JEGYZŐKÖNYV

Adatkezelés XML környezetben

Féléves feladat

Készítette: Radócz Nikolett

NeptunKód:WSI08I

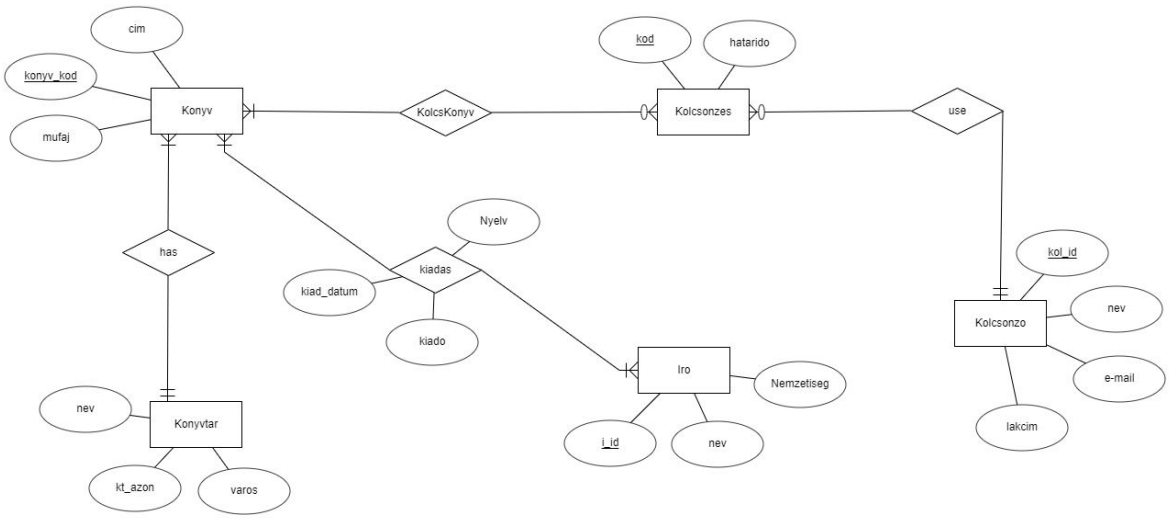
A feladat leírása:

A feladatom az Adatbázis rendszerek I. tárgy beadandó témája alapján készült el. A Könyvtárak, Könyv, Író, Kölcsönzés és Kölcsönző közötti kapcsolatokat mutatja be.

