

JEGYZŐKÖNYV

Adatkezelés XML környezetben
Féléves feladat

Készítette: **Szabó Péter**

Neptunkód: **ID4JQM**

Feladateleírás

Az Adatkezelés XML Környezetben című tárgy féléves feladatának jegyzőkönyvét egy **Iskola nyilvántartásáról** szóló adatbázis megtervezéséről írom.

Az adatbázis ER modell az Iskola, Osztály, Tanár és Diák egyedekből áll. Az Iskola és az osztályok (tartozik) között 1:1 kapcsolat áll fent. Az osztály és a tanár (tanít), illetve a diák (tanul) között pedig N:M kapcsolat áll fent.

Az ER modell leképzése után az adatbázist XDM modellre konvertáltam át, melyben nagy segítség volt a W3 Konzorcium irányelveinek megismerése. [1]

Az XDM modell alapján az érvényes XML dokumentum elkészítése következett. Az életszerű adatok generálásánál a képzelőerő fontos szerepet játszott a feladat elkészítésében.

Az XML dokumentum alapján az XMLSchema elkészítésében az ID és IDREF típusok mellett megvalósítottam a KEY és KEYREF elemeket is, így egyszerűbbé lehetett tenni az attribútumokra való hivatkozást.

A féléves feladat második részében az érvényes XML dokumentum DOM program általi feldolgozását valósítottam meg.

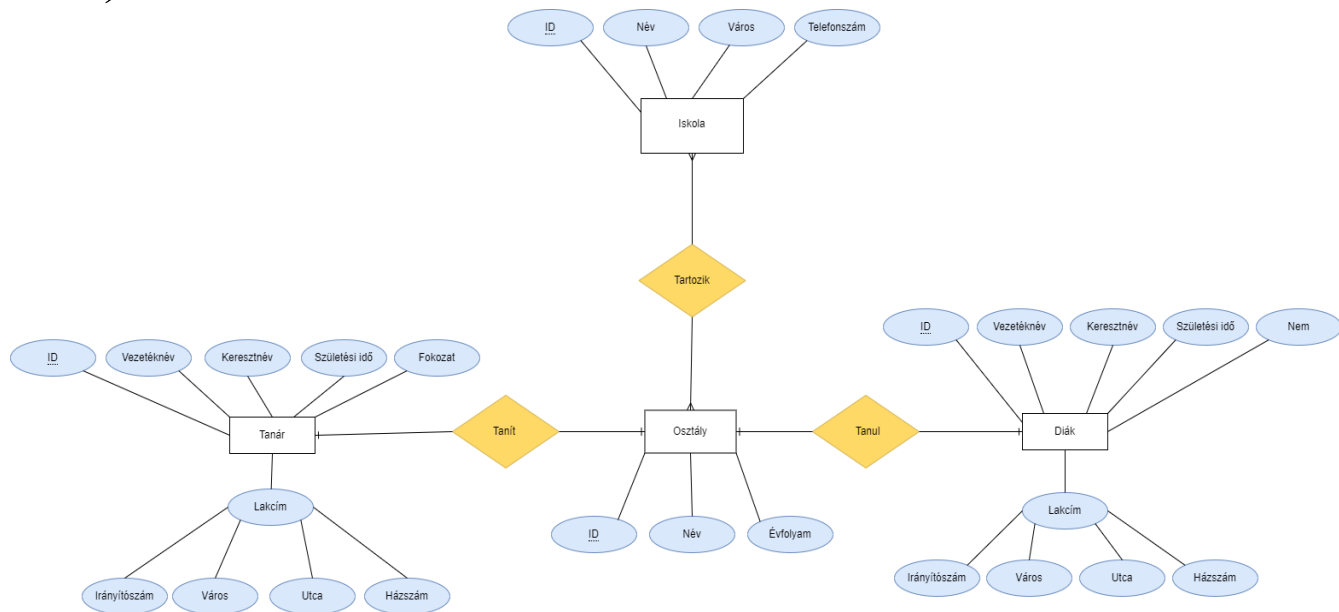
Az adatolvasás során a gyökérelem tulajdonságai olvasása után az osztály tulajdonságait dolgoztam fel. Külön függvényt írtam a tanárok és a diákok kiíratására. Az osztályokon végig járva az osztályokhoz tartozó diákok és tanárok listázását oldottam meg, így az adatok rendezetten kerültek kiíratásra.

Az adatmódosítás során egy konzolos alkalmazást készítettem mellyel az osztályok neveit szabadon lehet módosítani. Az alkalmazás listázza az osztályokat, majd kiírja az osztály jelenlegi nevét és adatolvasás után a konzolban megadott értékre állítja át az osztály nevét és azt beírja az XML fájlba.

