

XML and Relational Databases (2)

Exercise 1: Node IDs (=Sheet 7, Exercise 2)

1.1. Add NodeIDs to the *flights_data_no_nesting.xml* document, using one of the following types of IDs. **Assign IDs which maintain document order!**

- a) integer IDs
- b) double IDs
 - a. Dewey order

1.2. Write the new Nodes Ids for the document resulted after the execution of the following XUpdate (**only new nodes get new IDs, the other IDs remain unchanged. Document IDs overall should maintain document order.**)

```
let $doc := doc('flights_data_no_nesting.xml')
do update insert <Reservation>
  <date>2006-12-26</date>
  <flightRef>LX183</flightRef>
  <passRef>000111</passRef>
</Reservation>
before $doc/Reservation[1]
```

Exercise 2:

<pre><Passenger> <name>Santa Claus</name> <passportnumber>000111</passportnumber> <address> <city>North Pole</city> <street>Cold Str.</street> <number>01</number> <code>3456</code> </address> </Passenger></pre>	<pre><City> <name>London</name> <country>United Kingdom</country> </City> <City> <name>North Pole</name> <country>Nowhere</country> </City></pre>
--	---

2.1. Map the above XML fragments to an SQL/XML database. Look at variants that store parts of the data in “relational” form.

2.2 Reconstruct the complete XML structure of passenger

2.3. Get the streets of all Passengers as an XML element.

2.4. Determine all Passengers living in ‘North Pole’

2.5. Determine, for each country, **the number of Passengers** living in that country, as XML.

2.6. Determine, for each country, the **LIST** of Passengers and the list of Passengers, as XML.