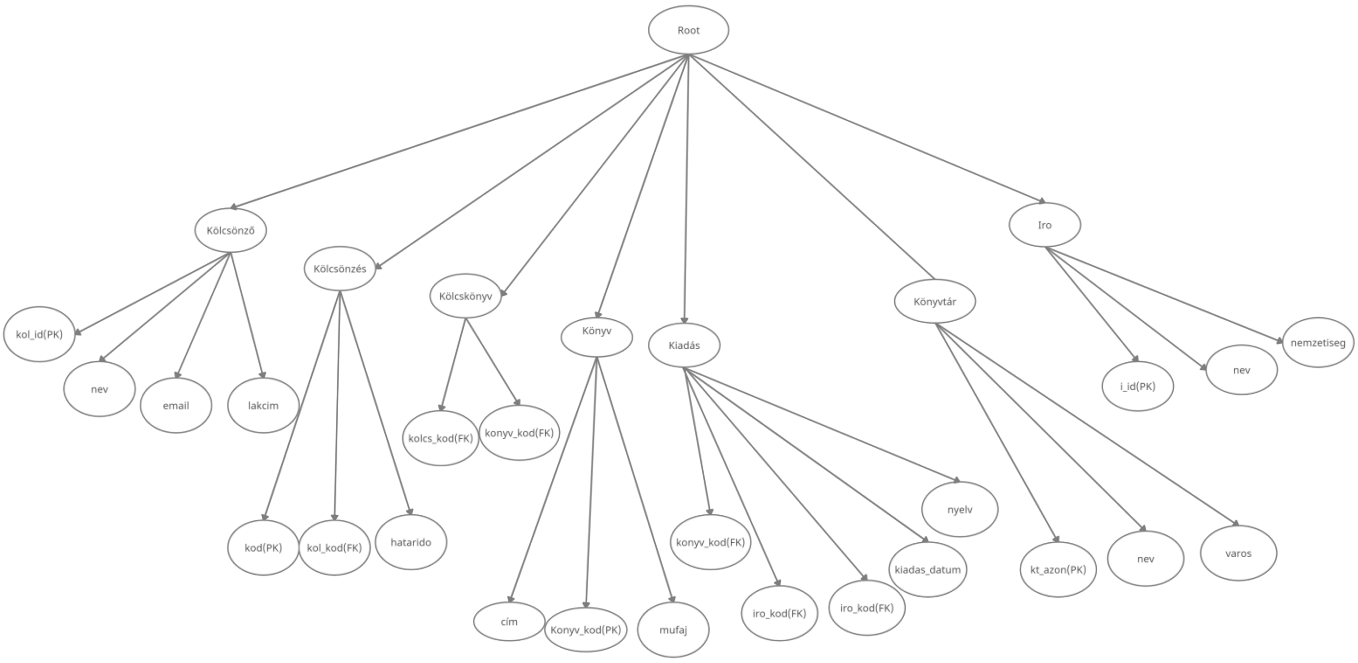
1. feladat

A feladat első részében az ER modellt, az XDM modellt az XML és az XMLSchema dokumentumot szemléltetem.

1.1 ER modell



1.2 XDM modell



1.2 XML kód

```xml
<?xml version="1.0" encoding="UTF-8"?>
<root xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:SchemaLocation="XMLSchemaWSI08I.xsd">

<kolcsonzo kol_id="01">
    <nev>Bene Anna</nev>
    <email>banna@gmail.com</email>
    <lakcim>Miskolc</lakcim>
</kolcsonzo>
<kolcsonzo kol_id="01">
    <nev>Kiss Jozsef</nev>
    <email>jozsi@gmail.com</email>
    <lakcim>Budapest</lakcim>
</kolcsonzo>


<kolcsonzes kod="01" kol_id="01">
    <hatarido>2020.12.20</hatarido>
</kolcsonzes>
<kolcsonzes kod="02" kol_id="02">
    <hatarido>2021.01.10</hatarido>
</kolcsonzes>


<iro i_id="01">
    <nev>Vavyan Fable</nev>
    <nemzetiseg>Magyar</nemzetiseg>
</iro>
<iro i_id="02">
    <nev>Jennifer A. Nelsen</nev>
    <nemzetiseg>Angol</nemzetiseg>
</iro>
<iro i_id="03">
    <nev>Tasmina Perry</nev>
    <nemzetiseg>Angol</nemzetiseg>
</iro>

```

```xml

<konyvtar kt_azon="01">
    <nev>II. Rákóczi Ferenc Megyei és Városi Könyvtár</nev>
    <varos>Miskolc</varos>
</konyvtar>


<konyv konyv_kod="01" kt_azon="01">
    <cim>Vészbejárat</cim>
    <műfaj>krimi</műfaj>
</konyv>
<konyv konyv_kod="02" kt_azon="01">
    <cim>Sztárok ügyvédje</cim>
    <műfaj>romantikus</műfaj>
</konyv>
<konyv konyv_kod="03" kt_azon="01">
    <cim>The Shadow Throne</cim>
    <műfaj>misztikus</műfaj>
</konyv>


<kiadas i_id="01" konyv_kod="01">
    <kiadas_datum>2000</kiadas_datum>
    <nyelv>magyar</nyelv>
</kiadas>
<kiadas i_id="02" konyv_kod="3">
    <kiadas_datum>2015</kiadas_datum>
    <nyelv>angol</nyelv>
</kiadas>
<kiadas i_id="02" konyv_kod="02">
    <kiadas_datum>2012</kiadas_datum>
    <nyelv>német</nyelv>
</kiadas>


<kolcskonyv kod="01" konyv_kod="03"/>
<kolcskonyv kod="01" konyv_kod="02"/>
<kolcskonyv kod="02" konyv_kod="01"/>
</root>
```

