

**Бази даних та інформаційні системи**

**ЛАБОРАТОРНА РОБОТА №8**

**XQuery - мова запитів XML-документів**

Виконав:

Ст. Яцуляк Андрій

Група ПМі-31

Оцінка

Перевірила:

доц. Малець Р.Б.

## Тема: Вивчення XQuery - мови запитів XML-документів

**Мета роботи:** Ознайомлення з синтаксисом мови XQuery та конструюванням запитів даних XML документа.

### Завдання

Розробити базу даних для системи автоматизації шкільної бібліотеки. Система веде облік читачів, які реєструються в бібліотеці і можуть позичати книги. Для читачів зберігається прізвище, ім'я, контактні дані (адреси та телефони) та рейтинг (наскільки вчасно повертають позичені книги). Крім того база даних містить інформацію про книги, які зберігаються в бібліотеці, їх статус (доступна, кому позичена, не видається, в ремонті і т.д.) Бібліотекар може додавати нових читачів, книги і їх деталі, а також здійснювати пошук книг за авторами, видавництвами, назвами, статусом і т.д.

### Хід роботи

1. Опрацював теоретичний матеріал.
2. XML-документ, створений в ЛР-6:

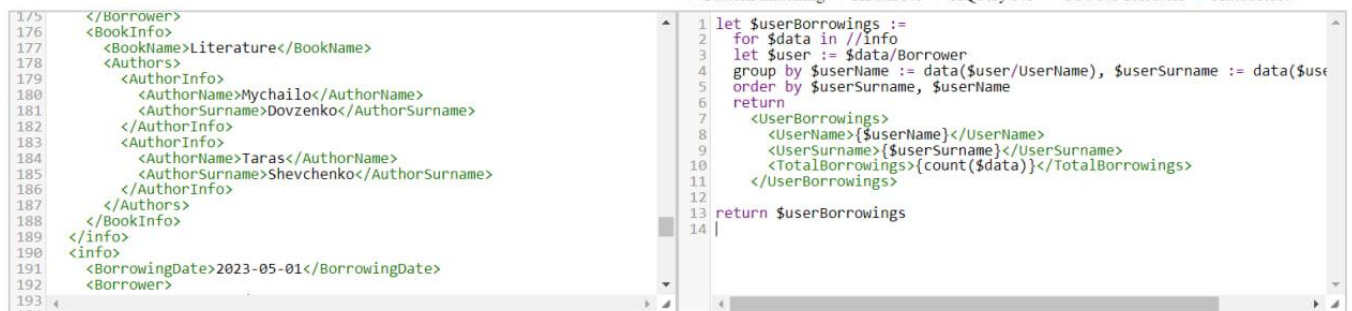
```
<Data>
  <info>
    <BorrowingDate>2022-04-01</BorrowingDate>
    <Borrower>
      <UserName>Pavlo</UserName>
      <UserSurname>Pavlenko</UserSurname>
    </Borrower>
    <BookInfo>
      <BookName>Newera</BookName>
      <Authors>
        <AuthorInfo>
          <AuthorName>Dmytro</AuthorName>
          <AuthorSurname>Vyshnevetskii</AuthorSurname>
        </AuthorInfo>
      </Authors>
    </BookInfo>
  </info>
  <info>
    <BorrowingDate>2022-11-03</BorrowingDate>
    <Borrower>
      <UserName>Andriy</UserName>
      <UserSurname>Yatsuliak</UserSurname>
    </Borrower>
    <BookInfo>
      <BookName>Mathematics</BookName>
      <Authors>
        <AuthorInfo>
          <AuthorName>Petro</AuthorName>
          <AuthorSurname>Yarmola</AuthorSurname>
        </AuthorInfo>
      </Authors>
    </BookInfo>
  </info>
  <info>
    <BorrowingDate>2022-11-04</BorrowingDate>
```

3. У відповідності до свого XML документа, написати декілька запитів XQuery:

1) Для кожного користувача виводить його ім'я і прізвище, а також кількість позичених книг:

```
let $userBorrowings :=
  for $data in //info
  let $user := $data/Borrower
  group by $userName := data($user/UserName), $userSurname := data($user/UserSurname)
  order by $userSurname, $userName
  return
    <UserBorrowings>
      <UserName>{$userName}</UserName>
      <UserSurname>{$userSurname}</UserSurname>
      <TotalBorrowings>{count($data)}</TotalBorrowings>
    </UserBorrowings>

return $userBorrowings
```



Надіслати ☐ disable auto refresh ☐ disable syntax highlighting

Output Options: Node format:  Output format:  ☐ Show types ☐ Hide variable names

Compatibility:

Old languages: ☐ XPath 2.0 ☐ XPath 3.0 ☐ XQuery 1.0 ☐ XQuery 3.0

New languages: ☐ XPath 4.0 ☐ XQuery 4.0

Result of the above expression applied to the above input file:



1	Anatoliy
2	Buyalskii
3	1
4	
5	
6	
7	Vasyl
8	Korol
9	1
10	
11	Vasyl
12	...

2) Виводиться користувач, який позичив або книгу "Biology", або книгу "Mathematics":

```
for $data in //info
where $data/BookInfo/BookName = "Biology" or $data/BookInfo/BookName = "Mathematics"
return
  concat(
    $data/Borrower/UserName,
    " ",
    $data/Borrower/UserSurname,
    " -- ",
    $data/BookInfo/BookName,
    " "
  )
```

```

175 </Borrower>
176 <BookInfo>
177 <BookName>Literature</BookName>
178 <Authors>
179 <AuthorInfo>
180 <AuthorName>Mychailo</AuthorName>
181 <AuthorSurname>Dovzenko</AuthorSurname>
182 </AuthorInfo>
183 <AuthorInfo>
184 <AuthorName>Taras</AuthorName>
185 <AuthorSurname>Shevchenko</AuthorSurname>
186 </AuthorInfo>
187 </Authors>
188 </BookInfo>
189 </info>
190 <info>
191 <BorrowingDate>2023-05-01</BorrowingDate>
192 <Borrower>
193

