Assignment A4

Professor: Jeremy Harper **TAs**: Bhavana Dhonvan, Latha Sree Rayala

Student Name: Deepak Rajput **Program**: MS Applied Data Science

Question 1. Investigate the Healthcare CCDA standard and the FHIR Standard. Write a one paragraph short summary of the difference between these XML standards. Include this in your paragraph.

Answer: FHIR (Fast healthcare Interoperability Resources) and C-CDA (Consolidated Clinical Document Architecture) are the two primary data formats used to electronically share medical records. Both are developed by HL7 (Health level 7) the standards development organization dedicated to digital health interoperability. Both are open source, supports structured and unstructured data and combine information into standardized documents types.
Basic differences: C-CDA requires special parsing to extract individual data points where as FHIR is easy to search and separate every aspect of patient encounters. C-CDA is best for industries which requires fast implementation, prefer to receive individual data files and already using CDA-compatible system. Where as FHIR is best for organizations which have well-defined use cases, know which specific data elements (like medications) they need and previously worked with healthcare data (especially JSON-based FHIR).
Tech. differences: C-CDA easy to implement but hard to parse where as FHIR is more intensive to implement but easier to parse. C-CDA particle store data for 2 days and FHIR store for 30 days. C-CDA use XML including PNGs and PDFs, whereas FHIR use JSON data formats. C-CDA is downloadable via REST API and FHIR follows REST FHIR R4 Spec. C-CDA being phased out and particle can transform into FHIR format and FHIR can largely codified as new standard.

Question 2. Update the following XML file to be a patient with your name and fictional details.

Answer: Required Patient details code with my name and dummy details when DTD declared inside XML file:

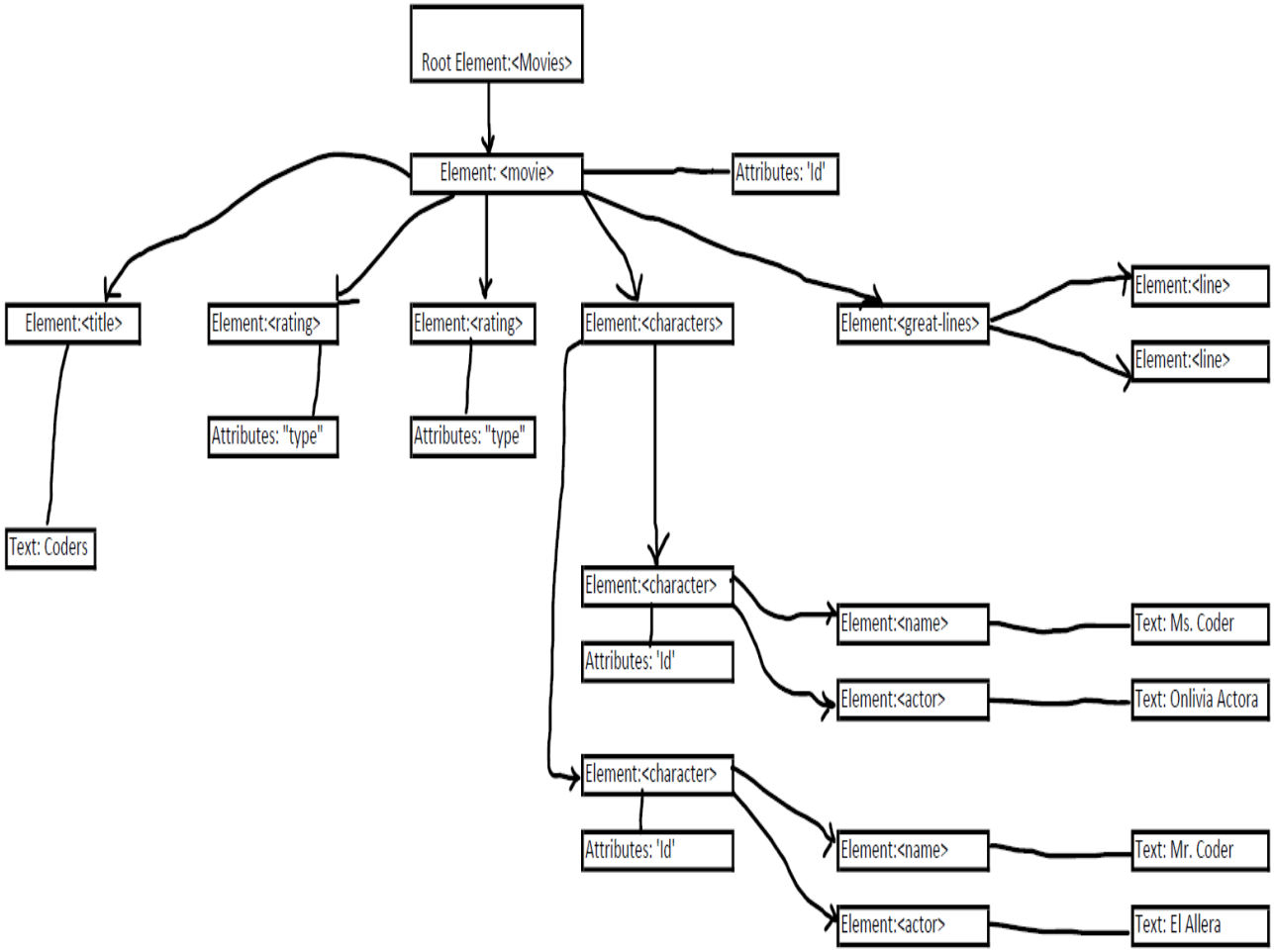
```
<?xml version="1.0"?>
<!DOCTYPE Patient_details [
<!ELEMENT Patient_details (Name,Age,Height,Weight)>
<!ELEMENT Name (#PCDATA)>
<!ELEMENT Age (#PCDATA)>
<!ELEMENT Height (#PCDATA)>
<!ELEMENT Weight (#PCDATA)>
]>
<Patient_details>
<Name>Deepak Rajput</Name>
<Age>32</Age>
<Height>165 cm</Height>
<Weight>72 kg</Weight>
</Patient_details>
```

