Задание за решения на задачи по Упражнение - Модул 4: XPath и XQuery

Изготвил: Марина Дабова, ФН: 62503, група 3, специалност СИ

Коментарите са обозначени с червен цвят

#### **Задача 1: Създайте файлове със следните XSL и XML документи и разгледайте XML документа с Internet Explorer или Mozilla Firefox. За атрибута select на елемента xsl:value-of заменете функцията text() с '.' и вижте новия резултат в Internet Explorer.**

XML документ:

<?xml version="1.0" encoding="ISO-8859-1"?>

<?xml-stylesheet type="text/xsl" href="catalogDisplay.xsl"?>

<catalog>

<cd category="Techno" id="8c0a600b">

<artist>TATU</artist>

<title>200 po vstrechnoy</title>

<year>2002</year>

<tracklist num="1">

<track length="4:10">Zachem ya? (Tell Me Why)</track>

<track length="3:34">Ya soshla s uma (Mad as Hell)</track>

<track length="4:40">Nas ne dogonyat (On The Run)</track>

<track length="4:39">Doschitay do sta (Countdown)</track>

<track length="3:21">30 minut (Another Minute)</track>

<track length="4:19">Ya tvoy vrag (Wake Up, Time To Die)</track>

<track length="4:20">Ya tvoya ne pervaya (First In First Out)</track>

<track length="3:55">Robot (Robotronik)</track>

<track length="3:20">Malchik-gay (Zoophilic Lolita)</track>

<track length="3:53">Nas ne dogonyat (HarDrum rmx)</track>

<track length="4:04">30 minut (HarDrum rmx)</track>

</tracklist>

</cd>

<cd category="Techno" id="cd100010">

<artist>Tatu</artist>

<title>Zvezdnaya seriya 2001</title>

<year>2001</year>

<tracklist num="2">

<track length="3:31">Ya soshla s uma</track>

<track length="3:53">Robot</track>

<track length="4:08">Zachem ya?</track>

<track length="4:36">Doschitaj do sta</track>

<track length="4:17">Ya tvoj vrag</track>

<track length="4:38">Nas ne dogonyat</track>

<track length="3:18">Mal'chik-gej</track>

<track length="3:18">30 minut</track>

<track length="4:18">Ya ne tvoya pervaya</track>

<track length="3:50">Nas ne dogonyat (hardrum remix)</track>

<track length="4:03">30 minut (hardrum remix)</track>

<track length="4:03">Ya soshla s uma (dj ram remix)</track>

<track length="4:10">Ya soshla s uma (hardrum remix)</track>

<track length="3:37">Ya soshla s uma (dj ram breakbeat remix)</track>

<track length="7:00">Ya soshla s uma (S.Galoyan remix)</track>

<track length="5:36">Ya soshla s uma (Video)</track>

</tracklist>

</cd>

</catalog>

XSL документ - catalogDisplay.xsl:

<?xml version="1.0" encoding="ISO-8859-1"?>

<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">

<xsl:template match="/">

<html>

<body>

<h2>My CD Collection</h2>

<table border="1">

<tr bgcolor="#9acd32">

<th align="left">Title</th>

<th align="left">Artist</th>

<th align="left">Year</th>

<th align="left">Category</th>

<th align="left">ID</th>

</tr>

<xsl:for-each select="catalog/cd">

<tr>

<td><xsl:value-of select="title"/></td>

<td><xsl:value-of select="artist"/></td>

<td><xsl:value-of select="year"/></td>

<td><xsl:value-of select="./@category"/></td>

<td><xsl:value-of select="./@id"/></td>

</tr>

<tr>

<td>

<xsl:for-each select="tracklist/track">

<li><xsl:value-of select='.'/></li> <!-- заменяме "text()" с "."-->

</xsl:for-each>

</td>

</tr>

</xsl:for-each>

</table>

</body>

</html>

</xsl:template>

</xsl:stylesheet>

След като земеним "text()" с "." и го сложим документа в Internet Explorer, може да видим, че резултата е един и същ.

