XQuery Tutorial

Student: Anke Bremer
Betreuer: Dr. Sosna

Inhalt

1 Gliederung Tutorial

2 Beispiele für Seitengestaltung
- XML Namespaces
- Pfadausdrücke

3 Ideen / Fragen / Vorschläge

Inhalt Tutorial

Gliederung Tutorial

1. Über das Tutorial
2. Einleitung
    - 2.1. Allgemeines zu XQuery
    - 2.2. XML
        - 2.2.1. Definition, W3C
        - 2.2.2. Aufbau von XML-Dokumenten (Baumstruktur, Elemente, Attribute, Kommentare, Markup)
        - 2.2.3. Namensräume in XML
        - 2.2.4. Validierung
        - 2.2.5. DTD
        - 2.2.6. XML – Schema
        - 2.2.7. Übungen

Gliederung Tutorial

2.3. XPath

2.3.1. Definition, W3C
2.3.2. Terminologie
2.3.3. Verwandschaftsbeziehungen von Knoten
2.3.4. Axen
2.3.5. Adressierung
2.3.6. Vorteile - Nachteile / Vergleich mit XQuery
2.3.7. Übungen

Gliederung Tutorial

3. XQuery

   3.1. Allgemein / Grundlagen
   3.2. Terminologie (atomare Werte, items, Konstruktoren,...)
   3.3. Datentypen
   3.4. Achsen in XQuery / Verwandschaftsbeziehungen

   3.5. Anfragen / Ausdrücke in XQuery

      3.5.1. Allgemein (Struktur, Lokalisierungsschritte, ...)
      3.5.2. Einfache Ausdrücke (Übungen)
      3.5.3. Pfadausdrücke (Übungen)
      3.5.4. Elementkonstruktoren (Übungen)
      3.5.5. Sequenzausdrücke (Übungen)

Gliederung Tutorial

3.5.6. FLWR – Ausdrücke

3.5.6.1. Allgemein

3.5.7.2. Klauseln (Übungen)

3.5.9.2.1. For-Klausel
3.5.9.2.2. Let-Klausel
3.5.9.2.3. Where-Klausel
3.5.9.2.4. Order-by-Klausel
3.5.9.2.5. Return-Klausel

3.5.9.3. Verbundberechnung

3.5.9.3.1. Symmetrischer. Verbund (Ü)
3.5.9.3.2. Einseitger äußerer Verbund (Ü)
3.5.9.3.3. Vollständig äußerer Verbund (Ü)

3.5.9.4. Gruppierung und Aggregation

Gliederung Tutorial

Beispiel 1

XML Namensräume

XML - Namensräume

XML – Namensräume: Dienen dazu Namenskonflikte und Mehrdeutigkeiten in XML-Dokumenten zu vermeiden. Potenziell mehrdeutige Namen werden dabei an einen URI (Uniform Resource Identifier) gebunden.

Namensräume müssen im XML-Dokument deklariert werden, bevor sie verwendet werden.

Der URI eines Namensraums wird einem sogenannten „Namensraumpräfix“ zugewiesen

Ein Element oder Attribut dieses Namensraumes wird dann über das Präfix angesprochen.

Beispiel 1:

```
<mitarbeiter typ="professor" abteilung="DB">
    <name></name>
    <adresse></adresse>
    <telefon></telefon>
    <gehalt></gehalt>
    <rechner>
        <name>Professors Rechner</name>
        <adresse>127.98.76.35</adresse>
    </rechner>
</mitarbeiter>
```

Das XML - Dokument enthält hier neben den **Namen und Adressen der Mitarbeiter** der Universität auch die **Daten (Name, Adresse) des Rechners**, an dem der Mitarbeiter arbeitet

Wenn alle Namen derer, die in der Uni arbeiten ausgegeben werden sollen, werden allerdings auch alle Namen der Rechner ausgegeben, da nach dem Element <name> gesucht wird.

XML - Namensräume

Um diese Namenskonflikte zu vermeiden, werden zwei Namensräume deklariert. (siehe Beispiel 2)

Beispiel 2:

```
<ma:mitarbeiterverzeichnis universität="Universität Leipzig"
        xmlns:ma="http://www.irgendeineAdresse.de/irgendwas"
        xmlns:edv="http://www.irgendeineAndereAdresse.de/irgendwasAnderes">

.......
<ma:mitarbeiter typ="professor" abteilung="DB">
        <ma:name></ma:name>
        <ma:adresse>
            <ma:strasse></ma:strasse>
            <ma:ort></ma:ort>
        </ma:adresse>
        <ma:telefon></ma:telefon>
        <ma:gehalt></ma:gehalt>
        <edv:rechner>
            <edv:name>Professors Rechner</edv:name>
            <edv:adresse>127.98.76.35</edv:adresse>
        </edv:rechner>
    </ma:mitarbeiter>
.....
</ma:mitarbeiterverzeichnis>
```

XML - Namensräume

Der **erste Namensraum** weist alle Elemente, die ihm zugewiesen werden, als **Mitarbeiterdaten** aus.

In dieser Zeile wird der Namensraum für die Mitarbeiterdaten deklariert:

```
xmlns:ma="http://www.irgendeineAdresse.de/irgendwas"
```

Dem Präfix **ma** wird der URI "http://www.irgendeineAdresse.de/irgendwas" zugewiesen.