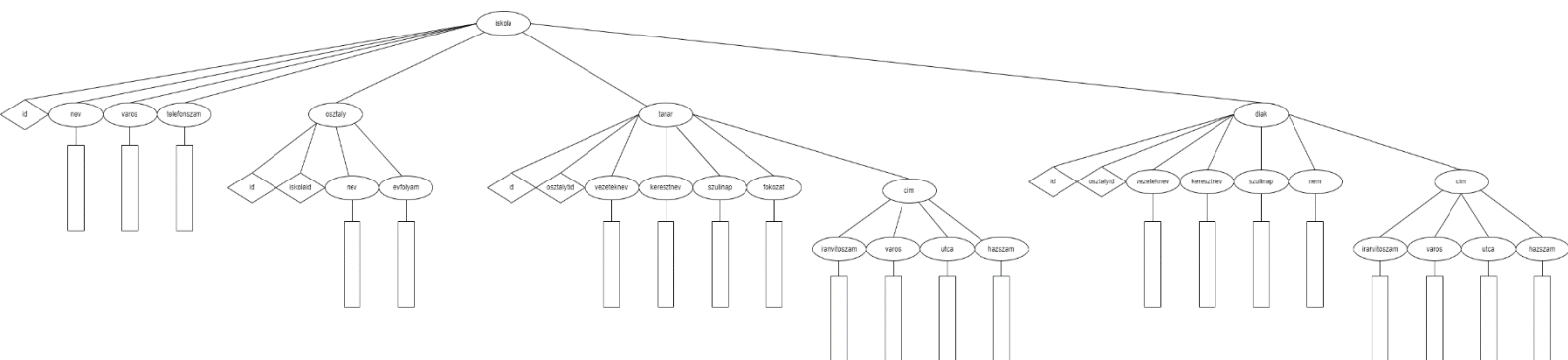
[1] <https://www.w3.org/XML/Datamodel.html>

1.Feladat

1.A) Az adatbázis ER modell



1.B) Adatbázis konvertálása XDM modellre



1.C) Az XDM modell alapján XML dokumentum készítése

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?><?xml-model href="XMLSchemaID4JQM.xsd" type="application/xml"
schematypens="http://www.w3.org/2001/XMLSchema"?><iskola id="IS001">
```

```
<nev>Móra Ferenc Iskola</nev>
<varos>Miskolc</varos>
<telefonszam>+362345079</telefonszam>
<tanarok>
  <tanar id="TAN001" osztalyid="CS101">
    <vezeteknev>Varga</vezeteknev>
    <keresztnev>Zsófia</keresztnev>
    <szulinap>1988-03-22</szulinap>
    <cim>
      <iranyitoszam>3535</iranyitoszam>
      <varos>Miskolc</varos>
      <utca>Virág</utca>
      <hazszam>1</hazszam>
    </cim>
    <fokozat>TAN1</fokozat>
  </tanar>
  <tanar id="TAN002" osztalyid="CS102">
    <vezeteknev>Szabó</vezeteknev>
    <keresztnev>Dzsénifer</keresztnev>
    <szulinap>1977-10-20</szulinap>
    <cim>
      <iranyitoszam>3544</iranyitoszam>
      <varos>Miskolc</varos>
      <utca>Jácint</utca>
      <hazszam>10</hazszam>
    </cim>
    <fokozat>TAN3</fokozat>
  </tanar>
  <tanar id="TAN003" osztalyid="CS103">
    <vezeteknev>Nagy</vezeteknev>
    <keresztnev>Ádám</keresztnev>
    <szulinap>1970-01-01</szulinap>
    <cim>
      <iranyitoszam>3520</iranyitoszam>
      <varos>Miskolc</varos>
      <utca>Valami</utca>
      <hazszam>6</hazszam>
    </cim>
    <fokozat>TAN2</fokozat>
  </tanar>
  <tanar id="TAN004" osztalyid="CS104">
    <vezeteknev>Antal</vezeteknev>
    <keresztnev>Csaba</keresztnev>
    <szulinap>1973-12-20</szulinap>
    <cim>
      <iranyitoszam>3535</iranyitoszam>
      <varos>Miskolc</varos>
      <utca>Nyár</utca>
      <hazszam>10</hazszam>
    </cim>
    <fokozat>TAN3</fokozat>
  </tanar>
</tanarok>
```

