VIT®
Vellore Institute of Technology
(Deemed to be University under section 3 of the UGC Act, 1956)

Programme	:	**B.Tech.**	Semester	:	**Fall 22-23**
Course	:	**CSE3002:** **Internet and Web Programming Lab**	Slot	:	**L9+L10**
Faculty	:	**Dr. M. Premalatha**	Marks	:	**10**

Date: 01-11-2022
Advait Deochakke
20BCE1143

Exercise –8

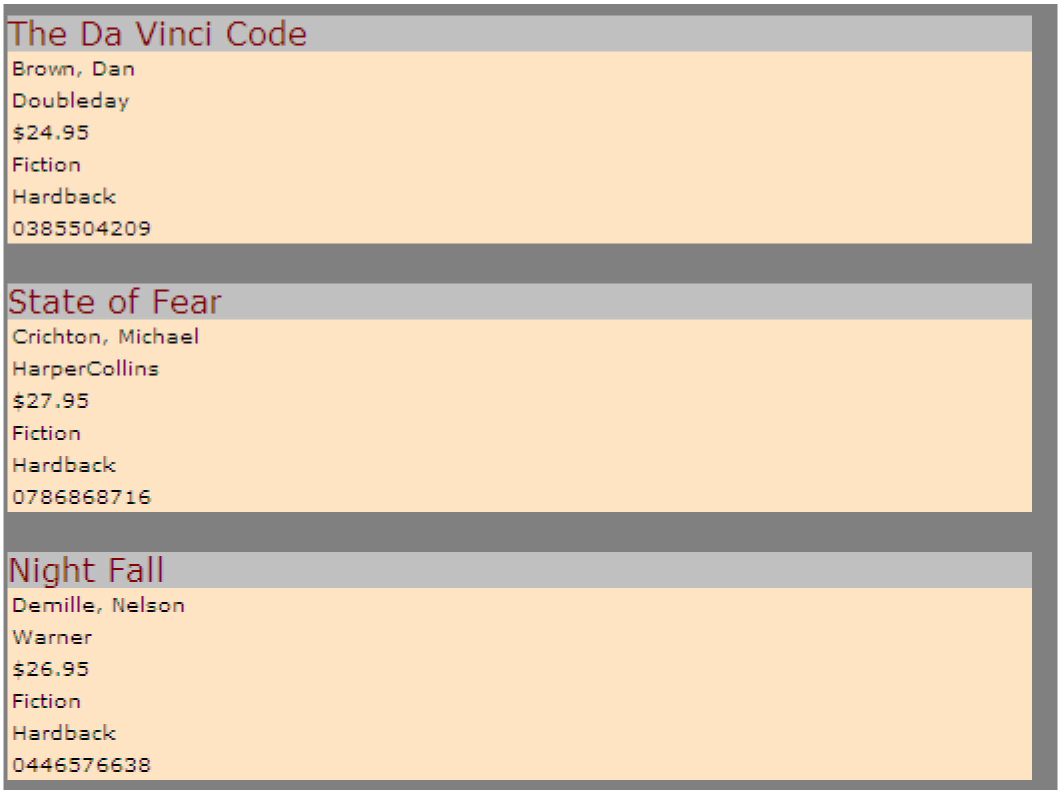
XML

Note:

i. **View your registration number in the title bar and snap a screenshot of output along with the title bar.**

