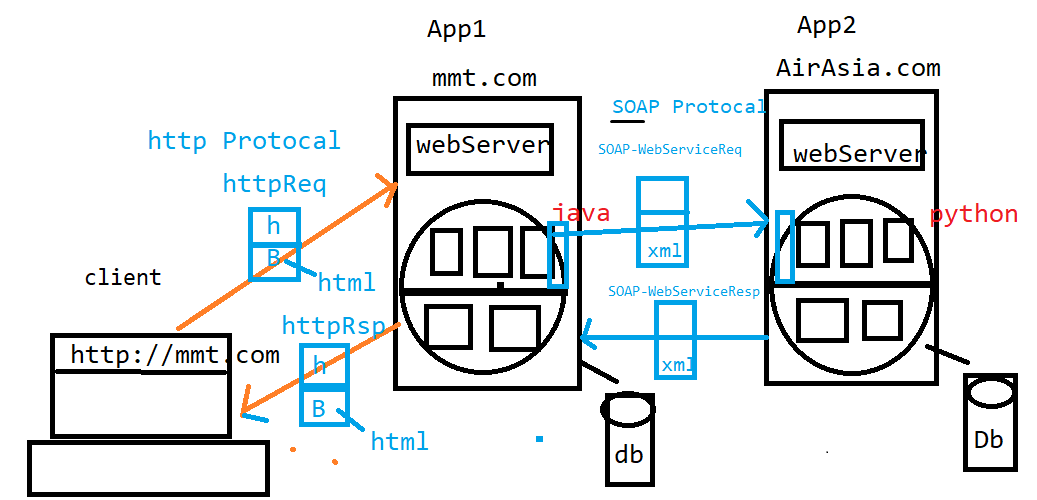
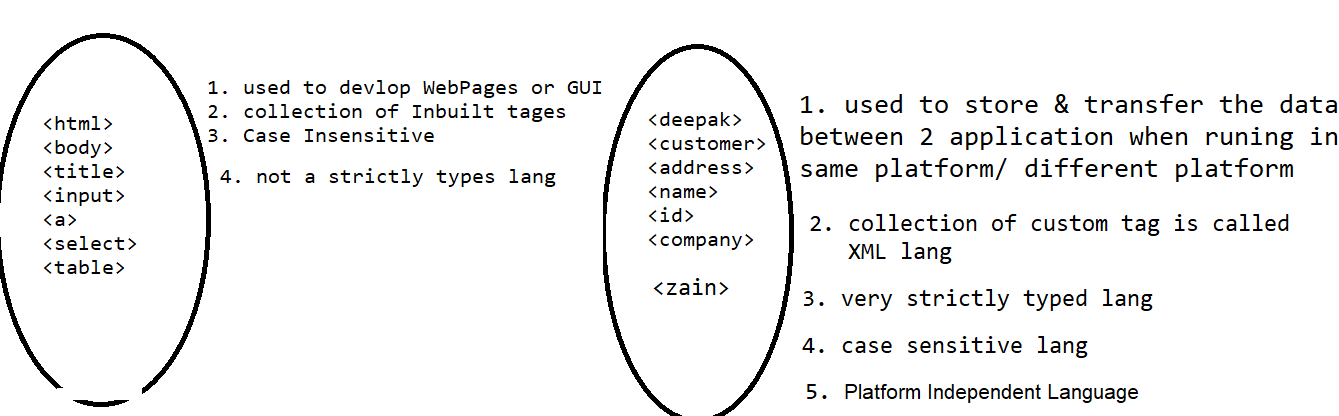
Extensible Mark-up Language (XML)



* XML is “Markup Language & Platform Independent Language" which helps to store and transport data.
* Different Applications which are developed using different technologies or same technologies can Transfer the Data among themselves with the help of XML.
* As the name implies it's an extension of HTML & hence XML looks similar to HTML but it’s not a HTML
* XML has User-defend Tags.
* XML tags are also called as “elements”.

HTML vs XML



XML Syntax

* XML is "Strictly Typed" Language hence
* Case-sensitive
* They cannot contain spaces
* For every element data, “data-type” should be defined,
* Every opening element should have corresponding closing element and also XML elements must be properly nested/closed
* They must start with a letter or underscore
* They are cannot start with the letters like xml or XML or Xml etc.
* MIME type (Content Type) of XML is "application/xml“
* File extension of XML is ".xml"

Rule 1: XML Prolog :

Below line is called as "XML prolog", which is optional. If it exists, it must be the First Line of XML

EG :

<?xml encoding=’UTF-8’ verion=1.1 scema=http://testing.xom…>

Rule 2: Xml Comments

* The syntax of XML comment is similar to that of HTML

Ex:

<!--This is a comment -->

Rule 3 : XML Structure

* Like HTML, XML follows a Tree Structure
* An XML tree starts at a "root element" and branches from

