ELECTIVE -I

XML & WEB SERVICES

- XML - Extensible Markup Language → text-based.

↓ derived from

Standard Generalised Markup Language (SGML)

- XML is a software & hardware independent tool for storing data.
- (a) markup language
  (b) store and transport data
  (c) self descriptive
  (d) W3C Recommendation.

- CHARACTERISTICS:

✱ XML is extensible → create own self descriptive tags.
✱ XML carries data, does not present it.
✱ XML is a public standard. → Developed by (W3C) and is open standard.

XML USAGE:

- Can work behind the scene to simplify the creation of HTML documents.
- Exchange information b/w organisation & systems.
- Used for offloading and reloading of databases.
- Used to encode any structured information.

- Can be used to store and arrange the data.
- can easily merge with style sheets
- virtual representation of any data is possible.

The Difference b/w

XML	HTML
• was designed to describe data with focus on what data is	Designed to display data, with focus on how data looks.
• Stores and transport data	Present and display data
• Has own tags	Predefined tags
• Case sensitive	Not case sensitive
• Documents form tree structures.	

Why XML important?

Provides rules to define any data.

COMPONENTS: (Save as .xml extension)

- XML document
- XML declaration
- XML elements < Test, attributes, other elements
- XML content

XML DELIMITER CHARACTER :

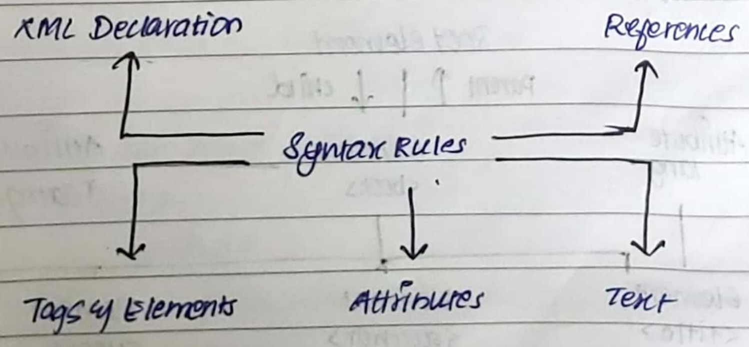
< - - Start of an XML markup lang
> - End of an XML markup lang
& - Start of an XML entity
; - End of an XML entity

XML SYNTAX :

XML document :

```
<?xml version = "1.0"?>
<contact_info>
        <name> Rajesh </name>
        <company> TCS </company>
        <phone> 93333 32354 </phone>
</contact_info>
```

Syntax rules :
→ Tags are case sensitive
→ All XML elements must have closing tag
→ Must be propesly nested
→ Must have root element (XML doc.)



XML Declaration:
<?xml version = "1.0" encoding = "UTF-8"?>

Tags & Elements : <element>

Root element : Can have only one root element.

```
<root>
      <x>      </x>
      <y>      </y>
</root>
```

- XML elements are case sensitive.

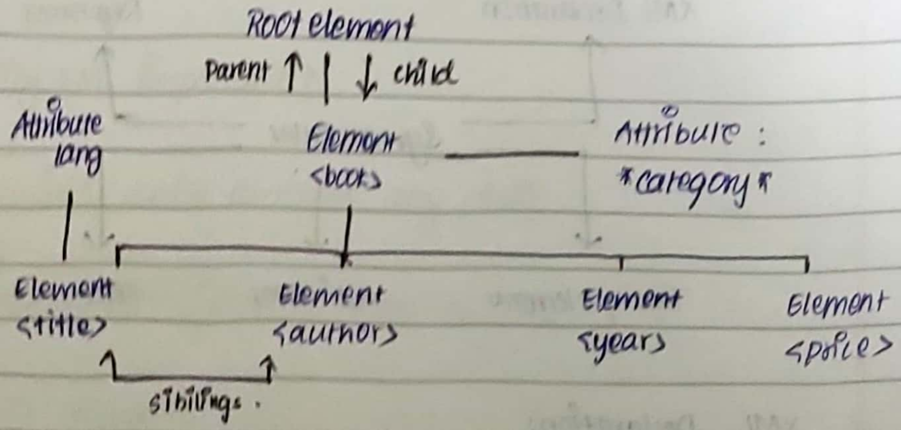
  <contact info> ⟷ <Contact info>

  are different

XML ATTRIBUTES :

- An attribute specifies a single property for the element using a name/value pair.
- An XML element can have one or more attribute
- Provide additional Information about an element.
- Provide info that is not a part of data
- Must be quoted.

