

JEGYZŐKÖNYV

Adatkezelés XML környezetben

Féléves feladat

A feladat címe

Készítette: Butella Bence Kristóf

Neptunkód: IVLJQO

Dátum:

A feladat leírása: A beadandó témája egy étel rendelő app adatbázisa ami kezeli a leadott rendeléseket a rendszer eltárolja az adott személy adatait illetve a rendelést hogy melyik étterem készítette és hogy mikor mit.

Étterem egyed tulajdonságai:

- ÉtteremID: Az étterem egyed elsődleges kulcsa, azonosítója.
- Név: Az étterem neve.
- Telefonszám: Az étterem telefonszáma.
- Weblap: Az étterem weblap elérhetőséget tárolom benne.

Vevő egyed tulajdonságai:

- Id: A vevő id-át tartalmazza mivel egy vevő több rendelés is adhat le így tudjuk meg különböztetni.
- Név: A vevő neve.
- Telefonszám: A vevő telefonszáma.
- Fizetés: Összetett elem ami a készpénzből áll össze meg a kártyás fizetésből.

Lakcím egyed:

- Id: Egy vevő több helyről is rendelheti az ételét ezért külön id-val kell rendelkeznie.
- Házsám: Fontos a rendelés pontosítása érdekében.
- Utca: Fontos a rendelés pontosítása érdekében.
- Város: Fontos a rendelés pontosítása érdekében mert nem biztos hogy azonos helyen van az étterem mint a rendelő lakcíme.

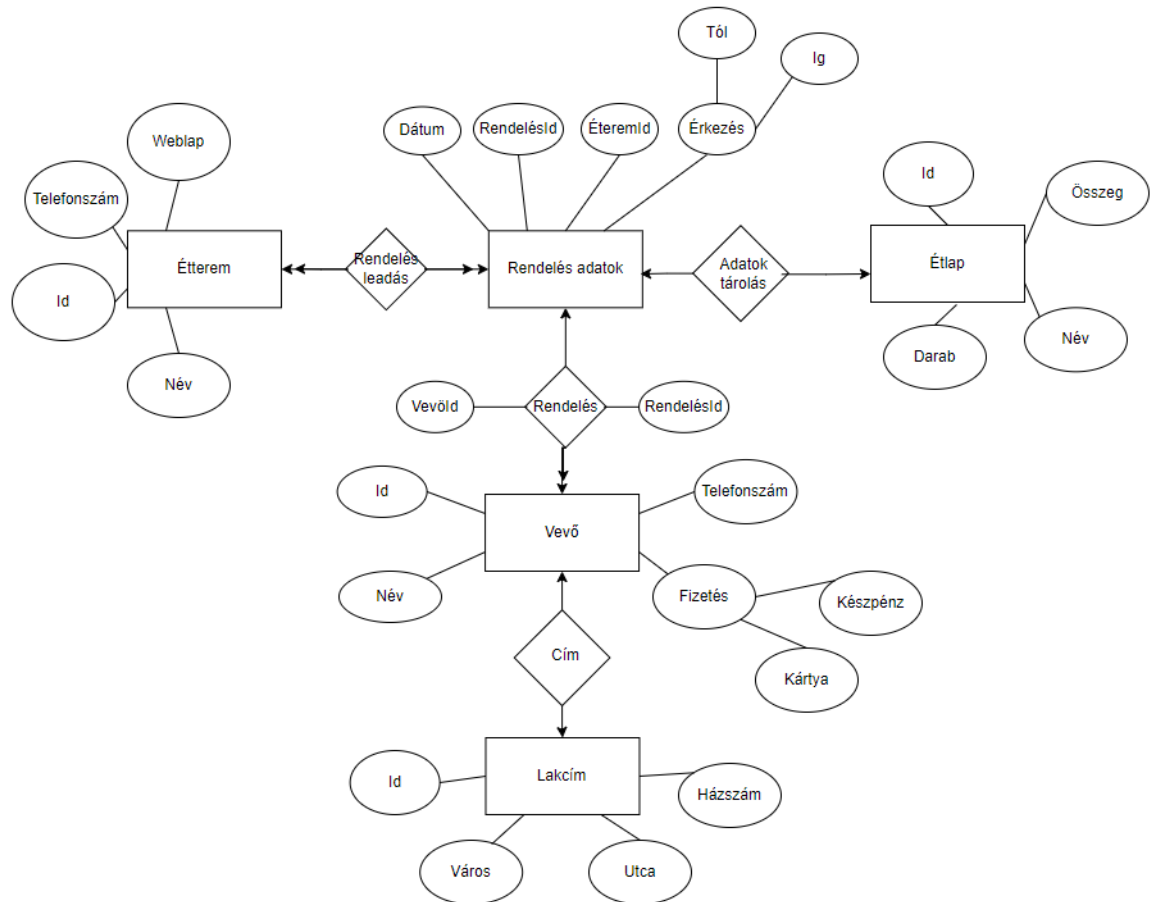
Étlap:

- Név: Az étel neve hogy mit rendelt a vevő.
- Darab: Meg rendelt ételek mennyisége.
- Összeg: Mennyibe került az adott rendel mennyiség.
- Id: Több rendelés leadás megkülönböztetése érdekében.

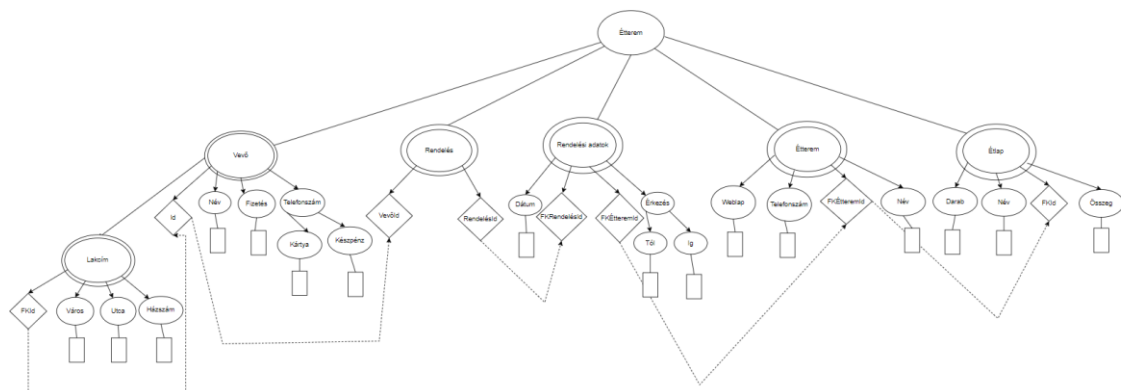
Rendelés adatok:

- Dátum: Mikor adtál le a rendelést.
- RendelésId: A rendelés Id-ja.
- ÉtteremId: Az étterem Id-ja.
- Érkezés: Összetett elem ami abból az érkezés idő intervallumijából áll mettől meddig fog meg érkezni az adott rendelés.

1. Feladat:
ER modell:



XDM modell:



XML modell alapján készült xml kód adatokkal:

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

```

<etterem xmlns="ivljqo" xsi:schemaLocation="ivljqo schema.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