```

</tanarok>
<diakok>
  <diak id="DI000" osztalyid="CS101">
    <vezeteknev>Varga</vezeteknev>
    <keresztnev>Béla</keresztnev>
    <szulinap>2005-10-20</szulinap>
    <nem>F</nem>
    <cim>
      <iranyitoszam>3535</iranyitoszam>
      <varos>Miskolc</varos>
      <utca>Kertész</utca>
      <hazszam>11</hazszam>
    </cim>
  </diak>
  <diak id="DI001" osztalyid="CS101">
    <vezeteknev>Nagy</vezeteknev>
    <keresztnev>Kinga</keresztnev>
    <szulinap>2001-01-04</szulinap>
    <nem>N</nem>
    <cim>
      <iranyitoszam>3535</iranyitoszam>
      <varos>Miskolc</varos>
      <utca>Eper</utca>
      <hazszam>9</hazszam>
    </cim>
  </diak>
  <diak id="DI002" osztalyid="CS101">
    <vezeteknev>Gábori</vezeteknev>
    <keresztnev>Tamás</keresztnev>
    <szulinap>2005-05-14</szulinap>
    <nem>F</nem>
    <cim>
      <iranyitoszam>3535</iranyitoszam>
      <varos>Miskolc</varos>
      <utca>Farkas</utca>
      <hazszam>30</hazszam>
    </cim>
  </diak>
  <diak id="DI003" osztalyid="CS101">
    <vezeteknev>Nagy</vezeteknev>
    <keresztnev>Gizell</keresztnev>
    <szulinap>2006-05-14</szulinap>
    <nem>N</nem>
    <cim>
      <iranyitoszam>3535</iranyitoszam>
      <varos>Miskolc</varos>
      <utca>Áprád</utca>
      <hazszam>10</hazszam>
    </cim>
  </diak>

```

```

<diak id="DI004" osztalyid="CS101">
  <vezeteknev>Kis</vezeteknev>
  <keresztnev>Gábor</keresztnev>
  <szulinap>2010-10-24</szulinap>
  <nem>F</nem>
  <cim>
    <iranyitoszam>3535</iranyitoszam>
    <varos>Miskolc</varos>
    <utca>Halom</utca>
    <hazszam>33</hazszam>
  </cim>
</diak>
<diak id="DI005" osztalyid="CS102">
  <vezeteknev>Balogh</vezeteknev>
  <keresztnev>Kinga</keresztnev>
  <szulinap>2005-05-14</szulinap>
  <nem>N</nem>
  <cim>
    <iranyitoszam>3535</iranyitoszam>
    <varos>Miskolc</varos>
    <utca>Eper</utca>
    <hazszam>15</hazszam>
  </cim>
</diak>
<diak id="DI006" osztalyid="CS102">
  <vezeteknev>Vándor</vezeteknev>
  <keresztnev>Áron</keresztnev>
  <szulinap>2005-05-14</szulinap>
  <nem>F</nem>
  <cim>
    <iranyitoszam>3535</iranyitoszam>
    <varos>Miskolc</varos>
    <utca>Banán</utca>
    <hazszam>35</hazszam>
  </cim>
</diak>
<diak id="DI007" osztalyid="CS102">
  <vezeteknev>Répás</vezeteknev>
  <keresztnev>Mónika</keresztnev>
  <szulinap>2005-05-14</szulinap>
  <nem>N</nem>
  <cim>
    <iranyitoszam>3535</iranyitoszam>
    <varos>Miskolc</varos>
    <utca>Szabó</utca>
    <hazszam>16</hazszam>
  </cim>
</diak>

```

```
<diak id="DI008" osztalyid="CS102">
  <vezeteknev>Vándor</vezeteknev>
  <keresztnev>Karcsi</keresztnev>
  <szulinap>2010-05-14</szulinap>
  <nem>F</nem>
  <cim>
    <iranyitoszam>3535</iranyitoszam>
    <varos>Miskolc</varos>
    <utca>Virágos</utca>
    <hazszam>15</hazszam>
  </cim>
</diak>
<diak id="DI009" osztalyid="CS102">
  <vezeteknev>Váradi</vezeteknev>
  <keresztnev>András</keresztnev>
  <szulinap>2007-04-14</szulinap>
  <nem>F</nem>
  <cim>
    <iranyitoszam>3535</iranyitoszam>
    <varos>Miskolc</varos>
    <utca>Vilmos</utca>
    <hazszam>15</hazszam>
  </cim>
</diak>
<diak id="DI010" osztalyid="CS103">
  <vezeteknev>Cirmos</vezeteknev>
  <keresztnev>Csilla</keresztnev>
  <szulinap>2007-11-08</szulinap>
  <nem>N</nem>
  <cim>
    <iranyitoszam>3535</iranyitoszam>
    <varos>Miskolc</varos>
    <utca>Kappa</utca>
    <hazszam>45</hazszam>
  </cim>
</diak>
<diak id="DI011" osztalyid="CS103">
  <vezeteknev>Farkas</vezeteknev>
  <keresztnev>Ferenc</keresztnev>
  <szulinap>2006-11-08</szulinap>
  <nem>F</nem>
  <cim>
    <iranyitoszam>3535</iranyitoszam>
    <varos>Miskolc</varos>
    <utca>Király</utca>
    <hazszam>27</hazszam>
  </cim>
</diak>
```



