



## **CS5004NT Emerging Programming Platforms and Technologies**

**Seen Exam**

**40%**

**2020-21 Autumn**

**Student Name:**

**London Met ID:**

**College ID:**

**Assignment Due Date:** May 27, 2021

**Assignment Submission Date:** May 27, 2021

*I confirm that I understand my assignment needs to be submitted online via Google Classroom under the relevant module page before the deadline for my assignment to be accepted and marked. I am fully aware that late submissions will be treated as non-submission and a mark of zero will be awarded.*

## TABLE OF CONTENT

TASK A .....	1
TREE DIAGRAM .....	1
LIST OF ELEMENT AND ATTRIBUTES .....	2
TASK B .....	3
TESTING.....	3
WELL-FORMED XML.....	3
VALIDATION OF XML WITH DTD .....	4
VALIDATION OF XML WITH XSD (RUSSIAN DOLL, Flat Catalog, Venetian Blind) .....	5
ENUMERATION RESTRICTION .....	6
RESTRICTION ON MIN AND MAX VALUE.....	7
RESTRICTION ON CONTACT NUMBER.....	8
RESTRICTION ON GENDER.....	9
DEVELOPMENT.....	11
XML .....	11
DTD.....	17
XSD RUSSIAN DOLL.....	19
XSD FLAT CATALOG .....	24
XSD VENETIAN BLIND.....	30

## TABLE OF FIGURE

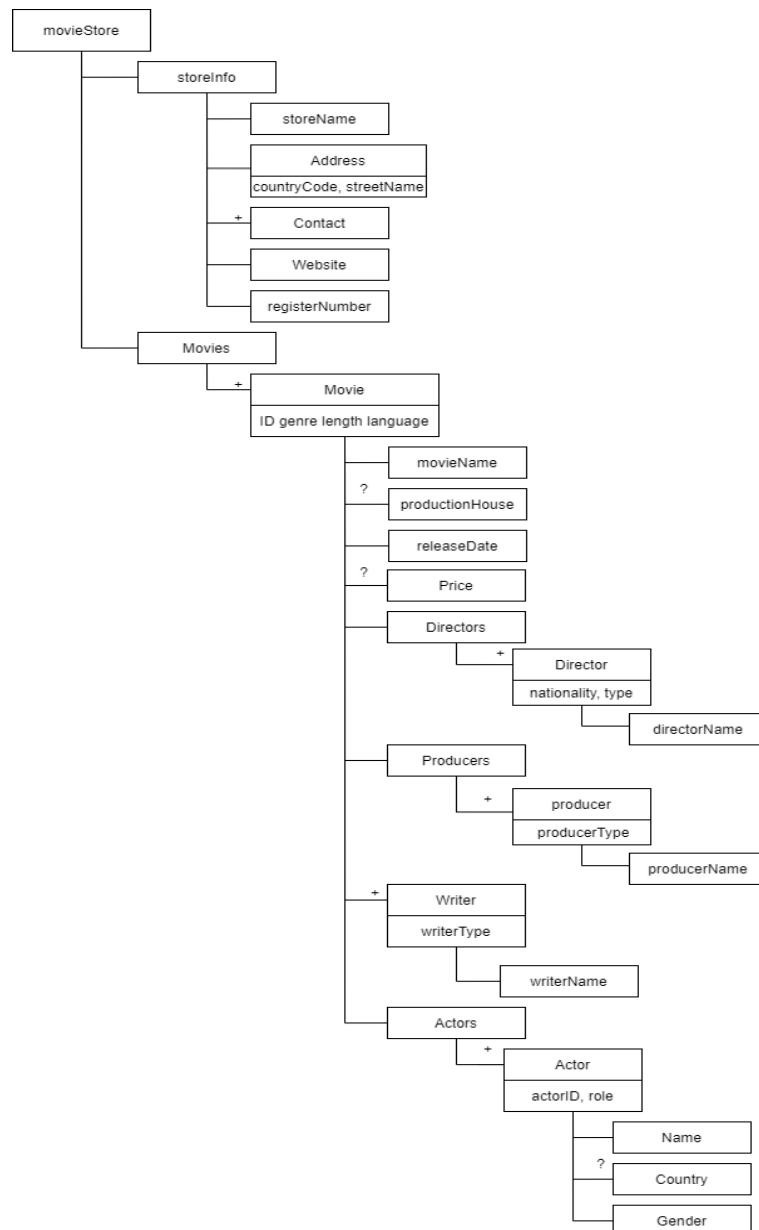
Figure 1 Tree Diagram .....	1
Figure 2 Well-formed XML .....	3
Figure 3 XML file .....	4
Figure 4 DTD File .....	4
Figure 5 No Error message on Xml and DTD Validation .....	4
Figure 6 XML file .....	5
Figure 7 Flat Catalog file .....	5
Figure 8 Russian Doll File .....	6
Figure 9 Venetian Blind File .....	6
Figure 10 Restriction on genre .....	7
Figure 11 Error message on wrong genre .....	7
Figure 12 Restriction on Price .....	8
Figure 13 Error message of maximum price .....	8
Figure 14 Restriction on Contact .....	9
Figure 15 Error message of invalid contact number .....	9
Figure 16 Restriction on Gender .....	10
Figure 17 Error message on wrong gender .....	10

