**Контрольные вопросы к лабораторной работе 1:**

**1.**XML-документ представляет собой обычный текстовый файл, в котором при помощи специальных маркеров создаются элементы данных, последовательность и вложенность которых определяет структуру документа и его содержание.

**2.**Каждый XML- документ размечается тегами.***Тег*** - это текст, заключенный в угловые скобки, который не относится к содержанию документа, а отмечает начало или конец какого-либо документа.

**3.**XML-документ состоит из двух основных частей: пролога и элемента Документ (корневого элемента). Помимо этого, корректно сформированный XML-документ может содержать комментарии, инструкции по обработке, пробелы.

**4.** 1.DTD

2.Схема XML

1. DTD описывает схему документа для конкретного языка разметки посредством набора объявлений (объектов-параметров, элементов и атрибутов), которые описывают его класс (или тип) с точки зрения синтаксических ограничений этого документа. Также DTD может объявлять конструкции, которые всегда необходимы для определения структуры документа, но, зато, могут влиять на интерпретацию определённых документов.

2. XML Schema — язык описания структуры XML-документа. Спецификация XML Schema является рекомендацией W3C.

Как большинство языков описания XML, XML Schema была задумана для определения правил, которым должен подчиняться документ. Но, в отличие от других языков, XML Schema была разработана так, чтобы её можно было использовать в создании программного обеспечения для обработки документов XML.

**5.** Тело документа XML состоит из элементов разметки (markup) и непосредственно содержимого документа - данных (content). XML - тэги предназначены для определения элементов документа, их атрибутов и других конструкций языка.

**6.** Правила использования имен элементов (типов элемента):

· имя должно начинаться с буквы или символа подчеркивания (\_),

· следующие после первого символы могут быть буквами, цифры, точкой, тире или подчеркиванием.

· Не следует использовать имена, начинающиеся с префикса "xml"(в любом сочетании строчных или прописных букв).

· Имя, записанное в начальном теге, должно в точности соответствовать имени в конечном теге;

· Соблюдение регистра существенно для имен элементов, как и для всего текста в описании разметки.

**7.** Комментарий начинается с символов<!--и заканчивается символами-->. Внутри комментариев не может содержаться двойное тире (--), символ левой угловой скобки (<) и знак амперсанда (&). Комментарии можно вставлять в любое место XML-документа, кроме описания тега.

**8.** Инструкции (инструкции по обработке) предназначены для XML-процессора, который будет обрабатывать документ. Общий вид инструкции:

<?наименование данные?>

**9.** Документ XML может содержать пустые строки, состоящие из одного или нескольких пробелов, символов табуляции, символа Enter. Можно свободно добавлять пробелы и переводы строк между:

· начальными и конечными тегами;

· комментариями;

· инструкциями по обработке.

Не должно быть пробелов между открывающейся угловой скобкой и имеем элемента.

**10.** Отображение XML-документов с использованием каскадной таблицы стилей.

Отображение XML-документов с использованием каскадной таблицы стилей осуществляется в два этапа:

1. создание файла таблицы стилей.

2. связывание таблицы стилей с XML-документом.

Таблица стилей состоит из одного или нескольких правил (иногда их называют набором правил). Правило содержит информацию по отображению определенного типа элемента в XML-документе

**11.** Селектор представляет собой имя типа элемента, к которому относится информация по отображению.

**12.** Контекстуальным (contextual) селектором называется селектор, в котором имя элемента может предваряться именами одного или нескольких элементов-предков (родительский, родительский плюс родительский родителя и т.д.). В этом случае правило будет применено только к элементам с этим именем, которые являются вложенными подобным образом. Между именами элементов в контекстном селекторе ставят только пробелы.

**13.** C помощью директивы @import в одну каскадную таблицу стилей встроить одну или несколько других таблиц стилей. Возможность импорта отдельных таблиц стилей позволяет хранить правила для связанных стилей в отдельных файлах, а затем объединять их при создании документов определенного типа. Директива @import должна располагаться в начале таблицы стилей перед правилами. В начале таблицы стилей можно разместить несколько директив @import.

**14.** Чтобы связать таблицу каскадных стилей с XML-документом, необходимо вставить в документ зарезервированную инструкцию по обработке xml-stylesheet. Эта инструкция по обработке имеет следующую обобщенную форму записи, где URLТаблСтил есть URL, задающий местонахождение файла таблицы стилей:

<?xml-stylesheet type="text/css" href="URLТаблСтил"?>

**15.** В таблицах каскадных стилей можно присваивать значения свойствам на нескольких различных уровнях. Основные уровни, на которых можно присваивать значение свойству - от высшего уровня приоритета к низшему.

1. Значение свойства в атрибуте STYLE для определенного элемента в XML-документе, имеет наивысший приоритет.

2. Если не установлено свойство в атрибуте STYLE, браузер использует значение свойства, объявленного в правиле CSS с контекстуальным селектором.

3. Если не объявлено значение определенного свойства в правиле, имеющем соответствующий контекстуальный селектор, браузер использует значение, объявленное в правиле с родовым селектором (т.е. селектором, который включает только имя элемента).

4. Если не объявлено значение для определенного свойства для элемента в родовом правиле, браузер использует установку свойства, объявленную для ближайшего элемента-предка (родителя, родителя родителя и т.д.).

5. Если таблица стилей не содержит установку свойства для какого-либо родительского элемента, браузер использует свою собственную установку. Такой установкой может быть значение по умолчанию, встроенное в браузер, либо значение, заданное пользователем браузера. Например, если не установлено значение для свойства font-family, браузер использует свое собственное значение этого свойства для отображения всех элементов. [В Internet Explorer 5 это шрифт Times New Roman, если только пользователь браузера не выберет другое семейство шрифтов, воспользовавшись командой Internet Options (Свойства обозревателя) из меню Tools (Сервис)].

6. Если для определенного свойства установлены конфликтующие значения на одном и том же уровне, В таком случае браузер использует последнюю установку, которую он обработал.

**16.** Свойство display управляет основным способом отображения текста элемента браузером. Вы можете назначить ему три ключевых слова CSS. Свойство display не наследуется дочерними элементами.

**17.** Возможные значения свойства:

· block. Браузер всегда помещает перевод строки перед и после текста элемента (который включает и текст, принадлежащий любым дочерним элементам). В результате текст элемента отображается в отдельном "блоке" с предшествующим текстом выше и последующим текстом ниже. Присвоение значения block позволяет форматировать текст с применением различных свойств обрамления к блоку текста, таких как поля, рамки и отступы. Элемент block похож на абзац в программе текстового процессора, который отделен пробелами от предшествующего и последующего текста, и для которого можно задавать отступы, рамки и т.д.;

· inline (по умолчанию). Браузер не вставляет перевод строки перед или после текста элемента (если только предшествующий текст достиг правой границы окна, и браузер должен перенести текст на следующую строку). Он будет вставлять перенос строки внутри текста элемента только с целью уместить текст в окне. Текст элемента, таким образом, может быть размещен в той же строке, что и предыдущий или последующий текст. Элемент inline аналогичен группе символов внутри абзаца в программе текстового процессора;

· none. Браузер не отображает элемент. Вы можете использовать эту установку для элементов, несущих информацию, которую вы не хотели бы помещать на экране.