1.3 XML Schema

```xml
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
<xs:element name="root">
    <xs:complexType>
        <xs:sequence>
<xs:element name="kolcsonzo" maxOccurs="unbounded" >
    <xs:complexType>
        <xs:sequence>
            <xs:element name="nev" type="xs:string"/>
            <xs:element name="email" type="xs:string"/>
            <xs:element name="lakcim" type="xs:string"/>
        </xs:sequence>
        <xs:attribute name="kol_id" type="xs:int"/>
    </xs:complexType>
</xs:element>

<xs:element name="kolcsonzes" maxOccurs="unbounded">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="hatarido" type="xs:date"/>
        </xs:sequence>
        <xs:attribute name="kod" type="xs:int"/>
        <xs:attribute name="kol_id" type="xs:int"/>
    </xs:complexType>
</xs:element>

<xs:element name="iro" maxOccurs="unbounded">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="nev" type="xs:string"/>
            <xs:element name="nemzetiseg" type="xs:string"/>
        </xs:sequence>
        <xs:attribute name="i_id" type="xs:int"/>
    </xs:complexType>
</xs:element>
```

```xml
<xs:element name="konyvtar" maxOccurs="unbounded">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="nev" type="xs:string"/>
            <xs:element name="varos" type="xs:string"/>
        </xs:sequence>
        <xs:attribute name="kt_azon" type="xs:int"/>
    </xs:complexType>
</xs:element>

<xs:element name="konyv" maxOccurs="unbounded">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="cim" type="xs:string"/>
            <xs:element name="műfaj" type="xs:string"/>
        </xs:sequence>
        <xs:attribute name="konyv_kod" type="xs:int"/>
    </xs:complexType>
</xs:element>

<xs:element name="kiadas" maxOccurs="unbounded">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="kiadas_datum"  type="xs:int"/>
            <xs:element name="nyelv"  type="xs:string"/>
        </xs:sequence>
        <xs:attribute name="i_id" type="xs:int"/>
        <xs:attribute name="konyv_kod" type="xs:int"/>
    </xs:complexType>
</xs:element>


<xs:element name="kolcskonyv">
    <xs:complexType>
        <xs:attribute name="kol_id" type="xs:int"/>
        <xs:attribute name="konyv_kod" type="xs:int"/>
    </xs:complexType>
</xs:element>

        </xs:sequence>
    </xs:complexType>

<xs:key name="key1">
    <xs:selector xpath="kolcsonzo"></xs:selector>
    <xs:field xpath="@kol_id"></xs:field>
</xs:key>

<xs:key name="key2">
    <xs:selector xpath="iro"></xs:selector>
    <xs:field xpath="@i_id"></xs:field>
</xs:key>

<xs:key name="key3">
    <xs:selector xpath="kolcsonzes"></xs:selector>
    <xs:field xpath="@kod"></xs:field>
</xs:key>

<xs:key name="key4">
    <xs:selector xpath="konyv"></xs:selector>
    <xs:field xpath="@konyv_kod"></xs:field>
</xs:key>

<xs:key name="key5">
    <xs:selector xpath="konyvvtar"></xs:selector>
    <xs:field xpath="@kt_azon"></xs:field>
</xs:key>

<xs:keyref refer="key1" name="ref1">
            <xs:selector xpath="kolcsonzes" />
            <xs:field xpath="@kol_id" />
</xs:keyref>

<xs:keyref refer="key2" name="ref2">
            <xs:selector xpath="kiadas" />
            <xs:field xpath="@i_id" />
</xs:keyref>


<xs:keyref refer="key4" name="ref3">
             <xs:selector xpath="kiadas" />
             <xs:field xpath="@konyv_kod" />
</xs:keyref>

<xs:keyref refer="key3" name="ref4">
             <xs:selector xpath="kolcskonyv" />
             <xs:field xpath="@kod" />
</xs:keyref>

<xs:keyref refer="key4" name="ref5">
             <xs:selector xpath="kolcskonyv" />
             <xs:field xpath="@konyv_kod" />
</xs:keyref>
<xs:keyref refer="key5" name="ref6">
             <xs:selector xpath="konyv" />
             <xs:field xpath="@kt_azon" />
</xs:keyref>

</xs:element>


</xs:schema>
```