## TABLE OF TABLES

Table 1 Test well-formed Xml.....	3
Table 2 Test Validation of xml with DTD .....	4
Table 3 Test validation of xml with all XSD .....	5
Table 4 Test Enumeration restrictions .....	6
Table 5 Test Restriction on min and max value .....	7
Table 6 Test Restriction on contact number .....	8
Table 7 Test Restriction on gender .....	9

**TASK A****TREE DIAGRAM**

In an XML tree diagram, the root element is at the top of an XML tree diagram, and the child elements are attached to the root elements with branches, similar to how leaves are connected to trees with branches. The tree diagram of the Movie store is shown below where the root element is store and child elements are also shown below.



*Figure 1 Tree Diagram*

**LIST OF ELEMENT AND ATTRIBUTES**

<b>ELEMENTS</b>	<b>ATTRIBUTES</b>
storeInfo	
Address	countryCode streetName
Contact	
Website	
registerNumber	
Movies	
Movie	ID genre length language
movieName	
productionHouse	
releaseDate	
Price	
Directors	
Director	nationality type
directorName	
Producers	
Producer	producerType
producerName	
Writer	writerType
writerName	
Actors	
Actor	actorID role
Name	
Country	
Gender	

**TASK B****TESTING****WELL-FORMED XML**

<b>Objective</b>	To check whether the XML is well-formed or not.
<b>Actions</b>	An XML file needs to be open in opera or any other website. It must be present in a tree structure way.
<b>Expected Result</b>	XML file should need to be open in a well-formed and tree structure.
<b>Actual Result</b>	The XML file was opened in a well-formed and tree structure.
<b>Conclusion</b>	Test successful.

*Table 1 Test well-formed Xml*

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```

▼<movieStore xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="19031886_venetianblind.
  ▼<storeInfo>
    <storeName>NoName Movie Store</storeName>
    <Address countryCode="N977" streetName="National Galli">Itahari</Address>
    <Contact>9824382947</Contact>
    <Contact>9865753452</Contact>
    <Website>www.NoNameMovieStore.com</Website>
    <registerNumber>111-111-111</registerNumber>
  </storeInfo>
  ▼<Movies>
    ▼<Movie ID="M1" genre="Action" length="114 Min" language="English">
      <movieName>Project Power</movieName>
      <productionHouse>Screen Arcade Supermarche</productionHouse>
      <releaseDate>August 14, 2020</releaseDate>
      <Price>3000</Price>
      ▼<Directors>
        ▼<Director nationality="British" type="Main-director">
          <directorName> Henry Joost</directorName>
        </Director>
      </Directors>
      ▼<Producers>
        ▼<Producer producerType="Executive Producer">
          <producerName>Ariel Schulman</producerName>
        </Producer>
        ▼<Producer producerType="co-producer">
          <producerName>Eric Newman</producerName>
        </Producer>
      </Producers>
    </Movie>
  </Movies>
</movieStore>

```

*Figure 2 Well-formed XML*

**VALIDATION OF XML WITH DTD**

<b>Objective</b>	To ensure that the XML file is validated against the DTD file.
<b>Actions</b>	Copy the entire content of the XML file and paste it into the XML validator site. Copy the entire content of the DTD file and paste it on the same XML validator site.
<b>Expected Result</b>	"No error message" should need to be displayed.
<b>Actual Result</b>	"No error message" was displayed.
<b>Conclusion</b>	Test successful.

*Table 2 Test Validation of xml with DTD*

Or upload it:

Choose File

19031886\_catalog.xml

The validation check is performed against any XML

*Figure 3 XML file*

Or upload it:

Choose File

19031886\_catalog.dtd

continue validation

*Figure 4 DTD File***No errors were found**

The following files have been uploaded so far:

XML document: [19031886\\_catalog.xml](#)[19031886\\_catalog.dtd](#)

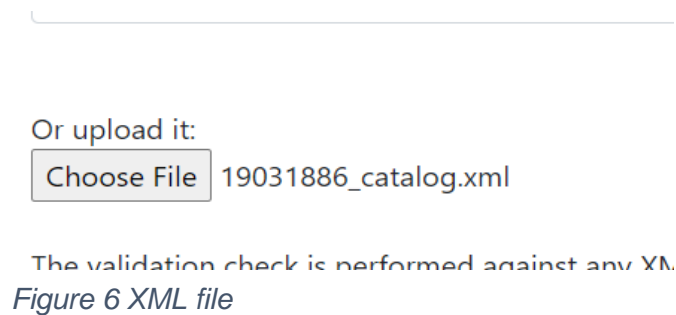
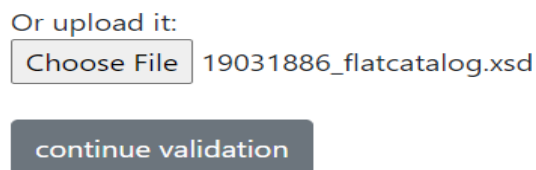
Click on any file name if you want to edit the file.

*Figure 5 No Error message on Xml and DTD Validation*



**VALIDATION OF XML WITH XSD (RUSSIAN DOLL, Flat Catalog, Venetian Blind)**

<b>Objective</b>	To ensure that the XML file is validated against the XSD file.
<b>Actions</b>	Copy the entire content of the XML file and paste it into the XML validator site. Copy the entire content of the XSD file and paste it on the same XML validator site.
<b>Expected Result</b>	"No error message" should need to be displayed.
<b>Actual Result</b>	"No error message" was displayed.
<b>Conclusion</b>	Test successful.

*Table 3 Test validation of xml with all XSD**Figure 6 XML file**Figure 7 Flat Catalog file*

Or upload it:

Choose File

19031886\_russiandoll.xsd

continue validation

*Figure 8 Russian Doll File*

Or upload it:

Choose File

19031886\_v...tianblind.xsd

continue validation

*Figure 9 Venetian Blind File*

## ENUMERATION RESTRICTION

<b>Objective</b>	Trying to provide different genres.
<b>Actions</b>	Genre=Horror
<b>Expected Result</b>	"Error message" should need to be displayed.
<b>Actual Result</b>	"Error message" was displayed.
<b>Conclusion</b>	Test successful.

*Table 4 Test Enumeration restrictions*



