

XSLT Programming

Instructions to run XSLT

You need to write an xslt program called course.xsl. You can run your program using the following command:

```
> java -classpath xalan\xalan.jar org.apache.xalan.xslt.Process -in course.xml -xsl course.xsl -out course.html
```

Before running the java program, you need to download xalan, the java implementation (processor) of XSLT, from xalan.apache.org. Alternatively, you can directly download from here the [xalan.jar file](#) and [serializer.jar file](#).

You need to put all the files in the directory where you run the command.

You can use other tools to run xslt, such as a browser. Most of the case the result will be the same as xalan. Since your assignment will be tested using xalan, it is safer for you to test on xalan before your submission.

Assignment specification

Given an XML document about courses in a university as [this](#), write an XSLT called course.xsl to extract the course ID, year, course title, prof, meeting days, time, and location, and present the data in an HTML file as in this sample [output](#). There are three requirements you need to take care of:

1. You should extract the courses that are offered since year 2003 (including 2003).
2. Some courses are taught by more than one prof. You need to list all the professors, separated by a comma ",". Please note that you should not print a comma at the very end of the professor list. you can consider to use `test="position()=last()"` in XSLT. Some courses are cross listed and hence have different course-ids. You need to list all the course IDs, separated by ",".
3. You do not need to handle special characters in the input;
4. When you click the course title, the course description will popup. This is implemented by JavaScript and you can reuse the javaScript as in the [output](#).

```
5.     function addDisplay (details) {  
6.         document.write(details);  
7.     }
```

Note that document.write in the above javaScript code simply print the string in its parameter to the page. What you need to consider is to pass the course description into javaScript function `addDisplay()`. The XSLT code used can include `<xsl:attribute name="href">`, which is introducing href attribute. Following that you can add the attribute value.