

Processamento Estruturado de Documentos 2001

LESI + LMCC (5º ano)

Ficha Teórico-Prática nº4

Data: 14 de Novembro de 2001

Exercício 1 (Sumários) *Especifique uma stylesheet XSL que transforma o documento dos sumários numa página HTML atendendo aos seguintes requisitos:*

1. *As aulas teóricas aparecem primeiro ordenadas cronologicamente;*
2. *As aulas práticas aparecem a seguir também ordenadas cronologicamente;*
3. *no topo da página aparece um índice cronológico de aulas.*

Passagem de Parâmetros para uma stylesheet XSL

Um parâmetro pode ser declarado globalmente (filho de *xsl:stylesheet*) ou localmente numa template.

A declaração de parâmetros faz-se recorrendo ao elemento *xsl:param* que tem a seguinte sintaxe:

```
<xsl:param
  name="nome-do-parâmetro"
  [select="expressão-xpath"]>
  conteúdo-textual-do-parâmetro
</xsl:param>
```

Um parâmetro pode ser inicializado com o resultado duma expressão XPath ou alternativamente com um conteúdo alfanumérico.

Os parâmetros globais são usados para passar valores da linha de comando para a stylesheet.

Por exemplo, se eu quiser passar uma palavra como argumento para a stylesheet, tenho de declarar o seguinte parâmetro:

```
<xsl:param
  name="palavra">
  vazio
</xsl:param>
```

Desta maneira fica disponível o parâmetro de nome *palavra* inicializado com um valor por omissão "vazio". Está acessível em qualquer ponto da stylesheet (é global) através da expressão "\$palavra".

Para passar valores na linha de comando tenho de actuar de acordo com a sintaxe esperada pelo processador:

Sablotron `sabcmd some-stylesheet.xml some-document.xml $palavra="ped"`

Saxon `saxon some-document.xml some-stylesheet.xml palavra="ped"`

Exercício 2 (Submissão de TPs) Foi montado um sistema para submissão electrónica de trabalhos práticos. Este sistema guarda a informação enviada num ficheiro XML com a seguinte estrutura:

```
...
<grupo>
  <aluno1>
    <numero1>19670</numero1>
    <nome1>Alexandra Pinto</nome1>
    <curso1>LESI</curso1>
  </aluno1>
  <aluno2>
    <numero2>22639</numero2>
    <nome2>André Sousa</nome2>
    <curso2>LESI</curso2>
  </aluno2>
  <aluno3>
    <numero3></numero3>
    <nome3></nome3>
    <curso3></curso3>
  </aluno3>
  <titulo>Processador genérico de documentos estruturados (XML)</titulo>
  <file>19670_22639_.pli</file>
</grupo>
...
```

Agora é necessário criar várias stylesheets para processar esta informação e criar páginas HTML onde seja fácil consultar a informação.

1. Especifique uma stylesheet que gera um novo ficheiro XML com a actualização do nome de algumas anotações: procederá à remoção do índice numérico nas etiquetas aluno, numero, nome e curso. A stylesheet deverá ainda eliminar os campos vazios.
2. Especifique uma stylesheet que produz uma lista de alunos ordenada alfabeticamente por nome. Coloque a informação relativa a cada um dos alunos numa linha numa tabela HTML.
3. Especifique uma stylesheet que produz uma lista de alunos ordenada por número de aluno. Coloque a informação relativa a cada um dos alunos numa linha numa tabela HTML.
4. Especifique uma stylesheet que produz uma lista das submissões repetidas. Essa lista deverá ser um novo documento XML que deverá indicar para cada situação: a posição da submissão/grupo na árvore documental original, qual a situação anómala (o número que está repetido por exemplo); pense num DTD para este documento resultado.
5. Desenvolva uma stylesheet de nome remove que recebe um parâmetro numérico n e elimina o n -ésimo grupo do documento XML, o resultado é o mesmo documento XML mas sem aquela subárvore.
6. Desenvolva uma stylesheet de nome consulta que recebe um parâmetro numérico n e dá como resultado um novo documento (HTML?) com a informação respeitante ao grupo na posição n no documento original.
7. Desenvolva uma stylesheet de nome take, que recebe um parâmetro numérico n , e que dá como resultado um documento formado pelo conteúdo dos primeiros n grupos no documento original.

8. Desenvolva uma *stylesheet* de nome *drop*, que recebe um parâmetro numérico *n*, e que dá como resultado um documento formado pelo conteúdo dos últimos grupos do documento original após a eliminação dos primeiros *n* grupos.

Exercício 3 (Bibliografia) Descarregue o ficheiro relativo à Bibliografia do JCR da página da disciplina e tente resolver as alíneas seguintes:

- a) Desenvolva uma *stylesheet XSL*, e uma *script XML::DT* para gerar uma versão *HTML* do documento atendendo aos seguintes pormenores:
- Cada entrada deverá estar individualizada das restantes (linhas horizontais, espaçamento, ...).
 - No topo da página, deverá aparecer um índice (disposto horizontalmente ou verticalmente) que permita um acesso directo a cada entrada. Utilize o identificador da entrada.
 - Tente criar um índice de autores em que a cada autor está associada a lista de identificadores das respectivas entradas. Pense na *script* ou na *stylesheet*, faça a que lhe parecer mais fácil. Depois tente fazer a outra.
- b) Se ainda tiver tempo especifique um *DTD* para este tipo de documentos.