

## XML

### Exercices dirigés / Travaux pratiques : XML, DOM CORRIGE

#### 1. Travaux dirigés

A. Proposez une correction pour rendre bien-formé le document XML suivant.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<course numéro="8"> (ce n'est pas une erreur, mais si on veut que numéro soit de type ID, il ne peut pas avoir des valeurs numériques, mais par exemple "c8" au lieu de "8")
  <gagnant temps=35>Dupont</gagne> (35 entre guillemets, </gagnant>)
  <2nd>Durand</2nd> (2nd ne peut pas être nom de balise, à remplacer par second par exemple)
  <les arbitres> <arbitre1 nom='Martin'> <arbitre2 nom='Legrand'> </les arbitres>
(les arbitres ne peut pas être balise à cause de l'espace, remplacer par les-arbitres par exemple ; les balises arbitre1 et arbitre2 doivent être fermées)
  <début format heure="hh:mm"> 13:00 <fin> 13:53 </début> </fin> (attribut format heure doit être par exemple format-heure à cause de l'espace ; les balises début et fin se chevauchent)
</course>
<course numéro="9"> (il faut une seule racine, on doit rajouter par exemple une balise courses qui contienne les deux balises course)
  <gagnant/>Rameau</gagnant> (gagnant ne doit pas se fermer tout de suite)
</course>
```

B. Donner les noms qualifiés et les noms universels de chaque élément et attribut du document XML suivant :

```
<?xml version="1.0"?>
<A>
  <B xmlns="http://a.b.c/" attb="abc">
    <C attc="bcd"/>
    <p:D xmlns:p="http://c.d.e/">
      <p:E atte="def">
        <F/>
        <p:G xmlns:p="http://f.g.h/">
          <p:H p:atth="ghi"/>
        </p:G>
      </p:E>
      <I p:atti="hij"/>
    </p:D>
  </B>
  <J attj="ijk"/>
  <q:K xmlns:q="http://j.k.l/">
    <L attl="klm">
      <p:L/>
    </L>
  </q:K>
</A>
```

Éléments : on prend l'espace de noms (pour le préfix de l'élément, ou sans préfix si l'élément n'en a pas) de l'ancêtre le plus proche

- A
- http://a.b.c/:B

- http://a.b.c/:C
- http://c.d.e/:D
- http://c.d.e/:E
- http://a.b.c/:F
- http://f.g.h/:G
- http://f.g.h/:H
- http://a.b.c/:I
- J
- http://j.k.l/:K
- L
- p:L est une erreur car le préfix p n'est pas défini pour p:L

Attributs : seuls les attributs avec préfix ont un espace de nom

- attb
- attc
- ate
- http://f.g.h/:atth
- http://c.d.e/:atti
- attj
- attl

C. Ecrire une méthode Java qui reçoit l'élément racine d'un document XML DOM et qui :

a. Retourne le nombre d'éléments dans le document

```
private static int compteElem(Node noeud){
    int resultat = 0;

    if (noeud.getNodeType() == Node.ELEMENT_NODE){
        resultat++;
    }
    //parcours récursif
    if (noeud.hasChildNodes()){
        NodeList enfants = noeud.getChildNodes();
        for(int i=0; i<enfants.getLength(); i++){
            //appel récursif
            int resenf = compteElem(enfants.item(i));

            //combiner resenf avec resultat
            resultat += resenf;
        }
    }
    return resultat;
}
```

b. Affiche seulement le texte du document

```
private static void afficheTexte(Node noeud){

    if (noeud.getNodeType() == Node.TEXT_NODE){
        System.out.print(noeud.getValue());
    }
    //parcours récursif
    if (noeud.hasChildNodes()){
        NodeList enfants = noeud.getChildNodes();
        for(int i=0; i<enfants.getLength(); i++){
            //appel récursif
            afficheTexte(enfants.item(i));
        }
    }
}
```

```
    }  
  }  
}
```

c. Affiche la somme des valeurs de tous les éléments « montant ».

```
private static double sommeMontants (Node r){  
    double somme = 0.0 ;  
    //trouver montants  
    Element er = (Element) r;  
    NodeList montants = er.getElementsByTagName("montant");  
  
    //parcours montants  
    for(int i=0; i<montants.getLength(); i++){  
        String val = montants.item(i).getTextContent();  
        //convertit en valeur numérique  
        double dval = Double.parseDouble(val) ;  
        somme = somme + dval ;  
    }  
  
    return somme ;  
}
```

Remarque : `Double.parseDouble` peut lever une exception, qui est laissée de côté dans cet exemple

## 2. Travaux pratiques

Soit le fichier XML *flux.xml*, disponible sur la page web du cours (<https://depinfo.u-cergy.fr/~vodislav/LP-WAM>). Téléchargez-le dans votre répertoire de travail.

- A. Le document est mal formé et plusieurs erreurs sont signalées. Corrigez ces erreurs de la façon la plus naturelle, pour rendre le document bien formé. Utilisez l'outil de validation XML en ligne du W3C (<http://validator.w3.org>) vérifiez le caractère bien formé de ce document. Utilisez l'onglet « Validate by Direct Input » en copiant son contenu dans la fenêtre proposée.
- B. En utilisant le modèle de programme *DOMTest.java* présent sur le site, écrivez les programmes suivants. Pour cela, il faut copier le modèle vers un fichier Java dans le répertoire de votre projet XML, ensuite modifier ce fichier pour réaliser les actions demandées.
  - 1) Programme qui affiche le nombre de nœuds texte dans le document *flux.xml* corrigé. Comment expliquez-vous le résultat par rapport à ce que vous pouvez compter en visualisant le document ?
  - 2) Programme qui affiche les titres des dépêches dans le fichier *flux.xml*