#### **Задача 2: За XML документа от задача 1 напишете XPATH изрази, които извеждат:**

**1. Стойностите на всички track елементи, чиито атрибут length е равен на '4:04' и са включени в елемент cd, имащ id равно на 8c0a600b**

catalog/cd[@id='8c0a600b']/tracklist/track[@length='4:04']

може да запишем и като:

//cd[@id='8c0a600b']//track[@length='4:04']

Първо си записваме пътя(представила съм го по два начина в зависимост от описването на пътя- дали целия път ще изпишем или не) . След това намираме всички със елемент cd, имащ атрибут id=8c0a600b и съответно в тази йерархия търсим атрибута track с атрибут length='4:04'.

Получаваме като резултат: 30 minut (HarDrum rmx)

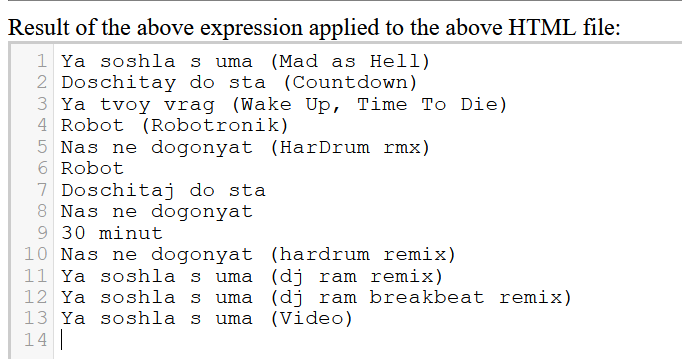
**2. Всички track елементи на четни/нечетни позиции**

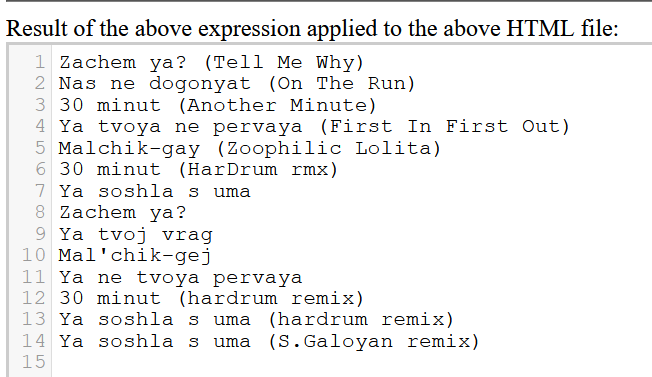
catalog/cd/tracklist/track[position() mod 2 = 0]

catalog/cd/tracklist/track[position() mod 2 = 1]

Тук използваме position(), за да намерих позицията като съответно чрез mod 2 ще си намерим четните и нечетните. Ако position() mod 2 е равно на 0 то ще ни изкара четните, а ако е равно на 1, ще изкара нечетните.

Резултата:

 <-за четни

<-за нечетни

**3.Стойностите на всички track елементи, чиято дължина на текста е по-голяма от 35**

catalog/cd/tracklist/track[string-length(text()) > 35]

Търсим си всички стойности на всички елементи track ,които имат дължина на текста по-голяма от 35. Използваме string-length() и text() , за да проверим дължината на текста.

Резултат:

Ya tvoya ne pervaya (First In First Out)

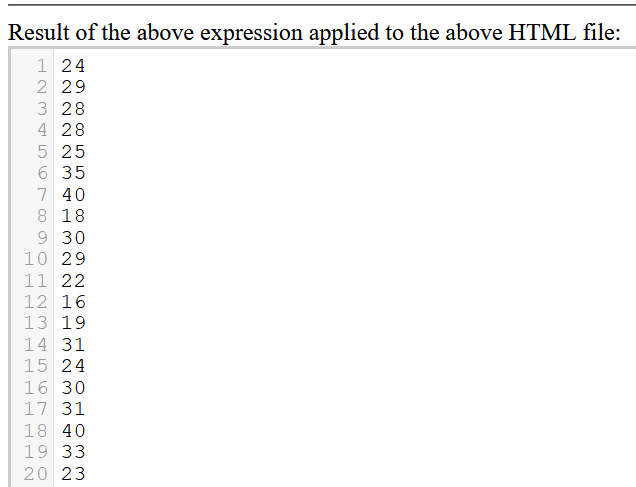
Ya soshla s uma (dj ram breakbeat remix)

**4.Дължината на текста на всички track елементи, чиято дължина на текста е по-голяма от 15**

catalog/cd/tracklist/track[string-length(text()) > 15]/string-length()

Първо си намираме за всеки елемент track с дължина на текста повече от 15(string-length(text()) > 15) и за всеки от тях изкарване дължината на текста (чрез string-length()).

