

ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO

LEI/LSIRC/LSIG

PEI/EOI

1º Semestre ■ Docente: Bruno Oliveira/Mariana Carvalho
Ficha Prática 4

Documentação complementar:

- XPATH
- Introdução e exemplos XPATH
- XQuery 1.0/2.0 Funções e Operadores
- XQuery 3.0 Funções e Operadores
- Exemplos

PARTE I

- 1. Considere o documento XML com o nome: cd_catalog, disponível na plataforma moodle. Escreva expressões XPath que permitam obter:
 - Todos os cds em que o elemento Country é igual a 'USA';
 - Todos os cds com um preço inferior a 10 euros;
 - Todos os cds lançados entre 1980 e 1990;
 - Título (title) de cada CD;
 - Os CDs dos artistas em que o nome comeca por 'B':
 - O número de CDs armazenados no documento;
 - O título (Title) e preço (Price) para os CDs dos anos 90.
- 2. Considere o documento XML com o nome: cd_catalog, disponível na plataforma moodle. Escreva transformações XQuery (utilizando a ferramenta: Oxygen XML Editor) que permitam obter:
 - 2.1. Todos os cds lançados entre 1980 e 1990, ordenados pelo ano de lançamento.

O resultado deverá surgir com o seguinte formato:

```
?xml version="1.0" encoding="UTF-8"?>
<CD>
    <TITLE>Greatest Hits</TITLE>
    <ARTIST>Dolly Parton</ARTIST>
    <COUNTRY>USA</COUNTRY>
    <COMPANY>RCA</COMPANY>
    <PRICE>9.90</PRICE>
    <YEAR>1982</YEAR>
    </CD>
    <CD>
(...)
```

PEI/EOI Página: 1 / 4

2.2. O título (Title) e preço (Price) para os CDs dos anos 90. O resultado deverá surgir com o seguinte formato:

2.3. Contar o número de CDs armazenados no documento. O resultado deverá surgir com o seguinte formato:

```
<?xml version="1.0" encoding="UTF-8"?>
<Total>26</Total>
```

2.4. A lista de todos os CDs (title, year, price) assim como a "idade" de cada CD (tendo em conta o ano atual). O resultado deverá ser apresentado utilizando um elemento: <AGE>. O resultado deverá surgir com o seguinte formato:

2.5. O número de autores (ARTIST) existentes. Retorne um documento XML válido.

PARTE II

- 1. Considere o documento XML com o nome: bookstore (disponível na plataforma moodle). Escreva expressões XPath que permitam obter:
 - A lista dos títulos (title) dos livros (book) escritos (lang) em "en-us";
 - Quantos livros (book) estão escritos (lang) em "en-us"? (dica: utilize a função count ())
 - A lista dos autores (author) com o apelido "Laurentiis";
 - A lista dos títulos (title) dos livros (book) com um preço (price) superior a 30;
 - O último nodo da bookstore; (dica: utilize a função last())
 - O primeiro nodo da bookstore; (dica: utilize a função position())
 - Devolver o número de elementos que o documento contém; (dica: utilize a função count ())
 - Do primeiro livro, devolver o seu segundo elemento;
 - Devolver os livros que contenham "ay" ou "ry" no seu título (title); (dica: utilize a função contains())

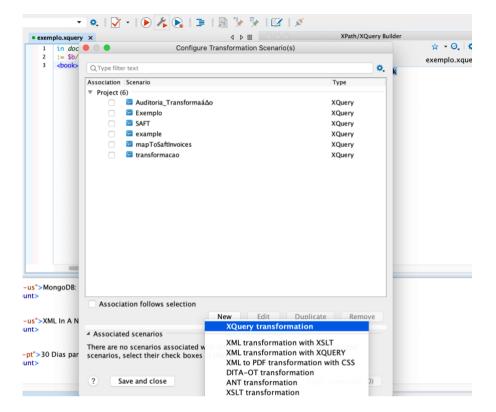
PEI/EOI Página: 2 / 4

- 2. Considere o documento XML com o nome: bookstore, disponível na plataforma moodle. Considere o documento XML com o nome: bookstore.xml, disponível na plataforma moodle. Escreva transformações XQuery (utilizando a ferramenta: Oxygen XML Editor) que permitam obter:
 - 2.1. Todos os livros com um preço entre 30 e 100 euros;
 - 2.2. Um documento com os títulos dos livros e o seu preço. Os dados dos livros devem ser apresentados de forma ordenada alfabeticamente pelo título;
 - 2.3. Apresentar todos os livros e o número de autores de cada um;
 - 2.4. Apresentar o título de cada livro utilizando um elemento: *<Titulo>*. Associado a este elemento, crie um atributo indicando a posição do livro na série retornada pela consulta;
 - 2.5. Apresentar todos os livros (todos os elementos). Para os livros da categoria: WEB, deve ser aplicado um desconto de 15%;
 - 2.6. Escreva uma transformação XQuery que permita obter um novo documento XML de acordo com a estrutura do documento em anexo: newBookStore.xml.
- Considere os documentos SAFT1 e SAFT2 disponibilizados na plataforma moodle. Coloque os dois documentos numa pasta com o nome: collection. Crie um ficheiro XQuery que contenha uma transformação que permita obter um documento XML com a lista de produtos existente em cada documento SAFT (distinga os produtos de acordo com o documento SAFT a que se encontram associados);

```
<?xml version="1.0" encoding="UTF-8"?>
<resumo>
    <saft nomeEmpresaEmissora="" data=" ">
        odutos>
            oduto>
              (\ldots)
           </produto>
        </produtos>
    </saft>
    <saft nomeEmpresaEmissora="" data=" ">
        cprodutos>
           oduto>
              (\ldots)
           </produto>
       </produtos>
    </saft>
```

Utilizando a ferramenta Oxygen XML Editor pode executar uma transformação XQuery conforme demonstra a seguinte imagem:

PEI/EOI Página: 3 / 4



Mais informação: https://www.oxygenxml.com/doc/versions/22.1/ug-editor/topics/xquery-transformation.html