**ASSIGNMENT NO. 6**

**Aim**

Create XML, XML schemas , DTD for any database application and implement min 10 queries using XQuery FLOWR expression and XPath

**Objective**

Create XML Schema, XML document and implement Queries using XQuery.

**Theory**

**XML**

XML is a software- and hardware-independent tool for storing and transporting data.

XML stands for EXtensible Markup Language

XML is a markup language much like HTML

XML was designed to store and transport data

XML was designed to be self-descriptive

XML is a W3C Recommendation

**XML vs HTML**

XML and HTML were designed with different goals:

XML was designed to carry data - with focus on what data is

HTML was designed to display data - with focus on how data looks

XML tags are not predefined like HTML tags are

The XML language has no predefined tags.

The tags in the example above (like <to> and <from>) are not defined in any XML standard. These tags are "invented" by the author of the XML document.

HTML works with predefined tags like <p>, <h1>, <table>, etc.

With XML, the author must define both the tags and the document structure.

**XML is Extensible**

Most XML applications will work as expected even if new data is added (or removed).

Imagine an application designed to display the original version of note.xml (<to> <from> <heading> <data>).

Then imagine a newer version of note.xml with added <date> and <hour> elements, and a removed <heading>.

The way XML is constructed, older version of the application can still work:

**XML simplifies things :**

* It simplifies data sharing
* It simplifies data transport
* It simplifies platform changes

It simplifies data availability

**XML Schema (Order.xsd)**

<xs:schema xmlns:xs='http://www.w3.org/2001/XMLSchema'>  
  
 <xs:element name="Customer">  
 <xs:complexType>  
 <xs:sequence>  
 <xs:element ref="orders" minOccurs='1' maxOccurs='unbounded'/>  
 </xs:sequence>  
 </xs:complexType>  
</xs:element>  
  
 <xs:element name="orders">  
 <xs:complexType>  
 <xs:sequence>  
 <xs:element ref="custid" minOccurs='0' maxOccurs='1'/>  
 <xs:element ref="custname" minOccurs='0' maxOccurs='1'/>  
 <xs:element ref="itemname" minOccurs='0' maxOccurs='1'/>  
 <xs:element ref="amount" minOccurs='0' maxOccurs='1'/>  
 </xs:sequence>  
 </xs:complexType>  
 </xs:element>  
  
 <xs:element name="custid" type='xs:string'/>  
 <xs:element name="custname" type='xs:string'/>  
 <xs:element name="itemname" type='xs:string'/>  
 <xs:element name="amount" type='xs:integer'/>  
   
</xs:schema>

**XML Document (Order.xml)**

<?xml version="1.0" encoding="UTF-8"?>  
<Customer xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation='Order.xsd'>  
 <orders>  
 <custid>c101</custid>  
 <custname>XYZ</custname>  
 <itemname>TV</itemname>  
 <amount>10000</amount>  
 </orders>  
   
 <orders>  
 <custid>c102</custid>  
 <custname>ABC</custname>  
 <itemname>Fridge</itemname>  
 <amount>25000</amount>  
 </orders>  
   
 <orders>  
 <custid>c103</custid>  
 <custname>ABC</custname>  
 <itemname>WM</itemname>  
 <amount>5000</amount>  
 </orders>  
 </Customer>

**XML Schema (Customer.xsd)**

<?xml version="1.0" encoding="UTF-8"?>  
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">   
 <xs:element name="Cust">  
 <xs:complexType>  
 <xs:sequence>  
 <xs:element ref="Custinfo" minOccurs='1' maxOccurs='unbounded'/>  
 </xs:sequence>  
 </xs:complexType>  
 </xs:element>  
   
 <xs:element name="Custinfo">  
 <xs:complexType>  
 <xs:sequence>  
 <xs:element ref="custid" minOccurs='0' maxOccurs='1'/>  
 <xs:element ref="custname" minOccurs='0' maxOccurs='1'/>  
 <xs:element ref="custaddr" minOccurs='0' maxOccurs='1'/>  
 </xs:sequence>  
 </xs:complexType>  
 </xs:element>  
   
 <xs:element name="custid" type='xs:string'/>  
 <xs:element name="custname" type='xs:string'/>  
 <xs:element name="custaddr" type='xs:string'/>  
 <xs:element name="amount" type='xs:integer'/>  
   
</xs:schema>

**XML Schema (Customer.xml)**

<?xml version="1.0" encoding="UTF-8"?>  
<Cust xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation='Customer.xsd'>  
 <Custinfo>  
 <custid>c101</custid>  
 <custname>XYZ</custname>  
 <custaddr>Pune</custaddr>  
 </Custinfo>  
   
 <Custinfo>  
 <custid>c104</custid>  
 <custname>PQR</custname>  
 <custaddr>Mumbai</custaddr>  
 </Custinfo>  
</Cust>

**XQuery**

1. **Get information of all customers**

for $x in *doc*("Order.xml") /Customer/orders  
return $x

1. **Get names of all customers**

for $x in *doc*("Order.xml") /Customer/orders  
return $x/custname

1. **Get information of customer c101**

for $x in *doc*("Order.xml") /Customer/orders  
where $x/custid = "c101"  
return $x

1. **Find information of all customers who purchased item whose cost is greater than 25000**

for $x in *doc*("Order.xml") /Customer/orders  
where $x/amount >25000  
return $x

1. **Sort all customer records by amount**

for $x in *doc*("Order.xml") /Customer/orders  
order by $x/amount  
return $x

1. **Display first four characters of all items**

for $x in *doc*("Order.xml") /Customer/orders  
let $iname := (*substring*($x/itemname,1,2))  
order by $x/custname  
return $iname

1. **Joining of two XML documents**

for $x in *doc*("Customer.xml") /Cust/Custinfo, $y in *doc*("Order.xml") /Customer/orders  
where $x/custid=$y/custid  
return $x