produits de moins de 10 euros
- b) afficher la liste des marques avec pour chacune le nombre de produit, le prix moyen en euros et s'il existe des lots ou pas ; pour les lots, on se basera uniquement sur la marque du lot en ignorant celles des produits le constituant.

**Q9 :** Ecrire un template XSL permettant d'afficher récursivement un lot (ses caractéristiques ainsi que le détail des produits le constituant).

## Fichier XSD : nomenclature.xsd

```
<xs:schema version="1.0" xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:element name="catalogue" type="tCatalogue" />

  <xs:element name="produit" type="tProduit" />

  <xs:complexType name="tCatalogue">
    <xs:sequence>
      <xs:element ref="produit"/>
    </xs:sequence>
    <xs:attribute name="datePublication" type="xs:date" />
  </xs:complexType>

  <xs:complexType name="tProduit">
    <xs:annotation>
      <xs:documentation>prix en centimes d'euros</xs:documentation>
    </xs:annotation>
    <xs:sequence>
      <xs:element name="description" type="xs:string" />
      <xs:choice>
        <xs:element name="stylo" type="tStylo" />
        <xs:element name="crayon" type="tCrayon" />
        <xs:element name="cahier" type="tCahier" />
        <xs:element name="gomme" type="tGomme" />
        <xs:element name="lot" type="tLot" />
      </xs:choice>
    </xs:sequence>
    <xs:attribute name="reference" type="xs:ID" use="required" />
    <xs:attribute name="marque" type="xs:string" use="required" />
    <xs:attribute name="prix" type="xs:unsignedInt" use="required" />
    <xs:attribute name="intitulé" type="xs:string" use="required" />
  </xs:complexType>

  <xs:complexType name="tStylo">
    <xs:attribute name="typeStylo" type="tTypeStylo" use="required" />
    <xs:attribute name="couleur" type="tCouleur" use="required" />
  </xs:complexType>

  <xs:simpleType name="tTypeStylo">
    <xs:restriction base="xs:string">
      <xs:enumeration value="bille" />
      <xs:enumeration value="plume" />
      <xs:enumeration value="feutre" />
    </xs:restriction>
  </xs:simpleType>

  <xs:simpleType name="tCouleur">
    <xs:restriction base="xs:string">
      <xs:enumeration value="jaune" />
      <xs:enumeration value="vert" />
      <xs:enumeration value="rouge" />
      <xs:enumeration value="bleu" />
      <xs:enumeration value="noir" />
      <xs:enumeration value="blanc" />
      <!-- TODO à compléter -->
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name="tCrayon">
    <xs:attribute name="dureté" type="tDureté" use="required" />
  </xs:complexType>
```

```

<xs:simpleType name="tDuret ">
  <xs:restriction base="xs:string">
    <xs:enumeration value="4B" />
    <xs:enumeration value="2B" />
    <xs:enumeration value="B" />
    <xs:enumeration value="HB" />
    <xs:enumeration value="H" />
    <xs:enumeration value="2H" />
    <xs:enumeration value="4H" />
  </xs:restriction>
</xs:simpleType>

<xs:complexType name="tCahier">
  <xs:annotation>
    <xs:documentation>grammage en grammes</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="dimension" type="tDimension" />
  </xs:sequence>
  <xs:attribute name="typeCahier" type="tTypeCahier" use="required" />
  <xs:attribute name="couleur" type="tCouleur" use="required" />
  <xs:attribute name="carreaux" type="tCarreaux" use="required" />
  <xs:attribute name="grammage" type="xs:unsignedShort" use="required" />
  <xs:attribute name="nombreFeuilles" type="xs:unsignedShort" use="required" />
</xs:complexType>

<xs:simpleType name="tTypeCahier">
  <xs:restriction base="xs:string">
    <xs:enumeration value="piqu " />
    <xs:enumeration value="spiral " />
  </xs:restriction>
</xs:simpleType>

<xs:simpleType name="tCarreaux">
  <xs:restriction base="xs:string">
    <xs:enumeration value="petits" />
    <xs:enumeration value="grands" />
    <xs:enumeration value="sans" />
  </xs:restriction>
</xs:simpleType>

<xs:complexType name="tDimension">
  <xs:annotation>
    <xs:documentation>dimensions en millim tres</xs:documentation>
  </xs:annotation>
  <xs:attribute name="largeur" type="xs:unsignedShort" />
  <xs:attribute name="hauteur" type="xs:unsignedShort" />
</xs:complexType>

<xs:complexType name="tGomme" />

<xs:complexType name="tLot">
  <xs:annotation>
    <xs:documentation>NB: par simplification, il n'y a pas de notion de
    quantit  associ e   chaque produit du lot</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="refProduit">
      <xs:complexType>
        <xs:attribute name="value" type="xs:IDREF" />
      </xs:complexType>
    </xs:element>
  </xs:sequence>
</xs:complexType>

</xs:schema>

```