Chapter 5

XML

5.1 Introduction of Markup Language

Markup language is concern about code and Tokens, which is arrange in a single document. which describe the data. In other words markup is used to organize Data. For Example HTML is a markup language which is used to develop web pages.

Example 5.1

```
        <html>

                <head>

                        <title> hello from HTML</title>

<head>

                <body><center><H1>HTML document<!H1></center>

This is an HTML document


</body>

</html>
```

In above example represents that HTML is using to interpret the important part of HTML pages such as header ,body and footer etc. This HTML page may be interpreted by Internet explorer, Google chrome and firefox. HTML markup may created using HTML tags which are predefined.

HTML tags directed to browser to represent data of document. In the same way XML is also used to store and represent data.

5.2 XML Introduction

XML language is developed by World Wide Web Consortium.XML is kind of markup language. XML Language is also used to represent and store the data as well as HTML. Data transformation at the internet is also accomplished by XML. XML Text is written in a such a way that human and machine can easily understand. It is self-descriptive language and a subset of standard generalized markup language (SGML).This kind of markup language is used in storage of large data.

5.2.1 Difference between HTML and XML

HTML	XML
HTML is an abbreviation for HyperText Markup Language.	XML stands for eXtensible Markup Language.
HTML was designed to display data and used to make attractive web pages.	XML used to transport and store data
HTML is a markup language itself.	XML provides a framework for defining markup languages.
HTML is case insensitive	XML is case sensitive.
HTML is used for designing a web -page to be rendered on the client side at client browser.	XML is used basically to transport data between the application and the database.
HTML has its own predefined tags.	custom tags can be defined and the tags are invented by the author of the XML document.
HTML is about displaying data,hence static.	XML is about carrying information, hence dynamic.

5.2.2 XML data Structure

XML is either storage of data or also capable to define structure of data. XML structure is also capable to work with Complex data.

For Example if we store large amount of account details in HTML, which increase changes of HTML errors.In XML we can create rules of syntax which instruct to definition of document.Due to this causes of errors in documents are almost removed. XML processor examines XML texts and two important factors are also examined by XML processor.which are as follows.

1. XML must be well formed.
2. XML must valid.

5.2.3 XML Features

1. Excellent for handling data with a complex structure or a typical data.
2. Text data description
3. Excellent for long-term data storage and data reusability
4. Human- and computer friendly format
5. Data described using markup language
6. Handles data in a tree structure having one and only one-root element

5.3 Development of XML document

5.3.1 XML declaration

XML document declaration is compulsory which represent as follows.

<? xml version =”1.0” encoding =”UTF-8”?>

In above declaration represent XML version and encoding is represent to useful characters encoding.

XML declaration helps to XML processor to parse XML documents. XML declaration is compulsory which is always written in first line of document.

5.3.2 Rules of XML Declaration

- If the XML declaration is present in the XML, it must be placed as the first line in the XML document.
- If the XML declaration is included, it must contain version number attribute.
- The Parameter names and values are case-sensitive.

- The names are always in lower case.
- Either single or double quotes may be used.
- The XML declaration has no closing tag i.e. </?xml>

5.3.3 XML Tags and elements

For XML elements three types of tags are used majorly.

1. Opening Tags
2. Closing tags
3. Empty Tag

1. Opening tags:

The beginning of every non-empty XML element is marked by a start tag. Following is an example of start tag –

Example: <localaddress>

2. Closing tags:

Every element that has a start tag should end with an end tag. Following is an example of end tag .

Example:</localaddress>

3. Empty Tags

The text that appears between start tag and end tag is called content. An element which has no content is termed as empty. An empty element can be represented in two ways as follows -

A start tag immediately followed by an end-tag as shown below -

<hr></hr>

A complete empty element tag is as shown below -

<hr />

Empty element tags may be used for any element which has no content.

5.3.4 XML Attributes

Various attributes are included in XML element.Attributes are created for various information of XML elements. XML attributes are consist of name and its value.

Syntax :-

```
        <element-name attribute1 attribute2 content…..>
        </element-name>
```

Example

```
    <student  firstname=”Mahesh”>
        <name>Sharma</name>
        <grade>A+</grade>
</student>
```

5.3.5 XML comments

XML comments are same as HTML comments. To understand XML documents comments are used. XML comments do not play any role in XML code execution.

Syntax –

Single line Comments

```
<!-----------Your Comment------------------->
```

Multi Line Comments

```
Comment <!------ ---------------------------->
```

Example –

```
<?XML Version = “1.0” encoding = “UTF-8”/>
<!—Student grades are updated monthly------->
<classlist>
```

```
    <student>
        <name> Ramesh</name>
        <grade>A+</grade>
    </student>
        <name>Girish</name>
        <grade>A-</grade>
    </student>
</classlist>
```

5.3.6 XML Entities

An XML documents are created by different storage units which are called entities. Each XML document has one entity which is called document entity.XML entity work for XML processor as starting point and placeholder.

