

Object Oriented Programming 2

Rolf Haenni & Andres Scheidegger

Exercises 4

1. Bank Account Information in XML

Open a new general (not Java) project in your Eclipse. In this project create a new XML file and enter the information corresponding to a bank account in XML (see Exercise 1-1 on slide 26 of presentation Topic4.1.pdf). Decide for each part of the information which form is better suited to hold the part, an element or an attribute.

2. Well-Formed Document

Copy the XML document from slide 39 of presentation Topic4.1.pdf (Exercise 1-2) to another XML file in your Eclipse. Correct the errors to get a well-formed XML document.

3. XML-Schema for Hotel

Copy the XML document of Exercise Hotel on slide 28 of the presentation Topic4.2.pdf to a new XML file in your Eclipse. Create a new XML-Schema file and write a schema for the validation of the Hotel document.

4. XML-Schema with Type Definitions for Hotel

Copy the XML document of Exercise Hotel 2 on slide 34 of the presentation Topic4.2.pdf to a new XML file in your Eclipse. Create a new XML-Schema file and write a schema for the validation of the Hotel document. This time use type definitions and reference them from the element definitions.

5. XML Schema for Fax

Write an XSD for the following XML document. The fax *header* should contain a *from*, a *to*, a *priority* and a *pages* part (each exactly one) in arbitrary order. All values should be restrictive as possible:

- *priority type* only “URGENT” and “NORMAL” should be permitted,
- attribute *faxno* from elements *from* and *to* only digits and spaces,
- *pages format* only “Letter” and “A4”,
- *checkinfo checksum* only digits and lowercase letters
- *checkinfo txtime* and *txdate* should accept only valid values of the given format.

```
<?xml version="1.0" encoding="UTF-8"?>
<fax>
  <header>
    <from name="MyCompany" faxno="012 345 67 89"/>
    <to name="YourCompany" faxno="098 765 43 21"/>
    <priority type="URGENT"/>
    <pages number="4" format="Letter" color="BW"/>
  </header>
  <body>
    Sehr geehrter Herr Bühler,
    Ich habe bei Ihnen gestern per Fax den Artikel 345 bestellt.
    Ich möchte diese Bestellung stornieren.
    Mit freundlichen Grüßen
    Hans Muster
  </body>
  <checkinfo checksum="2af4" txtime="04:35pm" txdate="2/17/07"/>
</fax>
```

6. XML Document and Schema for Persons Data

In topic 5 we will return to our person data application and use XML documents to persist the data. Therefore, we must define how an XML document containing person data should look like. Write first an example XML document and then derive an XML schema for its validation. You may first choose the Russian doll form for the schema and then refactor it by introducing types.

7. XML and XML Schema at Work (optional)

The swiss association eCH (<http://www.ech.ch>) is “a public-private platform to promote e-government in Switzerland. Its objective is to facilitate cooperation in electronic business transactions with the authorities. Its expert groups draw up national e-government standards.”¹

It provides XML schemas for the interchange of data between public authorities. A catalogue of such standards can be found on

<https://www.ech.ch/vechweb/page?p=categoryList&site=/documents/Alle/No>

As you can see, the standard for interchange of personal identifications is eCH-0044. the related XML schema can be accessed via the link <http://www.ech.ch/xmlns/eCH-0044/4/eCH-0044-4-1.xsd>

In your Eclipse write an valid instance document with your person identification.

¹ <https://www.egovernment.ch/en/organisation/projekt-und-leistungsverantwortliche/verein-ech/>