

Sl. 357 alternative:

<mycity >

<xsl:attribute name="lang" >

<xsl:value-of select="did" / >

</xsl:attribute >

⋮

</ >

Sl 360:

→ output

<county @code = 'D' >

neighbors = 'NLBLF CH A .. ' >

</ >

```

<xsl:stylesheet version="2.0"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
<xsl:template match="mondial">
<html><body><table>
<xsl:apply-templates select="country"/>
</table></body></html>
</xsl:template>
<xsl:template match="country">
<tr valign="top">
<td><xsl:value-of select="name"/></td>
<td><xsl:value-of select="@car_code"/></td>
<td align="right"><xsl:value-of select="population[last()]/></td>
<td align="right"><xsl:value-of select="@area"/></td>
<td valign="top">
<table>
<xsl:for-each select="id(@memberships)">
<tr> <td><xsl:value-of select="name"/> </td></tr>
</xsl:for-each>
</table>
</td>
</tr>
</xsl:template>
</xsl:stylesheet>

```

```

<xsl:stylesheet version="2.0"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
<xsl:template match="mondial">
<html><body><table>
<xsl:apply-templates select="country"/>
</table></body></html>
</xsl:template>
<xsl:template match="country">
<xsl:variable name="c" select="."/>
<xsl:for-each select="id(@memberships)">

<tr valign="top">
<td><xsl:value-of select="$c/name"/></td>
<td><xsl:value-of select="name"/> </td>
</tr>
</xsl:for-each>

</xsl:template>
</xsl:stylesheet>

```

join-loop

pairs (cont, org)

- XATE

```
</xsl:template>
```

```
<xsl:template match="country">
```

```
<xsl:apply-templates select="border">
```

```
<xsl:with-param name="c" select="."/>
```

```
</xsl:apply-templates>
```

```
</xsl:template>
```

```
<xsl:template match="border">
```

```
<xsl:param name="c"/>
```

```
<tr>
```

```
<td><xsl:value-of select="$c/@car_code"/></td>
```

```
<td><xsl:value-of select="@country"/></td>
```

```
</tr>
```

```
</xsl:template>
```

```
</xsl:stylesheet>
```

apply-templates with param :
↳ holds the complete country element



```
<xsl:with-param name="allparams" >
```

```
<firstparam> .. </>
```

```
<secondparam> .. </>
```

```
<thirdparam> .. </>
```

```
</>
```



access inside:

```
$allparams/firstparam
```

```
$allparams/secondparam
```

```
⋮
```

allparams:



≙ xquery's let := ΔΔΔ

```
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
version="2.0">
<xsl:template match="mondial">
<html><body> <h2>Membership Table</h2>
<table> <xsl:apply-templates select="organization"/>
</table></body></html>
</xsl:template>
<xsl:template match="organization">
<tr><td colspan="2"><b><xsl:value-of select="name"/></b></td></tr>
<xsl:apply-templates select="members">
<xsl:with-param name="the_org" select="name/text()" tunnel="yes"/>
</xsl:apply-templates>
</xsl:template>

<xsl:template match="members">
<xsl:param name="the_org"/>
<xsl:apply-templates select="id(@country)">
<xsl:with-param name="type" select="@type"/>
</xsl:apply-templates>
</xsl:template>

<xsl:template match="country">
<xsl:param name="the_org" tunnel="yes"/>
<xsl:param name="type"/>
<tr><td><xsl:value-of select="$the_org"/></td>
<td><xsl:value-of select="$type"/></td>
<td><xsl:value-of select="name/text()"/></td></tr>
</xsl:template>

</xsl:stylesheet>
```

*Solution to
SE 370 (1)*