XML Lab Course WS 2021/21

1. Unit: Querying with XPath and XQuery

For the exercises, it is useful to call saxon directly from the shell instead of using the DBIS XQuery Web-Interface, to get more detailed error messages.

Exercise 1.1 (Mondial - Headquarters of Organizations) Solve the first exercises as far as possible by XPath.

a) Give the names of all countries where some organization has its headquarter.

b) Give the names of all countries where no organization has its headquarter.

c) Give the names of all cities that have more than 1000000 inhabitants and where some organization has its headquarter.

d) Give the names of all cities that have more than 1000000 inhabitants and where no organization has its headquarter.

e) Give the names of all cities where an organization has its headquarter, and which are the capital of a member country of this organization.

Exercise 1.2 (Mondial - Country Radius)
The *radius* of a country is defined as the largest distance between the country's capital and anything (city, lake, mountain etc.) that is known to be located in that country.

a) State an XQuery query that returns for every country its radius; in descending order.
The distance between two pairs `(lat1,long1)` and `(lat2,long2)` can be computed as follows:

```
declare namespace math="http://www.w3.org/2005/xpath-functions/math";
declare function local:distance($lat1,$long1,$lat2,$long2) as xs:double
{ 6370*math:acos(math:cos($lat1 div 180*3.14)*math:cos($lat2 div 180*3.14)*
           math:cos(($long1 - $long2) div 180*3.14)
            + math:sin($lat1 div 180*3.14)*math:sin($lat2 div 180*3.14)) };
local:distance(47.98, 7.85, 51.53, 9.94)
```

b) Sketch how this query must be formulated in SQL against the relational variant of Mondial.

c) For what can the result of this query be useful?

Exercise 1.3 (Mondial - neighbor populations in descending order) Give for each country the sum of the population of its neighbors (in descending order, with those countries with no neighbors coming last).

Exercise 1.4 (Mondial - Lowest Highest Mountain)
Give the lowest mountain which is the highest one on its continent.

Exercise 1.5 (Mondial - Organizations and Continents)
List the names of all organizations with at least one member country on each continent.

Exercise 1.6 (Mondial - Non-Coverable Organizations)
Give the smallest (wrt. number of members) organization O_1 which is not covered by any other organization O_2 (i.e. for all other organizations O_2, O_1 has at least one member which is not a member of O_2).

Exercise 1.7 (Mondial - Country, Neighbor, Organization)

- Compute all pairs (country, organization) s.t. the country belongs to the organization, but all its neighbors do not belong to the organization.

- Compute all pairs (country, organization) s.t. the country does not belong to the organization, but all its neighbors belong to the organization.

Exercise 1.8 (Mondial - Sunrise in Berlin)
Give the names and longitude of all cities where on the 21st of September the sun raises later, but less than 10 minutes later than in Berlin.

Exercise 1.9 (Shared Islands)

a) For all islands that are shared between to or more countries, give the name, the area, and the number of such countries.
b) The database does not contain information, what portion of the island belongs to each of the countries.
   i) Extend the ER-Diagram to model this information (see Mondial Web page for it).
   ii) Consider how this can be realized in the Mondial XML database.
   iii) Extend the DTD and insert the appropriate information into the database. Data can e.g. be found at Wikipedia.
c) To validate your result first, list all countries (ordered descending by area) that are located to more than 90% on islands.
d) List all countries (ordered descending by area) that are located to more than 90% on islands, but where the capital is not located at a coast.

Exercise 1.10 (Web Data Extraction: Wikipedia) Use the *english* Wikipedia HTML pages for the following (the german Wikipedia pages use undeclared Umlaut-Entities, and cannot be XML-parsed):

a) Write an XQuery function that is invoked with the name of a mountain (e.g. "Mont Blanc") that returns a small XML fragment with data about that mountain.
b) Write an XQuery statement that invokes the function for all mountains in Mondial that are located in Germany.

Exercise 1.11 (User-defined Function: Functional Programming – Fibonacci)
This exercise deals with XQuery as a functional programming language. There is nothing about XML in it.

- Write a recursive XQuery function that computes the n-th Fibonacci Number (defined as $fib(n) := fib(n-1) + fib(n-2)$, $\text{fib}(0) := 0$, $\text{fib}(1) := 1$).
- Give the asymptotic complexity of your solution.
- Implement a linear algorithm in XQuery.

Exercise 1.12 (User-defined function: Recursive Network Length)
Write a recursive function that, given a river name, computes the total length of a river system of that river (i.e., its direct and indirect tributaries). Consider that rivers may flow into or through lakes.

Output for every river the total length of its network, in descending order.