```

```

1 for $data in //info
2 where $data/BookInfo/BookName = "Biology" or $data/BookInfo/BookName =
3 return
4 concat(
5   $data/Borrower/UserName,
6   " ",
7   $data/Borrower/UserSurname,
8   " -- ",
9   $data/BookInfo/BookName,
10  )
11

```

☐ Надіслати ☐ disable auto refresh ☐ disable syntax highlighting

**Output Options:** Node format:  Output format:  ☐ Show types ☐ Hide variable names

**Compatibility:**

**Old languages:** ☐ XPath 2.0 ☐ XPath 3.0 ☐ XQuery 1.0 ☐ XQuery 3.0

**New languages:** ☐ XPath 4.0 ☐ XQuery 4.0

Result of the above expression applied to the above input file:
 

```

1 Andriy Yatsuliak -- 'Mathematics'
2 Vasyl Korol -- 'Biology'
3 Andriy Melnyk -- 'Mathematics'
4

```

3) Для заданої книги за назвою, визначити, чи вона позичена, чи ні:

```

let $bookName := "Kobzar"
return
  if (some $data in //info satisfies $data/BookInfo/BookName = $bookName)
  then "The book is borrowed"
  else "The book isn't borrowed"

```

```

1 <Data>
2 <info>
3 <BorrowingDate>2022-04-01</BorrowingDate>
4 <Borrower>
5 <UserName>Pavlo</UserName>
6 <UserSurname>Pavlenko</UserSurname>
7 </Borrower>
8 <BookInfo>
9 <BookName>Newera</BookName>
10 <Authors>
11 <AuthorInfo>
12 <AuthorName>Dmytro</AuthorName>
13 <AuthorSurname>Vyshnevetski</AuthorSurname>
14 </AuthorInfo>
15 </Authors>
16 </BookInfo>
17 </info>
18 <info>
19

```

```

1 let $bookName := "Kobzar"
2 return
3 if (some $data in //info satisfies $data/BookInfo/BookName = $bookName)
4 then "The book is borrowed"
5 else "The book isn't borrowed"
6

```

☐ Надіслати ☐ disable auto refresh ☐ disable syntax highlighting

**Output Options:** Node format:  Output format:  ☐ Show types ☐ Hide variable names

**Compatibility:**

**Old languages:** ☐ XPath 2.0 ☐ XPath 3.0 ☐ XQuery 1.0 ☐ XQuery 3.0

**New languages:** ☐ XPath 4.0 ☐ XQuery 4.0

Result of the above expression applied to the above input file:
 

```

1 The book is borrowed

```

4) Вивести посортованих користувачів, які позичили книги вже у 2023 році:

```

for $data in //info
where xs:date($data/BorrowingDate) > xs:date("2023-01-01")
let $user := $data/Borrower
order by $user/UserSurname, $user/UserName
return concat($user/UserSurname, " ", $user/UserName)

```

<pre> 1 &lt;Data&gt; 2 &lt;info&gt; 3 &lt;BorrowingDate&gt;2022-04-01&lt;/BorrowingDate&gt; 4 &lt;Borrower&gt; 5 &lt;UserName&gt;Pavlo&lt;/UserName&gt; 6 &lt;UserSurname&gt;Pavlenko&lt;/UserSurname&gt; 7 &lt;/Borrower&gt; 8 &lt;BookInfo&gt; 9 &lt;BookName&gt;Newera&lt;/BookName&gt; 10 &lt;Authors&gt; 11 &lt;AuthorInfo&gt; 12 &lt;AuthorName&gt;Dmytro&lt;/AuthorName&gt; 13 &lt;AuthorSurname&gt;Vyshnevetskii&lt;/AuthorSurname&gt; 14 &lt;/AuthorInfo&gt; 15 &lt;/Authors&gt; 16 &lt;/BookInfo&gt; 17 &lt;/info&gt; 18 &lt;/Data&gt; 19 </pre>	<pre> 1 for \$data in //info 2 where xs:date(\$data/BorrowingDate) &gt; xs:date("2023-01-01") 3 let \$user := \$data/Borrower 4 order by \$user/UserSurname, \$user/UserName 5 return concat(\$user/UserSurname, " ", \$user/UserName) 6 </pre>
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Надіслати ☐ disable auto refresh ☐ disable syntax highlighting

Output Options: Node format:  Output format:  ☐ Show types ☐ Hide variable names

Compatibility:

Old languages: ☐ XPath 2.0 ☐ XPath 3.0 ☐ XQuery 1.0 ☐ XQuery 3.0

New languages: ☐ XPath 4.0 ☐ XQuery 4.0

Result of the above expression applied to the above input file:

```

1 Buyalskii Anatolii
2 Korol Vasyl
3 Kropyvnytskiy Vasyl
4 Kropyvnytskiy Vasyl
5 Melnyk Andriy
6 Miller Sophie
7 Pavlenko Pavlo
8 Shabanov Mykola
9 Skrypka Petro

```

**Висновок:** під час виконання лабораторно роботи я ознайомився з синтаксисом мови XSLT та перетворенням фрагмента XML документа з використанням XSLT.