CSE 4/560 Project 3: XML and XQuery

Due 23:59 EST 05/13/2020

This is an individual project for writing XQuery. There are 6 problems with 20 points in total. **Please note that academic integrity is strictly implemented and any violation will lead to a F grade in this course.**

1. Problem Statements

You are given the following DTD data format A that describes the information about authors and books. Assuming you have an XML document called books.xml that is valid against the given DTD, write the following queries in your solution files, use XQuery comments to indicate your answers to the questions, e.g., (: answer for 1.1 :). Use eXistDB or BaseX to test and verify your answers, you must use /db/books.xml as the path of the file books.xml. The file books.xml will be posted separately.

DTD data format A:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE biblio[
<!ELEMENT biblio (author*)>
<!ELEMENT author (name,book*)>
<!ELEMENT name (#PCDATA)>
<!ELEMENT book (title, category,rating,price)>
<!ELEMENT title (#PCDATA)>
<!ELEMENT category(#PCDATA)>
<!ELEMENT rating(#PCDATA)>
<!ELEMENT price (#PCDATA)>
<!ATTLIST book year CDATA #REQUIRED>
]>
```

1.1(2 pts) Find the names of all Jeff's co-authors and list them together with the titles of books that were co-authored.

Sample output format:
```
<book>
   <title>Big data analytics</title>
   <name>Jeff</name>
   <name>the other author</name>
</book>
……
```

1.2(2pts) Return all the author pairs who have co-authored two or more books together, list their co-authored books' information.

Sample output format:
```
<coauthor>
  <output>
     <name>author1</name>
     <name>author2</name>
     <book year="the book year">
       <title>the book title</title>
```

```
      <category>the book category</category>
      <rating>the book rating</rating>
      <price>the book price</price>
    </book>
  </output>
  <output>
    <name>author1</name>
    <name>author2</name>
    <book year="the book year">
      <title>the book title</title>
      <category>the book category</category>
      <rating>the book rating</rating>
      <price>the book price</price>
    </book>
  </output>
  ......
</coauthor>
......
```

1.3(2pts) Find the average book price of each category and global. If a category has higher than global average book price, list one most expensive book and its authors, for each of those categories.

```
Sample output format:
<result>
  <categories>
    <output>
          <category>DB</category>
          <title>Database systems</title>
          <price>1000</price>
          <name>author1</name>
          <name>author2</name>
          <name>......</name>
    </output>
  </categories>
  ......
</result>
```

1.4(4pts) Return all the book price and rating with book name and sort the price and rating from high to low separately.

```
Sample output format 1.4a:
<title>Applied Mathematics</title>
<price>100</price>
<title>Introduction to R programming<title/>
<price>200</price>
<title>Introduction to Python<title/>
<price>300</price>
<title>Big data analytics<title/>
<price>400</price>
...
Sample output format 1.4b:
<title>Applied Functional Analysis</title>
<rating>2</rating>
<title>Applied Mathematics</title>
<rating>2.1</rating>
<title>AWS: Security Best Practices on AWS</title>
<rating>2.7</rating>
<title>Introduction to R programming</title>
<rating>3.2</rating>
<title>Big data analytics</title>
<rating>3.5</rating>
...
```

1.5(4pts) The text book requirement in this class is based on 'category': one 'DB', one 'PL', one 'Science', one 'Others'. Return your plan for the book purchasing. The plan should follow some second rules (cheapest (1), best rating (1), assume you have $1800 how to get the best rating books (2))

Sample output format 1.5a:
```
<title>Big data analytics</title>
<price>400</price>
<category>DB</category>
<title>Applied Functional Analysis</title>
<price>400</price>
<category>Others</category>
<title>Introduction to R programming</title>
<price>200</price>
<category>PL</category>
<title>Applied Mathematics</title>
<price>100</price>
<category>Science</category>
```

Sample output format 1.5b:
```
<title>Database systems</title>
<rating>5</rating>
<category>DB</category>
<title>Pattern Recognition</title>
<rating>5</rating>
<category>Others</category>
<title>Introduction to Python</title>
<rating>4.7</rating>
<category>PL</category>
<title>Statistical Inference</title>
<rating>5</rating>
<category>Science</category>
```

1.6(6 pts) Define a DTD for an equivalent DTD data format B which stores the same information as A, but in which the authors are listed under their books. Write an XQuery query whose input is an XML document valid with respect to the DTD A and whose output is another XML document valid with respect to B.

2. Submission

Failure to comply with the submission specifications will incur penalties for EACH violation. The project is an INDIVIDUAL project: copying, collaboration or cooperation will be considered violations of academic integrity.

- What to submit: A zip file has to be submitted through the 'submit_cse460' (if you are CSE460 student) or 'submit_cse560' (if you are CSE560 student) submit script by 05/03/2020 23:59 EST. Only zip extension will be accepted, please **don't** use any other compression methods such as tar or 7zip. You can submit multiple times, note that only the last submission will be kept on the server.

- Zip file naming: Use *ubit_proj3* (**NO SPACE!**) for the filename, for example: *jsmith_proj3.zip*, where *jsmith* is the ubit of submitter. The project is an **INDIVIDUAL** project, so everyone needs to submit ONE zip file.

- Sub-structure of zip file: On unzipping the zip file, there should be a folder named *ubit_proj3*, under the folder *ubit_proj3*, there should be XQuery files, each should be your answers to one problem.

Reference:

https://www.w3schools.com/xml/xquery_intro.asp
http://www.datypic.com/books/xquery/examples.html