```
<diak id="DI012" osztalyid="CS103">
  <vezeteknev>Diadon</vezeteknev>
  <keresztnev>Csilla</keresztnev>
  <szulinap>2007-11-08</szulinap>
  <nem>N</nem>
  <cim>
    <iranyitoszam>3535</iranyitoszam>
    <varos>Miskolc</varos>
    <utca>Harapós</utca>
    <hazszam>95</hazszam>
  </cim>
</diak>
<diak id="DI013" osztalyid="CS103">
  <vezeteknev>Fricska</vezeteknev>
  <keresztnev>Zoltán</keresztnev>
  <szulinap>2007-11-08</szulinap>
  <nem>F</nem>
  <cim>
    <iranyitoszam>3535</iranyitoszam>
    <varos>Miskolc</varos>
    <utca>Kis</utca>
    <hazszam>45</hazszam>
  </cim>
</diak>
<diak id="DI014" osztalyid="CS103">
  <vezeteknev>Fesz</vezeteknev>
  <keresztnev>Andrea</keresztnev>
  <szulinap>2007-11-08</szulinap>
  <nem>N</nem>
  <cim>
    <iranyitoszam>3535</iranyitoszam>
    <varos>Miskolc</varos>
    <utca>Kormos</utca>
    <hazszam>65</hazszam>
  </cim>
</diak>
<diak id="DI015" osztalyid="CS104">
  <vezeteknev>Károlyi</vezeteknev>
  <keresztnev>Andrienn</keresztnev>
  <szulinap>2008-07-01</szulinap>
  <nem>N</nem>
  <cim>
    <iranyitoszam>3535</iranyitoszam>
    <varos>Miskolc</varos>
    <utca>Harminc</utca>
    <hazszam>30</hazszam>
  </cim>
</diak>
```

```

<diak id="DI016" osztalyid="CS104">
  <vezeteknev>Váradi</vezeteknev>
  <keresztnev>Miklós</keresztnev>
  <szulinap>2008-11-01</szulinap>
  <nem>F</nem>
  <cim>
    <iranyitoszam>3535</iranyitoszam>
    <varos>Miskolc</varos>
    <utca>Isten</utca>
    <hazszam>33</hazszam>
  </cim>
</diak>
<diak id="DI017" osztalyid="CS104">
  <vezeteknev>Mikszáth</vezeteknev>
  <keresztnev>Kálmán</keresztnev>
  <szulinap>2008-01-01</szulinap>
  <nem>F</nem>
  <cim>
    <iranyitoszam>3535</iranyitoszam>
    <varos>Miskolc</varos>
    <utca>Hatos</utca>
    <hazszam>10</hazszam>
  </cim>
</diak>
<diak id="DI018" osztalyid="CS104">
  <vezeteknev>Részegi</vezeteknev>
  <keresztnev>Vivien</keresztnev>
  <szulinap>2007-04-01</szulinap>
  <nem>N</nem>
  <cim>
    <iranyitoszam>3535</iranyitoszam>
    <varos>Miskolc</varos>
    <utca>Árpád</utca>
    <hazszam>30</hazszam>
  </cim>
</diak>
<diak id="DI019" osztalyid="CS104">
  <vezeteknev>Siker</vezeteknev>
  <keresztnev>Bence</keresztnev>
  <szulinap>2001-07-01</szulinap>
  <nem>F</nem>
  <cim>
    <iranyitoszam>3535</iranyitoszam>
    <varos>Miskolc</varos>
    <utca>Végh</utca>
    <hazszam>40</hazszam>
  </cim>
</diak>
</diakok>

```

