



Baze podataka 2

XPath i XQuery



XPath i XQuery

- Jezici za kreiranje upita nad XML dokumentima
- Koriste se za dobijanje i manipulisanje podacima iz XML baza podataka.
- Veliki broj ugrađenih funkcija
- Korisnik ima mogućnost definisanja sopstvenih funkcija
- Trenutne verzije (preporučene od W3C): XPath 3.1 i XQuery 3.1



XPath

- XPath koristi iskaze putanja za kretanje kroz logičku, hijerarhijsku strukturu XML dokumenta.
- Dizajniran je da radi sa jednim XML dokumentom. Vrednost vraćena XPath upitom je skup čvorova.
- XML dokumenti se tretiraju kao stabla čvorova sa dokument čvorom kao korenskim čvorom.



XPath

Operacija	Opis
<i>nazivcvora</i>	Dohvata sve čvorove sa nazivom <i>nazivcvora</i>
/	Selektuje čvorove počevši od korenog čvora
//	Selektuje čvorove koji se nalaze u podstablu od trenutne lokacije
.	Selektuje trenutni čvor
..	Selektuje roditelja od trenutnog čvora
@	Selektuje atribut
*	Dohvata čvorove sa bilo kojim nazivom
@*	Dohvata atribut sa bilo kojim nazivom
[i]	Dohvata i-te čvorove potomke
[@nazivatr]	Dohvata čvorove koji imaju atribut sa nazivom <i>nazivatr</i>
[uslov]	Dohvata čvorove koji zadovoljavaju neki uslov
Putanja1 Putanja2	Selektuje sve čvorove ko zadovoljavaju prvu putanju i sve čvorove koji zadovoljavaju drugu putanju



XPath

Primeri:

- `/bookstore`
- `/bookstore[1]`
- `/bookstore/book[position()<3]`
- `/bookstore/book[price>35.00]`
- `//title[@lang='en']`
- `//book/title | //book/price`



XPath Axes

- Axis definiše skup čvorova koji su u nekom odnosu sa trenutnim čvorom.

Axis	Opis
<i>ancestor</i>	Dohvata sve pretke trenutnog čvora
<i>ancestor-or-self</i>	Dohvata sve pretke trenutnog čvora i trenutni čvor
<i>descendant</i>	Dohvata sve potomke trenutnog čvora
<i>descendant-or-self</i>	Dohvata sve potomke trenutnog čvora i trenutni čvor
<i>child</i>	Dohvata sve direktne potomke trenutnog čvora
<i>parent</i>	Dohvata trenutni čvora
<i>self</i>	Dohvata čvorove sa bilo kojim nazivom
<i>following-sibling</i>	Dohvata sve bratske čvorove koji se u dokumentu nalaze posle njega
<i>preceding-sibling</i>	Dohvata sve bratske čvorove koji se u dokumentu nalaze pre njega
<i>attribute</i>	Dohvata sve attribute trenutnog čvora



XPath Axes

Primeri:

- `/bookstore/child::book`
- `/bookstore/book/attribute::*`
- `/bookstore/child::node()`
- `//price[text()<30]/parent::*`
- `//following-sibling::book[1]`



XQuery

- XQuery je nastao sa idejom da obezbedi upitni jezik koji ima istu širinu funkcionalnosti kao SQL nad relacionim bazama podataka. Jezici za kreiranje upita and XML dakumentima
- Izrazi u XQuery-u upadaju u 6 širokih tipova:
 - Izrazi putanje.
 - Konstruktori elemenata.
 - FLWOR izrazi.
 - Uslovni izrazi.
 - Kvantifikovani izrazi.
 - Izrazi koji u sebi uključuju korisnički definisane funkcije.



XQuery – izrazi putanje

- XQuery obezbeđuje izraze putanja koje su nadskup od onih u XPath-u.
- Funkcija `doc(nazivfajla)` se koristi za dohvaćanje fajla sa nazivom *nazivfajla*
- Funkcija `db:open(nazivbaze)` se koristi za dohvaćanje baze podataka

Primer:

```
doc("bookstore.xml")/bookstore/book[price<30]
```



XQuery – konstruktori elemenata

- Ponekad je neophodno za upit da kreira ili generiše elemente. Takvi elementi se mogu generisati direktno u upitu u okviru iskaza nazvanog konstruktori elemenata.

