

COIN-078B : Internet Programming with XML

Assignment 2

Objectives:

- Learn how to produce a paginated presentation of our XML information with [XSLT](#) and [XSLFO](#).
- Use the open-source [FOP \(Formatting Objects Processor\)](#) from Apache.org.

Note: Please follow the specifications below. Points will be knocked off if not followed.

Read [Assignment Guidelines & Requirements](#)

The Extensible Stylesheet Language Formatting Objects (XSLFO), combines the heritage of Cascading Stylesheets (CSS) and Document Style Semantics and Specification Language (DSSSL) in a well-thought-out and robust specification of formatting semantics for paginating information. Layout and typesetting controls give us the power to express our information on pages in a visually pleasing and perhaps meaningful set of conventions conveying information in the presentation itself. In this assignment you are going to write an XSLT program that will transform an XML document [elements.xml](#) (same one you used in Assignment 1) to an XSLFO document. By using the open-source [FOP \(Formatting Objects Processor\)](#) from Apache.org, you can run the XSLT and the XSLFO processors on your xml and xsl files in a single command. The output formats it currently supported include PDF, PCL, PS, SVG, XML (area tree representation), Print, AWT, MIF and TXT. The output target for this assignment is PDF. The PDF output should contain a title called "Periodic Table of Elements" on the first page and a table containing rows of elements. The table columns are **Name, Symbol, Atomic Number, Atomic Weight, Oxidation States, Density, Electron Configuration, Electro negativity, Atomic Radius, Atomic Volume, Ionization Potential, Specific Heat Capacity, Thermal Conductivity** respectively. If available the units should be extracted from the appropriate attribute and placed in parenthesis under each label. Give the table rows any alternating color you like. Use the XSLT choose command, position() function & mod operator to check odd/even row and add the appropriate background-color attribute value. The elements are sorted by name. The bottom-right of each page displays the page number. See the [screenshot](#).

You'll need to read Chapters 4, 8 and 9 of your textbook to be able to do this assignment.

Look at the textbook examples from page 117 to 131. Start with creating an XSL-FO template with title heading and table containing the specified columns and labels, a data row filled with temporary "data" text and page number at the bottom-right corner of the page. Save this file as **temp.fo**. Run FOP (FOP temp.fo temp.pdf) and check to see if the PDF output is displayed correctly in the Acrobat Reader. Using this file as template, follow the instructions in Assignment 1 on creating XSLT stylesheets with the appropriate template rules and applying them to the nodes of your XML document. Be sure to declare the proper namespaces. Save your XSLT file as **assign2.xsl** in the **styles** folder.

Cheatsheet: Read [XML Bible, 2nd Edition: XSL Formatting Objects](#) - Listing 19-5 (Using fo:page-number to place the page number in the footer) and Listing 19-10 (An XSL stylesheet that formats the elements as a table). **Hint:** Visualize region-after by temporarily applying a background color to it. **Important:** The margins you set for the region-body must be greater than or equal to the extents of the region-before and region-after (eg. 5mm for the margins of region-body and 4mm for the extent of region-after.)

Modify elements.xml to reference the XSLT file. Add `<?xml-stylesheet type="text/xsl" href="styles/assign2.xsl" ?>` following the xml declaration in the prolog. **Note:** use relative path to the xsl file (MSXML3 has problem accessing xsl file in absolute path). Save the XML file as **assign2.xml**. Look at the assign2.xml in IE6 and note what you see. Does IE6 support XSLFO? View the xslt output.

As a convenience, create a batch (DOS) or script (UNIX) file containing the FOP command and the appropriate parameters and path to the files. For example, C:\Apache\fop-0.20.5\fop -xsl styles\assign2.xsl -xml assign2.xml -pdf assign2.pdf. Read the [online instructions](#) on starting FOP. Then run the batch file to test your code. Make sure to view the PDF file in the Acrobat Reader.

Generate the XSLFO file by transforming the assign2.xml file with the assign2.xsl file, using [Xalan](#) that is included in the FOP download. In DOS command console, enter the following in the FOP directory:

```
xalan -in assign2.xml -xsl assign2.xsl -out assign2.fo
```

Note: Provide the correct path to each of the files. View the generated FO file. What do you see when you load this in the browser?

Upload the generated XSLFO file and the XSLT stylesheet into the **styles** folder, XML data (elements), documentation and PDF files into the **public_html** directory of your CTIS website. Add the links to all these files in your index.htm file. Make a copy of all the files in a portable storage to bring to class. **(Classroom section ONLY!)**

Submit a hard copy of your source code with the url where your pages can be accessed on the due date. (Classroom section ONLY!)

Don't forget to backup your local files.

Helpful Resources:

- [XSLT](#)
- [XPath](#)
- [XSLFO](#)
- [XSLT and XPath Sample Viewer](#)
- [XSLFO Reference](#)
- [Apache.org: FOP \(Formatting Objects Processor\)](#)
- [Apache FOP Compliance](#)

Due Week 4