Der **zweite Namensraum** weist alle Elemente, die ihm zugewiesen werden, als **Rechnerdaten** aus.

In dieser Zeile wird der Namensraum für die Rechnerdaten deklariert:

```
xmlns:edv="http://www.irgendeineAndereAdresse.de/irgendwasAnderes"
```

Dem Präfix **edv** wird der URI "http://www.irgendeineAndereAdresse.de/irgendwasAnderes" zugewiesen.

XML - Namensräume

Über dieses Präfix kann nun bestimmt werden, welcher Name und welche Adresse Mitarbeiterdaten sind.

Durch die Zuweisung der Namensraumpräfixe zu den einzelnen Elementen im XML - Dokument könne diese nun unterschieden werden

```
<ma:mitarbeiterverzeichnis universität="Universität Leipzig"

        xmlns:ma="http://www.irgendeineAdresse.de/irgendwas"
        xmlns:edv="http://www.irgendeineAndereAdresse.de/irgendwasAnderes">

.......
<ma:mitarbeiter typ="professor" abteilung="DB">
        <ma:name></ma:name>
        <ma:adresse>
            <ma:strasse></ma:strasse>
            <ma:ort></ma:ort>
        </ma:adresse>
        <ma:telefon></ma:telefon>
        <ma:gehalt></ma:gehalt>
        <edv:rechner>
            <edv:name>Professors Rechner</edv:name>
            <edv:adresse>127.98.76.35</edv:adresse>
        </edv:rechner>
    </ma:mitarbeiter>
.....
</ma:mitarbeiterverzeichnis>
```

Beispiel 2

Pfadausdrücke

XQuery - Pfadausdrücke

Über XQuery-Pfadausdrücke werden in einem Dokument enthaltene Knoten gesucht (z.B.. Element-, Attribut- und Textknoten)

Beim Angeben eines Pfads können Sie entweder die vollständige oder die abgekürzte Syntax verwenden

Ein **Pfadausdruck** besteht aus ein oder mehreren Schritten.

Ein **Lokalisierungsschritt** (Achsenschritt) eines Pfadausdrucks besteht aus:

1) Achse
2) Knotentest
3) Prädikaten (optional)

Syntax (vollständig): **Achse::Knotentest[Prädikat]**

Das Ergebnis eines Pfadausdrucks erscheint immer in der Dokumentreihenfolge ohne Duplikatknoten in der Ergebnissequenz.

XQuery - Pfadausdrücke

POPUPS

Achse:

- definiert die Bewegungsrichtung
- ein Achsenschritt in einem Pfadausdruck beginnt mit einem Kontextknoten und navigiert (durch die in der Achse angegebenen Richtung) zu den erreichbaren Knoten

Bsp:

xxx

Knotentest:

- ist eine Bedingung und die zweite Komponente eines Achsenschrittes in einem Pfadausdruck
- aus der durch die Achse vorgegebenen Richtung wird eine Menge von Knoten definiert
- aus dieser Menge werden nur die Knoten selektiert, die diese Bedingung erfüllen (der Knotentest liefert „wahr“ zurück)

Bsp:

xxx

XQuery - Pfadausdrücke

Ein **Pfadausdruck** kann **absolut** oder **relativ** sein.

Ein Pfadausdruck heißt dann absolut, wenn ihm ein '/' vorangestellt ist, ansonsten relativ.

Syntax Absoluter Pfadausdruck:

?? **doc(europe.xml) /aaa / bbb**

gibt alle bbb der aaa aus

Syntax Relativer Pfadausdruck:

?? **doc(europe.xml) xxx / yyy**

gibt alle yyy der xxx aus

XQuery - Pfadausdrücke

Beispiel 1: *absoluter Pfadausdruck*

doc(europe.xml) /child::country/child::Geography

Dieser Pfadausdruck enthält zwei durch einen Schrägstrich getrennte Lokalisierungsschritte.

Erster Schritt:

- die, dem Basisverzeichnis untergeordneten, <country> - Knoten werden abgerufen

Zweiter Schritt:

- der <country> - Knoten wird zum Kontextknoten
- die <Geography> - Knoten jedes gefundenen <country> - Knoten werden abgerufen

Ideen / Fragen / Vorschläge

Vorschläge für Funktionen?

Funktionen:

```
fn:abs($arg)
fn:ceiling($arg)
fn:floor($arg)
fn:round($arg)
fn:compare($arg, $arg, (collation)?)
fn:concat($arg, $arg, ...)
fn:substring($arg, start (,length)?)
fn:string-length($arg)
fn:contains($arg1, $arg2)
fn:starts-with($arg1, $arg2)
fn:matches($arg, $regExpr)
fn:replace($arg, $regExpr, $replacement)
fn:empty($seq)
fn:exists($seq)
fn:distinct-values($arg)
fn:day-from-date
```

Ideen / Fragen / Vorschläge

1. Übungsblätter (aktuelle und von den Vorjahren) zu XPath und XQuery

2. LOTS – Übungsblätter verlinken

3. Textform oder eher Stichpunkte?

Vielen Dank für

Ihre Aufmerksamkeit