

Assignment Two – XML

Weight: 10%**Deadline: 15:30 (BST) 28 March 2017**

In pairs, or as a group of 3, (split your groups from the first coursework into two)

1. Design an XML file and associated XML schema
2. Define XPath expressions to extract data
3. Design XSLT files to display the data in a browser

The material is all contained in the 3 XML lectures. An excellent piece of coursework will be well-presented and demonstrate the full range of the material discussed in the lectures.

The two groups from the original coursework should work independently of each other, devising their own schema and queries. You may want to focus on different parts of your original database to avoid any similarities.

1. Design an XML file to contain some of the data from your relational database project. The file should represent at least a one-to-many, and maybe a many-to-many relationship. *Choose the data to make an interesting XML file rather than strictly keeping to the data in your database, i.e. think about the XML hierarchical model as opposed to the relational model you had before.* Create a file with sample data. Choose suitable elements and enough data chosen to enable you to provide a range of XPath expressions in the 3rd section below. (5 marks)
2. Create an XML schema to describe the data in your XML file. Validate the XML file as well-formed by loading it into a browser and against your XML Schema at <http://www.xpathtester.com/validate> (5 marks)
3. Create XPath expressions of your choice (2 per person in the group) to obtain useful information. The results of these expressions should be data values, not elements. You should evaluate these using the XPath checking tool <http://www.xpathtester.com/xpath> (5 marks)
4. Create 1 XSLT file per person which transforms your original XML file to XHTML for display on a browser. (5 marks)

Marking 5 = outstanding, 4 = good, 3 = satisfactory, 2 = poor, 1 = very poor, 0 = nothing

Submission Instructions

The assignment is to be submitted through the two assignments on Vision.

1. Report through TurnItIn assignment. The report should contain:
 - The names of the people in the group.
 - Details of how the work was divided between the group members.
 - An overview in English of the contents of the XML file
 - For each XPath expression, a description in English, the expression, and the results (copied from running the query over the data).
2. An archive file as zip or tar.gz (no other archive formats will be marked) through the Grade Centre assignment. The archive file should contain:
 - XML file containing your data
 - XMLSchema file
 - For each XSL transformation
 - A link to open the XML and apply the transformation
 - The XSL file

Hints and Tips

- The most difficult bit is probably validating the XML file against the XML schema file – allow plenty of time.
- Just use a simple text editor NOT Word or Open Office Writer to create your XML files – you may get strange whitespace characters
- Don't leave any blank lines – (UNLIKE programming when blank spaces aid readability, they are invalid in XML) – if in doubt, remove other blank spaces.
- Don't use reserved XSL words like 'type' for your own element names.
- Write your own XML Schema rather than using a tool.

The tool versions that I have seen deduce the list of suitable elements from what you have provided, e.g. limiting the list of countries or suitable dates to only those in your XML file. While this sometimes makes sense (e.g. maybe you have `<gender>M</gender>` and `<gender>F</gender>` here a deduction that your only allowed values are M and F is ok), but it doesn't make sense in those other situations when your XML file just contains example values (like dates), but you'd like to be able to store any date not just those listed in your example file. **In short, a tool is likely to provide an impressive-looking, over-complicated and incorrect version.**

- Watch out for simple errors like misspelling of element names.
- Other XML Schema validation tools and XPath evaluation tools exist and may be used. Full details should be supplied.

Note that many XPath evaluation tools only support XPath 1.0; this means that many of the aggregation functions will not work.

Collaboration and Plagiarism

Coursework reports and code must be the group's own work. If some text or code in the coursework has been taken from other sources, these sources must be properly referenced. Failure to reference work that has been obtained from other sources or to copy the words and/or code of another student is plagiarism¹ and if detected, this will be reported to the School's Discipline Committee. If a group is found guilty of plagiarism, the penalty could involve voiding the course.

Students must never give hard or soft copies of their coursework reports or code to students in another group. Students must always refuse any request from another student not in their group for a copy of their report and/or code. It is expected that all group members will have read and write access to the report and code for their group.

Sharing a coursework report and/or code with another group is collusion, and if detected, this will be reported to the School's Discipline Committee. If found guilty of collusion, the penalty could involve voiding the course.

¹ Heriot-Watt guidelines on plagiarism
<https://www.hw.ac.uk/students/studies/examinations/plagiarism.htm>