```

<rendeles etteremid="string" rendelesid="string" />

<etterem etteremid="1">

<nev>Egyszer étterem</nev>

<telefonszam>06704564653</telefonszam>

<weblap>www.egyszeretterem.com</weblap>

</etterem>

<etterem etteremid="2">

<nev>Jó étterem</nev>

<telefonszam>06704569873</telefonszam>

<weblap>www.joetterem.com</weblap>

</etterem>

<etterem etteremid="3">

<nev>Volt étterem</nev>

<telefonszam>06704659873</telefonszam>

<weblap>www.voltetterem.com</weblap>

</etterem>

<vevo id="1">

<nev>Bence</nev>

<fizetes>

<keszpenz>10000</keszpenz>

<kartya>0</kartya>

</fizetes>

<telefonszam>06703333456</telefonszam>

<lakcim id="1">

<varos>Sátor</varos>

<iranyitoszam>3970</iranyitoszam>

<utca>Paloc</utca>

<hazszam>2</hazszam>

</lakcim>

</vevo>

<vevo id="2">

<nev>Zsófi</nev>

<fizetes>

<keszpenz>500</keszpenz>

<kartya>0</kartya>

</fizetes>

<telefonszam>06706343353</telefonszam>

<lakcim id="2">

<varos>Sátor</varos>

<iranyitoszam>3970</iranyitoszam>

<utca>Lolka</utca>

<hazszam>4</hazszam>

</lakcim>

</vevo>

<vevo id="3">

<nev>Sándor</nev>

<fizetes>

<keszpenz>1500</keszpenz>

<kartya>0</kartya>

</fizetes>

<telefonszam>06706343353</telefonszam>

<lakcim id="3">

<varos>Sátor</varos>

<iranyitoszam>3970</iranyitoszam>

<utca>Lolka</utca>

<hazszam>4</hazszam>

</lakcim>

</vevo>

<rendelesadatok rendelesid="1">

<datum>

<tol>2022.06.08.18:40</tol>

<ig>2022.06.08.19:30</ig>

```
</datum>
</rendelesadatok>
<rendelesadatok rendelesid="2">
  <datum>
    <tol>2022.07.09.18:40</tol>
    <ig>2022.07.09.19:30</ig>
  </datum>
</rendelesadatok>
<rendelesadatok rendelesid="3">
  <datum>
    <tol>2022.06.08.18:40</tol>
    <ig>2022.06.08.19:30</ig>
  </datum>
</rendelesadatok>
<etlap id="1">
  <nev>Pizza</nev>
  <darab>5</darab>
  <osszeg>10000</osszeg>
</etlap>
<etlap id="2">
  <nev>Hamburger</nev>
  <darab>1</darab>
  <osszeg>500</osszeg>
</etlap>
<etlap id="3">
  <nev>Gyros</nev>
  <darab>1</darab>
  <osszeg>1500</osszeg>
</etlap>
</etterem>
```

2. Feladat:

DomReadIVLJQO

Dom program adatkezeléshez a commentek a kódban olvashatóak:

Olvasás: a XMLIVLJQO.xml fájlból.

```
package hu.domparse.ivljqo;

import java.io.File;
import java.util.Iterator;

import javax.xml.parsers.DocumentBuilder;
import javax.xml.parsers.DocumentBuilderFactory;

import org.w3c.dom.Node;
import org.w3c.dom.NodeList;
import org.w3c.dom.Document;
import org.w3c.dom.Element;

public class DomReadIVLJQO {
    public static void main (String [] args) {

        try {
            //fájl helyének megadása
            File xml = new File("XMLIVLJQO.xml");
            //
            DocumentBuilderFactory dbf =
DocumentBuilderFactory.newInstance();

            //Dokumentum fa felépítése
            DocumentBuilder db = dbf.newDocumentBuilder();
            //xml beolvasás
            Document doc = db.parse(xml);
            //Gyökér beolvasása próbaként
            System.out.println(doc.getDocumentElement().getNodeName());

            //Minden adat kiírása
            //összeszedi az elemeket a fából
            NodeList nlVevo = doc.getElementsByTagName("vevo");
            NodeList nlEtterem = doc.getElementsByTagName("etterem");
            System.out.println("Vevők száma: " + nlVevo.getLength());
            System.out.println("Étterem száma: " + nlEtterem.getLength());

            //lista elemeire ciklus
            for (int i = 0; i < nlVevo.getLength(); i++) {
                Node n1 = nlVevo.item(i);

                if(n1.getNodeType() == Node.ELEMENT_NODE) {
                    Element e11 = (Element) n1;

                    //vevő adatainak kiírása
                    System.out.println((i+1) + ". vevő neve: " +
e11.getElementsByTagName("nev").item(0).getTextContent());
                    System.out.println((i+1) + ". vevő fizetése: " +
e11.getElementsByTagName("kartya").item(0).getTextContent());
                    System.out.println((i+1) + ". vevő készpénz: " +
e11.getElementsByTagName("keszpenz").item(0).getTextContent());
```

```

        System.out.println((i+1) + ". vevő telefonszáma: "
+ e11.getElementsByTagName("telefonszam").item(0).getTextContent());
        System.out.println((i+1) + ". vevő lakcíme: " +
e11.getElementsByTagName("iranyitoszam").item(0).getTextContent() + ". "
+
e11.getElementsByTagName("varos").item(0).getTextContent()
+ ", " +
e11.getElementsByTagName("utca").item(0).getTextContent() +
" utca " +
e11.getElementsByTagName("hazszam").item(0).getTextContent());
    }
}

//étterem adatainak kiírása
for (int i = 0; i < n1Etterem.getLength(); i++) {
    Node n2 = n1Etterem.item(i);

    if(n2.getNodeType() == Node.ELEMENT_NODE) {
        Element e12 = (Element) n2;
        System.out.println((i+1) + ". etterem neve: " +
e12.getElementsByTagName("nev").item(0).getTextContent()
+ " és weblap: " +
e12.getElementsByTagName("weblap").item(0).getTextContent());
    }
}
//kivételkezelés
} catch (Exception e) {
    System.out.println("Hiba történt: " +e.getMessage());
}
}
}

