**ABOUT XSL**

**XSL** (e**X**tensible **S**tylesheet **L**anguage) - это язык стилей для **XML**.

**XSLT** означает **XSL Transformations**.

Консорциум World Wide Web (W3C) начал разрабатывать XSL, потому что возникла необходимость в языке стилей на основе XML.

XSL = таблицы стилей для XML.

XML не использует предопределенные теги, и поэтому смысл каждого тега не совсем понятен. Например, элемент <table> может указывать таблицу HTML, предмет мебели или что-то еще - и браузеры не знают, как отображать его!

Итак, XSL описывает, как должны отображаться элементы XML.

XSL - больше, чем язык таблиц стилей. Он состоит из четырех частей:

1. **XSLT** - язык для преобразования XML-документов;
2. **XPath** - язык для навигации в XML-документах;
3. **XSL-FO** - язык для форматирования XML-документов (прекращено в 2013 году);
4. **XQuery** - язык для запросов к XML-документам.

С помощью CSS3 Paged Media Module W3C предоставил новый стандарт для форматирования документов. Таким образом, с 2013 года CSS3 предлагается в качестве замены XSL-FO.

Что такое XSLT?

* XSLT означает XSL Transformations;
* XSLT является самой важной частью XSL;
* XSLT преобразует XML-документ в другой XML-документ;
* XSLT использует XPath для навигации по XML-документам;
* XSLT является Рекомендацией W3C.

XSLT является самой важной частью XSL.

XSLT используется для преобразования XML-документа в другой документ XML или другого типа документа, который распознается браузером, например, HTML и XHTML. Обычно XSLT делает это, преобразовывая каждый элемент XML в элемент (X)HTML.

С помощью XSLT вы можете добавлять / удалять элементы и атрибуты в выходной файл или из него. Вы также можете переупорядочивать и сортировать элементы, выполнять тесты и принимать решения о том, какие элементы скрывать и отображать, и многое другое.

Общим способом описания процесса преобразования является то, что XSLT преобразует дерево-источник XML в дерево результатов XML.

XSLT использует XPath для поиска информации в XML-документе. XPath используется для навигации по элементам и атрибутам в XML-документах.

Правильная декларация таблиц стилей

Корневым элементом, объявляющим документ как таблицу стиля XSL, является

<xsl: stylesheet> или <xsl: transform>.

Примечание: <xsl: stylesheet> и <xsl: transform> полностью синонимичны!

Элемент <xsl: stylesheet> не только идентифицирует документ как XSL-таблицу стилей, но и служит хранилищем других элементов.

Правильный способ объявления таблицы стилей XSL в соответствии с Рекомендацией W3C XSLT:

<xsl: stylesheet version = "1.0"

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

Чтобы получить доступ к элементам XSLT, атрибутам и функциям, мы должны объявить пространство имен XSLT в верхней части документа.

xmlns: xsl = "http://www.w3.org/1999/XSL/Transform" указывает на официальное пространство имен XSLT W3C. Если вы используете это пространство имен, вы также должны включить атрибут version = "1.0".

Свяжите таблицу стилей XSL с XML-документом

Добавьте ссылку таблицы стилей XSL в свой XML-документ:

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

Если у вас есть совместимый с XSLT браузер, он красиво преобразует ваш XML в XHTML.

Элемент <xsl: template>

Элемент <xsl: template> используется для создания шаблонов.

Браузер использует шаблон для отображения определенной ветви элементов в иерархии XML-документа, с которым вы связываете таблицу стилей. Атрибут match шаблона указывает на определенную ветвь. Значение атрибута match носит название образца (pattern). Образец ("/") представляет корневой элемент всего XML-документа. Каждая XSL-таблица стилей должна содержать один и только один шаблон с атрибутом match, который имеет значение "/". Корневой образец ("/") не представляет элемент Документ (или корневой элемент) XML-документа. Он представляет весь документ, для которого элемент Документ является дочерним. (Т. е. он аналогичен корневому узлу Document в объектной модели документа DOM).

Значение атрибута match является выражением XPath.

Содержимое внутри элемента <xsl: template> определяет некоторый HTML для записи на выходе.

Элемент <xsl: значение-of>

Элемент <xsl: value-of> может использоваться для извлечения значения элемента XML и добавления его в выходной поток преобразования.

Примечание. Атрибут select - выражение XPath. Выражение XPath работает подобно навигации по файловой системе; передняя косая черта (/) выбирает подкаталоги.

Элемент <xsl: for-each>

Элемент XSL <xsl: for-each> может использоваться для выбора каждого элемента XML указанного набора узлов.

Примечание. Атрибут select - выражение XPath. Выражение XPath работает подобно навигации по файловой системе; передняя косая черта (/) выбирает подкаталоги.

Фильтрация вывода

Мы также можем отфильтровать вывод из файла XML, добавив критерий для атрибута select в элементе <xsl: for-each>. Пример:

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

Операторы фильтра:

* = (равный);
* ! = (не равно);
* & Lt; меньше, чем;
* & GT; больше, чем.

Элемент <xsl: sort>

Элемент <xsl: sort> используется для сортировки вывода.

Где разместить элемент <xsl: sort>?

Чтобы отсортировать вывод, просто добавьте элемент <xsl: sort> внутри элемента <xsl: for-each> в файле XSL.

Примечание. Атрибут select указывает, какой элемент XML нужно сортировать.

Элемент <xsl: if>

Элемент <xsl: if> используется для установки условного теста на содержимое XML-файла.

Синтаксис:

<xsl: if test = "expression">

   ...некоторый вывод, если выражение истинно...

Где поместить элемент <xsl: if>?

Чтобы добавить условный тест, добавьте элемент <xsl: if> внутри элемента <xsl: for-each> в файле XSL.