Резултат:



**5. Последния track елемент от всяко cd**

catalog/cd/tracklist/track[last()]

Използваме last() ,за да намерим последния track елемент от всяко cd

Резултат:

30 minut (HarDrum rmx)

Ya soshla s uma (Video)

**6.Петия track елемент от всяко cd**

catalog/cd/tracklist/track[position()=5]

Използваме position()=5,за да намерим петата позиция на track елемента от всяко cd

Резултат:

30 minut (Another Minute)

Ya tvoj vrag

**7.Броя на track елементите за всяко cd**

catalog/cd/tracklist/count(track)

Използваме count(),за да намерим броя на track за всяко cd

Резултат:

11

16

**8.Всички track елементи, които съдържат 'Ya soshla s uma'**

catalog/cd/tracklist/track[matches(text(),'Ya soshla s uma')]

Използваме синтаксиса matches(text(), 'regex'),за да намерим всички track елементи съдържащи 'Ya soshla s uma' .

Резултат:

Ya soshla s uma (Mad as Hell)

Ya soshla s uma

Ya soshla s uma (dj ram remix)

Ya soshla s uma (hardrum remix)

Ya soshla s uma (dj ram breakbeat remix)

Ya soshla s uma (S.Galoyan remix)

Ya soshla s uma (Video)

**9. Всички track елементи, които започват с буквата 'D'**

catalog/cd/tracklist/track[starts-with(text(), 'D')]

Използваме синтаксиса starts-with(text(), string),за да намерим всички track елементи,започващи с 'D'.

Резултат:

Doschitay do sta (Countdown)

Doschitaj do sta

**10. Всички track елементи, които завършват с израза 'sta'**

catalog/cd/tracklist/track[ends-with(text(), 'sta')]

Използваме синтаксиса ends-with(text(), string),за да намерим всички track елементи, завършващи с израза 'sta'.

Резултат:

Doschitaj do sta

**11. Стойностите на всички track елементи, разпечатани с главни букви**

catalog/cd/tracklist/track[upper-case(text())]

Използваме синтаксиса upper-case(text()).

Резултат:

Zachem ya? (Tell Me Why)

Ya soshla s uma (Mad as Hell)

Nas ne dogonyat (On The Run)

Doschitay do sta (Countdown)

30 minut (Another Minute)

Ya tvoy vrag (Wake Up, Time To Die)

Ya tvoya ne pervaya (First In First Out)

Robot (Robotronik)

Malchik-gay (Zoophilic Lolita)

Nas ne dogonyat (HarDrum rmx)

30 minut (HarDrum rmx)

Ya soshla s uma

Robot

Zachem ya?

Doschitaj do sta

Ya tvoj vrag

Nas ne dogonyat

Mal'chik-gej

30 minut

Ya ne tvoya pervaya

Nas ne dogonyat (hardrum remix)

30 minut (hardrum remix)

Ya soshla s uma (dj ram remix)

Ya soshla s uma (hardrum remix)

Ya soshla s uma (dj ram breakbeat remix)

Ya soshla s uma (S.Galoyan remix)

Ya soshla s uma (Video)

**12. Стойността на елемента year, който е под-елемент на елемента cd, съдържащ под-под-елемент track с length = '3:55' и имащ стойност 'Robot (Robotronik)'**

catalog/cd[tracklist/track[@length='3:55' and text()='Robot (Robotronik)']]/year

Тъй като елемента year е под-елемент на cd, трябва да започнем от него търсенето и съответно в него търсим елемента track с length = '3:55' и имащ стойност 'Robot (Robotronik)'

