CS 6314, Spring 2017
Dr. Mithun Balakrishna
Homework 4
Due March 26th, 2017 11:59 pm

A. Submission Instructions:

- Submit your solutions via eLearning.
- Please submit a single zip file with the following files:
  - XML Schema file (Question 1)
  - XML file (Question 2.a)
  - HTML file(s) (Questions 2.b & 2.c). **Please create a separate HTML page(s) for this homework**
  - Source code and executable (Question 3)
  - A ReadMe file with instructions on how/where to access (for Questions 1 & 2) and run (for Question 3) the solutions
- Late Submission Penalty:
  - up to 2 hours late — 10% deduction
  - 2 - 4 hours late — 20% deduction
  - 4 - 12 hours late — 35% deduction
  - 12 - 24 hours late — 50% deduction
  - 24 - 48 hours late — 75% deduction
  - more than 48 hours late — 100% deduction (zero credit)

B. Problems:

1. XML Schema (40 points)

Create a XML Schema (i.e. XSD) to describe the structure of an XML document that can contain the following information:

1. Product Information:
   a. Product Name: String datatype where first letter is a capital letter
   b. Product Description: String datatype where first letter is a capital letter
   c. Product Release Date: Date datatype
   d. Product Cost: the value should be in the decimal format with a currency type attribute (i.e. Dollars, Euros, etc.)
   e. Company Information: Reference to the defined "Company Information" XSD structure which is described next.
2. Company Information:
   i. Company Name: String datatype where first letter of each word is a capital letter
   ii. Company Address: A sequence of the following elements:

1. Street Name: String datatype of the following format: "Number Street_Name"
2. City: String datatype where first letter is a capital letter
3. Zip Code: Integer datatype with values from 11111 to 99999
4. Country: String datatype where all letters are capital letters

Each of the above elements/attributes occurs only once unless otherwise specified. Each optional element occurs at most once.

Please use freely available online XML Schema validators such as:

www.utilities-online.info/xsdvalidation

The page is to be created using a plain text editor such as WordPad, NotePad, Emacs, VI, etc. You ***CANNOT*** *use any graphical XML Schema authoring software applications.*

2. XML Document Creation and Display (40 points)

a. **(10 points)** Using the XML Schema defined in Question 1, create a XML document containing information for ten (10) products of your choice. Your XML should contain:
   i. 4 different companies
   ii. 2 companies from the same country
   iii. Atleast 2 products from 2 different companies with price more than 25 Dollars but less than 100 Dollars
   iv. Atleast 4 products from 3 different companies with release date after 2010

   The XML Schema should be used to validate the information populated into the XML document

b. **(15 points)** Display the all information contained in the XML document (created in Question 2.a) on a HTML page (in a tabular format) by traversing the XML DOM object using JavaScript

c. **(15 points)** Create XPath queries to display the following information contained in the XML document (created in Question 2.a) on a HTML page:
   i. Total cost of all the products created by companies from the same country
   ii. Product name, description, and release date of all products that cost more than 25 Dollars but less than 100 Dollars

iii. Name and address of companies that have published atleast one product after 2010

3. XML to JSON (20 points)

Below is an XML file. Your task is to programmatically transform this particular type of XML structure into an equivalent JSON notation. Submit your source code and executable with instructions on how to run the program. Please note that your code will be tested on a XML with same structure but with different values.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<glossary>
  <title>example glossary</title>
  <GlossDiv>
    <title>S</title>
    <GlossList>
      <GlossEntry ID="SGML" SortAs="SGML">
        <GlossTerm>Standard Generalized Markup Language</GlossTerm>
        <Acronym>SGML</Acronym>
        <Abbrev>ISO 8879:1986</Abbrev>
        <GlossDef>
          <para>A meta-markup language, used to create markup
languages.</para>
          <GlossSeeAlso OtherTerm="GML"/>
          <GlossSeeAlso OtherTerm="XML"/>
        </GlossDef>
        <GlossSee OtherTerm="markup"/>
      </GlossEntry>
    </GlossList>
  </GlossDiv>
</glossary>
```