

TD 1 : XML et DTD

Ex1. Syntaxe XML

Objectif: Cette section a pour objectif de vous initier aux documents XML

Q : Distinguez les noms XML correct des noms incorrect et corrigez les erreurs.

- `<Drivers_License_Number>98 NY 32</Drivers_License_Number>`

Correction: Correct

- `<Driver's_License_Number>98 NY 32</Driver's_License_Number>`

Correction: Incorrect (apostrophe)

- `<month-day-year>7/23/2001</month-day-year>`

Correction: Correct

- `<first name>Alan</first name>`

Correction: Incorrect (présence d'un espace)

- `<àçttûä>øåú</àçttûä>`

Correction: Correct

- `<first_name>Alan</first_name>`

Correction: Correct

- `<month/day/year>7/23/2001</month/day/year>`

Correction: Incorrect (à cause des /)

- `<_4-lane>I-610</_4-lane>`

Correction: Correct

- `<téléphone>011 33 91 55 27 55 27</téléphone>`

Correction: Correct

- `<4-lane>I-610</4-lane>`

Correction: Incorrect (un nom XML ne commence pas par un chiffre)

Q : Lisez les exemples suivants et vérifiez si les documents XML sont bien formés (et expliquez pourquoi si ce n'est pas le cas):

Correction: Il comprend un entête XML du type `<?xml version="1.0" ?>` Il existe un seul élément racine qui contient tous les autres éléments Les balises sont correctement imbriquées : les balises ouvrantes ont une balise fermante associée et il n'y a pas de chevauchement Le nom des balises est libre mais il contient au moins une lettre Les attributs des balises, lorsqu'ils existent (ce n'est pas le cas), ont obligatoirement une valeur qui doit toujours apparaître entre quotes

- `<?xml version="1.0" ?>`

```
<top>
  <item>Question 1<item answer="a">
  <item>Question 2<item answer="b">
  <item>Question 3<item answer="c">
</top>
```

Correction: `<item>` n est pas fermé des attributs ne se mettent pas dans la balise fermante

- `<?xml version="1.0" ?>`

```
<text>
  <font size='8pt'>petite police</font>
  <font size='24pt'>grande police</font>
</text>
```

Correction: Bien formé

- `<?xml version="1.0" ?>`

```
<top> <item val=2/>
  <item val=3/>
  <item val=12/>
</top>
```

Correction: Manque des guillemets pour les attributs

- ```
<?xml version="1.0" ?>
<text>
 Un petit texte
 Un grand texte
</text>
```

Correction: pas d'espaces dans les noms de balise (ou alors small est un attribut mais doit être suivi par affectation à une valeur)

- ```
<?xml version="1.0" ?>
<test>
  <elem att1= toto />
  <elem att2= titi />
</test>
```

Correction: Bien formé, les attributs ne sont pas nécessaires

- ```
<?xml version="1.0"?>
<a>
 <b a="toto">Et hop

<a>
 <b a="titi">Voil_a

```

Correction: Manque un élément Racine

- ```
<?xml version="1.0"?>
<programme titre="Internet">
<ul>
  <li>XML</li>
  <li>DTD</li>
  <li>API</li>
  <li>XSL</li>
</ul>
</programme>
```

Correction: les balises fermantes ne sont pas bonnes

- ```
<?xml version="1.0"?>
<a>
 <c/><d>

```

Correction: Mauvaise balise </b> </d>

- ```
<?xml version="1.0"?>
<!DOCTYPE text [ <!ELEMENT text EMPTY> ]>
<text><bf>Titre</bf></text>
```

Correction: Bien formé mais non valide (le DTD interne impose que l'élément text soit vide)

- ```
<?xml version="1.0" ?>
<niveaux>
 <truc>chose</truc>
 <niveau index="1">
 <truc> </truc>
 </niveau>
 <niveau index="2">
```

```
<truc attribut="chose"> </truc>
</niveau> </niveaux>
```

Correction: Bien Formé (truc peut être utilisé à plusieurs endroits)

Remarque: provient en partie du document <http://apiacoa.free.fr/publications/teaching/xml/annals/>

## Ex2. Créer une DTD

Objectif: Composition d'éléments

Q : Etudier les exemples d'adresse suivants et déduisez en les éléments élémentaires qui les constituent:

- Exemple 1 - L'adresse d'un particulier: Johnny Begood 121 rue Plein-Ciel Nancy 54000
- Exemple 2 - Une adresse professionnelle Lucid-it 14 boulevard du 21ème R.A. "Le Carré Rive Gauche" Nancy 54000

Correction: Identifiant Numero Rue des informations supplémentaires éventuelles (« le carré rive gauche ») une ville un code postal

Q : Ecrire un document XML bien formé contenant les informations précédentes

Correction:

```
<?xml version="1.0" encoding="UTF-8" ?>
<carnet>
 <entite>
 <identifiant>
 <prenom>Johnny</prenom>
 <nom>Begood</nom>
 </identifiant>
 <adresse>
 <numero>121</numero>
 <rue>rue Plein Ciel</rue>
 <ville>Nancy</ville>
 <cpostal>54000</cpostal>
 </adresse>
 </entite>
 <entite>
 <identifiant>
 <nom_societe> Lucid-it</nom_societe>
 </identifiant>
 <adresse>
 <numero>14</numero>
 <rue> boulevard du 21ème R.A.</rue>
 <complement> Le Carré Rive Gauche</complement>
 <ville>Nancy</ville>
 <cpostal>54000</cpostal>
 </adresse>
 </entite>
</carnet>
```

Q : Concevoir une DTD validant votre fichier XML

Correction:

```
<!ELEMENT carnet (entite*)>
<!ELEMENT entite (identifiant,adresse)>
<!ELEMENT identifiant ((prenom,nom)|nom_societe)>
<!ELEMENT nom (#PCDATA)>
<!ELEMENT prenom (#PCDATA)>
<!ELEMENT nom_societe (#PCDATA)>
```

```
<!ELEMENT adresse (numero,rue,complement?,ville,cpostal)>
<!ELEMENT numero (#PCDATA)>
<!ELEMENT rue (#PCDATA)>
<!ELEMENT complement (#PCDATA)>
<!ELEMENT ville (#PCDATA)>
<!ELEMENT cpostal (#PCDATA)>
```

### Ex3. Etude avancée

Objectif: Définition avancée de la structure d'un document

Q : Définir un élément TRUC avec les contraintes suivantes :

- L'élément TRUC ne peut avoir que des sous-éléments A, B et C
- Ces sous-éléments doivent apparaître dans cet ordre
- L'élément TRUC doit avoir au moins un de ces sous-éléments
- L'élément TRUC ne peut pas avoir de double. Si il y a un sous-élément A, TRUC ne peut avoir de second sous-élément A.

Exemples de parties d'un fichier XML associée à la DTD

• `<TRUC><A/><B/><C/></TRUC>`

est valide

• `<TRUC><B/><C/></TRUC>`

est valide

• `<TRUC></TRUC>`

est non valide

• `<TRUC><A/><B/><B/></TRUC>`

est non valide

• `<TRUC><B/><A/><C/></TRUC>`

est non valide

Correction:

```
<!ELEMENT TRUC ((A,B?,C?)|(A?,B,C?)|(A?,B?,C))>
```

### Ex4. Question subsidiaire

Objectif: Définition avancée de la structure d'un document

Q : On donne ci-dessous l'extrait d'un fichier BiBTeX, un système permettant de répertorier des références bibliographique.

```
@InProceedings{Dalzilios:learnbs,
author = "Dal Zilio, Silvano and Bernard, Thierry M.",
title = "Learning Binary Shapes as Compression and its Cellular Implementation",
booktitle = "ACCV '95 -- 2nd Asian Conference on Computer Vision",
year = 1995,
volume = 2,
pages = "616--620",
month = dec,
url = "http://www.cmi.univ-mrs.fr/~dalzilio/Papers/accv95.ps",
abstract = We present a methodology to learn how to
recognize binary shapes, based on the principle that
recognition may be understood as a process of
information compression.
```

