COHORT-2

ENTERPRISE PROGRAMMING

1. **To create a simple XML document to display the address book.**

**Solution:**

<?xml version="1.0"?>

[<addressbook>](file:///D:\2020-21_EVEN%20SEM\ENTERPIRSE%20PROGRAMMING\XML\simpleXMLdocument.xml)

[<details>](mailto:%3cdetails%3e)

<name>haran</name>

<email>[haranbabu@kluniversity.in</email](mailto:haranbabu@kluniversity.in%3c/email)>

<location>vijayawada</location>

<dob>23/11/80</dob>

</details>

[<details>](mailto:%3cdetails%3e)

<name>raj</name>

<email>[guruharan@gmail.com</email](mailto:guruharan@gmail.com%3c/email)>

<location>hyderabad</location>

<dob>23/11/85</dob>

</details>

</addressbook>

1. **To create a xml document and database for importing and exporting xml document into database.**
2. **To create a simple XSLT transformation from XSL to XML.**

**<?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>  
 <body>  
 <h2>My CD Collection</h2>  
 <table border="1">  
 <tr bgcolor="#9acd32">  
 <th>Title</th>  
 <th>Artist</th>  
 </tr>  
 <xsl:for-each select="catalog/cd">  
 <tr>  
 <td><xsl:value-of select="title"/></td>  
 <td><xsl:value-of select="artist"/></td>  
 </tr>  
 </xsl:for-each>  
 </table>  
 </body>  
 </html>  
</xsl:template>  
  
</xsl:stylesheet>

1. **Create a XSD and DTD for Student information as given below xml document.**

<student> --->root element

<regno>170024</regno> -->integer

<name>Ramu</name> -->string

<avg>92</avg> -->decimal

<dob>15-08-1992</dob> -->date

<time></time> -->time

<mobile no="9849984900"/> -->complexType ( -->ineteger)

<distinction>YES</distinction> -->boolean

</student>

Student.xsd

<?xml version="1.0" encoding="utf-8"?>

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

<xs:element name="student">

<xs:complexType>

<xs:sequence>

<xs:element name="regno" type="xs:unsignedInt" />

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

<xs:element name="avg" type="xs:unsignedByte" />

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

<xs:element name="time" />

<xs:element name="mobile\_no" type="xs:unsignedLong" />

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

</xs:sequence>

</xs:complexType>

</xs:element>

</xs:schema>

Student.dtd

<?xml encoding="UTF-8"?>

<!ELEMENT student (regno,name,avg,dob,time,mobile\_no,distinction)>

<!ATTLIST student

xmlns CDATA #FIXED ''>

<!ELEMENT regno (#PCDATA)>

<!ATTLIST regno

xmlns CDATA #FIXED ''>

<!ELEMENT name (#PCDATA)>

<!ATTLIST name

xmlns CDATA #FIXED ''>

<!ELEMENT avg (#PCDATA)>

<!ATTLIST avg

xmlns CDATA #FIXED ''>

<!ELEMENT dob (#PCDATA)>

<!ATTLIST dob

xmlns CDATA #FIXED ''>

<!ELEMENT time (#PCDATA)>

<!ATTLIST time

xmlns CDATA #FIXED ''>

<!ELEMENT mobile\_no (#PCDATA)>

<!ATTLIST mobile\_no

xmlns CDATA #FIXED ''>

<!ELEMENT distinction (#PCDATA)>

<!ATTLIST distinction

xmlns CDATA #FIXED ''>

**5.Create XML document which store the information about**

**Library/Bookstore. more than 5 books.**

Solution: bookdetails.xml:

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

<bookstore>

<book category="COOKING">

<title lang="en">Everyday Italian</title>

<author>Giada De Laurentiis</author>

<year>2005</year>

<price>30.00</price>

</book>

<book category="CHILDREN">

<title lang="en">Harry Potter</title>

<author>J K. Rowling</author>

<year>2005</year>

<price>29.99</price>

</book>

<book category="WEB">

<title lang="en">XQuery Kick Start</title>

<author>James McGovern</author>

<author>Per Bothner</author>

<author>Kurt Cagle</author>

<author>James Linn</author>

<author>Vaidyanathan Nagarajan</author>

<year>2003</year>

<price>49.99</price>

</book>

<book category="WEB">

<title lang="en">Learning XML</title>

<author>Erik T. Ray</author>

<year>2003</year>

<price>39.95</price>

</book>

</bookstore>

**6.Create XML document which store the information about**

**employee. more than 5.(Self descriptive)**

solution: employee.xml

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

<document>

<employee>

<name>

<name>haran</name>

</name>

<hiredate>October 15, 2005</hiredate>

<projects>

<project>

<product>Printer</product>

<id>111</id>

<price>$111.00</price>

</project>

<project>

<product>Laptop</product>

<id>222</id>

<price>$989.00</price>

</project>

</projects>

</employee>

<employee>

<name>

<name>rajesh</name>

</name>

<hiredate>October 20, 2005</hiredate>

<projects>

<project>

<product>Desktop</product>

<id>333</id>

<price>$2995.00</price>

</project>

<project>

<product>Scanner</product>

<id>444</id>

<price>$200.00</price>

</project>

</projects>

</employee>

<employee>

<name>

<lastname>Gable</lastname>

<firstname>Clark</firstname>

</name>

<hiredate>October 25, 2005</hiredate>

<projects>

<project>

<product>Keyboard</product>

<id>555</id>

<price>$129.00</price>

</project>

<project>

<product>Mouse</product>

<id>666</id>

<price>$25.00</price>

</project>

</projects>

</employee>

</document>