New Perspectives on XML Comprehensive, 3rd Edition

Additional Project

Tutorial 9

Alfonse Dawkins is a marketing manager at The Zocalo Fire Pit, an online distributor of gourmet meats and other dishes and beverages. He would like to identify the orders that represent the largest purchases. The customers who made these orders will be offered membership in a Customer Appreciation Club. This data is contained in two source XML files: ***zfpxcustomers.xml*** and***zfporders.xml***.

Alfonse has asked for your help in using XQuery to create a FLWOR query structure to identify the customers who made the largest purchases. The query should list the information on the customers who made orders of $200 or more so he can invite them to join the Customer Appreciation Club.

Complete the following:

1. Using your text editor, open the files***zfpxcustomers.xml*** and***zfporders.xml***.Review the contents to familiarize yourself with the structure of the XML files.Note that the order elements in the ***zfporders.xml*** file are related to the data in the ***zfpcustomers.xml*** fileby the custID attribute.
2. Using your text editor, open the ***zfp\_query1txt.xq*** file. Enter your name and the date in the comment section of the file, and save it as ***zfp\_query1.xq***.
3. Add a statement in the query prolog to declare the XQuery version as 1.0.
4. Alfonse's query will draw data from both files. Inside the results element, write code to apply a for clause to each document source to create variables that reference the lists of customers and orders. Name the variable to reference the customer elements in the ***zfpcustomers.xml*** file **$c**, and the variable to reference the order elements in the ***zfporders.xml*** file **$o**. For example, the code to create the $c variable is:

for $c in doc('zfpcustomers.xml')//customer

1. Alfonse wants to invite customers who placed orders totaling $200 or more to join the new Customer Appreciation Club. Write a let clause to declare the $totalvariable containing the sum of each order. (Hint: The $o variable refers to each order element. Each order element can contain multiple products, and each product contains qty and priceatrtributes.)
2. Next, write the where clause of the FLWOR structure. This clause should limit the results to orders where the $total variable is greater than or equal to 200, so only customers who placed orders for at least $200 are sent the offer.  
     
   The where clause should also join the $o and $c variables, allowing data to be drawn from both the ***zfporders.xm***l and ***zfpcustomers.xml*** files. (Hint: Note that in the ***zfporders.xml*** file custID appears as an element and not an attribute.)
3. Next, add an order by clause to the FLWOR structure, to sort the results in descending order by the value of the $totalvariable.
4. Next, add the return clause of the FLWOR structure. The return clause should produce the main content of the resulting XML document that will provide Alfonse with the customer information he needs to send invitations to the Customer Appreciation Club.  
     
   In the return clause, add code to create a members element. Within the members element, first add an orderTotal element that displays the member's order total. (Hint: Use the concat() and round-half-to-even() functions to round the $total value to two decimal places and format it with a dollar sign.)  
     
   Next within the members element, create a member element. The member element should contain the customer information. (Hint: You can use "{$c}" to return the customer information here.)
5. If you are using Saxon in Java command line mode, go to the folder containing your data files, and run the following command to generate the ***clubMembers.xml*** result document:

javanet.sf.saxon.Query !indent=yes -q:zfp\_query1.xq  
 -o:clubMembers.xml

Otherwise, use the commands appropriate to your XSLT 2.0 processor to run the transformation.

1. Use your web browser or text editor to open the ***clubMembers.xml*** file. Verify that the content of the file matches that shown in the figure below.



1. Submit the completed files to your instructor.