There are different types of entities in XML but here we will learn only character entities.

They are introduced to avoid the ambiguity while using some symbols. For example, an ambiguity is observed when less than (<) or greater than (>) symbol is used with the angle tag (< >). Character entities are basically used to delimit tags in XML. Following is a list of pre-defined character entities from XML specification. These can be used to express characters without ambiguity.

- Ampersand - &amp;
- Single quote - &apos;
- Greater than - &gt;
- Less than - &lt;
- Double quote - &quot;

In this way main characters are represented throuth character entities.

Types of Character Entities

There are three types of character entities -

1. Predefined Character Entities

2. Numbered Character Entities

3. Named Character Entities

5.3.7 XML Features

1. XML separates data from HTML

2. XML simplifies data transport

3. XML simplifies data sharing

4. XML is not dependent on operating system.

5. XML can be used to create new internet languages

5.5 XML DTD

The purpose of a DTD is to define the legal building blocks of an XML document. It defines the document structure with a list of legal elements. XML DTD may be used by following two ways.

An XML DTD can be either specified inside the document, or it can be kept in a separate document and its reference will be passed into original XML document.

Syntax –

```
<!doctype element DTP identifier

[

        Declaration 1

        Declaration2

        -----------------

]>
```

In the above syntax,

- The DTD starts with <!DOCTYPE delimiter>
- An element tells the parser to parse the document from the specified root element.
- DTD identifier is an identifier for the document type definition, which may be the path to a file on the system or URL to a file on the internet. If the DTD is pointing to external path, it is called External Subset.
- The square brackets [ ] enclose an optional list of entity declarations called Internal Subset.

DTD declaration is two types, as follows

1. Internal DTD declaration: If the DTD is declared inside the XML file, it must be wrapped inside the <!DOCTYPE> definition.

```
<?xml version="1.0"?>

<!DOCTYPE note [

<!ELEMENT note (to,from,heading,body)>

<!ELEMENT to (#PCDATA)>

<!ELEMENT from (#PCDATA)>

<!ELEMENT heading (#PCDATA)>

<!ELEMENT body (#PCDATA)>

]>

<note>

<to>Mahesh</to>

<from>Jaipur</from>

<heading>Reminder</heading>

<body>Please remember it</body>
```

```
</note>
```

In above example.

- !DOCTYPE note defines that the root element of this document is note.
- !ELEMENT note defines that the note element must contain four elements: "to,from,heading,body".

2. External DTD Declaration

If the DTD is declared in an external file, the <!DOCTYPE> definition must contain a reference to the DTD file:

```
<?xml version="1.0"?>

<!DOCTYPE note SYSTEM "note.dtd">

<note>

    <to>Mahesh</to>

    <from>Jaipur</from>

    <heading>Reminder</heading>

    <body>Please remember it</body>

</note>
```

Note-dtd

```
<!ELEMENT note (to,from,heading,body)>

<!ELEMENT to (#PCDATA)>

<!ELEMENT from (#PCDATA)>

<!ELEMENT heading (#PCDATA)>

<!ELEMENT body (#PCDATA)>
```

In above example Node-dtd is an external DTD file whose reference is given in XML file. Different attributes of Note-dtd like Note ,to, from ,heading, body are used in original XML document.

5.6 DTD Schema

XML document schema types are dependent on data types of data. On the basis of data types XML document schema categorized in following two ways.

1. Simple type

2. Complex type

XML Schema is commonly known as XML Schema Definition (XSD). It is used to describe and validate the structure and the content of XML data.

XML Schema Structure

XML schema defines the elements, attributes and data types. It is similar to a database schema that describes the data in a database.

Syntax :-

You need to declare a schema in your XML document as follows –

```
<XS : Schema XMLns : xs = http://www.abc.com/2001/XML Schema>
```

Example:

```
<?XML version = “1.0” encoding = “UTF-8”?>

<XS: Schema XMLS:XS =”http://www.abc.com/2001/XML Schema>

<XS: element name= “contact”>

<xs : ComplexType>

        <xs: sequence>

                <xs: elements name=”name”  type=”xs: string”/>

                <xs: elements name=”Company”  type=”xs: string”/>

                <xs: elements name=”phone”  type=”xs: int”/>

                </xs: sequence>

</xs :ComplexType>
```

```
</xs: element>

</xs: Schema>
```

5.6.1 Difference between XML Schema and DTD

XML Schema	DTD
1. XML Schema is designed using XML	1. DTD is developed using SGML Syntax
2. Supports datatypes for elements and attributes.	2. DTD doesn't support data types.
3. XML may be replaced using DOM.	3. Replacement is not possible using DOM.