2. feladat

2.1 DOM adatolvasás

```java
package hu.domparse.wsi08i;

import java.io.File;

public class DOMReadWSI08I {

    public static void main(String[] args) throws SAXException, IOException, ParserConfigurationException {
        //XML file objektum letrehozasa
        File xmlFile = new File("xmlWSI08I.xml");

        // Document Builder letrehozasa
        DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance();
        DocumentBuilder dBuilder = factory.newDocumentBuilder();


        Document doc = dBuilder.parse(xmlFile);

        //normalizalas
        doc.getDocumentElement().normalize();

        //Gyoker elem kiiratasa
        Element root = doc.getDocumentElement();
        System.out.println("Gyoker elem: " + root.getNodeName());


        NodeList NL1 = doc.getElementsByTagName("kolcsonzo");

        for (int i = 0; i < NL1.getLength(); i++) {
            Node nNode = NL1.item(i);

            if (nNode.getNodeType() == Node.ELEMENT_NODE) {

                System.out.println();

                Element elem = (Element) nNode;

                System.out.println("Elem:" + nNode.getNodeName());
                String kol_id = elem.getAttribute("kol_id");

                Node n1 = elem.getElementsByTagName("nev").item(0);
                String nev = n1.getTextContent();

                Node n2 = elem.getElementsByTagName("email").item(0);
                String email = n2.getTextContent();

                Node n3 = elem.getElementsByTagName("lakcim").item(0);
                String lakcim = n3.getTextContent();

                System.out.println("id: " + kol_id);
                System.out.println("Neve: " + nev);
                System.out.println("email: " + email);
                System.out.println("lakcim: " + lakcim);

            }
        }
NodeList NL2 = doc.getElementsByTagName("kolcsonzes");

        for (int i = 0; i < NL2.getLength(); i++) {
            Node nNode = NL2.item(i);

            if (nNode.getNodeType() == Node.ELEMENT_NODE) {

                System.out.println();

                Element elem = (Element) nNode;

                System.out.println("Elem:" + nNode.getNodeName());
                String kod = elem.getAttribute("kod");
                String kol_id = elem.getAttribute("kol_id");

                Node n1 = elem.getElementsByTagName("hatarido").item(0);
                String hat = n1.getTextContent();



                System.out.print("kod: " + kod +", ");
                System.out.println("kolcsonzes id: " + kol_id);
                System.out.println("hatarido: " + hat);


            }
```

```
        }

        NodeList NL3 = doc.getElementsByTagName("iro");

        for (int i = 0; i < NL3.getLength(); i++) {
            Node nNode = NL3.item(i);

            if (nNode.getNodeType() == Node.ELEMENT_NODE) {
                Element elem = (Element) nNode;
                System.out.println();

                System.out.println("Elem:" + nNode.getNodeName());
                String iid = elem.getAttribute("i_id");

                Node n1 = elem.getElementsByTagName("nev").item(0);
                String nev = n1.getTextContent();

                Node n2 = elem.getElementsByTagName("nemzetiseg").item(0);
                String nemz = n1.getTextContent();

                System.out.println("Iro id: " + iid);
                System.out.println("nev: " + nev);
                System.out.println("nemzetiseg: " + nemz);

            }
        }

        NodeList NL4 = doc.getElementsByTagName("konyvtar");