XML TREE :

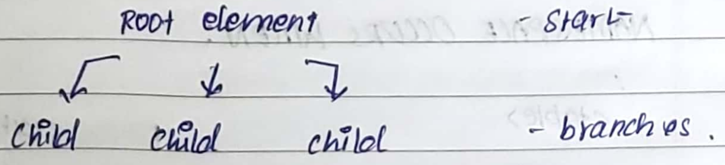


EXAMPLE:

```
<?xml version = "1.0" encoding = "UIF-8"?>
<bookstore>
      <book category ="cooking">
          <title lang ="en"> Everyday Italian </title>
          <author> Giada De Laurentis </author>
           <year> 2005 </year>
          <price > 30.00 </price>
      </book>
      <book category = "children">
      <title lang = "en"> Harry Potter </title>
       <author> J.K.Rowling </author>
       <year> 2005 </year>
       <price> 29.99 </price>
      </book>
</bookstore>
```

XML documents are element trees

XML tree



All elements can have subchild.

```
      <root>
            <child>
                  <subchild>        </subchild>
            </child>
      </root>
```

XML NAMESPACES:

DEFAULT NAMESPACES: Applies to all elements without a prefix.

```
<root xmlns = "https://www.w3.org/TR/html4/">
<table>
  <tr> ~~Apples </tr~~
    <td> Apples </td>
    <td> Bananas </td>
  </tr>
</table>
```

INHERITED NAMESPACES: Child element Enherit the namespace of their parent unless overriden.

```
<root>
  <child> content </child>
 </root>
```

NAMESPACE OCCURS WHEN:

```
<table>                          |   <table>
    <tr>                         |     <name> African coffee Table </name>
     <td> Apples </td>           |     <width> 80 </width>
     <td> Bananas </td>          |     <length> 120 </length>
   </tr>                         |   </table>
<table>                          |
```

If there XML fragments are added together there would be a name conflict. Both contain <table> element but with different content//

SOLVING NAME CONFLICT USING A PREFIX:

```
<h:table>
 <h:tr>
    <h:td> Apples </h:td>
    <h:td> Bananas </h:td>
 </h:tr>
</h:table>
```

```
<f:table>
   <f:name>African Coff. Table </f:name
  <f:width> 80 </f:width>
  <f:length>120 </f:length>
</f:table>
```

XML VALIDATOR. used for syntax checks of our XML doc.

Well formed :

(a) Have proper syntax.

- Must have root element
- Must have closing tag
- tags are case sensitive
- elements must be properly nested.
- attribute value must be quoted.

VALID XML DOCUMENTS:

Well formed doc is not same as Valid Doc.
But a valid XML document must be well formed.
In addition it must conform to a document type defn.

DTD
The Original - Document Type Definition.

XML schema.
An XML based alternative to DTD.

XML DTD :

- Basic building blocks of XML .
- Purpose : Define the structure of an XML document .
  Defines the structure with a list of legal elements .
- DTD is a vaild XML document + well formed .

```
<!DOCTYPE book                                  - root element of doc is book
[
<!ELEMENT book                                  - defines the book elements
                                                  must contain below elem.
  (title,author,price)> <!ELEMENT               - !Element - title
     title (#PCDATA)> <!ELEMENT author          - !Element author .
      (#PCDATA)> <!ELEMENM price                - !Element price .
      (#PCDATA)> ]>
```

2 TYPES OF DTD :

↓ Internal / Embedded DTD

↓ External DTD .

When to use a DTD/schema ?

- With a DTD we can verify that the data you receive from the outside world is valid .
- Can use DTD to verify our own data .

XML SCHEMA :

- Describes the structure of an XML doc . just like DTD
- XML schema defines the elements, attributes & datatypes .
- Supports namespaces .

Document that describes some rules or limits on structure of on XML file .

Purpose: Define a structure

TYPES:

simple: Define constraints on text values	complex: Define structure with nested elements.
Global Declaration	Local Type Declaration.
Can be reused throughout the schema	Defined within a specific element/attribute.

Schema Management: Organising and maintaining XML schemas

Instance Document: XML documents that address/adhere to the schema.

Benefits of using XML?

- Support interbusiness transactions
- Maintain data Integrity.
- Improve search efficiency
- Design flexible applications.

XSLT INTRODUCTION:

XSL (Extensible Stylesheet Language) is a styling language for XML.

XSLT stands for XSL Transformations.

HTML	XML
• Predefined tags	• No predefined tags
• CSS - stylesheet for HTML.	• XSL = stylesheet for XML

What is XSLT?

- Stands for XSL Transformations
- XSLT is the most important part of XSL.
- transforms XML document into another XML document
- XSLT uses XPath to navigate.
- Is a W3C Recommendation,

XML DOM PARSER:

- XML parser is s/w that can process or read XML documents to extract the data within them.

  Modern data formats: JSON & XML

- XML DOM defines a standard way for accessing & manipulating XML documents. It presents a XML doc as tree structure.

THE XML DOM:

All XML elements can be accessed through XML DOM.

XML DOM is

- A standard object model for XML
- A standard programming Interface for XML
- Platform and language - Independent
- A W3C standard.

XML DOM PROPERTIES:

x.modelName
x.nodevalue
x.parentNode
x.childNodes
x.attributes.

XML DOM METHODS:

- x.getElementByTagName(name) - getall Elements
- x.appendChild(node) - Insert childnode
- x.removeChild(node) - remove childnode