

XPath

Графеева Н.Г.

2017

Как выбирать данные из загруженных XML-документов?

- Существуют специальные ***стандартизированные языки***, ориентированные на выборку таких данных:
- XPath – язык для построения навигационных выражений
- XQuery – язык для построения запросов (использует XPath)

XPath

- XPath - это синтаксис для определения фрагментов XML-документа
- XPath использует специальные выражения для навигации по XML-документу
- XPath содержит библиотеку стандартных функций
- XPath рекомендован к использованию консорциумом W3C

Примечание: World Wide Web Consortium (W3C) международное сообщество, которое развивает открытые стандарты для обеспечения долгосрочного роста Интернета.

XPath - терминология

- Nodes
- Atomic values
- Parent
- Children
- Siblings
- Ancestors
- Descendants

Пример XML-документа

- `<?xml version="1.0" encoding="UTF-8"?>`

```
<bookstore>
```

```
  <book>
```

```
    <title lang="en">Harry Potter</title>
```

```
    <author>J K. Rowling</author>
```

```
    <year>2005</year>
```

```
    <price>29.99</price>
```

```
  </book>
```

-

```
  ...
```

```
</bookstore>
```

Nodes

<bookstore> (root element node)

<author>J K. Rowling</author> (element node)

lang="en" (attribute node)

Atomic values

- J K. Rowling

en

- 2005

Parent

- **book** -> title
- **book** -> author
- **book** -> year
- **book** -> price

Children

- **title** -> book
- **author** -> book
- **year** -> book
- **price** -> book

Siblings

- title
- author
- year
- price

Ancestors

- title -> **book**
- title -> **bookstore**

Descendants

- bookstore -> **book**
- bookstore -> **title**
- bookstore -> **author**
- bookstore -> **year**
- bookstore -> **price**

Выборка узлов

Expression	Description
<i>nodename</i>	Selects all nodes with the name " <i>nodename</i> "
/	Selects from the root node
//	Selects nodes in the document from the current node that match the selection no matter where they are
.	Selects the current node
..	Selects the parent of the current node
@	Selects attributes

Примеры (выборка узлов)

Path Expression	Result
bookstore	Selects all nodes with the name "bookstore"
/bookstore	Selects the root element bookstore Note: If the path starts with a slash (/) it always represents an absolute path to an element!
bookstore/book	Selects all book elements that are children of bookstore
//book	Selects all book elements no matter where they are in the document
bookstore//book	Selects all book elements that are descendant of the bookstore element, no matter where they are under the bookstore element
//@lang	Selects all attributes that are named lang

Примеры (предикаты)

Path Expression	Result
/bookstore/book[1]	Selects the first book element that is the child of the bookstore element. Note: In IE 5,6,7,8,9 first node is [0], but according to W3C, it is [1]. To solve this problem in IE, set the SelectionLanguage to XPath: <i>In JavaScript: xml.setProperty("SelectionLanguage","XPath");</i>
/bookstore/book[last()]	Selects the last book element that is the child of the bookstore element
/bookstore/book[last()-1]	Selects the last but one book element that is the child of the bookstore element
/bookstore/book[position()<3]	Selects the first two book elements that are children of the bookstore element
//title[@lang]	Selects all the title elements that have an attribute named lang
//title[@lang='eng']	Selects all the title elements that have an attribute named lang with a value of 'eng'
/bookstore/book[price>35.00]	Selects all the book elements of the bookstore element that have a price element with a value greater than 35.00
/bookstore/book[price>35.00]/title	Selects all the title elements of the book elements of the bookstore element that have a price element with a value greater than 35.00

Шаблоны

Wildcard	Description
*	Matches any element node
@*	Matches any attribute node
node()	Matches any node of any kind

Примеры (использование шаблонов)

Path Expression	Result
<code>/bookstore/*</code>	Selects all the child nodes of the bookstore element
<code>//*</code>	Selects all elements in the document
<code>//title[@*]</code>	Selects all title elements which have any attribute