        for (int i = 0; i < NL4.getLength(); i++) {
            Node nNode = NL4.item(i);

            if (nNode.getNodeType() == Node.ELEMENT_NODE) {
                Element elem = (Element) nNode;
                System.out.println();

                System.out.println("Elem:" + nNode.getNodeName());
                String ktazon = elem.getAttribute("kt_azon");

                Node n1 = elem.getElementsByTagName("nev").item(0);
                String nev = n1.getTextContent();

                Node n2 = elem.getElementsByTagName("varos").item(0);
                String varos = n1.getTextContent();

                System.out.println("Konyvtar azonosito: " + ktazon);
                System.out.println("nev: " + nev);
                System.out.println("varos: " + varos);

            }
        }

        NodeList NL5 = doc.getElementsByTagName("konyv");

        for (int i = 0; i < NL5.getLength(); i++) {
            Node nNode = NL5.item(i);

            if (nNode.getNodeType() == Node.ELEMENT_NODE) {
                Element elem = (Element) nNode;
                System.out.println();

                System.out.println("Elem:" + nNode.getNodeName());
                String kkod = elem.getAttribute("konyv_kod");
                String ktazon = elem.getAttribute("kt_azon");

                Node n1 = elem.getElementsByTagName("cim").item(0);
                String cim = n1.getTextContent();

                Node n2 = elem.getElementsByTagName("műfaj").item(0);
                String mufaj = n1.getTextContent();

                System.out.println("konyv kod: " + kkod);
                System.out.println("Konyvtar azonosito: " + ktazon);
                System.out.println("cim: " + cim);
                System.out.println("mufaj: " + mufaj);

            }
        }

        NodeList NL6 = doc.getElementsByTagName("kiadas");

        for (int i = 0; i < NL6.getLength(); i++) {
            Node nNode = NL6.item(i);

            if (nNode.getNodeType() == Node.ELEMENT_NODE) {
                Element elem = (Element) nNode;
                System.out.println();

                System.out.println("Elem:" + nNode.getNodeName());
                String iid = elem.getAttribute("i_id");
                String kkod = elem.getAttribute("konyv_kod");

                Node n1 = elem.getElementsByTagName("kiadas_datum").item(0);
                String kdatum = n1.getTextContent();

                Node n2 = elem.getElementsByTagName("nyelv").item(0);
                String nyelv = n1.getTextContent();

                System.out.println("Iro id: " + iid);
                System.out.println("Konyv kod: " + kkod);
                System.out.println("kiadas datum: " + kdatum);
                System.out.println("nyelv: " + nyelv);
```

```
195
196             }
197         }
198
199         NodeList NL7 = doc.getElementsByTagName("kolcskonyv");
200
201         for (int i = 0; i < NL7.getLength(); i++) {
202             Node nNode = NL7.item(i);
203
204             if (nNode.getNodeType() == Node.ELEMENT_NODE) {
205                 Element elem = (Element) nNode;
206                 System.out.println();
207
208                 System.out.println("Elem:" + nNode.getNodeName());
209                 String kod = elem.getAttribute("kod");
210                 String kkod = elem.getAttribute("konyv_kod");
211
212
213                 System.out.println("Kolcsonzes kodja:: " + kod);
214                 System.out.println("Konyv kod: " + kkod);
215
216             }
217         }
218
219     }
220
221 }
```

