**Лабораторная работа №2**

**Преобразование XML-документа**

**Теоретический материал**

**XSLT** (eXtensible Stylesheet Language Transformations) - расширяемый язык преобразования листов стилей. При использовании XSLT в среде Web синтаксический анализ может происходить либо на стороне пользователя (т.е. в пользовательском браузере), либо на стороне сервера.

Анализ XSLT на стороне клиента похож на процедуру применения каскадных стилей. В исходный документ нужно добавить тег

<?xml-stylesheet type="text/xsl" href="transform.xsl" ?>

Здесь **transform.xsl** - имя файла листа стилей XSLT.

**Шаблоны**

Каждый элемент XSLT начинается префиксом **xsl:**. Элемент **xsl:stylesheet** служит контейнером для листа стилей XSLT. Атрибут **version="1.0"** этого элемента определяет версию спецификации XSL.

Преобразования XSLT основаны на шаблонах. Шаблон определяется инструкцией **xsl:template**.

XSLT-процессор анализируют исходный документ и пытается найти подходящий XSL-шаблон. Если такой шаблон найден, то выполняются инструкции внутри него.

#### Обработка всегда начинается с шаблона, где **match="/"**. Это значение пути адресации соответствует корневому узлу.

### ***Исходный XML***

<?xml version="1.0" encoding="WINDOWS-1251" ?>

<?xml-stylesheet type="text/xsl" href="t02.xsl" ?>

<книга>

<название>Мастер и Маргарита.</название>

<автор>Михаил Булгаков</автор>

</книга>

### ***Преобразование XSLT***

<?xml version="1.0" encoding="WINDOWS-1251" ?>

<xsl:stylesheet version = "1.0"

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

<xsl:template match="/">

<h2>

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

</h2>

</xsl:template>

<xsl:template match="автор">

<i>

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

</i>

</xsl:template>

</xsl:stylesheet>

**РЕЗУЛЬТАТ ПРИМЕРА:**

## Мастер и Маргарита. Михаил Булгаков

Содержание элементов может быть извлечено из исходного документа двумя основными способами:

1. инструкцией **xsl:value-of**. В этом случае содержание элемента используется без какой-либо дальнейшей обработки.
2. инструкцией **xsl:apply-templates**. В этом случае XSLT-процессор продолжает обрабатывать выбранные элементы, для которых определен шаблон:

<?xml version="1.0" encoding="WINDOWS-1251" ?>

<xsl:stylesheet version = "1.0"

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

<xsl:template match="/">

<h2>

<xsl:apply-templates/>

</h2>

</xsl:template>

<xsl:template match="автор">

<i>

&#160;<xsl:value-of select="."/>

</i>

</xsl:template>

</xsl:stylesheet>

**РЕЗУЛЬТАТ ПРИМЕРА:**

## Мастер и Маргарита. *Михаил Булгаков*

В качестве значений атрибутов **match** и **select** используются выражения, синтаксис которых похож на маршрут файловой системы:

|  |  |
| --- | --- |
| **Выражение** | **Описание** |
| / | Корневой узел |
| . | Текущий узел |
| .. | Родительский узел текущего узла |
| fruit | Узел **fruit** |
| fruit/lime | Подузел **lime** узла **fruit** |
| fruit/\* | Все потомки узла **fruit** |
| /fruit | Узел **fruit**, являющийся прямым потомком корневого узла |
| @taste | Атрибут **taste** текущего узла |
| @\* | Все атрибуты текущего узла |
| fruit@taste | Атрибут **taste** узла **fruit** |
| fruit/lime@taste | Атрибут **taste** узла **lime**, являющегося подузлом узла **fruit** |
| ..@taste | Атрибут **taste** родительского узла |
| // | Любое количество промежуточных узлов |
| fruit//lime | Все узлы **lime**, имеющие предка **fruit** |
| | | Знак разделения конкретных узлов |
| lime|grape | Узел **lime** и узел **grape** |
| [] | Предикатное выражение |
| fruit[lime] | Узел **fruit**, имеющий потомка **lime** |
| fruit[lime="fine"] | Узел **fruit**, имеющий потомка **lime**, значение которого равно **fine** |
| fruit[@taste] | Узел **fruit**, имеющий атрибут **taste** |
| fruit[@taste="5"] | Узел **fruit**, имеющий атрибут **taste**, значение которого равно **5** |
| count(fruit/\*) | Количество потомков узла **fruit** |
| name() | Имя текущего узла |

### ***Исходный XML***

<?xml version="1.0" encoding="WINDOWS-1251" ?>

<?xml-stylesheet type="text/xsl" href="t07.xsl" ?>

<source>

<AAA id="a1">

<BBB id="b1"/>

<BBB id="b2"/>

</AAA>

<AAA id="a2">

<BBB id="b3"/>

<BBB id="b4"/>

<CCC id="c1">

<DDD id="d1"/>

</CCC>

<BBB id="b5">

<CCC id="c2"/>

</BBB>

</AAA>

</source>

### ***Преобразование XSLT***

<?xml version="1.0" encoding="WINDOWS-1251" ?>

<xsl:stylesheet version = "1.0"

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

<xsl:template match="/">

<xsl:apply-templates select="/source/AAA//CCC"/>

<xsl:apply-templates select="/source//AAA/BBB//\*"/>

</xsl:template>

<xsl:template match="AAA">

<div style="color:navy">

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

<xsl:text> id=</xsl:text>

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

</div>

</xsl:template>

<xsl:template match="BBB">

<div style="color:purple">

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

<xsl:text> id=</xsl:text>

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

</div>

</xsl:template>

<xsl:template match="CCC">

<div style="color:red">

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

<xsl:text> id=</xsl:text>

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

</div>

</xsl:template>

<xsl:template match="DDD">

<div style="color:blue">

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

<xsl:text> id=</xsl:text>

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

</div>

</xsl:template>

</xsl:stylesheet>

**РЕЗУЛЬТАТ ПРИМЕРА:**

CCC id=c1

CCC id=c2

CCC id=c2

С помощью режимов **mode** элемент может быть обработан многократно, причем каждый раз с различным результатом.

К атрибутам можно обращаться также, как и к элементам. Надо только поставить "**@**" перед именем атрибута. Атрибуты можно обрабатывать аналогично элементам.

**Задания к лабораторной работе**

**Задание 1**

1. Разработать XSLT преобразование XML документа (Лаб1) в документ XML другой структуры. В результате преобразования должен получиться новый XML документ с тем же содержанием, но в котором отсутствуют блоки правильных ответов.
2. Создать XML схему для проверки корректности полученного документа.
3. Создать преобразование для получения списка правильных ответов по вопросам.

**Задание 2**

Разработать XSLT преобразование XML документа в HTML документ. В результате преобразования документа должен получиться HTML документ, представляющий собой страницу для отображения в браузере. Страница должна представлять собой список вопросов с вариантами ответов и соответствующими элементами выбора: список, флажки, поле ввода.