

CO3098/CO7098

Coursework 1

XML Schema, XSLT and XML Parsing

Important Dates:

Handed out: 3-Oct-2017

Deadline: 26-Oct-2017 at 23:59 GMT

The deadline is strict and will not be changed. Please ensure that you submit your work in time.

- This coursework counts as 10% of your final mark.
- Please read guidelines on plagiarism in the study guide and course documentation.
- This coursework requires knowledge about XML Schema, XSLT and DOM/SAX Parsing
- Please put your plagiarism coversheet for this coursework in Box B (or submit a signed coversheet electronically to the blackboard)

Data Description

Consider an XML document (IServiceREST.xml):

```
<?xml version="1.0" encoding="UTF-8"?>
<service id="IServiceREST">
  <inherit>
    <parent>Remote</parent>
  </inherit>
  <package>uk.ac.le.cs.module</package>
  <include>java.rmi.Remote</include>
  <include>java.net.*</include>
  <include>java.io.*</include>
  <abstract_method id="getContent">
    <visibility>public</visibility>
    <parameters>
      <argument type="URL">url</argument>
      <argument type="int">timeout</argument>
    </parameters>
    <throw>
      <exception>MalformedURLException</exception>
      <exception>IOException</exception>
    </throw>
    <return>String</return>
  </abstract_method>
  <abstract_method id="deleteUser">
    <visibility>public</visibility>
    <parameters>
      <argument type="String">username</argument>
    </parameters>
    <return>void</return>
  </abstract_method>
</service>
```

This XML document provides an alternative representation of a **Java interface** below:

```
package uk.ac.le.cs.module;

import java.rmi.Remote;
import java.net.*;
import java.io.*;

interface IServiceREST extends Remote{

    public String getContent(URL url, int timeout)
        throws MalformedURLException, IOException;
    public void deleteUser(String username);
}
```

Tasks:

Task 1: [35 Marks]

Write an XML Schema (IServiceREST.xsd) that allows validation of the provided XML document. Note that the schema must be consistent with these rules in Java:

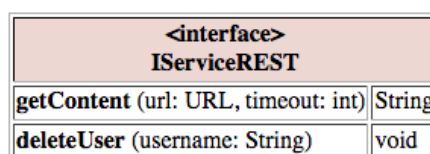
- An interface may extend **one or many** interfaces.
- There can be **only one** package statement
- There can be **zero to many** import statements
- An interface can have **zero to many** abstract methods
- A method only allows **one** value to be returned
- A method can only have **zero or one** visibility modifier *
- A method can accept **zero to many** parameters
- Exception(s) may be thrown from a method

* If a method has no visibility modifier then it is accessible only within its own package (default visibility); assuming all methods are non-static.

Task 2: [30 Marks]

Write an XSLT stylesheet (IServiceREST.xslt) that takes the provided XML document as input and generates an HTML document as follows:

UML class diagram



(**Note:** You may show the result in a different page layout or colour scheme. The page must display the name, return type and the parameter list of each method)

Task 3: [35 Marks]

Write a Java program that reads the XML above and produces an output as follows. Please choose an appropriate parsing technology (DOM/SAX).

```
public String getContent(URL url, int timeout)
    throws MalformedURLException, IOException;
public void deleteUser(String username);
```

(Note: You need to implement either XMLParserDOM.java or XMLParserSAX.java)

Submission

- Zip all files in a single zip file for submission.
 - IServiceREST.xsd
 - IServiceREST.xslt
 - XMLParserDOM.java (or XMLParserSAX.java, depending on your choice)
- The archive should be named **CO3098_CW1_email_id.zip** or **CO7098_CW1_email_id.zip** (e.g. CO3098_CW1_yh37.zip).

Your submission should also include a completed coursework plagiarism coversheet (print and signed PDF or image). You need to submit the zip file via Blackboard and you are allowed to re-submit as many times as you like **before** the deadline. Marks for any coursework which does not have the accompanying cover sheet will be withheld till you provide one.

Note: apart from the XML document above, we might use other XML instance documents as the test suite for assessing task 1, 2 and 3. Your schema and the XPATH used in your XSLT should also work for these XML documents)