2.2 DOM adatmódosítás

```
1  package hu.domparse.wsi08i;
2
3  import java.io.File;
4  import java.io.IOException;
5  import javax.xml.parsers.DocumentBuilder;
6  import javax.xml.parsers.DocumentBuilderFactory;
7  import javax.xml.parsers.ParserConfigurationException;
8  import javax.xml.xpath.XPath;
9  import javax.xml.xpath.XPathConstants;
10 import javax.xml.xpath.XPathExpressionException;
11 import javax.xml.xpath.XPathFactory;
12 import org.w3c.dom.Document;
13 import org.w3c.dom.Element;
14 import org.w3c.dom.Node;
15 import org.w3c.dom.NodeList;
16 import org.xml.sax.SAXException;
17 public class DOMModifyWSI08I {
18
19     public static void main(String[] args) throws ParserConfigurationException, SAXException, IOException, XPathExpressionException {
20         // meghatarozza az XML file helyét
21                 File xmlFile = new File("xmlWSI08I.xml");
22
23                 // letrehozza a Document Buildert
24                 DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance();
25                 DocumentBuilder builder = factory.newDocumentBuilder();
26
27                 // DOM objektumma konvertal
28                 Document doc = builder.parse(xmlFile);
29
30                 // normalizal
31                 doc.getDocumentElement().normalize();
32
33                 // xPath segitségével módositandó elem meghatározása
34                 XPath xPath = XPathFactory.newInstance().newXPath();
35
36                 // elem amit modositunk
37                 String expr1 = "/root/kolcsonzo[@kol_id = '01']/nev/text()";
38                 NodeList nodeList1 = (NodeList) xPath.compile(expr1).evaluate(doc, XPathConstants.NODESET);
39
40                 // Mire szeretnénk átirni
41                 nodeList1.item(0).setNodeValue("Radocz Nikolett");
```

```
42
43                 // Elérés megadása
44                 String expr2 = "/root/kolcsonzo[@kol_id = '01']";
45                 nodeList1 = (NodeList) xPath.compile(expr2).evaluate(doc, XPathConstants.NODESET);
46
47                 //masik elemet i modositunk
48                 String expr3 = "/root/iro[@i_id = '02']/nev/text()";
49                 NodeList nodeList2 = (NodeList) xPath.compile(expr3).evaluate(doc, XPathConstants.NODESET);
50
51                 nodeList2.item(0).setNodeValue("temp");
52
53                 String expr4 = "/root/iro[@i_id = '02']";
54                 nodeList2 = (NodeList) xPath.compile(expr4).evaluate(doc, XPathConstants.NODESET);
55
56                 for (int i = 0; i < nodeList1.getLength(); i++) {
57                     Node nNode = nodeList1.item(i);
58
59                     System.out.println("\nCurrent Element : " + nNode.getNodeName());
60
61                 // Ciklussal kilirattatjuk az értékeket
62                     if (nNode.getNodeType() == Node.ELEMENT_NODE) {
63                         Element eElement = (Element) nNode;
64
65                         System.out.println("Kolcsonzo id: " + eElement.getAttribute("kol_id"));
66
67                         System.out.println("Nev: " + eElement.getElementsByTagName("nev").item(0).getTextContent());
68
69                         System.out.println("E-mail: " + eElement.getElementsByTagName("email").item(0).getTextContent());
70
71                         System.out.println("Lakcim: " + eElement.getElementsByTagName("lakcim").item(0).getTextContent());
72                     }
73                 }
74
75                 for (int i = 0; i < nodeList2.getLength(); i++) {
76                     Node nNode = nodeList2.item(i);
77
78                     System.out.println("\nCurrent Element : " + nNode.getNodeName());
79
80                     if (nNode.getNodeType() == Node.ELEMENT_NODE) {
81                         Element eElement = (Element) nNode;
```

```
82
83                          System.out.println("Iro id: " + eElement.getAttribute("i_id"));
84
85                          System.out.println("Iro neve: " + eElement.getElementsByTagName("nev").item(0).getTextContent());
86
87                          System.out.println("Iro nemzetisege: " + eElement.getElementsByTagName("nemzetiseg").item(0).getTextContent());
88                      }
89                  }
90
91      }
92
93  }
94
```