```
<osztalyok>
  <osztaly id="CS101" iskolaid="IS001">
    <nev>A</nev>
    <evfolyam>Első</evfolyam>
  </osztaly>
  <osztaly id="CS102" iskolaid="IS001">
    <nev>B</nev>
    <evfolyam>Második</evfolyam>
  </osztaly>
  <osztaly id="CS103" iskolaid="IS001">
    <nev>C</nev>
    <evfolyam>Harmadik</evfolyam>
  </osztaly>
  <osztaly id="CS104" iskolaid="IS001">
    <nev>D</nev>
    <evfolyam>Negyedik</evfolyam>
  </osztaly>
</osztalyok>
</iskola>
```

1.D) Az XML dokumentum alapján XMLSchema készítése

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

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="iskola">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="nev" maxOccurs="1" type="xs:string"/>
        <xs:element name="varos" maxOccurs="1" type="xs:string"/>
        <xs:element name="telefonszam" maxOccurs="1" type="xs:string"/>
        <xs:element name="tanarok" maxOccurs="1">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="tanar" type="tanartype" maxOccurs="unbounded"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="diakok" maxOccurs="1">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="diak" type="diaktype" maxOccurs="unbounded"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="osztalyok" maxOccurs="1">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="osztaly" type="osztalytype" maxOccurs="unbounded"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
      <xs:attribute name="id" type="xs:ID" use="required"/>
    </xs:complexType>
    <xs:key name="iskolakey">
      <xs:selector xpath="."/>
      <xs:field xpath="@id"/>
    </xs:key>
    <xs:key name="osztalykey">
      <xs:selector xpath="osztalyok/osztaly"/>
      <xs:field xpath="@id"/>
    </xs:key>
    <xs:key name="tanarkey">
      <xs:selector xpath="tanarok/tanar"/>
      <xs:field xpath="@id"/>
    </xs:key>
    <xs:key name="diakkey">
      <xs:selector xpath="diakok/diak"/>
      <xs:field xpath="@id"/>
    </xs:key>
    <xs:keyref refer="iskolakey" name="osztalyref">
      <xs:selector xpath="csportok/osztaly"/>
      <xs:field xpath="@iskolaid"/>
    </xs:keyref>
    <xs:keyref refer="osztalykey" name="diakref">
      <xs:selector xpath="diakok/diak"/>
      <xs:field xpath="@osztalyid"/>
    </xs:keyref>
  </xs:element>
</xs:schema>
```

```
<xs:keyref refer="osztalykey" name="tanarref">
  <xs:selector xpath="tanarok/tanar"/>
  <xs:field xpath="@osztalyid"/>
</xs:keyref>
</xs:element>
<xs:complexType name="cimtype">
  <xs:sequence>
    <xs:element name="iranyitoszam" type="xs:integer"/>
    <xs:element name="varos" type="xs:string"/>
    <xs:element name="utca" type="xs:string"/>
    <xs:element name="hazszam" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="tanartype" >
  <xs:sequence>
    <xs:element name="vezeteknev" type="xs:string"/>
    <xs:element name="keresztnev" type="xs:string"/>
    <xs:element name="szulinap" type="xs:date"/>
    <xs:element name="cim" type="cimtype"/>
    <xs:element name="fokozat" type="xs:string"/>
  </xs:sequence>
  <xs:attribute name="id" type="xs:ID" use="required"/>
  <xs:attribute name="osztalyid" type="xs:IDREF" use="optional"/>
</xs:complexType>
<xs:complexType name="diaktype">
  <xs:sequence>
    <xs:element name="vezeteknev" type="xs:string"/>
    <xs:element name="keresztnev" type="xs:string"/>
    <xs:element name="szulinap" type="xs:date"/>
    <xs:element name="nem" type="xs:string"/>
    <xs:element name="cim" type="cimtype"/>
  </xs:sequence>
  <xs:attribute name="id" type="xs:ID" use="required"/>
  <xs:attribute name="osztalyid" type="xs:IDREF" use="optional"/>
</xs:complexType>
<xs:complexType name="osztalytype">
  <xs:sequence>
    <xs:element name="nev" type="xs:string"/>
    <xs:element name="evfolyam" type="xs:string"/>
  </xs:sequence>
  <xs:attribute name="id" type="xs:ID"/>
  <xs:attribute name="iskolaid" type="xs:IDREF"/>
</xs:complexType>
</xs:schema>
```