```

Futási eredmény:

```

etterem
Vevők száma: 3
Étterem száma: 4
1. vevő neve: Bence
1. vevő fizetése: 0
1. vevő készpénz: 10000
1. vevő telefonszáma: 06703333456
1. vevő lakcíme: 3970. Sátor, Paloc utca 2
2. vevő neve: Zsófi
2. vevő fizetése: 0
2. vevő készpénz: 500
2. vevő telefonszáma: 06706343353
2. vevő lakcíme: 3970. Sátor, Lolka utca 4
3. vevő neve: Sándor
3. vevő fizetése: 0
3. vevő készpénz: 1500
3. vevő telefonszáma: 06706343353
3. vevő lakcíme: 3970. Sátor, Lolka utca 4
1. etterem neve: Egyszer étterem és weblap: www.egyszeretterem.com

```

DomModifyVLJQO:

```
package hu.domparse.ivljqo;
```



```

import java.io.File;

import javax.xml.parsers.DocumentBuilder;
import javax.xml.parsers.DocumentBuilderFactory;
import javax.xml.transform.OutputKeys;
import javax.xml.transform.Transformer;
import javax.xml.transform.TransformerConfigurationException;
import javax.xml.transform.TransformerException;
import javax.xml.transform.TransformerFactory;
import javax.xml.transform.TransformerFactoryConfigurationError;
import javax.xml.transform.dom.DOMSource;
import javax.xml.transform.stream.StreamResult;

import org.w3c.dom.Document;
import org.w3c.dom.Element;
import org.w3c.dom.Node;
import org.w3c.dom.NodeList;

public class DomModifyIVLJQO {

    public static void main(String[] args) {

        try {
            //fájl helyének megadása
            String path = "XMLIVLJQO.xml";
            File xmlFile = new File (path);
            //
            DocumentBuilder db;
            DocumentBuilderFactory dbf =
DocumentBuilderFactory.newInstance();

            db = dbf.newDocumentBuilder();
            Document doc = db.parse(xmlFile);
            doc.getDocumentElement().normalize();

            // Módosítás
            updateElement(doc);

            // Törlés
            deleteElement(doc);

            // Element hozzáadása
            addElement(doc);

            // Módosított fájl létrehozása
            writeFile(doc);

        } catch (Exception e){
            e.printStackTrace();
        }
    }

    private static void writeFile(Document doc)
        throws TransformerFactoryConfigurationError,
TransformerConfigurationException, TransformerException {
        doc.getDocumentElement().normalize();
        TransformerFactory transformerFactory = TransformerFactory.newInstance();
        Transformer transformer = transformerFactory.newTransformer();