Резултат:

2002

**13. Среден брой track елементи от всички налични cd елементи**

count(catalog/cd/tracklist/track) div catalog/count(cd)

Тук първо намираме броя на елементите track (27)и след това трябва да разделим на броя на всички налични cd елементи(2)

Резултат:

13.5

**14. За всеки елемент cd изведете стойността на под-елементите му title и year, спазвайки следния модел: Заглавие: title\_value; Година на издаване: year\_value**

catalog/cd/concat('Заглавие:', title/text(), '; Година на издаване:', year/text())

Трябва да използваме concat, за да спазим дадения модел(: Заглавие: title\_value; Година на издаване: year\_value).

Резултат:

Заглавие:200 po vstrechnoy; Година на издаване:2002

Заглавие:Zvezdnaya seriya 2001; Година на издаване:2001

#### **Задача 3: За XML документа от задача 1, като използвате оси (axes), съставете XPATH изрази, които :**

**1. Избират елемента:tracklist и year и извеждат негова стойност или стойност на негов атрибут**

catalog/cd/child::tracklist/attribute::num

Чрез child::tracklist избираме елемента tracklist,който е под-елемент на cd, а той на catalog.След това трябва да изведем стойността на неговия атрибут „num“ и това правим чрез attribute::num

Резултат:

1

2

catalog/cd/child::year

Чрез child::year избираме елемента year,който е под-елемент на cd, а той на catalog.Като той има само стойност, няма атрибути, затова извеждаме нея.

Резултат:

2002

2001

**2. Избират атрибутите num и id съответно на елементите tracklist и cd**

catalog/cd/tracklist/attribute::num

Използваме attribute::num след като сме посочили пътя до tracklist.

Резултат:

1

2

catalog/cd/attribute::id

Използваме attribute::id след като сме посочили пътя до cd.

Резултат:

8c0a600b

cd100010

**3. Избират всички елементи track, които се намират преди track елемента със стойност 'Doschitay do sta (Countdown)'**

catalog/cd/tracklist/track[text()='Doschitay do sta (Countdown)']/preceding-sibling::\*

Задаваме пътя до track и търсим всички елементи track, които да са преди този със стойност 'Doschitay do sta (Countdown)'(-това го задаваме чрез text()='Doschitay do sta (Countdown)' ) като чрез preceding-sibling::\* избираме всички всички предни елементи

Резултат:

Zachem ya? (Tell Me Why)

Ya soshla s uma (Mad as Hell)

Nas ne dogonyat (On The Run)

**4. Избират всички елементи track, които се намират след track елемента със стойност 'Doschitay do sta (Countdown)'**

catalog/cd/tracklist/track[text()='Doschitay do sta (Countdown)']/following-sibling::\*

По аналогичен начин правим и тази задача като тук използваме following-sibling::\*, за да изберем всички следващи елементи track след този елемент track със стойност дадената.

Резултат:

30 minut (Another Minute)

Ya tvoy vrag (Wake Up, Time To Die)

Ya tvoya ne pervaya (First In First Out)

Robot (Robotronik)

Malchik-gay (Zoophilic Lolita)

Nas ne dogonyat (HarDrum rmx)

30 minut (HarDrum rmx)

**5. Избират всички елементи track, които се намират след track елемента със стойност 'Doschitay do sta (Countdown)' и имат стойност на атрибута length '4:04'**

catalog/cd/tracklist/track[text()='Doschitay do sta (Countdown)']/following-sibling::track[@length='4:04']

Тук е аналогично като в горната задача като не ни трябват всички елементи а самио тези със стойност '4:04' на атрибута length

Резултат:

30 minut (HarDrum rmx)

**6. Всички стойности на под-елементите на всички елементи cd в документа**

catalog/cd/descendant::\*

Първо си стигаме до елемента cd и след това използваме descendant::\* ,което оказва че взимаме всички потомци на cd

**7. Стойностите на всички елементи, които имат атрибут с име id**

catalog/cd/attribute::id

може да се запише и

//cd/attribute::id

а също и

//attribute::id

Резултат:

