1)

**AIM** : To write a XSD for the given xml

**DESCRIPTION** :

An XML Schema describes the structure of an XML document.

The XML Schema language is also referred to as XML Schema Definition (XSD).

One of the greatest strength of XML Schemas is the support for data types.

It is easier to describe allowable document content

It is easier to validate the correctness of data

It is easier to define data facets (restrictions on data)

It is easier to define data patterns (data formats)

It is easier to convert data between different data types

The <schema> element is the root element of every XML Schema.

XML Schemas define the elements of your XML files.

A simple element is an XML element that contains only text. It cannot contain any other elements or attributes.

A complex element is an XML element that contains other elements and/or attributes.

There are four kinds of complex elements:

empty elements

elements that contain only other elements

elements that contain only text

elements that contain both other elements and text

**PROGRAM CODE** :

<?xml version="1.0" encoding="ISO-8859-1"?>

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

<xs:element name="stock">

<xs:complexType>

<xs:sequence>

<xs:element name="new-car" minOccurs="0" maxOccurs="unbounded">

<xs:complexType>

<xs:sequence>

<xs:element name="model"/>

<xs:element name="price"></xs:element>

</xs:sequence>

</xs:complexType>

</xs:element>

<xs:element name="used-car" minOccurs="0" maxOccurs="unbounded">

<xs:complexType>

<xs:sequence>

<xs:element name="model" type="xs:string"/>

<xs:element name="price" type="xs:int"/>

<xs:element name="mileage" type="xs:int"/>

<xs:element name="condition" type="xs:int" minOccurs="0" maxOccurs="1"/>

</xs:sequence>

</xs:complexType>

</xs:element>

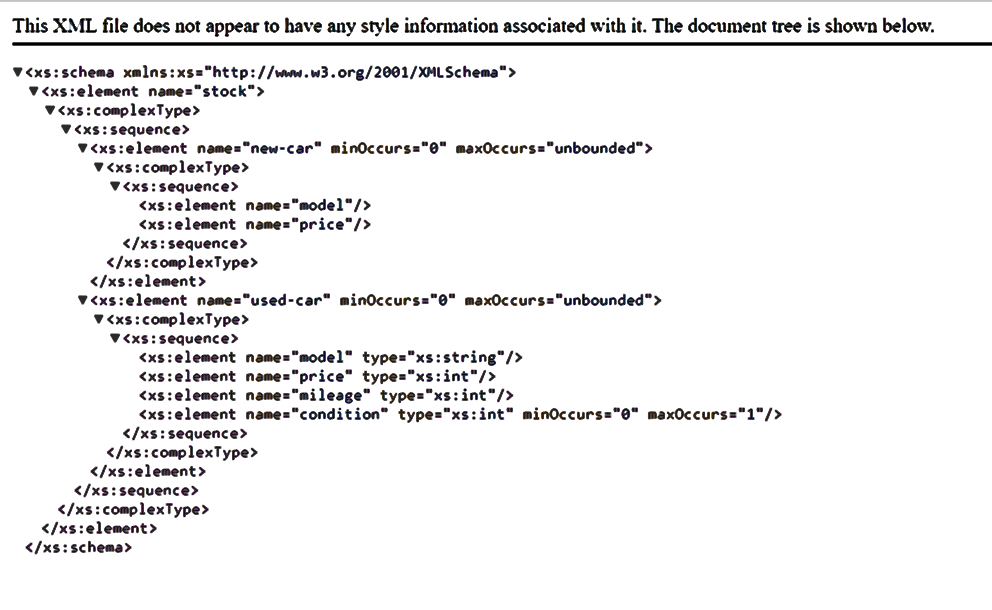
</xs:sequence>

</xs:complexType>

</xs:element>

</xs:schema>

**OUTPUT** :



**RESULT ANALYSIS** :

Learnt about the use of XML schema and validate XML against XSD.

Learnt about the types of xml elements such as Simple and Complex and how to use it in validation of xml document.

2)

**AIM** :

Design XSD and XML for the bibliography of a book- Refer PPT Pageno-29

**DESCRIPTION** :

The syntax for defining a simple element is:

<xs:element name="xxx" type="yyy"/>

Default and Fixed Values for Simple Elements

Simple elements may have a default value OR a fixed value specified.

A default value is automatically assigned to the element when no other value is specified.

<xs:element name="color" type="xs:string" default="red"/>

A fixed value is also automatically assigned to the element, and you cannot specify another value.

In the following example the fixed value is "red":

<xs:element name="color" type="xs:string" fixed="red"/>

The syntax for defining an attribute is:

<xs:attribute name="xxx" type="yyy"/>

Attributes may have a default value OR a fixed value specified.

A default value is automatically assigned to the attribute when no other value is specified.

In the following example the default value is "EN":

<xs:attribute name="lang" type="xs:string" default="EN"/>

A fixed value is also automatically assigned to the attribute, and you cannot specify another value.