"root element" will have "child elements“

* XML Consists of "Only One" root element

which is parent of all other elements child elements

child can have "sub elements / child elements

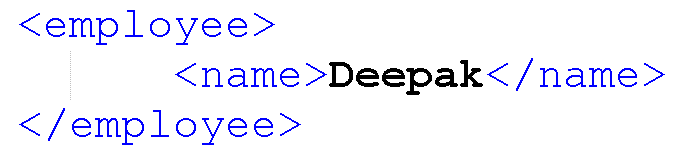
<root>

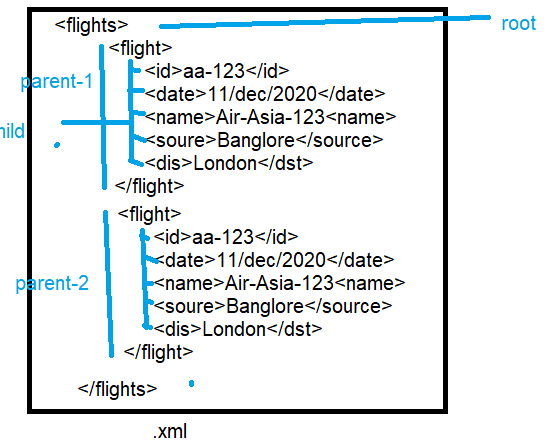
<child>

<subchild>.....</subchild>

</child>

</root>





Rule 4 : XML-Entity References

* Some characters have a special meaning in XML.
* If you place a character like "<" inside an XML element,
* it will generate an error because it represents the start of a new element

Ex:<message>salary<1000</message>

* To avoid this error, we can replace the "<" character withan "entity reference" as shown below

<message>salary &lt;1000</message>

There are 5 pre-defined entity references in XML:

&lt; < less than

&gt; > greater than

&amp; & ampersand

&apos; ‘ apostrophe

&quot; " quotation mar

Rule 5 : XML PCDATA: Parsed Character Data

* Text between start-element and end-element is called as PCDATA which will be examined by the parser

Example:-

<employee>Ram</employee>

The string "Ram" is considered as PCDATA

PCData : will be always consider as String

Rule 6 : XML-CDATA: Character Data

* If XML data contain many special characters, it is good practice to replace all of them. Instead we can use "CDATA (character data) section“

Example: -

<employee>emplyeSal >1000 & sal < 10 </employee> : Wrong

<![CDATA[<emplyeSal >1000 & sal < 10 </employee>]]>: Correct

Rule 7 : XML-Elements & Attributes

* XML element is everything from (including) the element's start tag to (including) the element's end tag
* An element can contain:

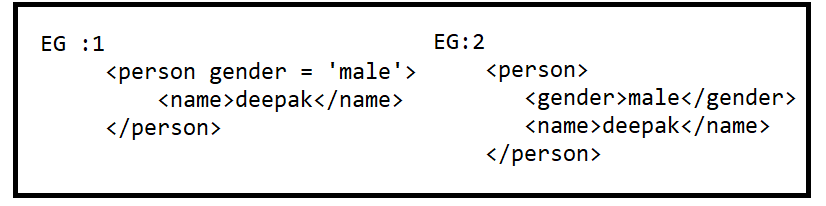
1. data

2. Attributes

3. other elements

4. All of the above

**🡺XML-Attributes**

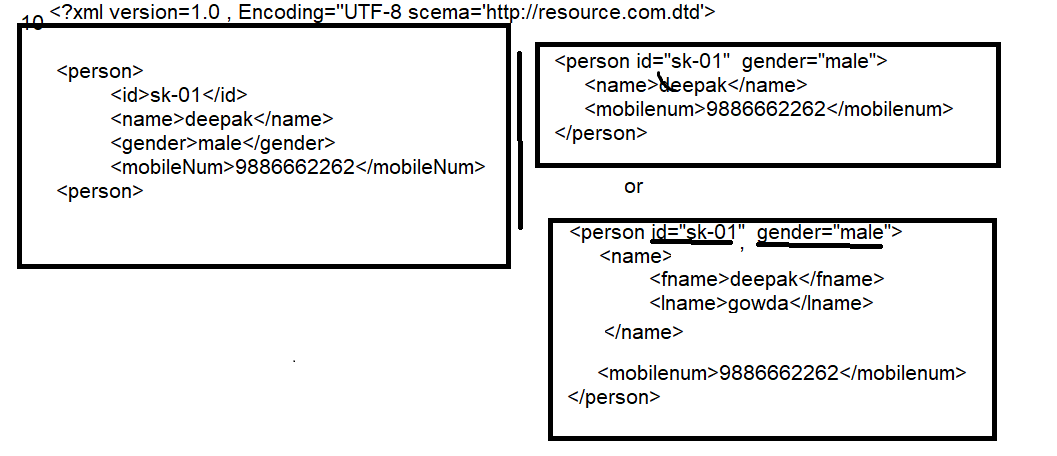
* Like HTML, XML elements can also have attributes, but **attributes can’t easily expandable like elements**
* XML Attributes Must be Quoted either single or double quotes can be use
* 

Example 1 gender is an attribute

Example 2 gender is an element

🡺XML Elements

Will go Elements when data extendable



Rule 8 : XML Schema's

* W.K.T XML helps us to store & transfer the data
* When sending data from one application to an another, it is essential that both applications have the same "expectations / agreement" about the content/data
* for example, A date like "03-11-2004" -in some countries, be interpreted as 3rd November and -in other countries as 11th March

There are two ways to define a Schema for XML

1.Document Type Definition (DTD)

2.XML Schema Definition (XSD)

XML-PARSAR(JAXB)

* JAXB is a Java API helps us to convert Java Object to XML & vice-versa
* The Process of converting Java Object to XML is called as "Marshalling" OR "Serialization“
* The Process of converting XML to Java Object is called as "Unmarshalling" OR "Deserialization