```

```

DOMSource ds = new DOMSource(doc);
StreamResult result = new StreamResult(new File("Updated_XMLIVLJQO.xml"));
transformer.setOutputProperty(OutputKeys.INDENT, "yes");
transformer.transform(ds, result);
System.out.println("XML módosítva");
}

//új elem hozzáadása az étteremhez
private static void addElement(Document doc) {
    NodeList etterem1 = doc.getElementsByTagName("etterem");
    Element etterem = null;

    //minden darabra
    for (int i = 0; i < etterem1.getLength(); i++) {
        etterem = (Element) etterem1.item(i);
        Element id = doc.createElement("kapacitas");
        id.appendChild(doc.createTextNode("500"));
        etterem.appendChild(id);
    }
}

//Egy elem törlése
private static void deleteElement(Document doc) {
    NodeList etterem = doc.getElementsByTagName("fizetes");
    Element fizetes = null;
    //minden elementre
    for (int i = 0; i < etterem.getLength(); i++) {
        fizetes = (Element) etterem.item(i);
        Node kartya = fizetes.getElementsByTagName("kartya").item(0);
        fizetes.removeChild(kartya);
    }
}

//Egy ügyfél nevének megváltoztatása (első elem használatával)
private static void updateElement(Document doc) {
    NodeList etterem = doc.getElementsByTagName("vevo");
    Element vevo = null;
    // minden elementre
    for (int i = 0; i < etterem.getLength(); i++) {
        vevo = (Element) etterem.item(i);
        Node nev = vevo.getElementsByTagName("nev").item(0).getFirstChild();
        //megfelelő ügyfél megkeresése és adat megváltoztatása
        if (vevo.getAttribute("id").contentEquals("1")) {
            nev.setNodeValue("Ádám");
        }
    }
}
}

```

Futási eredményt egy új xml fájlban jön létre ami az Update_XMLIVLJQO.xml

```
<?xml version="1.0" encoding="utf-8" standalone="no"?>
```

```

<etterem xmlns="ivljqo" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="ivljqo schema.xsd">

