XML practice **Lab Tasks**

1. Download the reviews_bad.xml and Employees.xml files from link on web page

2. Task #1: Use an XML editor or a text editor (make sure you can save as ASCII text) to edit the "reviews_bad.xml" XML file to correct the DTD and to correct the XML so that it validates using https://www.xmlvalidation.com

3. Task #2: Use an XML editor or a text editor to create an XML file with an embedded DTD (see Employees.xml in materials directory for an example of valid XML file with embedded DTD), using elements and a couple of attributes, to describe:

   - Your name (distinguishing first, middle, surname)
   - Student ID
   - Favourite music groups
   - Favourite County in Ireland
   - Expected date of graduation

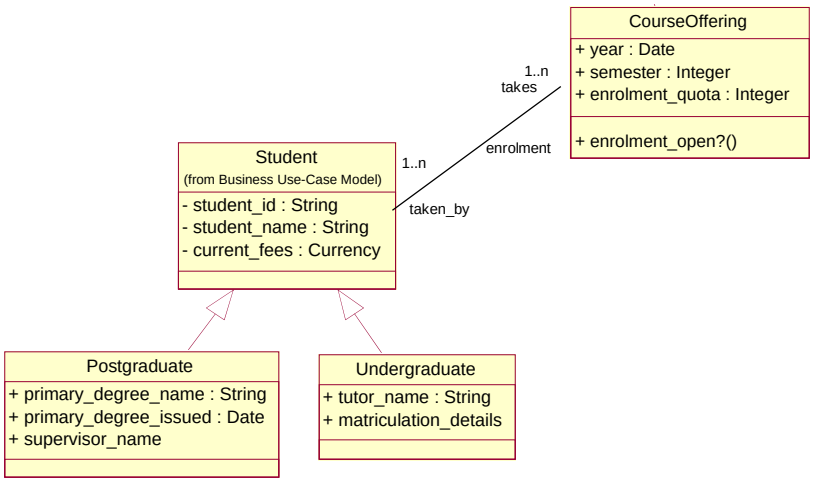
   Again use https://www.xmlvalidation.com to validate.

4. Task #3: Download Core Package Java BaseX to your laptop or your U: drive or to D: drive on PC (http://basex.org/products/download/all-downloads)

5. Task #4: Open up BaseX, create a database, and load the XML files you created in Tasks #1 and #2 and explore using the BaseX visualisation tools.

6. Task #5: Create two XML files (with embedded DTDs) that represent the diagram class model below. Just use ELEMENTS rather than any attributes. One XML representing Students (with a mixture of individual Undergrads and Postgrads included), and taking care to represent the inherited attributes. One XML representing CourseOfferings (especially taking care to represent the association between it and Students file) with a number of sample CourseOffering instances. You do **not** need to take the XMI approach discussed during lectures.

   Validate using the **https://www.xmlvalidation.com**
   Load into your BaseX database and visualise.



Task #6: Have a go at creating some of the XML needed for your group project

XPath practice

7. **Design XPath queries for the following.**
   All relate to the XML document bib.xml:

a. Return entire bib XML tree

You should get the entire XML document printed out including the root element

b. Get all the titles of books in the file (without using //) you should get:

<title>TCP/IP Illustrated</title>
<title>Advanced Programming in the Unix environment</title>
<title>Data on the Web</title>
<title>Economics of ... for Digital TV</title>

c. Get just the text from the first name elements of author You should get:

W.W.SergePeterDanDarcy

d. Return only the book element that has an editor You should get:

<book year="1999" > <title>Economics of ... for Digital TV</title>
<editor> <last>Gerbarg</last>
<first>Darcy</first>
<affiliation>CITI</affiliation>
</editor>
<publisher>Kluwer Academic Publishers</publisher>
<price>129.95</price>
</book>

e. Return only the books that are published after 1998 You should get:

<book year="2000" > <title>Data on the Web</title>

<author><last>Abiteboul</last>
<first>Serge</first>
</author>
<author><last>Buneman</last>
<first>Peter</first>
</author>
<author><last>Suciu</last>
<first>Dan</first>
</author>
<publisher>Morgan Kaufmann Publishers</publisher>
<price>39.95</price>
</book>
<book year="1999" > <title>Economics of ... for Digital TV</title>
<editor> <last>Gerbarg</last>
<first>Darcy</first>
<affiliation>CITI</affiliation>
</editor>
<publisher>Kluwer Academic Publishers</publisher>
<price>129.95</price>
</book>

f. Return the entire book element whose title is "Data on the Web" You should get:

