TP XML «1»

Goals: XML Documents, DTD/schema, style sheets CSS
Key words: XML, DTD, CSS

Important: TPs will be evaluated and scored (part of your final note/score). I will choose the TP to score (TP1, TP2, TP3 or a mix between both: eg. exercises 1 and 2 from TP1 and exercise 2 from TP2, etc.). It is of a prime importance that you submit your tps at the indicated deadline (to be defined by your TP teacher). After this deadline, no TP will be accepted.

Step 1: DTD definition and XML documents, Layout/Formatting by means of CSS.

Topic: "Au Cinéma!"

Les Brigades du Tigre
- Thriller (2h 5min)
- By Jérôme Cornuau
- VF (en French)
- With C. Cornillac, D. Kruger, E. Baer
- Year: 2005
- Press : 3/5 Spectators : 4/5

About the adventures of commissar Valentin and inspectors Terrasson and Pujol, members of the " Brigades mobiles", special squad of the French police before the first world war..

Wed, Sat : 14:00 | 16:40 | 19:40 | 22:10
Tue, Fri, Mon, Thu : 14:00 | 19:40 | 22:10
Sun : 10:20 | 14:00 | 16:40 | 20:20

In this exercise we propose to design a system that allows publishing cinema programs into the web. The program consists of description of each movie playing at the cinema for one week. As in the example, above, every movie is described by its title, duration, kind, movie maker, actors, year of production, language (French or OV) and a short paragraph for describing the story. This description is completed by the list of projection times (days, time).

Modeling and validating the XML data

1. Propose an XML tree for representing the precedent movie (Les brigades du tigres).
2. Which elements must be added for building an XML document representing the whole set of cinema programs? You XML document should contain at least 5 different movies.
3. Propose a DTD for validating the XML trees used for representing the cinema programs. Your DTD should respect the following constraints:
   - Press and spectators rating are optional.
   - Every movie must have a title
   - Every movie must have a projection time.
4. Propose a less restrictive DTD that would allow film descriptions given in any order.

Layout and display using CSS (Optional)

You have to write a CSS file in order to display your XML document. The following constraints have to be taken into account:

a. Every movie is displayed in a separated bloc as follows:
b. You will use the police : Times New Roman 12 points,
c. Titles are en bold, 14 points,
d. The movie maker and the year of production should be in red,
e. Actors are in blue, italic, and represent Hyperlinks to html pages describing the actor filmography.
f. For the movie scheduling, the student is free to give any (original) format of display.

PS: For CSS properties, examples, functions and facilities, please refer to:
http://www.w3schools.com/css/css_text.asp

Step 2: XML schema

a) Propose a well-formed XML document of what could be your address Book.
Each record, called « contact », should specify:

- Indication whether the contact is a person or a company, etc.,
- Its name,
- Surname (Only for persons),
- Phone number
- Address in terms of Street number and name, city, zip code and eventually a country.

b) Give the XML schema for the proposed XML documents (the one you just built). The XML schema should be specified in a separate file (external) with .xsd extension. Add a reference toward that schema within your XML document.

- You should validate your XML document using the following link:
http://tools.decisionsoft.com/schemaValidate/

c) Given the following XML schema:

```
<?xml version="1.0" encoding="UTF-8"?>

<bank xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="bank.xsd">
 <accounts>
  <savings_accounts>
   <savings_account id="a1" interest="0.03">
    <balance>2500</balance>
   </savings_account>
   <savings_account id="a2" interest="0.03">
    <balance>15075</balance>
   </savings_account>
  </savings_accounts>

  <checking_accounts>
   <checking_account id="a3">
    <balance>4025</balance>
   </checking_account>
   <checking_account id="a4">
    <balance>-125</balance>
   </checking_account>
   <checking_account id="a5">
    <balance>325</balance>
   </checking_account>
  </checking_accounts>
 </accounts>
  <customers>
  <customer id="c1">
   <name>Ben Richerdson</name>
   <address>Park Drive 2</address>
  </customer>
  <customer id="c2">
   <name>Marc Wretcher</name>
```

```
    <address>Mill Drive 75</address>
   </customer>
   <customer id="c3">
    <name>Angel Steady</name>
    <address>Lake Sight 15</address>
   </customer>
  </customers>
   <customer_accounts>
   <customer_account c_id="c1" ac_id="a2"/>
   <customer_account c_id="c1" ac_id="a3"/>
   <customer_account c_id="c2" ac_id="a4"/>
   <customer_account c_id="c3" ac_id="a1"/>
   <customer_account c_id="c3" ac_id="a5"/>
  </customer_accounts>

</bank>
```

Give an XML Schema for bank.xml satisfying the following requirements:

- There are two account types: checking and savings accounts
- The account id is unique in 'accounts'
- The customer id is unique in 'customers'
- 'c_id' refers to customers and 'ac_id' refers to accounts
- The account balance must be greater than "-500"
- Use inheritance for checking and savings accounts by deriving from a common account type.

You should validate your XML document/XML schema using the following link: http://tools.decisionsoft.com/schemaValidate/

Step 3:

Given the following XML file

```
<?xml version="1.0" encoding="utf-8"?>
<book isbn="0836217462">
 <title>
  Being a Dog Is a Full-Time Job
 </title>
 <author>Charles M. Schulz</author>
 <character>
  <name>Snoopy</name>
  <friend-of>Peppermint Patty</friend-of>
  <since>1950-10-04</since>
  <qualification>
    extroverted beagle
  </qualification>
 </character>
 <character>
  <name>Peppermint Patty</name>
  <since>1966-08-22</since>
  <qualification>bold, brash and tomboyish</qualification>
 </character>
</book>
```

We want to define a grammar for the XML document given above, using XML Schema. At this aim, start by:

1-Giving the XML schema prologue as well as the instruction to add in the XML file.

2-Give the XML Schema grammar used to instantiate the given XML document while taking into account the following constraints:

- The title (title), the author's name (author), the character's name (name) must not exceed 40 characters;
- A character (Character) may have more than one friend (friend-of), fiend's descriptions must not exceed 40 characters as well;
- The (since) tag is of type date;
- The ISBN's book is composed of exactly 10 figures (digit).