XPath Operators

Operator	Description	Example
	Computes two node-sets	//book //cd
+	Addition	6 + 4
-	Subtraction	6 - 4
*	Multiplication	6 * 4
div	Division	8 div 4
=	Equal	price=9.80
!=	Not equal	price!=9.80
<	Less than	price<9.80
<=	Less than or equal to	price<=9.80
>	Greater than	price>9.80
>=	Greater than or equal to	price>=9.80
or	or	price=9.80 or price=9.70
and	and	price>9.00 and price<9.90
mod	Modulus (division remainder)	5 mod 2

Примеры (выражения XPath)

- `/bookstore/book/title`
- `/bookstore/book[1]/title`
- `/bookstore/book/price[text()]`
- `/bookstore/book[price>35]/price`
- `/bookstore/book[price>35]/title`

Как использовать XPath в СУБД?

- В СУБД существуют специальные функции, ориентированные на выборку данных, которые используют выражения XPath.
- К примеру, функция XMLTable в ORACLE:
 - XMLTable('<XPath-expression> | <XQuery>'
 - PASSING <xml column>
 - COLUMNS <new column name> <column type> PATH '<XPath-expression>'
 - ...
 -)

Пример(XPath + ORACLE)

- Выборка всех имен и фамилий из xml-документа Employees
:
- SELECT t.id, x.*
- FROM employees t,
- XMLTABLE ('/Employees/Employee'
- PASSING t.data
- COLUMNS firstname VARCHAR2(30) PATH 'firstname',
- lastname VARCHAR2(30) PATH 'lastname') x
- WHERE t.id = 1;



Rows



Save

Run

```
SELECT t.id, x.*
   FROM employees t,
        XMLTABLE ('/Employees/Employee'
                  PASSING t.data
                  COLUMNS firstname VARCHAR2(30) PATH 'firstname',
                          lastname VARCHAR2(30) PATH 'lastname') x
  WHERE t.id = 1;
```

[Results](#) [Explain](#) [Describe](#) [Saved SQL](#) [History](#)

ID	FIRSTNAME	LASTNAME
1	John	Watson
1	Sherlock	Homes
1	Jim	Moriarty
1	Mycroft	Holmes

4 rows returned in 0.09 seconds

[Download](#)

Пример(XPath + ORACLE)

- Выборка имен всех сотрудников:
 - SELECT t.id, x.*
 - FROM employees t,
 - XMLTABLE ('/Employees/Employee/firstname'
 - PASSING t.data
 - COLUMNS firstname VARCHAR2 (30) PATH 'text()') x
 - WHERE t.id = 1;

Rows 10

Save

Run

```
SELECT t.id, x.*
  FROM employees t,
       XMLTABLE ('/Employees/Employee/firstname'
                PASSING t.data
                COLUMNS firstname VARCHAR2 (30) PATH 'text()') x
 WHERE t.id = 1;
```

Results Explain Describe Saved SQL History

ID	FIRSTNAME
1	John
1	Sherlock
1	Jim
1	Mycroft

4 rows returned in 0.07 seconds

[Download](#)

Пример(XPath + ORACLE)

- Имена и категории сотрудников:
- `SELECT emp.id, x.*`
- `FROM employees emp,`
- `XMLTABLE ('/Employees/Employee'`
- `PASSING emp.data`
- `COLUMNS firstname VARCHAR2(30) PATH 'firstname',`
- `type VARCHAR2(30) PATH '@type') x;`

SQL Commands - Windows Internet Explorer

http://apex.oracle.com/pls/apex/f?p=4500:1003:11175038592891::NO::

Файл Правка Вид Избранное Сервис Справка

Избранное SQL Commands Google

Oracle Application Express Workspace: GRAFEYEVA (Logout)