2.Feladat

2.A) Adatolvasás – DOMReadID4JQM.java

```
package hu.domparse.ID4JQM;

import java.io.*;
import javax.xml.parsers.*;
import org.w3c.dom.*;
import org.xml.sax.*;

public class DOMReadID4JQM {
    public static void main(String[] args) throws SAXException, IOException, ParserConfigurationException {

        // Az XML file objektumának létrehozása
        File xmlFile = new File("XMLID4JQM.xml");

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

        // Az XML fájl DOM objektummá való konvertálása.
        Document doc = dBuilder.parse(xmlFile);

        // A dokumentum normalizálása.
        doc.getDocumentElement().normalize();

        // A gyökér elem nevének meghatározása és az iskolához kapcsolódó elemek és attribútumok kiírása.
        System.out.println("A dokumentum gyökér eleme: " + doc.getDocumentElement().getNodeName());
        System.out.println(doc.getDocumentElement().getAttribute("id"));
        System.out.println(doc.getElementsByTagName("nev").item(0).getTextContent());
        System.out.println(doc.getElementsByTagName("varos").item(0).getTextContent());
        System.out.println(doc.getElementsByTagName("telefonszam").item(0).getTextContent());

        // Az összes <osztaly> elem beillesztése egy NodeList-be és az osztályok és attribútumaik kilistázása.
        NodeList nList = doc.getElementsByTagName("osztaly");
        System.out.println("\nA(z) iskola osztályai:\n");
        for (int i = 0; i < nList.getLength(); i++) {
            Node nNode = nList.item(i);

            if (nNode.getNodeType() == Node.ELEMENT_NODE) {
                System.out.println("\n-----\n");

                Element elem = (Element) nNode;
                String id = elem.getAttribute("id");
                listOsztaly(doc, id);
            }
        }
    }
}
```

```

/*
For ciklussal végig menni a NodeList-en és kiírni a dokumentum összes elemét és attribútumát.
Az osztályokon megy keresztül a ciklus.
Minden osztályhoz érve a osztály ID-t átadva meghívja a diák és a tanár kiíró függvényeket.
*/
for (int i = 0; i < nList.getLength(); i++) {
    Node nNode = nList.item(i);

    if (nNode.getNodeType() == Node.ELEMENT_NODE) {
        System.out.println("-----");
        System.out.println();
        System.out.println("-----");

        Element elem = (Element) nNode;

        String id = elem.getAttribute("id");

        String iskolaid = elem.getAttribute("iskolaid");

        Node node1 = elem.getElementsByTagName("nev").item(0);
        String nev = node1.getTextContent();

        Node node2 = elem.getElementsByTagName("evfolyam").item(0);
        String evfolyam = node2.getTextContent();

        System.out.println("Osztály ID: " + id);
        System.out.println("Iskola ID: " + iskolaid);
        System.out.println("\tOsztálynév: " + nev);
        System.out.println("\tÉvfolyam: " + evfolyam);

        System.out.println("\nA(z) " + nev + " osztály tanárai:\n");
        ListTanar(doc, id);

        System.out.println("\nA(z) " + nev + " osztály diákjai:\n");
        ListDiak(doc, id);
    }
}
}

```

```

//Az osztályokat és attributumait listázó metódus.
public static void ListOsztaly (Document doc, String osztalyid) {
    NodeList nList = doc.getElementsByTagName("osztaly");

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

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

            if (elem.getAttribute("id").toString().equals(osztalyid)) {
                String id = elem.getAttribute("id");
                String iskolaid = elem.getAttribute("iskolaid");

                Node node1 = elem.getElementsByTagName("nev").item(0);
                String osztalynev = node1.getTextContent();

                Node node2 = elem.getElementsByTagName("evfolyam").item(0);
                String evfolyam = node2.getTextContent();

                System.out.println("Osztály ID: " + id);
                System.out.println("\tIskola ID: " + iskolaid);
                System.out.println("\tOsztály név: " + osztalynev);
                System.out.println("\tÉvfolyam: " + evfolyam);

            }
        }
    }
}

```



```

//A diákokat és attribútumait listázó metódus.
public static void ListDiak (Document doc, String osztalyid) {
    NodeList nList = doc.getElementsByTagName("diak");

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

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

            if (elem.getAttribute("osztalyid").toString().equals(osztalyid)) {
                String id = elem.getAttribute("id");

                Node node1 = elem.getElementsByTagName("vezeteknev").item(0);
                String vezeteknev = node1.getTextContent();

                Node node2 = elem.getElementsByTagName("keresztnev").item(0);
                String keresztnév = node2.getTextContent();

                Node node3 = elem.getElementsByTagName("szulinap").item(0);
                String szulinap = node3.getTextContent();

                Node node4 = elem.getElementsByTagName("nem").item(0);
                String nem = node4.getTextContent();

                Node node5 = elem.getElementsByTagName("cim").item(0);
                String cim = node5.getTextContent();

                System.out.println("Diák ID: " + id);
                System.out.println("\tVezetéknév: " + vezeteknev);
                System.out.println("\tKeresztnév: " + keresztnév);
                System.out.println("\tSzületési idő: " + szulinap);
                System.out.println("\tNem: " + nem);
                System.out.println("\tLakcím: " + cim);
            }
        }
    }
}

