MADISON AREA TECHNICAL COLLEGE 152-121 Advanced Website Development

Project 3

Project Goals

- Know how to use namespaces.
- Understand how to validate XML using DTD.
- Understand how to use attribute lists.
- Know how to use cascading style sheets (CSS) to format an XML document.
- Understand how to create and use extensible style language transformations (XSLT) for transforming an XML document.
- Know how to reference elements for performing transformations using the XML tree.
- Learn how to access and display attributes using XSLT.
- Know how to loop through XML elements.
- Know how to use a series of XML templates to transform an XML document.
- How to use XSLT if and choose statements.
- Understand when and how to use XML expressions.
- Understand output methods.
- Know how to use xsl:text.
- Learn how to transform XML data into a file structure for appending to a database.
- Know how to validate an XML document using a schema.

Reading Assignment

- Tutorial 2, Working with Namespaces
- Tutorial 4, Working with Schemas
- Tutorial 5, Covering cascading style sheets
- Tutorial 6, working with XSLT (*Extensible Style Language Transformations)*
- Tutorial 7, Creating a Computational Style Sheet
- Appendix B, XSLT Elements and Attributes (reference)
- Appendix E, XML Schema Reference

Project Submission

- Zip up the files (**project3.xml, project3.xsd, project3x1.xsl, project3x2.xsl**) in the **project3** folder
- Submit the **project3.zip** file using the Blackboard submission link

REQUIREMENT #1

1) Open your **project2.xml** document and save a copy as **project3.xml**

2) Modify your **project3.xml** to make any corrections indicated by your instructor in your project 2 feedback

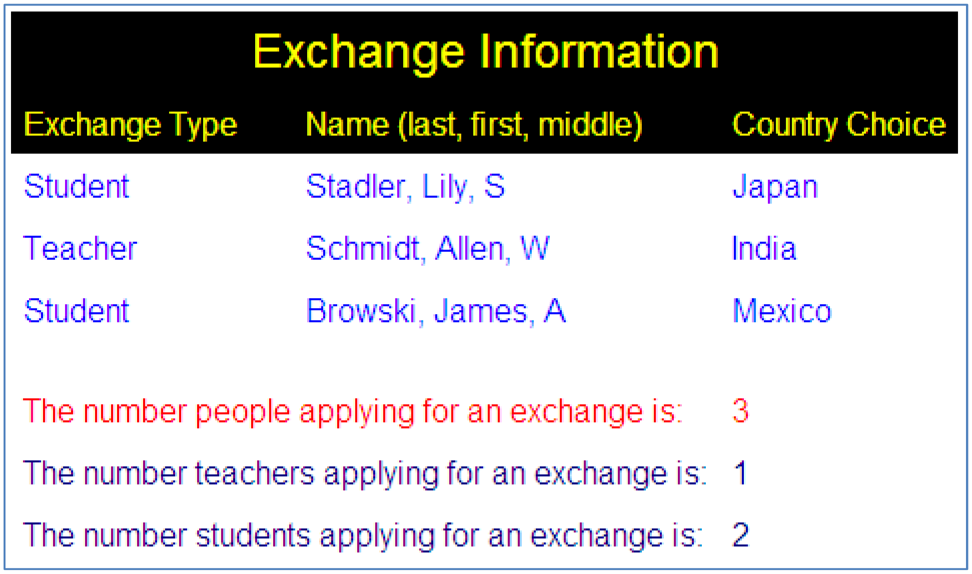
3) Replace any entity references with the actual values.

4) Create a **project3.xsd** schema for the **project3.xml** document by converting the internal DTD
   - a) Use Venetian Blind or Flat Catalog style
   - b) Establish minimum and maximum occurrences
     - i) Ensure there are at least 4 exchange elements
     - ii) Ensure there 2 or more language spoken elements
     - iii) Ensure there are exactly 5 country choice elements
     - iv) Use the minimum and maximum occurrences specified in the internal DTD to determine all other element restrictions
   - c) Include validation for the following:
     - i) Refer to the internal DTD to determine all parent/child element relationships to begin
     - ii) Validate the exchange type attribute can only have a value of: **S** (student) or **T** (teacher)
     - iii) Validate all name, mother name, and father name child elements have alphabetic contents
     - iv) Validate that the name, mother name, and father name each contains last name, first name, and optionally middle initial
     - v) Validate that the address contains street, apartment optionally, city, state or providence, zip or postal code, and country
     - vi) Validate that the school address contains street, apartment optionally, city, state or providence, zip or postal code, and country
     - vii) Validate either the state or province element is received. Contents of either must be alphabetic.
     - viii) Validate either the zip or postal code element is received. Contents of zip must be 5 digits or 5+4 digits. Contents of postal code must not be longer than 12 characters
     - ix) Validate the gender can only have a value of: **M** (male) or **F** (female)
     - x) Validate the Language spoken must have a fluency attribute containing a 1, 2, 3 or 4
     - xi) Validate the country choice must have a preference attribute containing a 1, 2, 3 or 4
     - xii) Validate any date element is given in month, day, year format and is a valid month (1-12), day (1-31), and year
     - xiii) Validate the age has a minimum value of 13
     - xiv) Validate the country is alphabetic
     - xv) Validate the school name is alphabetic
     - xvi) Validate the essay is no greater than 1,000 characters
     - xvii) Validate the phone number has a format of numeric and symbols: (999) 999-9999
     - xviii) Validate the exchange Number is an ID
     - xix) Validate the Fee Paid Currency attribute should be one of a list of enumerated currency types. To keep the project a reasonable size, you do not have to include all possible currency types, but you should include at least five currency types, including the ones used in your XML document
   - d) Remove the internal DTD from the **project3.xml**
   - a) Connect the **project3.xml** document to the newly created **project3.xsd**
   - e) Use Exchanger XML to validate the schema. If you are not using Exchanger XML editor, you must verify the document passes your schema validation by using one of the following validators:

     http://www.xmlvalidation.com/
     http://www.w3schools.com/xml/xml_validator.asp
   - b) Correct any errors and revalidate

REQUIREMENT #2

1) Create the following logic for the **project3x1.xsl** document:



Exchange Information		
Exchange Type	Name (last, first, middle)	Country Choice
Student	Stadler, Lily, S	Japan
Teacher	Schmidt, Allen, W	India
Student	Browski, James, A	Mexico
The number people applying for an exchange is:		3
The number teachers applying for an exchange is:		1
The number students applying for an exchange is:		2

a) Create an XSLT style sheet to format a summary list of exchange people. It should contain the following:
b) The listing should select and display **only** the country that is the *first* choice of the student or teacher.
c) You **must** use a table format. The heading and column headings should be in two rows, with a black background color and yellow font.
d) The top heading line should be: Exchange Information.
e) The second line should be column headings, and should include: Exchange Type, Name (last, first, middle), and Country Choice.
f) The exchange type attribute of an "S" should be converted into the text "Student" and an attribute of "T" should be converted into the text "Teacher". This should be in the first column.
g) The full name should be in the second column, with lastname, firstname and middle initial separated by a comma.
h) The Country should be in the third column
i) Double space after the list of exchange people and add the text (spanning the first two table cells) "The number people applying for an exchange is: " followed by a count of the number of exchange people in the third table cell.
j) Using the table format in point H, add a new row counting the number of teachers, and another row counting the number of students. These rows should be in a different color.
k) Any CSS should be internal to the XSLT (this is due to the project 3 tester)

REQUIREMENT #3

1) Create the following logic for the **project3x2.xsl** document:
   a) Transform the XML document into a text file (not an XML file) that have the correct length and formats specified below
   b) Do not include School Address, Languages Spoken, or Country Choices. These would be included as on different database tables

Element	Length
ID Number (the ID attribute for Exchange)	Remove the initial letter and pad the number to 8 digits
Last Name	20
First Name	15
Middle Initial	1 (Truncate if longer than 1 character, store a space if there is no middle initial)
Street	25
Apartment	12
City	35
State	15
Zip	10
Country	30
Telephone	10 All digits, without any formatting characters (remove all parenthesis)
Email Address	30
Application Date	8 (MMDDYYYY format), month and day should be padded on the left with a zero if they contain one digit
Available Date	8 (MMDDYYYY format), month and day should be padded on the left with a zero if they contain one digit
Date of Birth	8 (MMDDYYYY format), month and day should be padded on the left with a zero if they contain one digit This should contain zeros for teachers
Birth Country	30
Age	2 (This should contain two zeros for teachers)
Gender	1 (M or F - upper case letters)
School Name	35
Year In School	9 (This should contain spaces for teachers)
Emergency Telephone	10 All digits, without any formatting characters, such as parenthesis
Fee Paid	8 digits padded on the left with zeros
Currency	3 character standard currency code. Refer to the list of codes.
Mother Name	30 (This should contain spaces for teachers)
Father Name	30 (This should contain spaces for teachers)

project3x2.xsl extra credit:

- Create a named template that will pad a received field value with spaces up to a maximum indicated length
- Call the named template sending parameters for the field value and the maximum field length

2) Run all requirements using the project 3 tester:
   a) Project 3 Tester http://itins4.matcmadison.edu/IT/152121advweb/phpExample/validateFiles.php
   b) Use the Project 3 Test in Google Chrome, Firefox or IE browser and verify the code you added works when opening the document up within a browser.
   c) For viewing text output you may need to use the developer tools in the browser
      i) **IE9 in the Developer Tools View**: Ensure developer tools is installed. Using either F12 or the ⚙ settings icon, verify that all of the XML elements & attributes are there and contain the correct data.
      ii) **Firefox in the Firebug View**: Ensure Firebug is installed. Right-click and select Inspect Element. Use F12 and select the HTML tab, verify that all of the XML elements & attributes are there and contain the correct data.
      iii) **Google Chrome**: Right-click and select Inspect Element. Use F12 and select the Elements tab, verify that all of the XML elements & attributes are there and contain the correct data.