Required XML file:

```
<?xml version='1.0' standalone='yes'?>
<Patient_details>
<Name>Deepak Rajput</Name>
<Age>32</Age>
<Height>165 cm</Height>
<Weight>72 kg</Weight>
</Patient_details>
```

Question 3. Draw the tree structure of the XML file **For the given XML file linked below about the collections of movies.**

Answer:
Required reference Tree diagram for above XML file: as per same diagram need to draw for other 2 movies also but structure remains almost same for other 2 movies also.



Required Full Tree diagram for above XML file:

```
movies
    movie(id="m001")
        title
        rating(type="thumbs")
        rating(type="stars")
        characters
            character(id="c1001")
                name
                actor
            character(id="c1002")
                name
                actor
        great-lines
            line
            line
    movie(id="m002")
        title
        rating(type="stars")
        rating(type="thumbs")
        characters
            character(id="c2001")
                name
                actor
    movie(id="m003")
        title
        rating(type="thumbs")
        rating(type="stars")
        characters
            character(id="c3001")
                name
                actor
            character(id="c3002")
                name
                actor
            character(id="c3003")
                name
                actor
```

Question 4. Write a corresponding DTD for the given XML file.
Required DTD file:

```
<!DOCTYPE movies[
<!ELEMENT movies (movie+)>
<!ELEMENT movie (title, rating+, characters, great-lines)>
<!ATTLIST movie id CDATA #REQUIRED>
<!ELEMENT title (#PCDATA)>
<!ELEMENT rating (#PCDATA)>
<!ATTLIST rating type CDATA #REQUIRED>
<!ELEMENT characters (character+)>
<!ELEMENT character (name, actor)>
<!ATTLIST character id CDATA #REQUIRED>
<!ELEMENT name (#PCDATA)>
<!ELEMENT actor (#PCDATA)>
<!ELEMENT great-lines (line+)>
<!ELEMENT line (#PCDATA)>
]>
```

Question 5. Write a FHIR based XSLT file and test the output from w3Schools, to generate the report as in the image below I recommend leveraging the generic patient list example which can support multiple patients but you are welcome to choose other lists. http://hl7.org/fhir/stu3/list-example.xml.htmlLinks to an external site.

www.w3schools.com/xml/tryxslt.asp?xmlfile=cdcatalog&xsltfile=cdcatalog_choose

Answer: Used below XML file:

```
<?xml version="1.0" encoding="UTF-8"?>
<patients>
<patient>
<name>Deepak Rajput</name>
<age>34</age>
<sex>male</sex>
<contact>317-292-7890</contact>
</patient>
<patient>
<name>John smith</name>
<age>19</age>
<sex>male</sex>
<contact>317-232-7556</contact>
</patient>
<patient>
<name>sid mathew</name>
<age>23</age>
<sex>female</sex>
<contact>317-222-9999</contact>
</patient>
<patient>
<name>sony d</name>
<age>29</age>
<sex>female</sex>
<contact>123-456-9000</contact>
</patient>
</patients>
```

XSLT file for the same is given below:

```
<?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>Patients List</title>
<style>
table {
border-collapse: collapse;
width: 100%;
} th, td {
text-align: left;
padding: 8px;
border-bottom: 1px solid #ddd;
border-top: 1px solid #ddd;
border-right: 1px solid #ddd;
} th {
background-color: lightgreen;
}
tr.female {
background-color: pink;
}
</style>
</head>
<body>
<h2 style="background-color: lightgreen; padding: 10px;">Patients List</h2>
<table>
<tr>
<th>Name</th>
<th>Age</th>
<th>Sex</th>
<th>contact</th>
</tr>
<xsl:for-each select="patients/patient">
<tr>
<xsl:if test="sex = 'female'">
<xsl:attribute name="class">female</xsl:attribute>
</xsl:if>
<td><xsl:value-of select="name"/></td>
<td><xsl:value-of select="age"/></td>
<td><xsl:value-of select="sex"/></td>
<td><xsl:value-of select="contact"/></td>
</tr>
</xsl:for-each>
</table>
</body>
</html>
</xsl:template>
</xsl:stylesheet>
```

Result:

Patients List

Name	Age	Sex	contact
Deepak Rajput	34	male	317-292-7890
John smith	19	male	317-232-7556
sid mathew	23	female	317-222-9999
sony d	29	female	123-456-9000