Faculty of Science, Engineering and Technology

Fundamentals of Data Management

Pass Task 2.1: Semi-Structured Data: XML and JSON

Overview

In this tutorial, you'll create and query an XML document and translate it to JSON.

Purpose

Learn how to create and navigate tree-structured data formats. Acquaint yourself with the XML and JSON formats.

Task

Use a text editor (such as Notepad++ to construct your answers to the questions below. Test your XQuery expressions on the web site shown under 'Resources'.

Time

This task should be completed in your second lab class and submitted for feedback in the second lab or at the beginning of lab 3.

Resources

- Online module (from Canvas)
- Online resources, e.g.
    - https://www.w3schools.com/xml/xquery_example.asp (XQuery)
    - http://www.tutorialspoint.com/xquery/index.htm (XQuery)
- Web site for trialling XQuery expressions:
    - http://videlibri.sourceforge.net/cgi-bin/xidelcgi

Feedback

Discuss your solutions with the tutorial instructor.

Next

Get started on module 3.

Pass Task 2.1 — Submission Details and Assessment Criteria

Document your solutions using a word processor and upload as pdf to Doubtfire. The tutors will discuss the solution with you in the lab.

Example 2.1.1 – Structuring Information

Jane is an enthusiastic gardener. In recent years, she has begun to plant trees, and she is recording which trees thrive in which conditions. In 2005, she planted an Ash tree in a sunny, dry spot exposed to wind. It grew very slowly, but did not die. In 2006, she planted a lemon-scented gum in a sunny, dry spot sheltered from wind. In 2007 she planted a Japanese Maple in a wet spot with little sun and lots of exposure to wind. It died soon after.

How can this information be expressed in XML?

Obviously, there can be an unlimited number of trees, so there must be several tree elements in the document:

```
<tree> </tree>
<tree> </tree>
```

The trees are of a certain variety – ash, gum tree, etc. The variety pertains to the tree, so it has to be either an element or an attribute of the tree element:

```
<tree variety=”Ash”> </tree>
```

or

```
<tree>
  <variety>Ash</variety>
</tree>
```

There are environmental conditions such as sun, wind, soil. They are specific to each tree, because the tree has been planted in these conditions, so we have to put them into the tree element (somehow, somewhere).

We could do:

```
<tree>
  <variety>Ash</variety>
  <environment>sunny,dry,exposed</environment>
</tree>
```

This is a baseline approach that would make querying difficult: we have to parse the entire value of the environment element, split it into substrings and find the appropriate value in the right substring.

This might be more useful:

```
<tree>
  <variety>Ash</variety>
  <environment>
    <sun>very sunny</sun>
    <soil>dry</soil>
    <wind>exposed</wind>
  </environment>
</tree>
```

We also have a year the tree was planted in. The year is clearly part of the tree information, but not the environment (unless we have several trees of the same variety that were planted in different years in differen environments, but let’s not complicate this too much). So we would have:

```
<tree>
  <variety>Ash</variety>
  <planted>2006</planted>
  <environment>
    <sun>very sunny</sun>
    <soil>dry</soil>
    <wind>exposed</wind>
  </environment>
</tree>
```

We would also have to record the outcome, whether the tree was growing well.

```
<tree>
  <variety>Ash</variety>
  <planted>2006</planted>
  <environment>
    <sun>very sunny</sun>
    <soil>dry</soil>
    <wind>exposed</wind>
  </environment>
  <outcome>fair</outcome>
</tree>
```

We have captured all the relevant information, and all we have to do is repeat the exercise for the other trees. Then, we have to invent a root tag – a tag that encases all others as the document root: <janes_trees>, for example.

Note: You cannot use apostrophes (‘) or spaces in xml tag names.

Subtask 2.1.1

Express the following information in an XML document:

This is the reading list of James Nguyen. James read “Fifty shades of grey” by E. L. James in May 2016, did not like it very much, then “The grass is singing” by Doris Lessing in June 2016, enjoyed it quite a bit, then read Bill Bryson’s “A short history of nearly everything” in July 2016 and found it very informative. James plans to read “JSON in 24 hours” by Peter Settler later in the year for work, then perhaps Peter Hoeg’s “Miss Smilla’s feeling for snow” for entertainment.

Work by the example above. There is more than one possible outcome. Check question 2.1.2 to ensure you make a document that let you find the required information.

Document the result and upload.