```

<xs:attribute name="genre" use="required">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="Action"/>
      <xs:enumeration value="Adventure"/>
      <xs:enumeration value="Comedy"/>
      <xs:enumeration value="Fantasy"/>
      <xs:enumeration value="Romantic"/>
    </xs:restriction>
  </xs:simpleType>
</xs:attribute>

```

Figure 10 Restriction on genre

**2 errors have been found!**

Click on  to jump to the error. In the document, you can point at  with your mouse to see the error message.

**Errors in the XML document:**

- ✖ 114: 68 cvc-attribute.3: The value 'Horror' of attribute 'genre' on element 'Movie' is not valid with respect to its type, 'null'.
- ✖ 114: 68 cvc-enumeration-valid: Value 'Horror' is not facet-valid with respect to enumeration '[Action Adventure, Comedy, Fantasy, Romantic]'. It must be a value from the enumeration.

**XML document:**

Figure 11 Error message on wrong genre

**RESTRICTION ON MIN AND MAX VALUE**

<b>Objective</b>	Trying to give price value more than 5000.
<b>Actions</b>	<Price>10000</Price>
<b>Expected Result</b>	"Error message" should need to be displayed.
<b>Actual Result</b>	"Error message" was displayed.
<b>Conclusion</b>	Test successful.

Table 5 Test Restriction on min and max value



```

<xs:element name="Price" type="xs:string"/>
<xs:simpleType>
  <xs:restriction base="xs:positiveInteger">
    <xs:minInclusive value="1000" />
    <xs:maxInclusive value="5000" />
  </xs:restriction>
</xs:simpleType>
</xs:element>

```

Figure 12 Restriction on Price

**2 errors have been found!**

Click on  to jump to the error. In the document, you can point at  with your mouse to see the error message.

**Errors in the XML document:**



-  16: 24 cvc-maxInclusive-valid: Value '10000' is not facet-valid with respect to maxInclusive '5000' for type 'null'.
-  16: 24 cvc-type.3.1.3: The value '10000' of element 'Price' is not valid.

Figure 13 Error message of maximum price

**RESTRICTION ON CONTACT NUMBER**

<b>Objective</b>	Trying to give the which does not start with 98 and after 98 more than 8 digits.
<b>Actions</b>	<pre> &lt;Contact&gt;9824382947&lt;/Contact&gt; &lt;Contact&gt;0365753452&lt;/Contact&gt; </pre>
<b>Expected Result</b>	"Error message" should need to be displayed.
<b>Actual Result</b>	"Error message" was displayed.
<b>Conclusion</b>	Test successful.

Table 6 Test Restriction on contact number



