

CSCI 3020 ASSIGNMENT 6

DUE MONDAY, NOVEMBER 30, 11:59 PM

GOALS

This assignment has you write an XSLT stylesheet.

NOTE: This is the last assignment. Any open Dropbox, including this one, will close, on **Wednesday, Dec 2 11:59PM**. No other submissions will be accepted, so that we can get things graded in time for final grades.

TIPS AND GETTING HELP

- Read these instructions carefully
- Read and follow the submission instructions.
- Post general questions on the discussion board in D2L. Please refrain from posting more than 1 line of code, otherwise your post may be removed.
- Send me email (nicholsonja@apsu.edu) or come to my office. I'll be happy to help. When emailing me send me the entire file.

BACKGROUND

Your security clearances finally came through and you have been hired by the CIA and are now working in the one of the Agency's many IT departments. Although you jumped through a bunch of hoops to get those active security clearances, your first job is completely unclassified. You have been assigned to the office that collects information for and publishes the CIA World Factbook. They want you to add some flexibility to their publishing system.

Your project manager has given you three files

- **world_factbook_2008.xml**
 - This is the XML file your office generates throughout the year.
 - **You should not edit this file.**
 - Maurice Moss, who happens to have the oddest hairstyle in the office, manages the contents of this XML file.
- **fb_config.xsd**
 - This is an XML Schema.
 - **You should not edit this file.**
 - This file's purpose is to help you edit the **fb_config.xml** file correctly. If you validate **fb_config.xml** after editing it, this schema will ensure that your edits are correct.
 - This schema is written and maintained by another developer on the team, Roy Trenneman, who keeps mumbling about turning your computer on and off.

- **fb_config.xml**

- This is the configuration file defined by Roy's schema, **fb_config.xsd**:

```
<?xml version="1.0" encoding="UTF-8"?>
<config
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="fb_config.xsd">

    <order      output="asc"  />
    <background output="true" />
    <location   output="true" />
    <climate    output="true" />
    <population output="true" />

</config>
```

- **You have to edit this file.**
- The only items you should edit are the values of the attribute OUTPUT for each element. No other changes should be made.
- All the elements and attributes are required. The legal values of the OUTPUT attributes are
 - For the ORDER element, the OUTPUT attribute can have the two values
 - **asc** - this is short for ascending, i.e., alphabetical from A-Z
 - **dec** - this is short for descending, i.e., alphabetical from Z-A
 - The OUTPUT values for the BACKGROUND, LOCATION, CLIMATE, and POPULATION elements all have the same possible values
 - **true**
 - **1**
 - **false**
 - **0**

Your boss, Jen Barber, who has a lot of experience with computers, has asked that you to create an XSLT stylesheet. The three files above will be used as part of the system.

The stylesheet will use the **fb_config.xml** file to decide which parts of the **world_factbook_2008.xml** to include in the output. The ORDER attribute will determine whether the names of the countries are from A-Z or from Z-A, and the remaining elements will determine which pieces of data in the **world_factbook_2008.xml** are processed or ignored.

Right now, this is a prototype so there are only 4 elements to output. But if your stylesheet works as Jen hopes, then Roy will eventually extend the language in the **fb_config.xml** file to use all the data in Moss's input file.

INSTRUCTIONS

You will use jEdit and will need to install the XSLT Plugin as described in class/video.

Put all three files in the ZIP in a directory, and then in the same directory create a XSLT stylesheet file named **wfb.xsl**. Near the top of the wfb.xsl add a comment that looks like this:

```
<!--  
    NAME:  
    SECTION:  
    Assignment 6  
-->
```

filling in the information with your info.

You should read the **fb_config.xml** into a variable using the **document()** function.

Using the values in the **fb_config.xml** file, read the **world_factbook_2008.xml** file and output the data according to these guidelines:

1. The output should be HTML. Use this in your main template

```
<html>  
<head>  
    <title>  
        <xsl:text>CIA World Factbook</xsl:text>  
    </title>  
</head>  
<body>  
  
</body>  
</html>
```
2. In the template, before the **<html>** tag, add this XML in the stylesheet.

```
<xsl:text disable-output-escaping='yes'>  
    &lt;!DOCTYPE html> &#xA;  
</xsl:text>
```

This will ensure that the DOCTYPE statement for html5 is added to the output as part of your html. If done correctly your html output will look essentially like this (spacing might be different):

```
<!DOCTYPE html>  
<html>  
  <head>  
    <META http-equiv="Content-Type" content="text/html; charset=UTF-8">  
    <title>CIA World Factbook</title>  
  </head>
```

3. In the body, output an H1 tag. The element should look like this:

```
<h1>CIA World 2008 Factbook</h1>
```

Do not hard-code the year. It should be pulled from the YEAR attribute in the WFB element of the world_factbook_2008.xml

4. The remaining processing of the world_factbook_2008.xml involves extracting information about places. The decisions are based on the values in the **fb_config.xml** file.

- a. Each PLACE's output should be placed in a DIV tag with a border:

```
<div style="border:1px solid black;">
  <!-- data for a single PLACE. -->
</div>
```

- b. The order of the PLACE's can change in the output according to the **fb_config.xml** file.

- When the **fb_config.xml** file's ORDER element's attribute OUTPUT is set to **asc**, then the PLACES should be output alphabetically from A-Z
- When the **fb_config.xml** file's ORDER element's attribute OUTPUT is set to **dec**, then the PLACES should be output in reverse alphabetical order from A-Z

- c. Inside the DIV, the first element should be an H2. The text in the H2 should be the PLACE's NAME. For example,

```
<h2>Antarctica</h2>
```

The PLACE's NAME should always be output.

