

XML-Praktikum WS 2016/17

1. Versuch: Anfragen mit XPath und XQuery

Für die Bearbeitung ist es sinnvoll, saxon direkt per Shell zu benutzen anstelle des DBIS XQuery Web-Interfaces, um aussagekräftigere Fehlermeldungen zu bekommen.

Aufgabe 1.1 (Mondial - Headquarters of Organizations) Lösen Sie die Teilaufgaben soweit möglich mit XPath, den Rest mit XQuery.

- Geben Sie die Namen aller Länder, in denen eine Organisation ihren Sitz hat, an.
- Geben Sie die Namen aller Länder, in denen keine Organisation ihren Sitz hat, an.
- Geben Sie die Namen aller Städte mit mehr als 1000000 Einwohnern, die Sitz einer Organisation sind, an.
- Geben Sie die Namen aller Städte mit mehr als 1000000 Einwohnern, die nicht Sitz einer Organisation sind, an.
- Geben Sie die Namen aller Städte an, die Sitz einer Organisation und gleichzeitig Hauptstadt eines Mitgliedslandes dieser Organisation sind.

Aufgabe 1.2 (Mondial - Country Radius)

Der *Radius* eines Landes wird als die größte Entfernung zwischen seiner Hauptstadt und einem als zu Land gehörend bekannten Punkt definiert.

- Geben Sie für jedes Land seinen Namen und seinen Radius an.

Die Entfernung zweier Koordinatenpaare (b_1, l_1) und (b_2, l_2) kann man wie folgt berechnen:

```
$dist :=  
6370* $\text{math:acos}(\text{math:cos}(\$b_1 \text{ div } 180*3.14)*\text{math:cos}(\$b_2 \text{ div } 180*3.14)*$   
 $\text{math:cos}((\$l_1 - \$l_2) \text{ div } 180*3.14)$   
 $+ \text{math:sin}(\$b_1 \text{ div } 180*3.14)*\text{math:sin}(\$b_2 \text{ div } 180*3.14))$ 
```

- Wie müsste man diese Anfrage in SQL formulieren?
- Wozu kann das Ergebnis dieser Anfrage nützlich sein?

Aufgabe 1.3 (Mondial - neighbor populations in descending order) Geben Sie für jedes Land die Gesamtbevölkerung aller Nachbarstaaten absteigend sortiert an. Länder ohne Nachbarn (die meisten Inselstaaten) sollen am Ende der Sortierung aufgezählt werden.

Überlegen Sie, wie man diese Anfrage in SQL formulieren müsste. Was fällt auf?

Aufgabe 1.4 (Mondial - Lowest Highest Mountain)

Bestimmen Sie den niedrigsten Berg, der der höchste Berg des Kontinents ist, auf dem er sich befindet.

Aufgabe 1.5 (Mondial - Organizations and Continents)

Bestimmen Sie alle Organisationen, die auf jedem Kontinent mindestens ein Mitgliedsland haben.

Aufgabe 1.6 (Mondial - Non-Coverable Organizations)

Geben Sie die (nach Mitgliederzahl) kleinste Organisation O_1 an, die von keiner anderen Organisation O_2 überdeckt wird (d.h. für alle anderen Organisationen O_2 gilt, dass es mindestens ein Mitgliedsland in O_1 dass nicht in O_2 ist).

... weiter auf der Rückseite ...

Aufgabe 1.7 (Web Data Extraction: Germany-View)

Erstellen Sie einen "Deutschland-View": Verwenden Sie die Web-Ressource

<http://www.geohive.com/cntry/germany.aspx>

um XML-Dokument zu erzeugen, das folgender DTD genügen soll:

```
<!ELEMENT country (name,population,provinces,cities)>
<!ELEMENT name (#PCDATA)>
<!ELEMENT population (#PCDATA)>
<!ELEMENT provinces (province*)>
<!ELEMENT cities (city*)>
<!ELEMENT province (name,area,population)>
<!ELEMENT area (#PCDATA)>
<!ELEMENT city (name,population)>

<!ATTLIST country capital IDREF #IMPLIED>
<!ATTLIST province capital IDREF #IMPLIED>
<!ATTLIST city id ID #REQUIRED>
```

Empfehlung: die XML(HTML)-Quelldatei erst auf den eigenen Rechner zu kopieren.

Wenden Sie Ihr Programm auf die Seite für Frankreich an.

Aufgabe 1.8 (Web Data Extraction: Wikipedia)

- Schreiben Sie eine XQuery-Funktion, die mit dem Namen eines Berges aufgerufen wird (z.B. "Mont Blanc"), und ein kleines XML-Fragment mit Daten über diesen Berg ausgibt.
- Schreiben Sie eine XQuery-Statement, das diese Funktion für alle in Mondial enthaltenen Berge, die in Deutschland liegen, aufruft.

Aufgabe 1.9 (User-defined Function: Functional Programming – Fibonacci)

In dieser Aufgabe wird XQuery als funktionale Programmiersprache betrachtet. XML kommt dabei nicht vor.

- Schreiben Sie eine rekursive XQuery-Funktion, die die n -te Fibonacci-Zahl (definiert als $fib(n) := fib(n-1) + fib(n-2)$, $fib(0) := 0$, $fib(1) := 1$) berechnet.
- Geben Sie die asymptotische Komplexität Ihrer Lösung an.
- Implementieren Sie einen linearen Algorithmus in XQuery.

Aufgabe 1.10 (User-defined function: Recursive Network Length)

Schreiben Sie eine rekursive Funktion, die die Gesamtlänge des Flusssystemes eines gegebenen Flusses berechnet.

Beachten Sie, dass Flüsse auch in bzw. durch Seen fließen.