```

<xs:element name="Contact">
  <xs:simpleType>
    <xs:restriction base="xs:integer">
      <xs:pattern value="98[0-9]{8}" />
    </xs:restriction>
  </xs:simpleType>
</xs:element>

```

Figure 14 Restriction on Contact

**2 errors have been found!**

Click on  to jump to the error. In the document, you can point at  with your mouse to see the error message.

**Errors in the XML document:**



-  7:32 cvc-pattern-valid: Value '0365753452' is not facet-valid with respect to pattern '98[0-9]{8}' for type 'null'.
-  7:32 cvc-type.3.1.3: The value '0365753452' of element 'Contact' is not valid.

Figure 15 Error message of invalid contact number

**RESTRICTION ON GENDER**



<b>Objective</b>	Trying to give the gender value different from male and female.
<b>Actions</b>	<Gender>Boy</Gender>
<b>Expected Result</b>	"Error message" should need to be displayed.
<b>Actual Result</b>	"Error message" was displayed.
<b>Conclusion</b>	Test successful.

Table 7 Test Restriction on gender

```
<xs:element name="Gender">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:pattern value="Male|Female" />
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

Figure 16 Restriction on Gender

### 2 errors have been found!

Click on  to jump to the error. In the document, you can point at  with your mouse to see the error message.

#### Errors in the XML document:



-  44: 26 cvc-pattern-valid: Value 'Boy' is not facet-valid with respect to pattern 'Male|Female' for type 'null'.
-  44: 26 cvc-type.3.1.3: The value 'Boy' of element 'Gender' is not valid.

Figure 17 Error message on wrong gender

**DEVELOPMENT****XML**

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<movieStore xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="19031886_venetianblind.xsd">
```

```
  <storeInfo>
```

```
    <storeName>NoName Movie Store</storeName>
```

```
      <Address countryCode="N977" streetName="National
Galli">Itahari</Address>
```

```
      <Contact>9824382947</Contact>
```

```
      <Contact>9865753452</Contact>
```

```
      <Website>www.NoNameMovieStore.com</Website>
```

```
      <registerNumber>111-111-111</registerNumber>
```

```
    </storeInfo>
```

```
    <Movies>
```

```
      <Movie ID="M1" genre="Action" length="114 Min" language="English">
```

```
        <movieName>Project Power</movieName>
```

```
        <productionHouse>Screen Arcade
Supermarche</productionHouse>
```

```
        <releaseDate>August 14, 2020</releaseDate>
```

```
        <Price>3000</Price>
```

```
        <Directors>
```

```
          <Director nationality="British" type="Main-director">
```

```
            <directorName> Henry Joost</directorName>
```

```
          </Director>
```

```
        </Directors>
```

```
        <Producers>
```

```
          <Producer producerType="Executive Producer">
```

```
            <producerName>Ariel Schulman</producerName>
```

```
          </Producer>
```

```
<Producer producerType="co-producer">
    <producerName>Eric Newman</producerName>
</Producer>
</Producers>
<Writer writerType="Script Writer">
    <writerName>Mattson Tomlin</writerName>
</Writer>
<Writer writerType="Dialogue Writer">
    <writerName>Joseph Trapanese</writerName>
</Writer>
<Actors>
    <Actor actorID="A1" role="Star">
        <Name>Joseph Gordon-Levitt</Name>
        <Gender>Male</Gender>
    </Actor>
    <Actor actorID="A2" role="co-Star">
        <Name>Rodrigo Santoro</Name>
        <Country>USA</Country>
        <Gender>Boy</Gender>
    </Actor>
</Actors>
</Movie>
<Movie ID="M2" genre="Adventure" length="101 Min"
language="English">
    <movieName>Mortal Kombat</movieName>
    <productionHouse>Threshold Entertainment</productionHouse>
    <releaseDate>August 18, 1995</releaseDate>
    <Price>2000</Price>
    <Directors>
```



```
<Director nationality="British" type="Main-director">
    <directorName>Paul Anderson</directorName>
</Director>

<Director nationality="Korean" type="Assistant-director">
    <directorName>Robin Shou</directorName>
</Director>
</Directors>
<Producers>
    <Producer producerType="Executive Producer">
        <producerName>Lawrence
Kasanoff</producerName>
    </Producer>
</Producers>
<Writer writerType="Script Writer">
    <writerName>Kevin Droney</writerName>
</Writer>
<Actors>
    <Actor actorID="A3" role="Star">
        <Name>Christopher Lambert</Name>
        <Gender>Male</Gender>
    </Actor>
    <Actor actorID="A4" role="co-Star">
        <Name>Talisa Soto</Name>
        <Country>USA</Country>
        <Gender>Female</Gender>
    </Actor>
</Actors>
</Movie>
<Movie ID="M3" genre="Comedy" length="101 Min" language="Hindi">
```

