

# Processamento Estruturado de Documentos

LMCC & LESI, Universidade do Minho

Ano lectivo 2000/2001

Ficha Teórico-Prática N°6

José Carlos Ramalho

22 de Novembro de 2000

## 1 Transformação de Documentos com XSL

1.1 *Faça o download do documento gloss.xml, trata-se de um dicionário de termos arqueológicos. Especifique uma stylesheet em XSL que converta o dicionário num documento XML com a seguinte estrutura:*

```
<?xml version="1.0" encoding="iso-8859-1"?>
<gloss>
  <termo>
    <chave>Ábaco</chave>
    <classe>definição</classe>
    <descricao>Parte plana que remata superiormente o capitel.</descricao>
  </termo>
  <termo>
    <chave>Abóbada</chave>
    <classe>definição</classe>
    <descricao>Cobertura curva de um espaço delimitado por duas paredes, obtida pelo des
  </termo>
  ...
</gloss>
```

*Tenha em atenção a formatação (a indentação por espaços das diferentes etiquetas).*

**Solução**

---

```
<?xml version="1.0" encoding="iso-8859-1"?>
<xsl:stylesheet
  version="1.0"
```

```

xmlns:xsl="http://www.w3.org/1999/XSL/Transform">

<xsl:output version="1.0" encoding="iso-8859-1"/>

<xsl:template match="/">
  <xsl:element name="gloss">
    <xsl:text>
</xsl:text>
    <xsl:for-each select="//CHAVE">
      <xsl:element name="termo">
        <xsl:text>
</xsl:text>
        <xsl:element name="chave">
          <xsl:value-of select="."/>
        </xsl:element>
        <xsl:text>
</xsl:text>
        <xsl:element name="classe">
          <xsl:value-of select="../CLASSE"/>
        </xsl:element>
        <xsl:text>
</xsl:text>
        <xsl:element name="descricao">
          <xsl:value-of select="../DESC"/>
        </xsl:element>
        <xsl:text>
</xsl:text>
      </xsl:element>
    </xsl:for-each>
  </xsl:element>
</xsl:template>

</xsl:stylesheet>

```

---

**1.2** *Especifique uma nova stylesheet XSL para transformar o glossário obtido num novo documento XML mas agora com a estrutura seguinte:*

```

<gloss>
  <chaves>
    <chave>...</chave>

```

```

    ...
</chaves>
<classes>
  <classe>...</classe>
  ...
</classes>
<descs>
  <desc>...</desc>
  ...
</descs>
</gloss>

```

### *Solução*

---

```

<?xml version="1.0" encoding="iso-8859-1"?>
<xsl:stylesheet
  version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform">

<xsl:output version="1.0" encoding="iso-8859-1"/>

<xsl:template name="nl">
  <xsl:text>
</xsl:text>
</xsl:template>

<xsl:template name="indent">
  <xsl:text>   </xsl:text>
</xsl:template>

<xsl:template match="/">
  <xsl:call-template name="nl"/>
  <xsl:element name="gloss">

    <xsl:call-template name="nl"/>
    <xsl:call-template name="indent"/>
    <xsl:element name="chaves">
      <xsl:for-each select="/gloss/termo">
        <xsl:call-template name="nl"/>
        <xsl:call-template name="indent"/>
        <xsl:call-template name="indent"/>
        <xsl:element name="chave">
          <xsl:value-of select="chave"/>

```

```

        </xsl:element>
    </xsl:for-each>
</xsl:element>

<xsl:call-template name="nl"/>
<xsl:call-template name="indent"/>
<xsl:element name="classes">
    <xsl:for-each select="/gloss/termo">
        <xsl:call-template name="nl"/>
        <xsl:call-template name="indent"/>
        <xsl:call-template name="indent"/>
        <xsl:element name="classe">
            <xsl:value-of select="classe"/>
        </xsl:element>
    </xsl:for-each>
</xsl:element>

<xsl:call-template name="nl"/>
<xsl:call-template name="indent"/>
<xsl:element name="descs">
    <xsl:for-each select="/gloss/termo">
        <xsl:call-template name="nl"/>
        <xsl:call-template name="indent"/>
        <xsl:call-template name="indent"/>
        <xsl:element name="desc">
            <xsl:value-of select="descricao"/>
        </xsl:element>
    </xsl:for-each>
</xsl:element>

    <xsl:call-template name="nl"/>
</xsl:element>
</xsl:template>

</xsl:stylesheet>

```

---

**1.3** Para terminar especifique uma nova stylesheet para transformar o glossário obtido no último exercício num glossário igual ao resultante do primeiro exercício.

**Solução**

---

```

<?xml version="1.0" encoding="ISO-8859-1"?>
<xsl:stylesheet

```

```

    version="1.0"
    xmlns:xsl="http://www.w3.org/1999/XSL/Transform" >

<xsl:output version="1.0" encoding="ISO-8859-1"/>

<xsl:template match="/">
  <xsl:element name="gloss">

    <xsl:for-each select="//chave">
      <xsl:variable name="count" select="position()"/>
      <xsl:element name="termo">
        <xsl:element name="chave"><xsl:value-of select="."/></xsl:element>
        <xsl:element name="classe"><xsl:value-of select="//classe[$count]"/></xsl:element>
        <xsl:element name="descricao"><xsl:value-of select="//desc[$count]"/></xsl:element>
      </xsl:element>
    </xsl:for-each>

  </xsl:element>
</xsl:template>

</xsl:stylesheet>

```

---

## Aplicação das stylesheets no eremita

Teste as várias stylesheets com o Xalan:

```
$testXSL -IN doc.xml -XSL doc.xsl
```