**1. Что такое XSL? Чем является XSLT?**

XSL (eXtensible Stylesheet Language) —язык таблиц стилей для XML. XSL служит языком трансформирования документов XML и состоит из XML-словаря семантики форматирования

XSLT (eXtensible Stylesheet Language Transformations) — это декларативное описание преобразования (трансформации) любого XML-документа. Спецификация XSLT входит в состав XSL и является рекомендацией W3C

**2. Каково основное назначение технологии XSLT?**

Преобразование XML-документов из одной структуры в другую. XSLT предоставляет средства для создания правил преобразования, которые могут изменять структуру, содержимое и формат XML-документов

**3. Для чего предназначен <xsl:template>?**

**<xsl:template>** указывает, как должны преобразовываться части документа XML.

**4. Что означает значение match="/"?**

Значение «/» атрибут match используется, чтобы определить шаблон для всего XML документа целиком.

**5. Как подключить XSLT к xml?**

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

**6. Для чего предназначено <xsl:stylesheet>? \**

**<xsl:stylesheet>** определяет, что данный документ является таблицей стилей XSLT с атрибутами номера версии и пространства имен XSLT.

**7. Что означает <xsl:apply-templates>?**

**<xsl:apply-templates>** применяет некий шаблон к текущему элементу или к дочернему узлу текущего элемента

**8. Для чего и какие атрибуты имеет <xsl:sort>?**

**<xsl:sort>** используется для сортировки выходных данных и располагается внутри элемента **<xsl:for-each>**

– ***select*** — обязательный атрибут, значением которого является

ключевое выражение, которое вычисляется для каждого узла обрабатываемого множества, преобразуется в строку и затем используется как значение ключа при сортировке.

– ***order*** – необязательный атрибут, определяет порядок, в кото-

ром узлы должны сортироваться по своим ключам. Принимает только два значения – "*ascending*"(восходящий порядок сортировки), "*descending*" (нисходящий порядок).

– ***lang*** – необязательный атрибут, определяет язык ключей

сортировки.

– ***data-type*** – необязательный атрибут, определяет тип данных,

которые несут строковые значения ключей

**9. С помощью какого элемента можно осуществить сортировку с условиями?**

**<xsl:choose>** используется вместе с элементами **<xsl:when>** и **<xsl:otherwise>**, чтобы определить проверку на выполнение условия

**10. Для чего используется элемент <xsl:otherwise>?**

**<xsl:otherwise>** определяет поведение элемента в случае не прохождения проверки условия

**11. В чем заключается предназначение <xsl:when>?**

**<xsl:when>** определяет поведение элемента в случае прохождения проверки условия

**12. Что относится к XSL?**

XSL (Extensible Stylesheet Language) представляет собой семейство стандартов, включающих в себя три основных компонента:

XSLT (Extensible Stylesheet Language Transformations): XSLT используется для преобразования XML-документов в другие форматы, такие как HTML, текстовые файлы или другие XML-структуры. Он определяет правила и инструкции для преобразования одного XML-документа в другой формат.

XPath (XML Path Language): XPath является частью XSL, но может использоваться независимо от XSLT. Это язык запросов для выбора узлов в XML-документе. Он позволяет указывать пути к конкретным узлам или наборам узлов в XML-документе и используется в XSLT для выбора и обработки данных.

XSL-FO (Extensible Stylesheet Language Formatting Objects): XSL-FO используется для форматирования XML-документов для вывода на печать или другие медиа. Он определяет правила форматирования, такие как размещение текста, создание колонок, настройку страниц и другие параметры, необходимые для создания документов для печати или отображения.

**13. Как строятся шаблоны преобразований в XSLT?**

Шаблоны преобразований в XSLT строятся с использованием элемента **<xsl:template>**. С ним используются атрибуты:

**match**: Определяет, на какой части XML-документа будет применяться данный шаблон

**select**: Определяет, какие данные будут выбраны для обработки внутри шаблона.

**14. Каков алгоритм преобразования XML-документа с помощью языка XSLT?**

Процесс преобразования XML-документа с помощью XSLT включает несколько шагов:

**Загрузка XML и XSLT:** Процессор XSLT загружает исходный XML-документ, который нужно преобразовать, а также файл XSLT (трансформационные правила).

**Начальное сопоставление с шаблоном:** Процессор XSLT начинает сопоставление с корневым шаблоном или тем, который находится на наивысшем уровне вложенности в XSLT-документе.

**Сопоставление с элементами:** Процессор XSLT сопоставляет элементы XML с соответствующими шаблонами в XSLT-документе, используя атрибут **match** или **select**.

**Применение шаблонов:** Если найдено совпадение между элементом XML и шаблоном в XSLT, процессор XSLT применяет инструкции, определенные в этом шаблоне, к соответствующим элементам XML.

**Рекурсивное применение шаблонов:** Если в XSLT используется инструкция **<xsl:apply-templates>**, процессор XSLT рекурсивно применяет шаблоны к дочерним элементам или узлам в соответствии с логикой выбора шаблонов.

**Формирование результирующего документа:** По мере прохождения по XML-документу и применения шаблонов процессор XSLT формирует результирующий документ на основе выполненных преобразований.

**Завершение преобразования:** После завершения обработки всего XML-документа процессор XSLT возвращает результирующий документ в требуемом формате, который может быть HTML, XML или другим форматом, сформированным на основе правил преобразования из XSLT.

**15. Для чего предназначено <xsl:value-of>?**

**<xsl:value-of>** используется для извлечения значения ото-

бранного XML элемента и добавления его в выходной поток пре-

образовываемого документа

**16. Для чего используются <xsl:for-each>? Какие элементы XSL могут быть внутри него?**

**<xsl:for-each>** может использоваться для выбора каждого

XML элемента заданного узлового набора

Внутри **<xsl:for-each>** могут быть **условные** операторы, **<xsl:template>** для выбора и применения шаблонов к дочерним узлам или элементам текущего контекста, любые инструкции XSLT для форматирования и вывода данных **(<xsl:value-of>, <xsl:if>, <xsl:choose>)**