ii. **Screenshot is required for every modifications of the web page.**

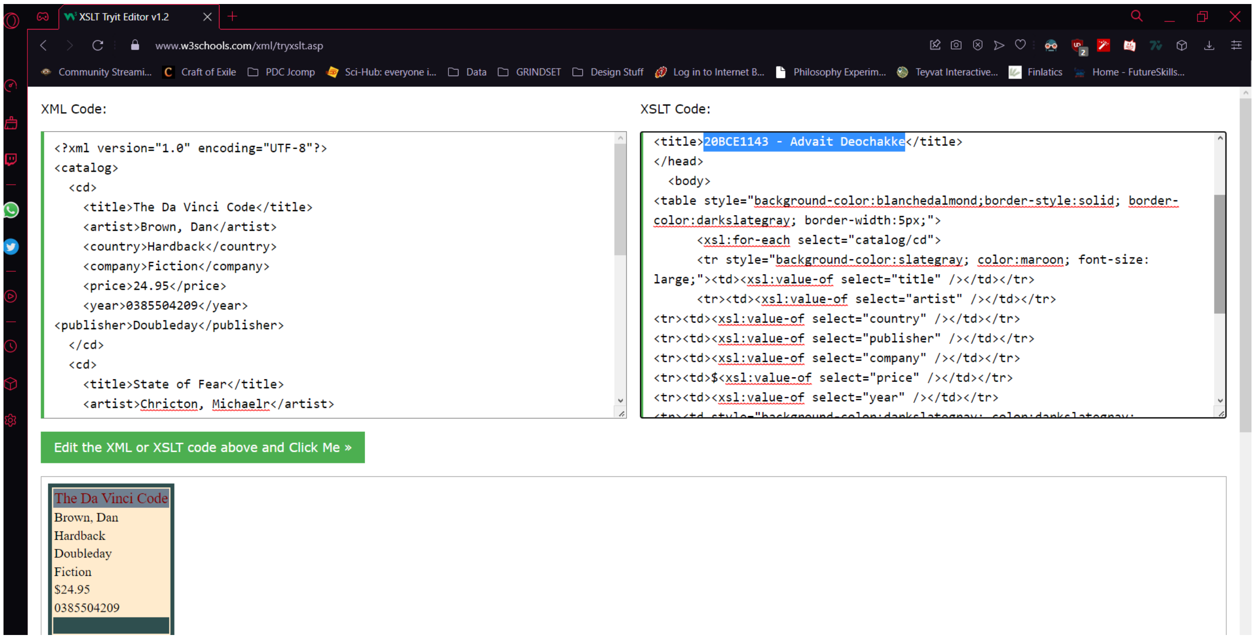
1. Apply styles as below using XML & CSS



The Da Vinci Code
Brown, Dan
Doubleday
$24.95
Fiction
Hardback
0385504209

State of Fear
Crichton, Michael
HarperCollins
$27.95
Fiction
Hardback
0786868716

Night Fall
Demille, Nelson
Warner
$26.95
Fiction
Hardback
0446576638



XML Code:

```
<?xml version="1.0" encoding="UTF-8"?>
<catalog>
  <cd>
    <title>The Da Vinci Code</title>
    <artist>Brown, Dan</artist>
    <country>Hardback</country>
    <company>Fiction</company>
    <price>24.95</price>
    <year>0385504209</year>
<publisher>Doubleday</publisher>
  </cd>
  <cd>
    <title>State of Fear</title>
    <artist>Chricton, Michaelr</artist>
```

XSLT Code:

```
<title>20BCE1143 - Advait Deochakke</title>
</head>
  <body>
<table style="background-color:blanchedalmond;border-style:solid; border-
color:darkslategray; border-width:5px;">
      <xsl:for-each select="catalog/cd">
      <tr style="background-color:slategray; color:maroon; font-size:
large;"><td><xsl:value-of select="title" /></td></tr>
      <tr><td><xsl:value-of select="artist" /></td></tr>
<tr><td><xsl:value-of select="country" /></td></tr>
<tr><td><xsl:value-of select="publisher" /></td></tr>
<tr><td><xsl:value-of select="company" /></td></tr>
<tr><td>$<xsl:value-of select="price" /></td></tr>
<tr><td><xsl:value-of select="year" /></td></tr>
<tr><td style="background-color:darkslategray; color:darkslategray;
```

Edit the XML or XSLT code above and Click Me »

The Da Vinci Code
Brown, Dan
Hardback
Doubleday
Fiction
$24.95
0385504209

XML Code:

```
<?xml version="1.0" encoding="UTF-8"?>
<catalog>
  <cd>
    <title>The Da Vinci Code</title>
    <artist>Brown, Dan</artist>
    <country>Hardback</country>
    <company>Fiction</company>
    <price>24.95</price>
    <year>0385504209</year>
<publisher>Doubleday</publisher>
  </cd>
  <cd>
    <title>State of Fear</title>
    <artist>Chricton, Michaelr</artist>
<country>Hardback</country>
    <company>Fiction</company>
    <price>27.95</price>
    <year>0786886871</year>
<publisher>HarperCollins</publisher>
  </cd>
  <cd>
    <title>Nightfall</title>
    <artist>Damille, Nelson</artist>
    <country>Hardback</country>
    <company>Fiction</company>
  <price>24.95</price>
  <year>046657638</year>
<publisher>Warner</publisher>
 </cd>
</catalog>
```