<book year="2000" > <title>Data on the Web</title>
<author><last>Abiteboul</last>
<first>Serge</first>
</author>
<author><last>Buneman</last>
<first>Peter</first>
</author>
<author><last>Suciu</last>
<first>Dan</first>
</author>
<publisher>Morgan Kaufmann Publishers</publisher>
<price>39.95</price>
</book>

g. Alter the last query to just return the second author You should get:

<author>
  <last>Buneman</last>
  <first>Peter</first>
</author>

h. Return those books which are priced between 50 and 100 only You should get:

<book year="1994" > <title>TCP/IP Illustrated</title>
<author><last>Stevens</last>
<first>W.</first>
</author>
<publisher>Addison-Wesley</publisher>

<price> 65.95</price>
</book>
<book year="1992" > <title>Advanced Programming in the Unix environment</title>
<author><last>Stevens</last>
<first>W.</first>
</author>
<publisher>Addison-Wesley</publisher>
<price>65.95</price>
</book>

i. **Return all those books that are NOT published by Addison-Wesley You should get:**

<book year="2000" > <title>Data on the Web</title>
<author><last>Abiteboul</last>
<first>Serge</first>
</author>
<author><last>Buneman</last>
<first>Peter</first>
</author>
<author><last>Suciu</last>
<first>Dan</first>
</author>
<publisher>Morgan Kaufmann Publishers</publisher>
<price>39.95</price>
</book>
<book year="1999" > <title>Economics of ... for Digital TV</title>
<editor> <last>Gerbarg</last>
<first>Darcy</first>
<affiliation>CITI</affiliation>
</editor>
<publisher>Kluwer Academic Publishers</publisher>
<price>129.95</price>
</book>

XQuery practice

8. In the Query Editor, try to design **XQueries**, against the XML files you have downloaded, to do the following:

   (**Remember that if you have not imported the XML files into BaseX, then use doc("filename") in the XQuery to indicate what XML file you are querying and make sure that document is in the folder from which you invoked BaseX**)

a. With bib as a root ("bib.xml"), list books published by Addison-Wesley after 1991, including their year and title. You should get:

<bib>

```
    <book year="1994">
      <title>TCP/IP Illustrated</title>
    </book>
    <book year="1992">
      <title>Advanced Programming in the Unix environment</title>
    </book>
  </bib>
```

b. List the titles and years of all books with bib as a root (in bib.xml) published by Addison-Wesley after 1991, in alphabetic order. You should get:

```
<bib>
  <book year="1992">
    <title>Advanced Programming in the Unix environment</title>
  </book>
  <book year="1994">
    <title>TCP/IP Illustrated</title>
  </book>
</bib>
```

c. With root /bib find all titles that contain the word "the", regardless of the level of nesting. You should get:

```
<results>
  <title>Advanced Programming in the Unix environment</title>
      <title>Data on the Web</title>
</results>
```

d. For each book representing bookstore called “bstore1” [under root /bib (bib.xml)] and under respresenting bookstore called “bstore2” [under root /reviews (reviews.xml)], list the title of the book and its price from each source. You should get:

```
<books-with-prices>
  <book-with-prices>
    <title>TCP/IP Illustrated</title>
    <price-bstore2>65.95</price-bstore2>
    <price-bstore1>65.95</price-bstore1>
  </book-with-prices>
  <book-with-prices>
    <title>Advanced Programming in the Unix environment</title>
    <price-bstore2>65.95</price-bstore2>
    <price-bstore1>65.95</price-bstore1>
  </book-with-prices>
  <book-with-prices>
    <title>Data on the Web</title>
    <price-bstore2>34.95</price-bstore2>
    <price-bstore1>39.95</price-bstore1>
  </book-with-prices>
</books-with-prices>
```

e. Under the root /summary_prices (“overview_prices.xml”) find the minimum price for each book, in the form of a "minprice" element with the book title as its title attribute.

```
<results>
   <minprice title="Advanced Programming in the Unix environment">
      <price>65.95</price>
   </minprice>
   <minprice title="TCP/IP Illustrated">
      <price>65.95</price>
   </minprice>
   <minprice title="Data on the Web">
      <price>34.95</price>
   </minprice>
</results>
```

f. For each book under root bib (“bib.xml”) that has at least one author, list the title and first two authors, and an empty "et-al" element if the book has additional authors.

```
<bib>
   <book>
      <title>TCP/IP Illustrated</title>
      <author>
         <last>Stevens</last>
         <first>W.</first>
      </author>
   </book>
   <book>
      <title>Advanced Programming in the Unix environment</title>
      <author>
         <last>Stevens</last>
         <first>W.</first>
      </author>
   </book>
   <book>
      <title>Data on the Web</title>
      <author>
         <last>Abiteboul</last>
         <first>Serge</first>
      </author>
      <author>
         <last>Buneman</last>
         <first>Peter</first>
      </author>
      <et-al/>
   </book>
</bib>
```