- d. When the value of the **fb_config.xml** file's BACKGROUND element's attribute OUTPUT is **true** or **1**, then output the text of the **INTRODUCTION** element's **Background** child in a P tag. Use the following as an example:

```
<p>
  <b>Background: </b>Speculation over the existence of a
  "southern land" was not confirmed until the early 1820s when
  ... an Antarctic Treaty was negotiated that neither denies nor
  gives recognition to existing territorial claims; signed in
  1959, it entered into force in 1961.
</p>
```

When the value of the **fb_config.xml** file's BACKGROUND element's attribute OUTPUT is **false** or **0**, do not output anything for the background.

- e. When the value of the **fb_config.xml** file's LOCATION element's attribute OUTPUT is **true** or **1**, then output the text of the **GEOGRAPHY** element's **Location** child in a P tag. Use the following as an example:

```
<p>
  <b>Location: </b>continent mostly south of the Antarctic
  Circle
</p>
```

When the value of the **fb_config.xml** file's LOCATION element's attribute OUTPUT is **false** or **0**, do not output anything for the location.

- f. When the value of the **fb_config.xml** file's CLIMATE element's attribute OUTPUT is **true** or **1**, then output the text of the **GEOGRAPHY** element's **Climate** child in a P tag. Use the following as an example:

```
<p>
    <b>Climate: </b>severe low temperatures vary with latitude,
    elevation, and distance from the ocean
</p>
```

When the value of the **fb_config.xml** file's CLIMATE element's attribute OUTPUT is **false** or **0**, do not output anything for the climate.

- g. When the value of the **fb_config.xml** file's POPULATION element's attribute OUTPUT is **true** or **1**, then output the following
- The text of the **PEOPLE** element's **Population** child in a P tag.
 - After the P tag, make a UL tag, and inside the UL tag add three LI tags that output the following elements:
 1. **PEOPLE/Age_structure/age_0_14_years**
 2. **PEOPLE/Age_structure/age_15_64_years**
 3. **PEOPLE/Age_structure/age_65_years_and_over**

Use the following as an example:

```
<p>
    <b>Population: </b>84,522 (July 2008 est.)
</p>
<ul>
    <li>
        <b>Age 0 to 14: </b>27.2% (male 11,670/female 11,318)
    </li>
    <li>
        <b>Age 15 to 64: </b>66.3% (male 26,138/female 29,859)
    </li>
    <li>
        <b>Age 65 and older: </b>6.6% (male 2,408/female 3,129)
        (2008 est.)
    </li>
</ul>
```

When the value of the **fb_config.xml** file's POPULATION element's attribute OUTPUT is **false** or **0**, do not output anything for the climate.

5. Some additional rules:

- a. If an element is empty, do not output anything for that element even if the matching OUTPUT element in the **fb_config.xml** is true.
- b. Do not output any empty P tags, LI tags, or UL tags. For example, don't generate this:

```
<p></p>
```

If the content isn't there, don't output the tags.
- c. Other than the name, all the other outputs have labels inside B tags. Those labels are required.
- d. You must have at least two **xsl:template** elements in wfb.xml. Adding more will make your code easier to understand.
- e. You don't have to worry about whitespace and line breaks. They will make it easier to debug your code, but you won't be graded on them.

SAMPLE OUTPUT

- Values of the **fb_config.xml** file used to generate **sample_output_1.html**

```
<?xml version="1.0" encoding="UTF-8"?>
<config
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="fb_config.xsd">

  <order      output="asc" />
  <background output="false" />
  <location   output="false" />
  <climate    output="false" />
  <population output="false" />

</config>
```

Since the order's output is **asc**, the names are from A – Z. All the other output values are **false**, so nothing else is output.

- Values of the **fb_config.xml** file used to generate **sample_output_2.html**

```
<?xml version="1.0" encoding="UTF-8"?>
<config
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="fb_config.xsd">

  <order      output="dec" />
  <background output="false" />
  <location   output="false" />
  <climate    output="false" />
  <population output="true" />

</config>
```

Since the order is **dec**, the place names are output from Z – A. Only population is true, so only it is added to the output for places. Take a look at the Falkland Islands and French Guiana to see examples of places that had empty population fields.

- Values of the **fb_config.xml** file used to generate **sample_output_3.html**

```
<?xml version="1.0" encoding="UTF-8"?>
<config
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="fb_config.xsd">

  <order      output="asc" />
  <background output="true" />
  <location   output="true" />
  <climate    output="true" />
  <population output="true" />

</config>
```

Since the order's output is **asc**, the names are from A – Z. All the other output values are true, so all the required data is displayed.

Don't make or use any filenames or directory names with spaces. jEdit will be very unhappy with spaces in names. This include all the directories starting at the root – on windows "C:\\" and on OS X/Linux "/". If ANY directory in the complete path of the file has a space in it, jEdit will not work.

To set up XSLT processing in jEdit, open jEdit's XSLT Processor

- For the Source XML, select **File**, and then find and locate the **world_factbook_2008.xml**
- For the Stylesheet, select **File**, find and locate the **wfb.xsl**
- For the output, select **File**, and then go to the directory where the **world_factbook_2008.xml** and **wfb.xsl** are located. Add the name **output.html**

TIP

When you first start, ignore the **fb_config.xml** file. Concentrate on outputting the correct HTML first. Once you go that, go back and read the **fb_config.xml** file and use it to perform the necessary logic.

SUBMITTING

Upload two files file into the DropBox.

- Your wfb.xsl
- Two examples of output.html that used two different configurations in **fb_config.xml**

Do not upload any other files.

You may resubmit until the dropbox closes. No matter what the dropbox will close Dec 2, 11:59PM