

[← XSLT <xsl:value-of> 元素](#)[XSLT <xsl:sort> 元素 →](#)

XSLT <xsl:for-each> 元素

<xsl:for-each> 元素允许您在 XSLT 中进行循环。

<xsl:for-each> 元素

XSL <xsl:for-each> 元素可用于选取指定的节点集中的每个 XML 元素：

实例

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
<xsl:template match="/">
<html>
<body>
<h2>My CD Collection</h2>
<table border="1">
<tr bgcolor="#9acd32">
<th>Title</th>
<th>Artist</th>
</tr>
<xsl:for-each select="catalog/cd">
<tr>
<td><xsl:value-of select="title"/></td>
<td><xsl:value-of select="artist"/></td>
</tr>
</xsl:for-each>
</table>
</body>
</html>
</xsl:template>
</xsl:stylesheet>
```

[尝试一下 »](#)

注意：select 属性的值是一个 XPath 表达式。这个 XPath 表达式的工作方式类似于定位某个文件系统，在其中正斜杠 (/) 可选择子目录。

过滤输出结果

通过在 <xsl:for-each> 元素中添加一个选择属性的判别式，我们也可以过滤从 XML 文件输出的结果。

<xsl:for-each select="catalog/cd[artist='Bob Dylan']">

合法的过滤运算符：

- = (等于)
- != (不等于)

- < (小于)
- > (大于)

看看调整后的 XSL 样式表：

实例

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
<xsl:template match="/">
<html>
<body>
<h2>My CD Collection</h2>
<table border="1">
<tr bgcolor="#9acd32">
<th>Title</th>
<th>Artist</th>
</tr>
<xsl:for-each select="catalog/cd[artist='Bob Dylan']">
<tr>
<td><xsl:value-of select="title"/></td>
<td><xsl:value-of select="artist"/></td>
</tr>
</xsl:for-each>
</table>
</body>
</html>
</xsl:template>
</xsl:stylesheet>
```

尝试一下 »

← XSLT <xsl:value-of> 元素

XSLT <xsl:sort> 元素 →

✎ 点我分享笔记