In the following example the fixed value is "EN":

<xs:attribute name="lang" type="xs:string" fixed="EN"/>

Attributes are optional by default. To specify that the attribute is required, use the "use" attribute:

<xs:attribute name="lang" type="xs:string" use="required"/>

**PROGRAM CODE**:

<?xml version="1.0" encoding="ISO-8859-1"?>

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

<xs:element name="bibliography">

<xs:complexType>

<xs:sequence>

<xs:element name="book">

<xs:complexType>

<xs:sequence>

<xs:element name="title" type="xs:string"/>

<xs:element name="author" type="xs:string"/>

<xs:element name="publisher" type="xs:string"/>

<xs:element name="year" type="xs:int"/>

<xs:element name="section">

<xs:complexType>

<xs:sequence>

<xs:element name="title" type="xs:string"/>

<xs:element name="content">

<xs:complexType>

<xs:sequence>

<xs:element name="i"/>

</xs:sequence>

</xs:complexType>

</xs:element>

</xs:sequence>

</xs:complexType>

</xs:element>

</xs:sequence>

<xs:attribute name="ISBN" type="xs:ID"/>

<xs:attribute name="price" type="xs:int"/>

</xs:complexType>

</xs:element>

</xs:sequence>

</xs:complexType>

</xs:element>

</xs:schema>

**OUTPUT** :



**RESULT ANALYSIS**:

Learnt about the use of XML schema and validate XML against XSD.

Learnt about the types of xml elements such as Simple and Complex and how to use it in validation of xml document. Also, understood about the declaration of attributes and restrictions on values of XML elements.

3)

**AIM** : To write a DTD and XML for a collection of receipies.

**DESCRIPTION** :

Restrictions are used to define acceptable values for XML elements or attributes. Restrictions on XML elements are called facets.

A complex element is an XML element that contains other elements and/or attributes.

There are four kinds of complex elements:

empty elements

elements that contain only other elements

elements that contain only text

elements that contain both other elements and text

Syntax:

Empty Element

<xs:element name="product">

<xs:complexType>

<xs:complexContent>

<xs:restriction base="xs:integer">

<xs:attribute name="prodid" type="xs:positiveInteger"/>

</xs:restriction>

</xs:complexContent>

</xs:complexType>

</xs:element>

Elements Only

<xs:element name="person">

<xs:complexType>

<xs:sequence>

<xs:element name="firstname" type="xs:string"/>

<xs:element name="lastname" type="xs:string"/>

</xs:sequence>

</xs:complexType>

</xs:element>

Text-Only

<xs:element name="shoesize">

<xs:complexType>

<xs:simpleContent>

<xs:extension base="xs:integer">

<xs:attribute name="country" type="xs:string" />

</xs:extension>

</xs:simpleContent>

</xs:complexType>

</xs:element>

Mixed

<xs:element name="letter">

<xs:complexType mixed="true">

<xs:sequence>

<xs:element name="name" type="xs:string"/>

<xs:element name="orderid" type="xs:positiveInteger"/>

<xs:element name="shipdate" type="xs:date"/>

</xs:sequence>

</xs:complexType>

</xs:element>

**PROGRAM CODE** :

<?xml version="1.0" encoding="ISO-8859-1"?>

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

<xs:element name="collection">

<xs:complexType>

<xs:sequence>

<xs:element name="description" type="xs:string"/>

<xs:element name="recipe">

<xs:complexType>

<xs:sequence>

<xs:element name="title" type="xs:string"/>

<xs:element name="ingredient">

<xs:complexType>

<xs:attribute name="name" type="xs:string"/>

<xs:attribute name="amount" type="xs:int"/>

<xs:attribute name="unit" type="xs:int"/>

</xs:complexType>

</xs:element>

<xs:element name="preparation">

<xs:complexType>

<xs:all>

<xs:element name="step" type="xs:string"/>

</xs:all>

</xs:complexType>

</xs:element>

<xs:element name="comment" type="xs:string"/>

<xs:element name="nutrtion">

<xs:complexType>

<xs:attribute name='protein' type="xs:int"/>

<xs:attribute name='carbohydrates' type="xs:int"/>

<xs:attribute name='fat' type="xs:int"/>

<xs:attribute name='alcohol' type="xs:int"/>

<xs:attribute name='calories' type="xs:int"/>

</xs:complexType>

</xs:element>

</xs:sequence>

</xs:complexType>

</xs:element>

</xs:sequence>

</xs:complexType>

</xs:element>

</xs:schema>

**OUTPUT** :



**RESULT ANALYSIS** :

Learnt about the use of XML schema and validate XML against XSD.

Learnt about the types of xml elements such as Simple and Complex and how to use it in validation of xml document. Also, understood about the declaration of attributes and restrictions on values of XML elements.