8c0a600b

cd100010

**8. Стойностите на всички елементи, които имат какъвто и да било атрибут**

//attribute::\*

\*-ката оказва всички

**9. Атрибута num с максимална стойност**

//tracklist[@num=max(//\*/@num)]/@num

num е атрибут на tracklist, затова трябва да стигнем до него. В случая съм показала краткия път. След това трябва да го изберем като трябва да намерим максималната му стойност затова използваме max() като в него посочваме отново пътя до num.\* оказва всичките атрибути num(защото в cd има два tracklist-а с различни атрибути num и различни стойности и трябва да вземем най-голямата от тях)

Резултат:

2

#### **Задача 4: За XML документа от задача 1, съставете XQUERY израз:**

**1. Селектиращ всички стойности на елемента track, който е под-елемент на tracklist, имащ атрибут num равен на 1.  
   А. Подредете резултата от 1. по азбучен ред  
   Б. Подредете резултата от 1. в обратен азбучен ред**

**Решение:**

for $o in catalog/cd/tracklist

where $o/@num = "1"

return $o/track/text()

A:

for $o in catalog/cd/tracklist

where $o/@num = "1"

order by $o ascending

return $o/track/text()

B:

for $o in catalog/cd/tracklist

where $o/@num = "1"

order by $o descending

return $o/track/text()

Използваме FLWR и в случая клаузите FOR, WHERE и RETURN както и ORDER BY клаузата, за да ги подредим съответно по азбучен ред или не.

**2. Създаващ следната структура:  
<records>  
   <record cd\_ID="CD\_ID\_VALUE" artist="ARTIST\_VALUE">  
       <info>Title: TITLE\_VАLUE, Year: YEAR\_VALUE, Track numbers: COUNT\_OF\_TRACKS</info>  
   </record>  
<records>  
в която стойностите на CD\_ID\_VALUE, ARTIST\_VALUE, TITLE\_VАLUE, YEAR\_VALUE, COUNT\_OF\_TRACKS отговарят съответно на стойностите на атрибута id на елемента cd, на елемента artist, на елемента title, на елемента year, на броя на елементите track за съответния елемент cd**

**Решение:**

element records

{

for $o in catalog/cd

return

element record {

attribute cd\_ID {$o/@id}, attribute artist {$o/artist},

element info {

text { "Title:"}, text {$o/title}, text { ", Year:"} ,text{$o/year}, text { ", Track numbers:"}, text{$o/tracklist/track}

}

}

}

Използвайки упътването решавам задачата като за стойностите на атрибута id на елемента cd записваме в { } $o/@id. За останалите стойностти по аналогичен начин спрямо елементите. По специален е за на броя на елементите track за съответния елемент cd за когото записваме {$o/tracklist/track}

**3. Създаващ списък със стойността на всички track елементи от всички cd елементи, следващ модела:  
<tracks>  
   <track> TRACK\_NAME\_1</track>  
   <track> TRACK\_NAME\_2</track>  
   <track> .........................................</track>  
   <track> TRACK\_NAME\_N</track>  
<tracks**>

**Решение:**

for $o in .

return

element tracks

{

for $m in $o/catalog/cd/tracklist/track/text()

return element track {$m}

}

Използваме FLWR. Връща текста от дадения път- името на всеки track

#### **Задача 5: Решете задача 4 като дефинирате една XQUERY функция и след това я използвате**

**Решение:**

declare function local:toc($catalog as element()) as element()\*

{

for $m in $catalog/cd/tracklist

where $m/@num = "1"

return $m/track/text()

};

{

for $s in doc(.)/catalog

return local:toc($s)

}

Използваме FLWR и по даденото упътване решаваме задачата:

declare function local:toc($book-or-section as element()) as element()\*  
{  
   for $section in $book-or-section/section  
   return  
      <section>  
       { $section/@\* , $section/title , local:toc($section) }  
      </section>  
};  
<toc>  
{  
   for $s in doc("book.xml")/book return local:toc($s)  
}  
</toc>