Primer:

```
<book bookid = '{$id}'>  
  {$ime},{$autor}  
</book>
```



XQuery – FLWOR iskazi

- FLWOR se izgovara kao "flower".
- Ovaj izraz je upit koji se sastoji od FOR, LET, WHERE, ORDER BY, RETURN i GROUP BY klauze.

Primer 1:

```
for $x in doc("bookstore.xml")/bookstore/book
where $x/price>30
order by $x/title
return $x/title
```



XQuery – FLWOR iskazi

Primer 2 (group by):

```
for $i in doc("order.xml")//item
```

```
let $d := $i/@dept
```

```
group by $d
```

```
order by $d
```

```
return <department name="{ $d }"
```

```
totQuantity="{sum($i/@quantity)}"/>
```



XQuery – FLWOR iskazi

Primer 3 (inner join):

```
for $item in doc("order.xml")//item,  
    $prod in doc("catalog.xml")//product  
where $item/@num = $prod/number  
return <item num="{ $item/@num} " name="{ $prod/name}"/>
```

ili

```
for $item in doc("order.xml")//item,  
    $prod in doc("catalog.xml")//product[number= $item/@num]  
return <item num="{ $item/@num} " name="{ $prod/name}"/>
```



XQuery – FLWOR iskazi

Primer 4 (outer join):

```
for $prodnum in doc("catalog.xml")//product/number,  
    $price allowing empty in doc("prices.xml")  
    //prices/priceList/prod[@num = $prodnum]  
return <product number="{ $prodnum}" price="{ $price/price}"/>
```

ili

```
for $prod in doc("catalog.xml")//product  
return <product number="{ $prod/number}" price=" {  
    for $price in doc("prices.xml")//prod[@num=$prod/number]  
    return $price/price }"></product>
```



XQuery – uslovni izrazi

- Uslovni izrazi ocenjuju test izraze i onda vraćaju jedan od dva rezultujuća izraza. Ako je vrednost test izraza tačno onda se vraća kao rezultat vrednost prvog rezultujućeg izraza, u suprotnom, vraća se vrednost drugog.

Primer:

```
for $x in doc("bookstore.xml")/bookstore/book
return if ($x/@category="CHILDREN")
    then <child>{data($x/title)}</child>
    else <adult>{data($x/title)}</adult>
```



XQuery – kvantifikovani izrazi

- SOME klauza i EVERY klauza - ekvivalentne kvantifikatorima koji se koriste u matematici i logici.

Primer:

```
for $x in doc("bookstore.xml") /bookstore/book
where some $p in $x/author
    satisfies (contains($p, "Per"))
return $x/title
```




XQuery – Ugradjene funkcije

- Ugradjene funkcije:

<http://www.xqueryfunctions.com/>

- Neke od njih su:

- fn:substring
- fn:matches
- fn:string-length
- fn:ceiling
- fn:floor
- fn:empty
- fn:number



XQuery – korisnički definisane funkcije

- Osim toga što je podržana centralna biblioteka funkcija sličnih onima u XPath-u, XQuery takođe daje mogućnost korisnicima da definišu funkcije koje će proširiti ovu biblioteku.
- Sintaksa:

```
declare function prefix:function_name($parameter as datatype)
  as returnDatatype
{
  ...function code here...
};
```



XQuery – korisnički definisane funkcije

Primer:

```
declare function local:minPrice($price as xs:decimal?, $discount as  
  xs:decimal?)
```

```
{
```

```
  let $disc := ($price * $ discount) div 100
```

```
  return ($ price - $disc)
```

```
};
```

```
<minPrice>
```

```
  {local:minPrice(200,10)}
```

```
</minPrice>
```



XQuery – konstruisanje elementa

Primeri:

- `<book>{data($x/title)}</book>`
- `<book>{ $x/title/text()}</book>`
- `element {'book'} { $x/title/text() }`
- `element {'book'} {concat($x/title/@lang,": ",$x/title/text())}`
- `element {'book'} {$x/title/@lang||" : "||$x/title/text() }`
- `<book>{attribute lang {$x/title/@lang}} {$x/title/text()}</book>`
- `<book>{attribute lang {$x/title/@lang}, $x/title/text()}</book>`
- `element {'book'} {attribute {'lang'} {$x/title/@lang}, $x/title/text() }`



Alat BaseX

- <https://basex.org/>