Home Application Builder SQL Workshop Team Development Administration

SQL Workshop SQL Commands Schema GRAFEYEVA

Rows 10 Save Run

```
SELECT emp.id, x.*
FROM employees emp,
XMLTABLE ('/Employees/Employee'
PASSING emp.data
COLUMNS firstname VARCHAR2(30) PATH 'firstname',
type VARCHAR2(30) PATH '@type') x;
```

Results Explain Describe Saved SQL History

ID	FIRSTNAME	TYPE
1	John	admin
1	Sherlock	admin
1	Jim	user
1	Mycroft	user

4 rows returned in 0.02 seconds [Download](#)

Application Express 4.2.4.00.08

Готово Интернет 100%

Пример

- Фамилия и имя сотрудника с номером 2222:
- ```
SELECT t.id, x.*
```
- ```
FROM employees t,
```
- ```
XMLTABLE ('/Employees/Employee[@emplid=2222]'
```
- ```
PASSING t.data
```
- ```
COLUMNS firstname VARCHAR2(30) PATH 'firstname',
```
- ```
lastname VARCHAR2(30) PATH 'lastname') x
```
- ```
WHERE t.id = 1;
```

Rows 10

Save

Run

```
SELECT t.id, x.*
FROM employees t,
XMLTABLE ('/Employees/Employee[@emplid=2222]'
PASSING t.data
COLUMNS firstname VARCHAR2(30) PATH 'firstname',
lastname VARCHAR2(30) PATH 'lastname') x
WHERE t.id = 1;
```

Results Explain Describe Saved SQL History

| ID | FIRSTNAME | LASTNAME |
|----|-----------|----------|
| 1  | Sherlock  | Homes    |

1 rows returned in 0.12 seconds

[Download](#)

# Пример

- Фамилии и имена администраторов:
  - SELECT t.id, x.\*
  - FROM employees t,
  - XMLTABLE ('/Employees/Employee[@type="admin"]'
  - PASSING t.data
  - COLUMNS firstname VARCHAR2(30) PATH 'firstname',
  - lastname VARCHAR2(30) PATH 'lastname') x
  - WHERE t.id = 1;

Rows 10

Save

Run

```
SELECT t.id, x.*
 FROM employees t,
 XMLTABLE ('/Employees/Employee[@type="admin"]'
 PASSING t.data
 COLUMNS firstname VARCHAR2(30) PATH 'firstname',
 lastname VARCHAR2(30) PATH 'lastname') x
 WHERE t.id = 1;
```

Results Explain Describe Saved SQL History

| ID | FIRSTNAME | LASTNAME |
|----|-----------|----------|
| 1  | John      | Watson   |
| 1  | Sherlock  | Homes    |

2 rows returned in 0.10 seconds

[Download](#)

# Пример

- Фамилии и имена сотрудников старше 40 лет:
- ```
SELECT x.*
```
- ```
FROM employees t,
```
- ```
XMLTABLE ('/Employees/Employee[age>40]'
```
- ```
PASSING t.data
```
- ```
COLUMNS firstname VARCHAR2(30) PATH 'firstname',
```
- ```
lastname VARCHAR2(30) PATH 'lastname',
```
- ```
age VARCHAR2(30) PATH 'age') x
```
- ```
WHERE t.id = 1;
```

Rows 10

Save

Run

```
SELECT x.*
 FROM employees t,
 XMLTABLE ('/Employees/Employee[age>40]'
 PASSING t.data
 COLUMNS firstname VARCHAR2(30) PATH 'firstname',
 lastname VARCHAR2(30) PATH 'lastname',
 age VARCHAR2(30) PATH 'age') x
 WHERE t.id = 1;
```

Results Explain Describe Saved SQL History

| FIRSTNAME | LASTNAME | AGE |
|-----------|----------|-----|
| Jim       | Moriarty | 52  |
| Mycroft   | Holmes   | 41  |

2 rows returned in 0.09 seconds

[Download](#)



# Полезные ссылки

- <http://www.w3schools.com/xml>
- <http://www.w3schools.com/xpath>