XSLT Code:

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
<xsl:template match="/">
 <html>
<head>
<title>20BCE1143 - Advait Deochakke</title>
</head>
 <body>
<table style="background-color:blanchedalmond;border-style:solid; border-color:darkslategray;
border-width:5px;">
    <xsl:for-each select="catalog/cd">
    <tr style="background-color:slategray; color:maroon; font-size: large;"><td><xsl:value-of
select="title" /></td></tr>
    <tr><td><xsl:value-of select="artist" /></td></tr>
<tr><td><xsl:value-of select="country" /></td></tr>
<tr><td><xsl:value-of select="publisher" /></td></tr>
<tr><td><xsl:value-of select="company" /></td></tr>
<tr><td>$<xsl:value-of select="price" /></td></tr>
<tr><td><xsl:value-of select="year" /></td></tr>
<tr><td style="background-color:darkslategray; color:darkslategray; ">blank</td></tr>
    </xsl:for-each>
  </table>
 </body>
 </html>
</xsl:template>
</xsl:stylesheet>
```

2. Apply XML DTD and XML Schema for the above Books Database

DTD :

```
<!DOCTYPE CATALOG [

<!ELEMENT CATALOG (CD+)>
<!ELEMENT CD (title+, artist+, country+, company+, price?, year+, publisher+)>
<!ATTLIST CD
  title (#PCDATA)
  artist (#pcdata)
  country (Hardback|Paperback|Ebook) "Hardback"
  company (Fiction|Thriller|Biography|Literature) "Fiction"
  price (#PCDATA)
  year (#pcdata)
  publisher (HarperCollins|Doubleday|Warner)
>
]>
```

Schema :

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

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

<!-- definition of simple elements -->
<xs:element name="title" type="xs:string"/>
<xs:element name="artist" type="xs:string"/>
<xs:element name="country" type="xs:string"/>
<xs:element name="company" type="xs:string"/>
<xs:element name="price" type="xs:decimal"/>
<xs:element name="year" type="xs:integer"/>
<xs:element name="publisher" type="xs:string"/>

<!-- definition of complex elements -->
<xs:element name="cd">
 <xs:complexType>
  <xs:sequence>
   <xs:element ref="title" maxOccurs="1" minOccurs="1"/>
   <xs:element ref="artist" maxOccurs="1" minOccurs="1"/>
   <xs:element ref="country" maxOccurs="1" minOccurs="1"/>
   <xs:element ref="company" maxOccurs="1" minOccurs="1"/>
   <xs:element ref="price" maxOccurs="1" minOccurs="0"/>
   <xs:element ref="year" maxOccurs="1" minOccurs="1"/>
   <xs:element ref="publisher" maxOccurs="1" minOccurs="1"/>
  </xs:sequence>
 </xs:complexType>
</xs:element>

<xs:element name="catalog">
 <xs:complexType>
  <xs:sequence>
   <xs:element ref="cd" maxOccurs="unbounded"/>
  </xs:sequence>
 </xs:complexType>
</xs:element>

</xs:schema>
```

3. Consider the following XML DTD

```
<?xml version="1.0"?>


<!-- 1. XML-Document -->
<!DOCTYPE catalog SYSTEM "cat.dtd">
<catalog>
<product>
<productname>cupboard</productname>
<storagenumber>3421</storagenumber>
<dimension unit="cm">
<width>120</width>
<heigth>180</heigth>
<depth>34</depth>
</dimension>
<description>oak veneer</description>
</product>
</catalog>
<?xml version="1.0"?>


<!-- 2. XML-Document -->
<!DOCTYPE catalog SYSTEM "cat.dtd">
<catalog>
<product>
<productname>mirror</productname>
```

```
<storagenumber>3455</storagenumber>
<dimension unit="m">
<width>1</width>
<heigth>1,5</heigth>
</dimension>
<description>blue frame</description>
<supplier>Schmid Co.</supplier>
<supplier>Mayer GmbH</supplier>
</product>
</catalog>
```

Define a possible XML Schema for the 2.XML documents.