```
<movieName>Roohi</movieName>
<productionHouse>Maddock Films</productionHouse>
<releaseDate>11 March 2021</releaseDate>
<Directors>
    <Director nationality="Indain" type="Main-director">
        <directorName>Hardik Mehta</directorName>
    </Director>
</Directors>
<Producers>
    <Producer producerType="Executive Producer">
        <producerName>Dinesh Vijan</producerName>
    </Producer>
    <Producer producerType="co-Producer">
        <producerName>Mrighdeep Singh
Lamba</producerName>
    </Producer>
</Producers>
<Writer writerType="Script Writer">
    <writerName>Mrighdeep Singh Lamba</writerName>
</Writer>
<Actors>
    <Actor actorID="A5" role="Star">
        <Name>    Rajkummar Rao</Name>
        <Country>India</Country>
        <Gender>Male</Gender>
    </Actor>
    <Actor actorID="A6" role="co-Star">
        <Name>Janhvi Kapoor</Name>
        <Country>India</Country>
```

```
<Gender>Female</Gender>
</Actor>
</Actors>
</Movie>
<Movie ID="M4" genre="Romantic" length="119 Min" language="Nepali">
  <movieName>    ए मेरो हजुर ३</movieName>
  <releaseDate>12 April 2019 </releaseDate>
  <Directors>
    <Director nationality="Nepali" type="Main-director">
      <directorName>Jharana Thapa</directorName>
    </Director>
  </Directors>
  <Producers>
    <Producer producerType="Executive Producer">
      <producerName>Sunil Kumar
Thapa</producerName>
    </Producer>
  </Producers>
  <Writer writerType="Script Writer">
    <writerName>Suman Poudel</writerName>
  </Writer>
  <Actors>
    <Actor actorID="A7" role="Star">
      <Name>Anmol K.C</Name>
      <Gender>Male</Gender>
    </Actor>
    <Actor actorID="A8" role="co-Star">
      <Name>Suhana Thapa</Name>
```

```
<Gender>Male</Gender>
</Actor>
</Actors>
</Movie>
<Movie ID="M5" genre="Fantasy" length="164 Min" language="Tamil">
  <movieName>Kaashmora</movieName>
  <productionHouse>Dream Warrior Pictures</productionHouse>
  <releaseDate>28 October 2016</releaseDate>
  <Directors>
    <Director nationality="Indain" type="Main-director">
      <directorName>Gokul</directorName>
    </Director>
  </Directors>
  <Producers>
    <Producer producerType="Executive Producer">
      <producerName>S R Prakashbabu</producerName>
    </Producer>
    <Producer producerType="co-Producer">
      <producerName>V. J. Sabu Joseph</producerName>
    </Producer>
  </Producers>
  <Writer writerType="Script Writer">
    <writerName>Yashwant mahilwar</writerName>
  </Writer>
  <Writer writerType="Dialogue Writer">
    <writerName>John Mahendran</writerName>
  </Writer>
  <Actors>
```

```

        <Actor actorID="A9" role="Star">
            <Name>Karthi</Name>
            <Country>India</Country>
            <Gender>Male</Gender>
        </Actor>
        <Actor actorID="A10" role="co-Star">
            <Name>Sri Divya</Name>
            <Country>India</Country>
            <Gender>Female</Gender>
        </Actor>
    </Actors>
</Movie>
</Movies>
</movieStore>

```

**DTD**

```

<!ELEMENT movieStore (storeInfo,Movies)>
<!ELEMENT storeInfo (storeName,Address,Contact+,Website,registerNumber)>
<!ELEMENT storeName (#PCDATA)>
<!ELEMENT Address (#PCDATA)>
<!ATTLIST Address countryCode ID #REQUIRED>
<!ATTLIST Address streetName CDATA #REQUIRED>
<!ELEMENT Contact (#PCDATA)>
<!ELEMENT Website (#PCDATA)>
<!ELEMENT registerNumber (#PCDATA)>
<!ELEMENT Movies (Movie+)>
<!ELEMENT Movie
(movieName,productionHouse?,releaseDate,Price?,Directors,Producers,Writer+,Actors)
>
<!ATTLIST Movie ID ID #REQUIRED>

```

