

XQUERY - HTML FORMAT

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

Copyright © tutorialspoint.com

XQuery can also be easily used to transform an XML document into an HTML page. Take a look at the following example to understand how XQuery does it.

Example

We will use the same books.xml file. The following example uses XQuery extract data from books.xml and create an HTML table containing the titles of all the books along with their respective prices.

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>
```

Given below is the Xquery expression that is to be executed on the above XML document.

books.xqy

```
let $books := (doc("books.xml")/books/book)
return <table><tr><th>Title</th><th>Price</th></tr>
{
  for $x in $books
  order by $x/price
  return <tr><td>{data($x/title)}</td><td>{data($x/price)}</td></tr>
}
</table>
</results>
```

Result

```

<table>
  <tr>
    <th>Title</th>
    <th>Price</th>
  </tr>
  <tr>
    <td>Learn XPath in 24 hours</td>
    <td>16.50</td>
  </tr>
  <tr>
    <td>Learn Java in 24 Hours</td>
    <td>30.00</td>
  </tr>
  <tr>
    <td>Learn XQuery in 24 hours</td>
    <td>50.00</td>
  </tr>
  <tr>
    <td>Learn .Net in 24 hours</td>
    <td>70.50</td>
  </tr>
</table>

```

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

Here we've used the following XQuery expressions –

- data function to evaluate the value of the title element, and
- { } operator to tell the XQuery processor to consider data as a function. If { } operator is not used, then data will be treated as normal text.

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