

# xslt语法:XSLT的元素语法

疯狂代码 <http://CrazyCoder.cn/>     ĵ:<http://CrazyCoder.cn/WebPlatformAndTool/Article11265.html>

通过前面两章介绍我们已经对XSLT基本概念和它转换过程有了些了解下面我们起来学习XSLT具体语法说到语法总是比较枯燥您可以大体上浏览遍等您真正需要使用XSLT时候再仔细研究它们

## 3.XSLT元素语法

### 3.1xsl:template和xsl:apply-templates

### 3.2xsl:value-of

### 3.3xsl:for-each

### 3.4xsl:

### 3.5Xxsl:choose,when,otherwise

### 3.6xsl:sort

### 3.1xsl:template和xsl:apply-templates

模板(template)是XSLT中最重要概念的XSLT文件就是由个个模板组成任何个XSLT文件至少包含个模板模板概念就象是搭积木；你如果是员也可以将模板看作个思路方法个类或者个模块它们可以被拼装组合也可以单独成块区别模板控制区别输出格式

模板(template)由两部分组成:匹配模式(matchpattern)和执行简单讲模式定义XML源文档中哪个节点将被模板处理执行则定义输出是什么格式两部分对应语法为xsl:template和xsl:apply-templates

xsl:template语法是:

```
<xsl:template
```

```
match=pattern
```

```
name=qname
```

```
priority=number
```

```
mode=qname>
```

```
<!--执行内容-->
```

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

xsl:template作用是定义个新模板属性中name,priority,和mode用来区别匹配同节点区别模板它们不是常用属

性match属性则控制模板匹配模式(pattern)匹配模式是用来定位XML源文档中哪个节点被模板处理个模板匹配个节点我们用个例子来帮助理解:

假设我们要处理个包含章节和段落文档我们用para元素定义段落用chapter元素定义章节我们来看看match属性可能值下面语句写法介绍说明模板匹配所有para元素

```
<xsl:templatematch="para">
```

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

下面语句写法介绍说明模板匹配所有para元素和所有chapter元素:

```
<xsl:templatematch="(chapter|para)">
```

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

下面语句写法介绍说明模板匹配所有父节点为chapter元素para元素:

```
<xsl:templatematch="chapter//para">
```

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

下面语句写法介绍说明模板匹配根节点:

```
<xsl:templatematch="/">
```

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

我们再来看apply-templates语法:

```
<xsl:apply-templates
```

```
select=node-expression
```

```
mode=qname>
```

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

xsl:apply-templates用来执行那个节点被模板具体处理你可以将它理解为中子select属性用来定义确切节点名称xsl:apply-templates总是包含在xsl:template元素中象这样:

```
<xsl:templatematch="/">
```

```
<xsl:apply-templatesselect="para"/>
```

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

这段代码介绍说明模板匹配整个文档(根节点)具体执行时处理根节点下所有para元素

```
<xsl:templatematch="para">
```

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

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

而这段代码则表示模板匹配para节点所有para下子元素都将被处理

### 3.2xsl:value-of

XSL:value-of用来将源文档中元素文本值写到输出文档中例如:

有个个人资料XML文档:

```
<?xmlversion="1.0"encoding="iso-8859-1"?>
```

```
<PERSON>
```

```
<name>ajie</name>
```

```
<age>28</age>
```

```
</PERSON>
```

我如果想在输出文档中显示上面这个XML源文档中name元素值可以这样写XSLT代码:

```
<xsl:templatematch="PERSON">
```

```
<xsl:value-of select="name"/>
```

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

执行后你会看到"ajie"被单独显示出来其中match="PERSON"定义模板匹配PERSON节点xsl:value-of语法介绍说明需要输出个节点值而select="name"则定义需要被输出元素为name看这个过程是不是和数据库里查询个人名字很象？当然xsl:value-of查询还有更多更复杂语法是涉及寻找和定位功能我们会放在后面XPath语法中在仔细讲解

同样功能还有xsl:copy-of使用方法样就不重复解释了

### 3.3xsl:for-each

xsl:for-each语法允许你循环处理被选择节点例如:有个含多个个人资料XML文档:

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

```
<PEOPLE>
```

```
<PERSON>
```

```
<name>ajie</name>
```

```
<age>28</age>
```

```
</PERSON>
```

```
<PERSON>
```

```
<name>tom</name>
```

```
<age>24</age>
```

```
</PERSON>
```

```
<PERSON>
```

```
<name>miake</name>
```

```
<age>30</age>
```

```
</PERSON>
```

```
</PEOPLE>
```

我需要显示所有人姓名则可以将XSLT代码写成:

```
<xsl:templatematch="PEOPLE">
```

```
<xsl:for-eachselect="child::PERSON">
```

```
<xsl:value-ofselect="name"/>
```

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

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

3.4xsl:

xsl:类似普通语言条件语句允许设定节点满足某个条件时被模板处理xsl:语法格式为:

```
<xsl:test=布尔表达式>
```

templatebody

</xsl:>

例如:

```
<xsl:templatematch="PEOPLE">
```

```
<xsl:test="@name">
```

```
<p> <xsl:value-ofselect="@name"/> </p>
```

```
</xsl:>
```

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

这段代码意思是检测PEOPLE节点下所有元素如果发现有<name>元素则将<name>元素值输出其中@符号是通配符表示节点下所有元素

### 3.5xsl:choose,xsl:when和xsl:otherwise

xsl:语法没有属性如果我们要进行多项选择那么就要使用xsl:choose/xsl:when/xsl:otherwise系列流程控制语法了具体使用请看下面XSL文件例子:

```
<xsl:templatematch="PEOPLE">
```

```
<xsl:choose>
```

```

<xsl:when test="@name=ajie">

<B> <xsl:value-of select="@name"/> </B>

</xsl:when>

<xsl:when test="@name">

<I> <xsl:value-of select="@name"/> </I>

</xsl:when>

<xsl:otherwise>

Nonameavailable

</xsl:otherwise>

<xsl:choose>

</xsl:template>

```

介绍说明:首先在PEOPLE节点下寻找<name>属性值为ajie元素如果找到将ajie用粗体输出；如果没有发现值为ajie<name>元素则将所有<name>元素值都用斜体输出；如果没有发现任何<name>元素则显示"No nameavailable"

### 3.6xsl:sort

在XSLT中可以对XML源文档元素进行重新排序排序语法就是xsl:sort举例:下面代码就是将文档元素按name排序

```

<xsl:templatematch="PEOPLE">

```

```
<xsl:apply-templatesselect="PERSON">
```

```
<xsl:sortselect="@name"/>
```

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

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

以上是XSLT元素主要语法还有很多其他语法例如:import,,element,attribute,number,param等等语法在这里就不解释我们目的是让您对XSLT语法有基本概念理解XSLT作为种转换语言强大功能 2009-2-12 3:58:48  
疯狂代码 <http://CrazyCoder.cn/>