```
<!ATTLIST Movie genre CDATA #REQUIRED>
<!ATTLIST Movie length CDATA #REQUIRED>
<!ATTLIST Movie language CDATA #REQUIRED>
<!ELEMENT movieName (#PCDATA)>
<!ELEMENT productionHouse (#PCDATA)>
<!ELEMENT releaseDate (#PCDATA)>
<!ELEMENT Price (#PCDATA)>
<!ELEMENT Directors (Director+)>
<!ELEMENT Director (directorName)>
<!ELEMENT directorName (#PCDATA)>
<!ATTLIST Director nationality CDATA #REQUIRED>
<!ATTLIST Director type CDATA #REQUIRED>
<!ELEMENT Producers (Producer+)>
<!ELEMENT Producer (producerName)>
<!ELEMENT producerName (#PCDATA)>
<!ATTLIST Producer producerType CDATA #REQUIRED>
<!ELEMENT Writer (writerName)>
<!ELEMENT writerName (#PCDATA)>
<!ATTLIST Writer writerType CDATA #REQUIRED>
<!ELEMENT Actors (Actor+)>
<!ELEMENT Actor (Name,Country?,Gender)>
<!ELEMENT Name (#PCDATA)>
<!ELEMENT Country (#PCDATA)>
<!ELEMENT Gender (#PCDATA)>
<!ATTLIST Actor actorID ID #REQUIRED>
<!ATTLIST Actor role CDATA #REQUIRED>
```

**XSD RUSSIAN DOLL**

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="movieStore">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="storeInfo">
          <xs:complexType>
            <xs:sequence>
              <xs:element type="xs:string" name="storeName" />
              <xs:element name="Address">
                <xs:complexType>
                  <xs:simpleContent>
                    <xs:extension base="xs:string">
                      <xs:attribute type="xs:string" name="countryCode" />
                      <xs:attribute type="xs:string" name="streetName" />
                    </xs:extension>
                  </xs:simpleContent>
                </xs:complexType>
              </xs:element>
              <xs:element name="Contact" maxOccurs="unbounded">
                <xs:simpleType>
                  <xs:restriction base="xs:integer">
                    <xs:pattern value="98[0-9]{8}" />
                  </xs:restriction>
                </xs:simpleType>
              </xs:element>
              <xs:element type="xs:string" name="Website" />
              <xs:element name="registerNumber">
```

```

<xs:simpleType>
  <xs:restriction base="xs:string">
    <xs:pattern value="[0-9]{3}-[0-9]{3}-[0-9]{3}" />
  </xs:restriction>
</xs:simpleType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="Movies">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="Movie" maxOccurs="unbounded">
        <xs:complexType>
          <xs:sequence>
            <xs:element type="xs:string" name="movieName" />
            <xs:element type="xs:string" name="productionHouse"
minOccurs="0" />
            <xs:element type="xs:string" name="releaseDate" />
            <xs:element name="Price" minOccurs="0">
              <xs:simpleType>
                <xs:restriction base="xs:positiveInteger">
                  <xs:minInclusive value="1000" />
                  <xs:maxInclusive value="5000" />
                </xs:restriction>
              </xs:simpleType>
            </xs:element>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```



```

        <xs:sequence>
            <xs:element name="Director"
maxOccurs="unbounded">
                <xs:complexType>
                    <xs:sequence>
                        <xs:element type="xs:string"
name="directorName" />
                    </xs:sequence>
                    <xs:attribute type="xs:string" name="nationality"
use="required" />
                    <xs:attribute type="xs:string" name="type"
use="required" />
                </xs:complexType>
            </xs:element>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="Producers">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="Producer"
maxOccurs="unbounded">
                <xs:complexType>
                    <xs:sequence>
                        <xs:element type="xs:string"
name="producerName" />
                    </xs:sequence>
                    <xs:attribute type="xs:string"
name="producerType" use="required" />
                </xs:complexType>
            </xs:element>

```

```

        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="Writer" maxOccurs="unbounded"
minOccurs="0">
    <xs:complexType>
        <xs:sequence>
            <xs:element type="xs:string" name="writerName" />
        </xs:sequence>
        <xs:attribute type="xs:string" name="writerType"
use="required" />
    </xs:complexType>
</xs:element>
<xs:element name="Actors">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="Actor" maxOccurs="unbounded">
                <xs:complexType>
                    <xs:sequence>
                        <xs:element type="xs:string" name="Name" />
                        <xs:element type="xs:string" name="Country"
minOccurs="0" />
                        <xs:element name="Gender">
                            <xs:simpleType>
                                <xs:restriction base="xs:string">
                                    <xs:pattern value="Male|Female" />
                                </xs:restriction>
                            </xs:simpleType>
                        </xs:element>

```