```

```

//A tanárokat és attribútumait listázó metódus.
public static void ListTanar (Document doc, String osztalyid) {
    NodeList nList = doc.getElementsByTagName("tanar");

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

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

            if (elem.getAttribute("osztalyid").toString().equals(osztalyid)) {
                String id = elem.getAttribute("id");

                Node node1 = elem.getElementsByTagName("vezeteknev").item(0);
                String vezeteknev = node1.getTextContent();

                Node node2 = elem.getElementsByTagName("keresztnev").item(0);
                String keresztnév = node2.getTextContent();

                Node node3 = elem.getElementsByTagName("szulinap").item(0);
                String szulinap = node3.getTextContent();

                Node node4 = elem.getElementsByTagName("fokozat").item(0);
                String fokozat = node4.getTextContent();

                Node node5 = elem.getElementsByTagName("cim").item(0);
                String cim = node5.getTextContent();

                System.out.println("Tanár ID: " + id);
                System.out.println("\tVezetéknév: " + vezeteknev);
                System.out.println("\tKeresztnév: " + keresztnév);
                System.out.println("\tSzületési idő: " + szulinap);
                System.out.println("\tFokozat: " + fokozat);
                System.out.println("\tLakcím: " + cim);
            }
        }
    }
}

```

2.B) Adatmódosítás – DOMModifyID4JQM.java

```
package hu.domparse.ID4JQM;

import java.io.*;
import java.util.Scanner;
import javax.xml.parsers.*;
import javax.xml.transform.*;
import javax.xml.transform.dom.*;
import javax.xml.transform.stream.*;

import org.w3c.dom.*;
import org.xml.sax.*;

public class DOMModifyID4JQM {
    public static void main(String[] args) throws SAXException, IOException,
        ParserConfigurationException, TransformerConfigurationException, TransformerException {
        // A forrás és cél XML file objektumának létrehozása.
        File xmlFile = new File("XMLID4JQM.xml");
        //File xmlOutputFile = new File("XMLID4JQMResults.xml");

        // Scanner olvasó példányosítása.
        Scanner in = new Scanner(System.in);

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

        // Az XML fájl DOM objektummá való konvertálása.
        Document doc = dBuilder.parse(xmlFile);

        // A dokumentum normalizálása.
        doc.getDocumentElement().normalize();

        // Az <osztaly> tag elemek kiválasztása.
        NodeList nList = doc.getElementsByTagName("osztaly");
        System.out.println("Osztály nevek felülírása:\n");

        // Jelenlegi osztályok kiírása.
        for (int i = 0; i < nList.getLength(); i++) {
            Node nNode = nList.item(i);
            Element elem = (Element) nNode;
            String id = elem.getAttribute("id");

            System.out.println((i+1)+". sorszámú osztály adatai: ");
            DOMReadID4JQM.ListOsztaly(doc, id);
            System.out.println("-----");
        }
    }
}
```

```

//Osztály nevének módosítása.
for (int i = 0; i < nList.getLength(); i++) {
    // Az <osztaly> tag (i)-ik eleme.
    Node nNode = nList.item(i);

    Element elem = (Element) nNode;

    // A név node kiválasztása.
    Node node1 = elem.getElementsByTagName("nev").item(0);
    String nev = node1.getTextContent();

    // Adat bekérése és a node értékének megváltoztatása.
    System.out.println("Az osztály jelenlegi neve: " + nev);
    System.out.println("Adja meg az osztály új nevét: ");
    String newname = in.next();
    node1.setTextContent(newname);
}

// A Scanner bezárása.
in.close();

// A módosított dokuemtum kiírása file-ba.
TransformerFactory transformerFactory = TransformerFactory.newInstance();
Transformer transformer = transformerFactory.newTransformer();

// DOMSource, mint a módosított adatok ideiglenes tárolója a kiíráshoz.
DOMSource source = new DOMSource(doc);

// A kiírandó file előkészítése.
StreamResult result = new StreamResult(new File("XMLID4JQM.xml"));

// A file kiírása.
transformer.transform(source, result);
}