COMMON SCHEMA :

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

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

<!-- definition of simple elements -->
<xs:element name="height" type="xs:decimal"/>
<xs:element name="width" type="xs:decimal"/>
<xs:element name="depth" type="xs:decimal"/>
<xs:element name="productname" type="xs:string"/>
<xs:element name="storagenumber" type="xs:integer"/>
<xs:element name="description" type="xs:string"/>
<xs:element name="supplier" type="xs:string"/>

<!-- definition of attributes -->
<xs:attribute name="unit" type="xs:string"/>

<!-- definition of complex elements -->

<xs:element name="dimension">
 <xs:complexType>
  <xs:sequence>
   <xs:element ref="height" use="required"/>
   <xs:element ref="width" maxOccurs="1" minOccurs="0"/>
   <xs:element ref="depth" maxOccurs="1" minOccurs="0"/>
   <xs:attribute ref="unit" use="required"/>
  </xs:sequence>
 </xs:complexType>
</xs:element>

<xs:element name="product">
 <xs:complexType>
```

```
  <xs:sequence>
   <xs:element ref="productname" use="required"/>
   <xs:element ref="storagenumber" use="required"/>
   <xs:element ref="dimension" use="required"/>
   <xs:element ref="description"/>
   <xs:element ref="supplier"/>
  </xs:sequence>
 </xs:complexType>
</xs:element>

<xs:element name="catalog">
 <xs:complexType>
  <xs:sequence>
   <xs:element ref="product" maxOccurs="unbounded"/>
  </xs:sequence>
 </xs:complexType>
</xs:element>

</xs:schema>
```

4. Consider now an extension of our original DTD, where the element juicer has attributes.
   <!ELEMENT juicers (juicer)*>
   <!ELEMENT juicer (name, image, description, warranty?,
   weight?, cost+, retailer)>
   <!ATTLIST juicer id ID #REQUIRED
   electric (true|false) #REQUIRED
   type (press | gear | centrifugal) #REQUIRED>
   <!ELEMENT name (#PCDATA)>
   <!ELEMENT image (#PCDATA)>
   <!ELEMENT description (#PCDATA)>
   <!ELEMENT warranty (#PCDATA)>
   <!ELEMENT weight (#PCDATA)>
   <!ELEMENT cost (#PCDATA)>
   <!ELEMENT retailer (#PCDATA)>
   Create corresponding attributes in the XML schema.

New Simple Elements :

```
<!-- definition of new simple elements -->
```

```
<xs:element name="weight" type="xs:decimal"/>
```

```
<xs:element name="cost" type="xs:decimal"/>
<xs:element name="warranty" type="xs:string"/>
<xs:element name="retailer" type="xs:string"/>
<xs:element name="image" type="xsd:base64Binary"/>

<!-- definition of new attributes -->
<xs:attribute name="id" type="xs:string">
  <xs:simpleType>
    <xs:restriction base="xs:integer">
      <xs:pattern value="[0-9]+"/>
    </xs:restriction>
  </xs:simpleType>
</xs:attribute>
<xs:attribute name="electric" type="xs:string">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:pattern value="(\btrue\b|\bfalse\b){1}"/>
    </xs:restriction>
  </xs:simpleType>
</xs:attribute>
<xs:attribute name="type" type="xs:string">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:pattern value="(\bpress\b|\bgear\b|\bcentrifugal\b){1}"/>
    </xs:restriction>
  </xs:simpleType>
</xs:attribute>

<!-- definition of new complex elements -->

<xs:element name="juicer">
 <xs:complexType>
  <xs:sequence>
   <xs:element ref="productname"/>
   <xs:element ref="image"/>
   <xs:element ref="description"/>
   <xs:element ref="warranty" maxOccurs="1"/>
   <xs:element ref="weight" maxOccurs="1"/>
   <xs:element ref="cost" minOccurs="1"/>
   <xs:element ref="retailer"/>
   <xs:attribute ref="ID" use="required"/>
   <xs:attribute ref="electric" use="required"/>
   <xs:attribute ref="type" use="required"/>
  </xs:sequence>
 </xs:complexType>
```

```
</xs:element>

<xs:element name="juicers">
 <xs:complexType>
  <xs:sequence>
   <xs:element ref="juicer" maxOccurs="unbounded"/>
  </xs:sequence>
 </xs:complexType>
</xs:element>

<xs:element name="catalog">
 <xs:complexType>
  <xs:sequence>
   <xs:element ref="product" maxOccurs="unbounded"/>
   <xs:element ref="juicers" maxOccurs="unbounded"/>
  </xs:sequence>
 </xs:complexType>
</xs:element>
```