```

        </xs:sequence>
        <xs:attribute type="xs:ID" name="actorID"
use="required" />
        <xs:attribute type="xs:string" name="role"
use="required" />
    </xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
<xs:attribute type="xs:ID" name="ID" use="required" />
<xs:attribute name="genre" use="required">
    <xs:simpleType>
        <xs:restriction base="xs:string">
            <xs:enumeration value="Action" />
            <xs:enumeration value="Adventure" />
            <xs:enumeration value="Comedy" />
            <xs:enumeration value="Fantasy" />
            <xs:enumeration value="Romantic" />
        </xs:restriction>
    </xs:simpleType>
</xs:attribute>
<xs:attribute type="xs:string" name="length" use="required" />
<xs:attribute type="xs:string" name="language" use="required" />
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
```

```
        </xs:element>
    </xs:sequence>
</xs:complexType>
</xs:element>
</xs:schema>
```

### **XSD FLAT CATALOG**

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
    <xs:element name="storeName" type="xs:string" />
    <xs:element name="Address">
        <xs:complexType>
            <xs:simpleContent>
                <xs:extension base="xs:string">
                    <xs:attribute type="xs:string" name="countryCode" />
                    <xs:attribute type="xs:string" name="streetName" />
                </xs:extension>
            </xs:simpleContent>
        </xs:complexType>
    </xs:element>
    <xs:element name="Contact">
        <xs:simpleType>
            <xs:restriction base="xs:integer">
                <xs:pattern value="98[0-9]{8}" />
            </xs:restriction>
        </xs:simpleType>
    </xs:element>
    <xs:element name="Website" type="xs:string" />
    <xs:element name="registerNumber">
        <xs:simpleType>
```

```
<xs:restriction base="xs:string">
  <xs:pattern value="[0-9]{3}-[0-9]{3}-[0-9]{3}" />
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="directorName" type="xs:string" />
<xs:element name="Director">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="directorName" />
    </xs:sequence>
    <xs:attribute type="xs:string" name="nationality" use="required" />
    <xs:attribute type="xs:string" name="type" use="required" />
  </xs:complexType>
</xs:element>
<xs:element name="producerName" type="xs:string" />
<xs:element name="Producer">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="producerName" />
    </xs:sequence>
    <xs:attribute type="xs:string" name="producerType" use="required" />
  </xs:complexType>
</xs:element>
<xs:element name="writerName" type="xs:string" />
<xs:element name="Name" type="xs:string" />
<xs:element name="Gender">
  <xs:simpleType>
```

```
<xs:restriction base="xs:string">
  <xs:pattern value="Male|Female" />
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="Country" type="xs:string" />
<xs:element name="Actor">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="Name" />
      <xs:element ref="Country" minOccurs="0" />
      <xs:element ref="Gender" />
    </xs:sequence>
    <xs:attribute type="xs:string" name="actorID" use="required" />
    <xs:attribute type="xs:string" name="role" use="required" />
  </xs:complexType>
</xs:element>
<xs:element name="movieName" type="xs:string" />
<xs:element name="productionHouse" type="xs:string" />
<xs:element name="releaseDate" type="xs:string" />
<xs:element name="Price">
  <xs:simpleType>
    <xs:restriction base="xs:positiveInteger">
      <xs:minInclusive value="1000" />
      <xs:maxInclusive value="5000" />
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

```
<xs:element name="Directors">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="Director" maxOccurs="unbounded" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="Producers">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="Producer" maxOccurs="unbounded" minOccurs="0" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="Writer">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="writerName" />
    </xs:sequence>
    <xs:attribute type="xs:string" name="writerType" use="required" />
  </xs:complexType>
</xs:element>
<xs:element name="Actors">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="Actor" maxOccurs="unbounded" />
    </xs:sequence>
  </xs:complexType>
```

```
</xs:element>
<xs:element name="Movie">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="movieName" />
      <xs:element ref="productionHouse" minOccurs="0" />
      <xs:element ref="releaseDate" />
      <xs:element ref="Price" minOccurs="0" />
      <xs:element ref="Directors" />
      <xs:element ref="Producers" />
      <xs:element ref="Writer" maxOccurs="unbounded" />
      <xs:element ref="Actors" />
    </xs:sequence>
    <xs:attribute type="xs:string" name="ID" use="required" />
    <xs:attribute name="genre" use="required">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:enumeration value="Action" />
          <xs:enumeration value="Adventure" />
          <xs:enumeration value="Comedy" />
          <xs:enumeration value="Fantasy" />
          <xs:enumeration value="Romantic" />
        </xs:restriction>
      </xs:simpleType>
    </xs:attribute>
    <xs:attribute type="xs:string" name="length" use="required" />
    <xs:attribute type="xs:string" name="language" use="required" />
  </xs:complexType>
```



