Name : Vaibhav Tomar

Register Number : 2447157

Class : MCA - A

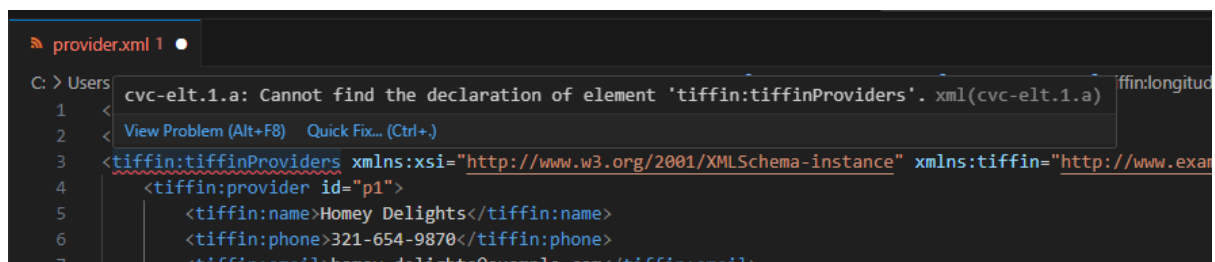**Course : Web Stack Development**

Date of submission : 11 August 2024

Lab Exercise 5

<u>Summary report of transformation and validation process :</u>

Issues encountered :

1. Root element not defined
    Reason – namespace not defined in xsi:schemaLocation

provider.xml

```xml
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet href="provider.xsl" type="text/xsl" ?>
<tiffin:tiffinProviders xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:tiffin="http://www.example.com/provider" xsi:schemaLocation="http://www.example.com/provider
provider.xsd">

<tiffin:provider id="p1">
    <tiffin:name>Healthy Tiffin Service</tiffin:name>
    <tiffin:phone>123-456-7890</tiffin:phone>
    <tiffin:email>contact@healthytiffin.com</tiffin:email>
    <tiffin:address>123 Tiffin Street, Food City</tiffin:address>
    <tiffin:price>15.99</tiffin:price>
    <tiffin:rating>4.5</tiffin:rating>
    <tiffin:longitude>45.123456</tiffin:longitude>
    <tiffin:latitude>-75.123456</tiffin:latitude>
    <tiffin:menu>
        <tiffin:menuName>Vegetarian Special</tiffin:menuName>
        <tiffin:description>A healthy vegetarian meal plan</tiffin:description>
        <tiffin:item>
            <tiffin:itemName>Paneer Butter Masala</tiffin:itemName>
            <tiffin:itemDiet>Vegetarian</tiffin:itemDiet>
            <tiffin:itemPrice>8.99</tiffin:itemPrice>
        </tiffin:item>
        <tiffin:item>
            <tiffin:itemName>Mixed Veg Curry</tiffin:itemName>
            <tiffin:itemDiet>Vegetarian</tiffin:itemDiet>
            <tiffin:itemPrice>7.99</tiffin:itemPrice>
        </tiffin:item>
    </tiffin:menu>
</tiffin:provider>

<tiffin:provider id="p2">
    <tiffin:name>Quick Bites Tiffin</tiffin:name>
```

```
        <tiffin:phone>987-654-3210</tiffin:phone>
        <tiffin:email>info@quickbitestiffin.com</tiffin:email>
        <tiffin:address>456 Snack Lane, Fastfood Town</tiffin:address>
        <tiffin:price>12.50</tiffin:price>
        <tiffin:rating>4.2</tiffin:rating>
        <tiffin:longitude>46.654321</tiffin:longitude>
        <tiffin:latitude>-76.654321</tiffin:latitude>
        <tiffin:menu>
            <tiffin:menuName>Non-Veg Combo</tiffin:menuName>
            <tiffin:description>Delicious non-vegetarian dishes</tiffin:description>
            <tiffin:item>
                <tiffin:itemName>Chicken Curry</tiffin:itemName>
                <tiffin:itemDiet>Non-Vegetarian</tiffin:itemDiet>
                <tiffin:itemPrice>9.99</tiffin:itemPrice>
            </tiffin:item>
            <tiffin:item>
                <tiffin:itemName>Fish Fry</tiffin:itemName>
                <tiffin:itemDiet>Non-Vegetarian</tiffin:itemDiet>
                <tiffin:itemPrice>10.50</tiffin:itemPrice>
            </tiffin:item>
        </tiffin:menu>
    </tiffin:provider>
</tiffin:provider>
```
…..6 more similar records of <tiffin:provider></tiffin:provider>

