Homework 17: Working With XML

Goal: This assignment has three parts: To look at how an app uses XML, to use XML yourself, and to develop new XML tags.

Part 1: Look at how an app uses XML

The goal in this part is to become familiar with how the separation of data and processing takes place between an XML specification of data and the processing of that data. We will use picture viewer software as an example. An important aspect is:

Requirement: It's best if you use your own computer for this assignment; also, you will need at least nine JPG images on it. The pictures can be personal because you are the only one who will see them.

The steps of this part of the assignment are:
1. Install software
2. Set up the XML file
3. Look At the file gallery.xml
4. Experiment with Customize
5. Save an image of the XML file to turn in

The details follow.

Install Software

Install simpleviewer on your computer: http://www.simpleviewer.net/simpleviewer/ (One file works for PC, Mac and Linux; follow the usual procedure for your machine.) If you do not have Adobe Air installed on your machine, you will need to install that too: http://get.adobe.com/air/.

Set Up the XML File

Start the simpleviewer and load at least 9 images. (This is a sweet app, so you may want to load more, but for this assignment, just do 9-16.) This requires dragging if they are visible, say on the desktop, or by browsing if they are somewhere else.

Look At gallery.xml

After loading the images, use your text editor (notepad++ or TextWranger), open the gallery.xml file and look at what is stored there. Notice the tag structure of the file, that is, what tags are used and which are inside of which others?

Experiment With Customize

Now return to the Customize step of the setup. Try the different styles and pick one. What to notice: All of these styles build off of the same data. That is, the processing has changed, but the file of data is the way it was set up during the Images phase.

Save A Copy of gallery.xml

You will need to turn in the tagged file, so save a copy now. That's the end of Part 1.

Part 2: Use XML To Build A Data File

The goal of this part is to create an XML file of information of interest to you.

The steps of this part are as follows:
1. Create a folder, and inside of it, create another folder called im, for images.
2. Into the outer folder place the .xml and .xsl files from the Calendar page.
3. Open mydiary.xml using the browser and using the text editor.
4. Edit the mydiary.xml so that it is customized to yourself.

When step 4 is completed you must have at least **five days of diary entries**.

Create Folders

Create a folder on your desktop to hold your diary; perhaps mydiary would be a good name. It also needs a folder inside of it for images, and im is a good name for that folder.

Get Files

On the calendar page are files that must be moved into the folder. Get the mydiary_xml.txt file and the mydiary_xsl.txt files and put them in the folder, renamed mydiary.xml and mydiary.xsl. Into the images folder, move the three .jpg image files and the .png file provided on the Calendar.

Open Files

Open the mydiary.xml file with a browser and with a text editor. Also open the mydiary.xsl file with a text editor. Compare the three files – the data file (.xml in text editor), the processing file (.xsl) to style it, and the result (.xml displayed in the browser). Notice how the data is formatted by the instructions of the style file.

Edit To Build A Diary

Now REMOVE the entries from the mydiary.xml file and images from im, and add new ones to customize it to yourself. (You may also want to change the mydiary.xsl file if you don't like the formatting, such as background color.) You should have fun with building your diary, but be sure to use a good mix of tags. DO NOT BECOME TOO PERSONAL BECAUSE YOU NEED TO TURN THIS IN.

You are finished with this part when you have five days worth of custom entries and you have used all of the XML tags you're given. Make a PDF of the resulting browser page to turn in.

Part 3: Create New Tags and Their Processing

The goal of this part is to add tags to the XML and write styling templates for the XSL. You will create a title tag, and then create tags for displaying poetry.

The steps for the assignment are as follows:

1. Add a `<title>` tag to some day of your diary.
2. Programming by analogy (that is, notice how the XSL templates are constructed), style the `<title>` tag using `<h2>` tags.
3. Develop the XML tags to display a poem: You need at least three.
4. Style the new poetry tags using XSL and programming by analogy again.

Details below.

Add A <title> Tag

Pick a day in your diary that needs some emphasis and add a `<title>` tag to the XML file, such as `<title>`A Really Terrific Day`</title>` that was used on 27 Feb of Part 2. Obviously, you can use the title more than once.

Style Using XSL

Noticing how the XSL templates work, program by analogy to style the `<title>` tag using HTML `<h2>` tags. This is very detailed because there is no allowance for typos, but all you need is to match the word "title", place the `<h2>`, cause the inside tags to be processed with an `<xsl:apply-templates/>` tag (note that it's self-terminating), and then place the `</h2>`. Try it out!

Develop Poetry Tags

To reasonably display poetry, you need several tags. You need an affinity tag to enclose the whole thing; you need an identity tag for the title (but you have that from above), an identity tag for the author, and an identity tag for a line of poetry. Develop those tags. Here is a sample poem to practice with

[Hint: There are several ways in HTML to shift the author's name right a few spaces, but one very simply way is simply to write
`<pre>` -- `</pre>` using the preformat tags.]

Using your new tags, place a poem of your choice into some entry of your diary. (You cannot use the Celery poem!)

> **Celery**
>
> Celery, raw
> Develops the jaw,
> Celery, stewed,
> Is more quietly chewed.
>
> *-- Ogden Nash*

Style The Poetry Tags

As before, write XSL styling templates for the new poetry tags. These should be straightforward, maybe easier than the `<title>` tag styling.

To Turn In

Create a new folder and place (a) your XML photo file (Part 1), (b) the PDF (Part 2), (c) the entire folder used in Part 3. Zip them together; submit them to the class drop box.

Celebrate … That was your LAST assignment!!! You did it!!!