```
</xs:element>
<xs:element name="storeInfo">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="storeName" />
      <xs:element ref="Address" />
      <xs:element ref="Contact" maxOccurs="unbounded" />
      <xs:element ref="Website" />
      <xs:element ref="registerNumber" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="Movies">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="Movie" maxOccurs="unbounded" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="movieStore">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="storeInfo" />
      <xs:element ref="Movies" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:schema>
```

**XSD VENETIAN BLIND**

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="movieStore" type="movieStoreType" />
  <xs:complexType name="AddressType">
    <xs:simpleContent>
      <xs:extension base="xs:string">
        <xs:attribute type="xs:string" name="countryCode" />
        <xs:attribute type="xs:string" name="streetName" />
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
  <xs:complexType name="storeInfoType">
    <xs:sequence>
      <xs:element type="xs:string" name="storeName" />
      <xs:element type="AddressType" name="Address" />
      <xs:element name="Contact" maxOccurs="unbounded">
        <xs:simpleType>
          <xs:restriction base="xs:integer">
            <xs:pattern value="98[0-9]{8}" />
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
      <xs:element type="xs:string" name="Website" />
      <xs:element name="registerNumber">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:pattern value="[0-9]{3}-[0-9]{3}-[0-9]{3}" />
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:schema>
```

```
        </xs:simpleType>
    </xs:element>
</xs:sequence>
</xs:complexType>
<xs:complexType name="DirectorType">
    <xs:sequence>
        <xs:element type="xs:string" name="directorName" />
    </xs:sequence>
    <xs:attribute type="xs:string" name="nationality" use="required" />
    <xs:attribute type="xs:string" name="type" use="required" />
</xs:complexType>
<xs:complexType name="DirectorsType">
    <xs:sequence>
        <xs:element type="DirectorType" name="Director" maxOccurs="unbounded" />
    </xs:sequence>
</xs:complexType>
<xs:complexType name="ProducerType">
    <xs:sequence>
        <xs:element type="xs:string" name="producerName" />
    </xs:sequence>
    <xs:attribute type="xs:string" name="producerType" use="required" />
</xs:complexType>
<xs:complexType name="ProducersType">
    <xs:sequence>
        <xs:element type="ProducerType" name="Producer" maxOccurs="unbounded"
/>
    </xs:sequence>
</xs:complexType>
<xs:complexType name="WriterType">
```

```
<xs:sequence>
  <xs:element type="xs:string" name="writerName" />
</xs:sequence>
<xs:attribute type="xs:string" name="writerType" use="required" />
</xs:complexType>
<xs:complexType name="ActorType">
  <xs:sequence>
    <xs:element type="xs:string" name="Name" />
    <xs:element type="xs:string" name="Country" minOccurs="0" />
    <xs:element name="Gender">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:pattern value="Male|Female" />
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
  </xs:sequence>
  <xs:attribute type="xs:string" name="actorID" use="required" />
  <xs:attribute type="xs:string" name="role" use="required" />
</xs:complexType>
<xs:complexType name="ActorsType">
  <xs:sequence>
    <xs:element type="ActorType" name="Actor" maxOccurs="unbounded" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="MovieType">
  <xs:sequence>
    <xs:element type="xs:string" name="movieName" />
```

```
<xs:element type="xs:string" name="productionHouse" minOccurs="0" />
<xs:element type="xs:string" name="releaseDate" />
<xs:element name="Price" minOccurs="0">
  <xs:simpleType>
    <xs:restriction base="xs:positiveInteger">
      <xs:minInclusive value="100" />
      <xs:maxInclusive value="5000" />
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element type="DirectorsType" name="Directors" />
<xs:element type="ProducersType" name="Producers" />
<xs:element type="WriterType" name="Writer" maxOccurs="unbounded" />
<xs:element type="ActorsType" name="Actors" />
</xs:sequence>
<xs:attribute type="xs:string" name="ID" use="required" />
<xs:attribute name="genre" use="required">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="Action" />
      <xs:enumeration value="Adventure" />
      <xs:enumeration value="Comedy" />
      <xs:enumeration value="Fantasy" />
      <xs:enumeration value="Romantic" />
    </xs:restriction>
  </xs:simpleType>
</xs:attribute>
<xs:attribute type="xs:string" name="length" use="required" />
```

```
<xs:attribute type="xs:string" name="language" use="required" />
</xs:complexType>
<xs:complexType name="MoviesType">
  <xs:sequence>
    <xs:element type="MovieType" name="Movie" maxOccurs="unbounded" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="movieStoreType">
  <xs:sequence>
    <xs:element type="storeInfoType" name="storeInfo" />
    <xs:element type="MoviesType" name="Movies" />
  </xs:sequence>
</xs:complexType>
</xs:schema>
```