XML

INSTRUCTIONS

1. This project must be done and submitted individually.
2. Submit one ZIP file, `<student number>.zip` (for example: "`A012345L.zip`"), containing the following files to the folder "XML Submissions" in Luminus Files by **Friday 29 March at 18:30**.
   - Your report (PDF file).
   - An SQL file `q1.sql` for Question 1.
   - An XML document `q1.xml` with internal DTD for Question 1 and 2.
   - Three XSLT files namely `query-1.xsl`, `query-2.xsl` and `query-3.xsl` for Question 3.
3. Make sure to write your **name** and your **student number** on the front page of the report.
4. After the deadline and until **Friday 5 April at 18.30**, you can submit to the folder "XML Late Submissions" in Luminus Files (penalties apply).
5. Structure your report according to the questions and parts. Use clear headings for the questions and parts, e.g. `Question 1` and `Question 1.b`.
6. Make sure to clearly highlight the answer to the question from other discussions and comments.

In this assignment, we use the Warehouse, Item, and Stock schema from the Introduction slide number 15 (IVLE `Files/Lecture Notes/Introduction/introduction.pdf`). You need to download the required SQL files (`warehouse.sql`, `item.sql`, and `stock.sql`) from the Luminus Files and import it into your PostgreSQL.

Question 1 [5 marks]

Export the data in the database, i.e. the data in the tables `warehouse`, `item` and `stock`, into an XML document. For this you should use PostgreSQL XML functions[1]. You should create an XML document that satisfies the following properties:

- The document's root element is named `warehouses`.
- Each warehouse record is represented by an element `warehouse` that is a sub-element of `warehouses`. There might be no warehouse.
- Each `warehouse` element consists of a sequence of four sub-elements: `id`, `name`, `address`, and `items` corresponding to the five attributes of the warehouse record. The address element consists of three sub-elements `street`, `city` and `country`. The `items` element contains one sub-element `item` for each of the items stocked in the warehouse. There might be item in the warehouse.
- Each `item` element consists of a sequence of five sub-elements: `id`, `im_id`, `name`, `price`, and `qty`. The first four sub-elements correspond to the four attributes of the item record and the `qty` element represents the quantity of that item stocked in the warehouse.

(a) Create an SQL file `q1.sql` containing the queries used to create the XML document with the properties mentioned above. Submit the result of the query as an XML document `q1.xml`. You can use the `COPY ([query]) TO 'q1.xml'` to export the result of a query into a file. Submit both the SQL file `q1.sql` and the XML document `q1.xml`. (3)

(b) Determine whether the XML document is lossless or lossy. If it is lossless, explain why in your report. If it is lossy, explain why and write an SQL query that shows the missing data in your report. (2)

Question 2 [3 marks]

The XML documents of question 1 should be submitted with an internal DTD. Make the DTD as tight as possible (It should validate this document and documents following the same design logic and it should not validate documents not following the same design logic). Validate the XML document with its DTD using `XML copy editor`. This software is available in your Virtual Machine. In your report, provide a screenshot showing that your XML is valid using one of the two software.

Question 3 [12 marks]

Write a separate XSLT stylesheet for each of the following queries. The XSLT stylesheet must work with the `XML copy editor` and Mozilla Firefox. For each query, the corresponding XSLT stylesheet should display the XML document as an HTML page presenting the result of the query in a table or a list.

(a) For each warehouse in Singapore, list the items that are available in the warehouse in quantity larger than 975. Print the name of each warehouse followed by the list of the names of the items it contains and the quantity available. (4)

(b) For each warehouse in Singapore or Malaysia, display the name of the warehouse and the name of the items available in the largest quantity in the warehouse. (4)

(c) Print the total quantity of items called "Sunscreen" available in Indonesia. (4)

XSLT template matching with `value-of` and `apply-template` is preferred to imperative control structures (e.g. `xsl:for-each`).

Submit the three XSLT files, `query-a.xsl`, `query-b.xsl` and `query-c.xsl`, corresponding to the three queries, respectively.

– END OF PAPER –

[1] `https://www.postgresql.org/docs/current/static/functions-xml.html` , you might need to use functions such as: `xmlelement`, `xmlconcat`, `xmlagg`, etc.