5.7 XML Validation

Validation is a process by which an XML document is validated. An XML document is said to be valid if its contents match with the elements, attributes and associated document type declaration (DTD), and if the document complies with the constraints expressed in it.

Validation is dealt in two ways by the XML parser. They are -

1 Well formed XML document

2 Valid XML document

1. Well formed XML Document

An XML document is said to be well-formed if it adheres to the following rules

- ♦ XML documents must have a root element
- ♦ XML elements must have a closing tag
- ♦ XML tags are case sensitive
- ♦ XML elements must be properly nested
- ♦ XML attribute values must be quoted

Example

Unformed XML Code:-

```
<email>
   <to>Mr. John
      <body> Hello there !</to>
      </body>
</email>
```

Well Formed XML Code

Example:-

```
        <email>
                <to>Mr. Garcia </to>
                <body> Hello there </body>
        </email>
```

2. Valid XML Document

If an XML document is well-formed and has an associated Document Type Declaration (DTD), then it is said to be a valid XML document.

5.8 XML Namespace

At the time of XML document creation, it may be possible that two elements have same name which increase chances of name conflicts.

Example

```
<?XML version = “1.0” encoding = “ISO-8859-15”?>
<html>
        <body>
                <P>Welcome</P>
                </body>
                <body>
        <height>6th </height>

        <Weight>155 </Weight>

        </body>

    </html>
```

In above example application <body> is used as per requirements which demonstrate human body but <body> is used two times in same document. As per rules of XML any element may not be used two times. To resolve this issue namespaces are developed.

Namespace is a mechanism by which element and attribute name can be assigned to a group. XML Namespaces provide a method to avoid element name conflicts.

Namespace declaration

A Namespace is declared using reserved attributes.

Example

```
<elements XMLs : name = “URL”>
```

- The Namespace starts with the keyword xmlns.
- The word name is the Namespace prefix.
- The URL is the Namespace identifier.

Example

```
<root>

    <h: table XMLs : h “http://www.abc.com/TR/>

    <h :tr>

    <h:td>Aries</h:td>

    <h:td>Bingo</h:td>

    </h:tr>
```

```
</h:table>
        <h: table XMLs : f "http://www.xyz.com/furniture/>
<h :name>
        <f:width>80 </width>
        <f:length>120</f:length>
</f:table>
</root>
```

In above example h and f are two different attributes of namespace which are added in single xml element (h: table).h and f are representing different URL whose definitions are different.

Default Namespace : It is such kind of namespace who does not use namespace prefix.

Default namespace may be declared using xmlns and without using prefix.

```
<xhtml xmlns=http://www.w3c.org/1999/xhtml>
```

Important points

1. XML is developed by wide web consortium (W3c). It is subset of SGML (Standard generally markup language).
2. XML is used for storage of document data and information exchange.
3. XML have three types of tags – opening tags, ending tags and empty tags
4. XML documents is created by various storage units which are called Entities
5. XML have such symbols which may not be used as contents or XML text therefore these symbols are used to define XML Text and contents.
6. XML DTD is used in creation of valid structure of XML.
7. XML Schema definition is used for basic XML structure, describe XML Contents and to validate the XML documents.
8. XML validation is the process to understand XML document structure.

Exercises

Objective Type Questions

1. XML Means?

a) X-Markup Language b) Extensible markup language

c) Extra Markup language d) Example markup language

2. XML data is described by

a) Through XML Description node b) By use of XSL

c) By Use of DTD d) None of the above

3. Which syntx is used for describe the version of XML

a) <XML version=”1.0”/> b) <?XML version=”1.0”?>

c) <?XML version=”1.0”/> d) <XML version=”1.0”?/>

4. DTD Means

a) Dynamic Type Definition b) Document Type definition

c) Do the Dance d) Direct Type Definition

5. XML data is used to store and _____________.

a) Data Exchange b) XML creation.

c) XML verification d) None of the above.

6. XML is similar to-

a) Java Script b) Cprogramming

c) CSS d) HTML

7. XML does not work for-

a) Information sharing b) To Store data

c) Exchanges of information. d) Structure of Information

8. XML valid syntax is_____

a) Mature b) Well Parameterized

c) Well Formed d) None of the above

9. XML documents is used to validate-

a) CGG b) DTD

c) Jquery d) Parses

10. XML developed for-

a) By W3C b) By W2C

c) By HTML d) None of the above

Short Type Questions.

1. How to define syntax of XML version.
2. Describe briefly XML and HTML.
3. What is mean of valid XML document?
4. What is DTD?
5. What is use of XML attribute in XML?

Essay Type Questions.

1. Describe XML main work and advantages.
2. Describe DTD with example.
3. Describe XML Schema.

Answer Key

1. B	2. C	3. B	4. B	5. A
6. D	7. C	8. C	9. B	10 A