```

<rendeles etteremid="string" rendelesid="string"/>

<etterem etteremid="1">

<nev>Egyszer étterem</nev>

<telefonszam>06704564653</telefonszam>

<weblap>www.egyszeretterem.com</weblap>

<kapacitas>500</kapacitas>

</etterem>

<etterem etteremid="2">

<nev>Jó étterem</nev>

<telefonszam>06704569873</telefonszam>

<weblap>www.joetterem.com</weblap>

<kapacitas>500</kapacitas>

</etterem>

<etterem etteremid="3">

<nev>Volt étterem</nev>

<telefonszam>06704659873</telefonszam>

<weblap>www.voltetterem.com</weblap>

<kapacitas>500</kapacitas>

</etterem>

<vevo id="1">

<nev>Ádám</nev>

<fizetes>

<keszpenz>10000</keszpenz>

</fizetes>

<telefonszam>06703333456</telefonszam>

<lakcim id="1">

<varos>Sátor</varos>

<iranyitoszam>3970</iranyitoszam>

<utca>Paloc</utca>

<hazszam>2</hazszam>

</lakcim>

</vevo>

<vevo id="2">

<nev>Zsófi</nev>

<fizetes>

<keszpenz>500</keszpenz>

</fizetes>

<telefonszam>06706343353</telefonszam>

<lakcim id="2">

<varos>Sátor</varos>

<iranyitoszam>3970</iranyitoszam>

<utca>Lolka</utca>

<hazszam>4</hazszam>

</lakcim>

</vevo>

<vevo id="3">

<nev>Sándor</nev>

<fizetes>

<keszpenz>1500</keszpenz>

</fizetes>

<telefonszam>06706343353</telefonszam>

<lakcim id="3">

<varos>Sátor</varos>

<iranyitoszam>3970</iranyitoszam>

<utca>Lolka</utca>

<hazszam>4</hazszam>

</lakcim>

</vevo>

<rendelesadatok rendelesid="1">

<datum>

<tol>2022.06.08.18:40</tol>

<ig>2022.06.08.19:30</ig>

</datum>

</rendelesadatok>

<rendelesadatok rendelesid="2">

<datum>

<tol>2022.07.09.18:40</tol>

<ig>2022.07.09.19:30</ig>

</datum>

</rendelesadatok>

<rendelesadatok rendelesid="3">

<datum>

<tol>2022.06.08.18:40</tol>

<ig>2022.06.08.19:30</ig>

</datum>

</rendelesadatok>

<etlap id="1">

<nev>Pizza</nev>

<darab>5</darab>

<osszeg>10000</osszeg>

</etlap>

<etlap id="2">

<nev>Hamburger</nev>

<darab>1</darab>

<osszeg>500</osszeg>

</etlap>

<etlap id="3">

<nev>Gyros</nev>

<darab>1</darab>

<osszeg>1500</osszeg>

</etlap>

<kapacitas>500</kapacitas>

</etterem>

DomQueryIVLJQO lekérdezés XML fájlból:


```

        .item(0)
        .gettextContent());
System.out.println("készpenz : "
    + eElement
    .getElementsByTagName("készpenz")
    .item(0)
    .gettextContent());
System.out.println("kartya : "
    + eElement
    .getElementsByTagName("kartya")
    .item(0)
    .gettextContent());
System.out.println("telefonszám : "
    + eElement
    .getElementsByTagName("telefonszam")
    .item(0)
    .gettextContent());
System.out.println("irányítószám : "
    + eElement
    .getElementsByTagName("iranyitoszam")
    .item(0)
    .gettextContent());
System.out.println("varos : "
    + eElement
    .getElementsByTagName("varos")
    .item(0)
    .gettextContent());
System.out.println("utca : "
    + eElement
    .getElementsByTagName("utca")
    .item(0)
    .gettextContent());
System.out.println("házzsam : "
    + eElement
    .getElementsByTagName("hazzsam")
    .item(0)
    .gettextContent());
    }
}
//Egy Étterem lekérdezés
String expression2 = "/etterem/etterem[@etteremid = "+szam+"]";
NodeList nodeList2 = (NodeList) XPath.compile(expression2).evaluate(
    doc, XPathConstants.NODESET);

for (int i = 0; i < nodeList2.getLength(); i++) {
    Node nNode2 = nodeList2.item(i);
    System.out.println("\nJelenlegi Element : " + nNode2.getNodeName());
    //Éttrem adatainka kírátása
    if (nNode2.getNodeType() == Node.ELEMENT_NODE) {
        Element eElement = (Element) nNode2;
        System.out.println("Etterem id : "
            + eElement.getAttribute("etteremid"));
        System.out.println("nev : "
            + eElement
            .getElementsByTagName("nev")
            .item(0)
            .gettextContent());
        System.out.println("telefonszám : "
            + eElement

```

```

        .getElementsByTagName("telefonszam")
        .item(0)
        .getContents();
System.out.println("Weblap : "
    + eElement
        .getElementsByTagName("weblap")
        .item(0)
        .getContents());
    }
}
//Késszpénzek összegzése
NodeList nlVevo = doc.getElementsByTagName("vevo");
int osszeg=0;
for (int i = 0; i < nlVevo.getLength(); i++) {

    int t[] = new int[nlVevo.getLength()];
    Node n1 = nlVevo.item(i);

    if(n1.getNodeType() == Node.ELEMENT_NODE) {
        Element e11 = (Element) n1;

        System.out.println("\n"+(i+1) + ". vevő készpénz:
" + e11.getElementsByTagName("keszpenz").item(0).getContents());

        t[i]=Integer.parseInt(e11.getElementsByTagName("keszpenz").item(0).getContents());

        }osszeg=osszeg+t[i];

    }
    System.out.println("\nKifizette készpénz Összeg: "+osszeg);
} catch (ParserConfigurationException e) {
    e.printStackTrace();
} catch (SAXException e) {
    e.printStackTrace();
} catch (IOException e) {
    e.printStackTrace();
} catch (XPathExpressionException e) {
    e.printStackTrace();
}
}
}

```

Futási
eredmény:

```

Console × Progress
<terminated> DomQuery\VLJQO [Java Application] C:\Program Files\Java\jdk-19\bin\javaw.exe (2022. nov. 29. 20:37:03 – 20:37:07) [pid: 9468]
Adjon meg egy id számot
1
Jelenlegi Element :vevo
Vevő id : 1
nev : Bence
keszpenz : 10000
kartya : 0
telefonszam : 06703333456
iranyitoszam : 3970
varos : Sator
utca : Paloc
hazszam : 2

Jelenlegi Element :etterem
Etterem id : 1
nev : Egyszer etterem
telefonszam : 06704564653
Weblap : www.egyszeretterem.com

1. vevő készpénz: 10000
2. vevő készpénz: 500
3. vevő készpénz: 1500
Kifizette készpénz Összeg: 12000

```