Dont voici une traduction XML possible:

```
<?xml version='1.0'?>
<bibtex-file>
<inproceedings>
<key>Dalzilios:learnbs</key>
<author>
```

```

<name>Dal Zilio, Silvano</name>
<name>Bernard, Thierry M.</name>
</author>
<title>Learning Binary Shapes as Compression and its Cellular Implementation</title>
<booktitle>
 <short>ACCV '95</short>
 <long>2nd Asian Conference on Computer Vision</long>
</booktitle>
<year>1995</year>
<volume>2</volume>
<pages first="616" last="620"/>
<month mtype="short">dec</month>
<url ftype="ps" href="http://www.cmi.univ-mrs.fr/~dalzilio/Papers/accv95.ps"/>
<abstract>
 We present a methodology to learn how to recognize binary shapes,
 based on the principle that recognition may be understood as a
 process of information compression.
</abstract>
</inproceedings>
</bibtex-file>

```

```

Correction: <!ENTITY % opt_fields "year?, volume?, pages?, month?, url?, abstract?, note?">
<!ENTITY % req_fields "author, title">
<!ENTITY % key_atts "key ID #REQUIRED">
<!ELEMENT bibtex-file (article|inproceedings|book|techreport|phdthesis|unpublished|misc)*>
<!ELEMENT key (#PCDATA)>
<!ELEMENT author (name+)>
<!ELEMENT name (#PCDATA)>
<!ELEMENT title (#PCDATA)>
<!ELEMENT booktitle (short, long?)>
<!ELEMENT short (#PCDATA)>
<!ELEMENT long (#PCDATA)>
<!ELEMENT year (#PCDATA)>
<!ELEMENT volume (#PCDATA)>
<!ELEMENT pages EMPTY>
<!ATTLIST pages
 first NMTOKEN #REQUIRED
 last NMTOKEN #REQUIRED>
<!ELEMENT month (#PCDATA)>
<!ATTLIST month
 mtype (short | long) "short">
<!ELEMENT url EMPTY>
<!ATTLIST url
 ftype (ps|pdf|html) #REQUIRED
 href CDATA #REQUIRED>
<!ELEMENT abstract (#PCDATA)>
<!ELEMENT number (#PCDATA)>
<!ELEMENT note (#PCDATA)>
<!ELEMENT publisher (#PCDATA)>
<!ELEMENT institution (#PCDATA)>
<!ELEMENT school (#PCDATA)>
<!ELEMENT howpublished (#PCDATA)>
<!ELEMENT address (#PCDATA)>
<!ELEMENT inproceedings (%req_fields;, booktitle, %opt_fields;)>

```

```
<!ATTLIST inproceedings %key_atts;>
<!ELEMENT article (%req_fields;, journal, %opt_fields;)>
<!ATTLIST article %key_atts;>
<!ELEMENT book (%req_fields;, publisher, %opt_fields;)>
<!ATTLIST book %key_atts;>
<!ELEMENT techreport (%req_fields;, institution?, number?, %opt_fields;)>
<!ATTLIST techreport %key_atts;>
<!ELEMENT phdthesis (%req_fields;, school?, %opt_fields;)>
<!ATTLIST phdthesis %key_atts;>
<!ELEMENT unpublished (%req_fields;, %opt_fields;)>
<!ATTLIST unpublished %key_atts;>
<!ELEMENT misc (%req_fields;, howpublished, %opt_fields;)>
<!ATTLIST misc %key_atts;>
```

Q : Ecrivez une DTD pour ce fichier xml en prenant conscience que:

- il peut y avoir un nombre infini d'auteurs
- l'élément booktitle peut ne pas contenir d'élément fils "long"
- une url peut soit pointer sur un fichier de type "ps", "pdf" ou "html"
- un mois peut être soit "short" ou "long"

Q : Reprendre la DTD en considérant que bibtex-file peut contenir d'autres types de publications (autres que inproceedings). Vous utiliserez des entités pour alléger le fichier DTD.

- Article : champs requis= (author, title, journal) et optionels= (year, volume, pages, month, url, abstract, note)
- PhdThesis : champs requis= (author, title) et optionels= (school, year, volume, pages, month, url, abstract, note)
- Book : champs requis= (author, title, publisher) et optionels= (year, volume, pages, month, url, abstract, note)
- Techreport : champs requis= (author, title) et optionels= (institution, number, year, volume, pages, month, url, abstract, note)
- Unpublished : champs requis= (author, title) et optionels= (year, volume, pages, month, url, abstract, note)
- Misc : champs requis= (author, title, howpublished) et optionels= (year, volume, pages, month, url, abstract, note)