provider.xsd

```xml
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" elementFormDefault="qualified" attributeFormDefault="unqualified"
targetNamespace="http://www.example.com/provider" xmlns:tiffin="http://www.example.com/provider" >

    <!-- Define a pattern for phone numbers -->
    <xs:simpleType name="phoneType">
        <xs:restriction base="xs:string">
            <xs:pattern value="\d{3}-\d{3}-\d{4}"/>
        </xs:restriction>
    </xs:simpleType>

    <!-- Define a pattern for email addresses -->
    <xs:simpleType name="emailType">
        <xs:restriction base="xs:string">
            <xs:pattern value="[a-zA-Z0-9._%+-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}"/>
        </xs:restriction>
    </xs:simpleType>

    <!-- Define a restriction for price -->
    <xs:simpleType name="priceType">
        <xs:restriction base="xs:decimal">
            <xs:minExclusive value="0"/>
            <xs:maxExclusive value="200"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:complexType name="itemType">
        <xs:sequence>
            <xs:element name="itemName" type="xs:string"/>
            <xs:element name="itemDiet" type="xs:string"/>
            <xs:element name="itemPrice" type="xs:decimal"/>
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="menuType">
        <xs:sequence>
```

```
            <xs:element name="menuName"/>
            <xs:element name="description"/>
            <xs:element name="item" type="tiffin:itemType" maxOccurs="unbounded"/>
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="tiffinProvidersType">
            <xs:sequence>
                <xs:element name="provider" maxOccurs="unbounded">
                <xs:complexType>
                    <xs:sequence>
                        <xs:element name="name" type="xs:string"/>
                        <xs:element name="phone" type="tiffin:phoneType"/>
                        <xs:element name="email" type="tiffin:emailType"/>
                        <xs:element name="address" type="xs:string"/>
                        <xs:element name="price" type="tiffin:priceType"/>
                        <xs:element name="rating" type="xs:decimal"/>
                        <xs:element name="longitude" type="xs:decimal"/>
                        <xs:element name="latitude" type="xs:decimal"/>
                        <xs:element name="menu" type="tiffin:menuType"/>
                    </xs:sequence>
                    <xs:attribute name="id" type="xs:ID" use="required"/>
                </xs:complexType>
            </xs:element>
            </xs:sequence>
    </xs:complexType>
    <xs:element name="tiffinProviders" type="tiffin:tiffinProvidersType"/>
</xs:schema>
```

Cases where xml violated XSD defined rules and shows errors

1. When phone number has more than 10 numbers xml gives error

2. When email does not follow the specified pattern

3. When price not in range 0-200 Rs.


```
viders xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:tiffin="http://www.example.com/provider" xsi:schemaLocation="http://www.example.com/provider
der id="n1">
ame>H  cvc-maxExclusive-valid: Value '1246547' is not facet-valid with respect to maxExclusive '200.0' for type 'priceType'. xml(cvc-maxExclusive-valid)
hone>
mail>   cvc-type.3.1.3: The value '1246547' of element 'tiffin:price' is not valid. xml(cvc-type.3.1.3)
ddres
rice>   View Problem (Alt+F8)   No quick fixes available
rice>1246547</tiffin:price>
ating>4.7</tiffin:rating>
ongitude>77.1025</tiffin:longitude>
atitude>28.7041</tiffin:latitude>
ider>
der id="p2">
ame>Gourmet Tiffin</tiffin:name>
hone>322-655-9871</tiffin:phone>
```

Documentation of the Solution

The solution aims to manage and display information about tiffin service providers in a structured and human-readable format. This is accomplished by defining an XML Schema (XSD) to enforce data structure, using an XML document to store data, and applying an XSLT stylesheet to transform the XML data into an HTML format for easy viewing.

1. *XSD Schema (provider.xsd)*

Purpose: The XSD schema defines the structure and data types of the XML document that holds the tiffin service providers' information. It ensures that the XML data follows a specific format, including restrictions on phone numbers, email addresses, prices, and other attributes.

Key Elements:

- phoneType and emailType: Custom types to validate phone numbers and email addresses using regular expressions.
- priceType: Restricts prices to values between 0 and 200.
- itemType: Represents individual menu items, including name, diet type, and price.
- menuType: Contains details about a menu, including its name, description, and items.
- tiffinProvidersType: The root structure that holds all providers, including their details, such as name, contact information, location, and menu.

Validation: The XSD schema is used to validate the XML document, ensuring that the data conforms to the expected structure and types.

2. *XML Document (provider.xml)*

Purpose: The XML document stores the data of various tiffin providers, including their contact information, pricing, ratings, and detailed menus.

Structure:

- Each provider is represented as a <tiffin:provider> element with a unique id.
- Inside each provider, elements like <tiffin:name>, <tiffin:phone>, and <tiffin:menu> provide detailed information.
- Menus are defined with <tiffin:item> elements, each containing the name, diet type, and price of a menu item.

Schema Reference: The XML document references the XSD schema using xsi:schemaLocation, ensuring that the document can be validated against the schema.

3. XSL Stylesheet (provider.xsl)

Purpose: The XSLT stylesheet transforms the XML data into an HTML format, making it easy to read and navigate. The HTML is styled using CSS embedded within the XSLT.

Key Components:

- Table Generation: The stylesheet generates an HTML table displaying provider details like ID, name, phone, email, address, price, rating, longitude, and latitude.
- Menu Display: For each provider, the stylesheet generates an additional section that lists the menu items with their names, diet type, and price.
- Styling: CSS is used to enhance the readability of the generated HTML, with styles applied to tables, headings, and rows.

Transformation: The XSLT stylesheet is applied to the XML document to produce an HTML file, which can be viewed in a web browser.