```

2.C) Adatlekérdezés – DOMQueryID4JQM.java

```
package hu.domparse.ID4JQM;

import java.io.File;
import java.io.IOException;
import java.util.Scanner;

import javax.xml.parsers.DocumentBuilder;
import javax.xml.parsers.DocumentBuilderFactory;
import javax.xml.parsers.ParserConfigurationException;
import javax.xml.transform.TransformerException;

import org.w3c.dom.Document;
import org.w3c.dom.Element;
import org.w3c.dom.Node;
import org.w3c.dom.NodeList;
import org.xml.sax.SAXException;

public class DOMQueryID4JQM {

    public static void main(String[] args) throws ParserConfigurationException, IOException, SAXException, TransformerException {
        // TODO Auto-generated method stub
        File xmlFile = new File("XMLID4JQM.xml"); //Xml fájl bekérése
        DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance(); //Olvasás lehetővé tétele
        DocumentBuilder dBuilder = factory.newDocumentBuilder();
        Document doc = dBuilder.parse(xmlFile);
        doc.getDocumentElement().normalize();

        System.out.println("Gyökér elem: " + doc.getDocumentElement().getNodeName()); //Gyökér elem kiírása
        System.out.println("-----");
        LoadQuery(doc);
    }

    public static void LoadQuery(Document doc) throws TransformerException {
        NodeList diaknodeList = doc.getElementsByTagName("diak"); // Diákok listája
        NodeList osztalynList = doc.getElementsByTagName("osztaly"); //Osztályok listája

        Element element = null;
        Node nNode = null;
        //Diákok ID és osztály ID szerinti kilistázása, hogy később tudjunk választani diákok lekérdezésre
        for (int i = 0; i < diaknodeList.getLength(); i++) {
            nNode = diaknodeList.item(i);
            element = (Element) nNode;
            String id = element.getAttribute("id");
            String osztid = element.getAttribute("osztalyid");
            String vnev = element.getElementsByTagName("vezeteknev").item(0).getTextContent();
            String knev = element.getElementsByTagName("keresztnev").item(0).getTextContent();
            System.out.println("\n" + (i + 1) + ") ID: " + id + "\nOsztály ID: " + osztid + "\nVezetékeve: " + vnev + "\nKeresztnév: " + knev + "\n");
        }

        Scanner in = new Scanner(System.in);

        System.out.println("Adja meg a diák ID-t:"); // Diák kiválasztása scanner inputról ID szerint
        String diakid = in.next();
    }
}
```

```

        for (int i = 0; i < diaknodelist.getLength(); i++) {
            nNode = diaknodelist.item(i);
            element = (Element) nNode;
            int db = element.getElementsByTagName("vezeteknev").getLength();
            if(nNode.getNodeType() == Node.ELEMENT_NODE) { // Ha megfelelő ID-t adtunk meg akkor elindítja a lekérdező metódust (QueryXML)

                if(element.getAttribute("id").equals(diakid)) {
                    QueryXML(db,doc,element,osztalynList);
                }
            }
        }
    }
    in.close();
}
// A lekérdezett diák adatainak kiíratási metódusa
public static void QueryXML(int db,Document doc, Element element, NodeList osztalynList) {
    Node nNode = null;
    Element elem = null;
    String id = element.getAttribute("id");

    System.out.println("\nDiák adatai:\n");
    System.out.println("Diák ID: " + id);
    for (int i = 0; i < db; i++) {

        nNode = osztalynList.item(i); //Az átadott osztály nodelistából kiválasztjuk az i. elemet
        elem = (Element) nNode;

        //Stringbe szedett kiíratása az elemeknek és attributumoknak a gyökér elemből (iskola) és a diák és osztály elemből
        String diakcim = element.getElementsByTagName("cim").item(i).getTextContent();
        String diaknem = element.getElementsByTagName("nem").item(i).getTextContent();
        String osztid = element.getAttribute("osztalyid");
        String osztalynev = elem.getElementsByTagName("nev").item(i).getTextContent();
        String szulinap = element.getElementsByTagName("szulinap").item(i).getTextContent();
        String vname = element.getElementsByTagName("vezeteknev").item(i).getTextContent();
        String kname = element.getElementsByTagName("keresztnev").item(i).getTextContent();
        String iskolaneve = doc.getElementsByTagName("nev").item(i).getTextContent();

        System.out.println("Iskola neve: "+iskolaneve+"\nOsztályának ID-je: "+osztid+"\nOsztályának neve "+osztalynev+"\nVezetékneve: "+vname+
            "\nKeresztneve: "+kname+"\nSzületési ideje: "+szulinap+"\nNeme: "+diaknem+"\nLakcíme: "+diakcim);
    }
}
}

```