

XQUERY - FIRST APPLICATION

http://www.tutorialspoint.com/xquery/xquery_first_application.htm

Copyright © tutorialspoint.com

Example

Following is a sample XML document containing the records of a bookstore of various books.

books.xml

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

  <book category="JAVA">
    <title lang="en">Learn Java in 24 Hours</title>
    <author>Robert</author>
    <year>2005</year>
    <price>30.00</price>
  </book>

  <book category="DOTNET">
    <title lang="en">Learn .Net in 24 hours</title>
    <author>Peter</author>
    <year>2011</year>
    <price>70.50</price>
  </book>

  <book category="XML">
    <title lang="en">Learn XQuery in 24 hours</title>
    <author>Robert</author>
    <author>Peter</author>
    <year>2013</year>
    <price>50.00</price>
  </book>

  <book category="XML">
    <title lang="en">Learn XPath in 24 hours</title>
    <author>Jay Ban</author>
    <year>2010</year>
    <price>16.50</price>
  </book>

</books>
```

Following is a sample Xquery document containing the query expression to be executed on the above XML document. The purpose is to get the title elements of those XML nodes where the price is greater than 30.

books.xqy

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

Result

```
<title lang="en">Learn .Net in 24 hours</title>
<title lang="en">Learn XQuery in 24 hours</title>
```

Verify Result

To verify the result, replace the contents of *books.xqy* (given in the [Environment Setup](#) chapter) with the above XQuery expression and execute the XQueryTester java program.

XQuery Expressions

Let us understand each piece of the above XQuery expression.

Use of functions

```
doc("books.xml")
```

doc is one of the XQuery functions that is used to locate the XML source. Here we've passed "books.xml". Considering the relative path, books.xml should lie in the same path where books.xqy is present.

Use of XPath expressions

```
doc("books.xml")/books/book
```

XQuery uses XPath expressions heavily to locate the required portion of XML on which search is to be made. Here we've chosen all the book nodes available under books node.

Iterate the objects

```
for $x in doc("books.xml")/books/book
```

XQuery treats xml data as objects. In the above example, \$x represents the selected node, while the for loop iterates over the collection of nodes.

Apply the condition

```
where $x/price>30
```

As \$x represents the selected node, "/" is used to get the value of the required element; "where" clause is used to put a condition on the search results.

Return the result

```
return $x/title
```

As \$x represents the selected node, "/" is used to get the value of the required element, price; "return" clause is used to return the elements from the search results.

Loading [Mathjax]/jax/output/HTML-CSS/fonts/TeX/fontdata.js