Lab Assignment 3 on XML

CS311WP: Dr T Veeraiah

Semi-structure Data Processing: Transforming XML data to CSV format

For Lab3, You can write in your choice of any languages in any platform.

The Semi-Structured Data Formats: XML and JSON are used for Data Exchange betwee web application platforms.
The server side needs to design/create XML Elements to encode the database to send and build XML documents from them.
The Client side needs to parse the XML documents from the Server to extract the meta data information with the values to store them in their database server to retrieve.

In this Lab, Parse XML Document using any Parser that supports DOM (Document Object Model *)* to Extract Schema and Data from XML Documents and Transform them into a Flat Text File Format (CSV/TSV) to Create Tables in a Relational DB

A way to retrieve information from XML 1les is to use an XML parser. An XML parser is, quite simply, software that reads an XML 1le and makes available the data in it. You ca write your own XML parser that supports the XML Document Object Model (DOM) to build a tree structure that represents XML schema and Data.

The DOM tree that is built by the XML parser from a given XML document consists of di0erent Node types and edges (Parent 5/6Child hierarchy) as covered in class. You may choose to use an existing parser (for example, DOM Parser, SAX Parser) that supports the XML Document Object Model (DOM). The DOM de1nes a standard set of commands that parsers should expose so you can access HTML and XML document content from your application programs.

An XML parser that supports the DOM will take the data in an XML document and expose it via a set of objects that you can program against. You can learn how to acces and manipulate XML documents via the XML DOM implementation, as exposed by the Microsoft® XML Parser (Msxml.dll) to download below or any other available XML parser that supports DOM. You can use a SAX Parser with Java as you can 1nd. Learn how to use DOM Parser below and download one of any available then add the library into your programming platform like Visual Studio. Read through for more details. You can use XPATH in your program. See instructions on how to set up XPath in the Lab Section on the class webpage.

http://www.w3.org/TR/2000/REC-xml-20001006
http://www.w3.org/DOM/
http://www.dmoz.org/Computers/Programming/Internet/W3C_DOM/
http://www.dmoz.org/Computers/Programming/Languages/Java/XML/Class_Libraries/Parsers/
http://www.saxproject.org/
http://www.microsoft.com/en-us/download/details.aspx?id=3988
http://msdn.microsoft.com/en-us/library/aa468547.aspx

For sample code examples to extract data from a XML document using DOM,

http://www.w3schools.com/dom/dom_examples.asp
http://www.mkyong.com/java/how-to-read-xml-file-in-java-dom-parser/

http://www.mightywebdeveloper.com/coding/mysql-to-xml-php/

http://www.codediesel.com/php/converting-mysql-queries-to-xml/

Write a program that parses a given XML document in a file (in next page) using any existing XML parser that supports DOM to do the followings:

a) Extract data from the document and
b) Transforms the data into structured text files in CSV/TBS and/or into tables in a SQL database.

For this task,
1) Check the links given above to learn about DOM and an existing XML parser (ex: DOM parse, MSXML parser, or SAX Parser)
1) Download any available XML parser that supports DOM library (ex: DOM/MSXML into VS or DOM/SAX Parser for Java) to set up
2) Write a program that loads the given XML documents in a file (in next page) a input and parses to extract terminal data to write into an output file.
3) Write Stored Procedures to read data from the files and then store them in the tables in database
4) For this task, you need to have a mapping strategy to carry the relationships i XML to write data into the output files then store them into tables

This lab is to learn how to handle a multivalued columns, nested and irregular data in semi-structured data to transform them to a correct relational database scheme, which is a common task in real life applications.

There are many ways to transform XML to Table structures. You are to design a correct database scheme to convert the XML data to table structures.

Two common ways could be:
I. One Way:
1) Create a big dirty table in CSV in your program and
2) Create multiple tables in correct scheme reading from the big table in a Stored Procedure in a SQL Server.
II. Another Way
1) Design a scheme (multiple CSV 𝕝le formats) and create multiple CSV 𝕝les in your program and
2) Create database tables directly from each CSV in a Stored Procedure in a SQL Serve

Design your CSV 𝕝le formats. There is no one strict 𝕝le format as a solution. Think abou what would be a good database scheme (structure) to transform those irregular multi valued data or nested data to a table (or connected multiple tables) so that you can retrieve them from a database easily and e3 ciently without losing information. As long as it is transformed to a correct database scheme without losing data, it would be good for this lab.

Implement and Submit any one method as in the followings:

On Blackboard:

1. Lab Report (in doc 𝕝le) that shows the followings:

1) Your set up/platform procedure,

2) The executions to generate the output 𝕝les in a structured any text 𝕝le format

(in CSV, TSV),

3) Your outputs in text.

4) Screen captures of your SQL tables created from the converted 𝕝les, and

5) Your source codes/scripts.

2. Submit your zip 𝕝le including report 𝕝le in doc, all your codes/scripts, and your output 𝕝les in one zip 𝕝le.

<bibs>