

JEGYZŐKÖNYV

Adatkezelés XML környezetben

Féléves feladat

Könyvtár nyilvántartó rendszer

Készítette: **Tóth Tibor Dávid**

Neptunkód: **WTX6LW**

A feladat leírása:

A feladatom témaköre egy könyvtár felépítése, és az ehhez kapcsolódó egyedek és kapcsolatok kialakítása. Ahogy az ER modellen is látszik, megtalálható a Kölcsönző egyed, amely a nevéből adódóan könyveket kölcsönöz a könyvtárból. Tulajdonságai az ID (egyedi azonosító, amely a többi egyednél is hasonlóan fog szerepelni), név, valamint a lakhely és elérhetőség, amelyek összetett tulajdonságok és felépülnek (város, utca, házszám, valamint email, és telefonszámból).

A könyv egyed ID, cím, ár, szerző és oldalszám tulajdonságokból épül fel. A könyvtáros egyed ID, név, az összetett elérhetőség, valamint a többértékű kitüntetések, mivel egy könyvtárost több ilyen jutalomban is részesülhet a jól végzett munkájáért.

A termék egyednek ID, név és ár tulajdonságai vannak.

Végezetül a gyártó, amely ID, név és elérhetőséggel bír.

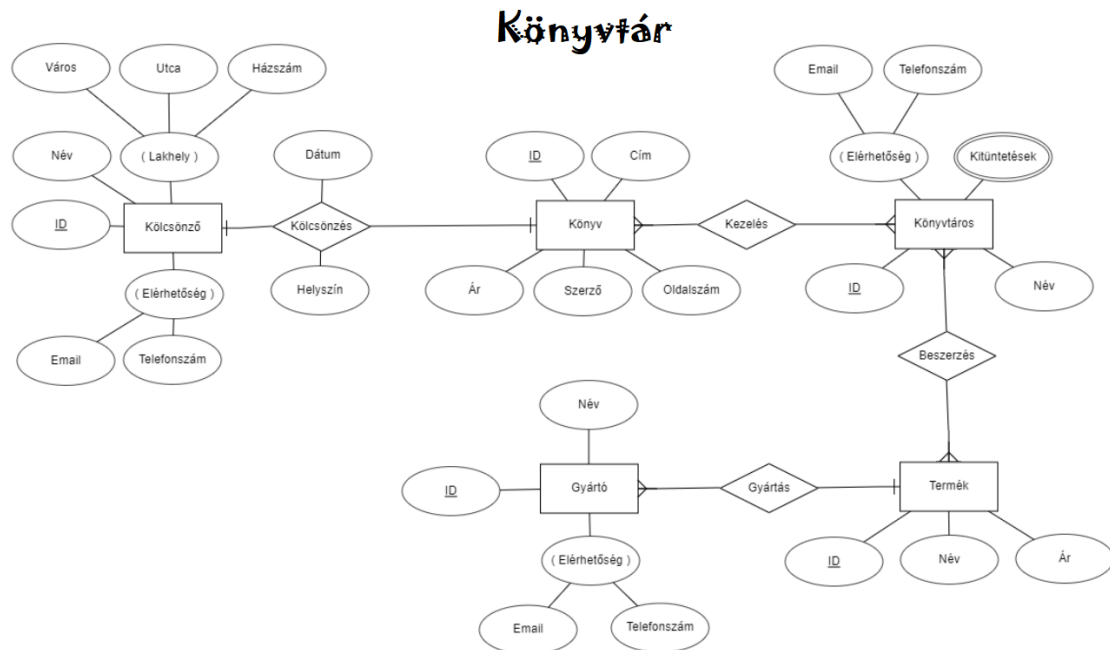
Kapcsolatok: Kölcsönző és Könyv között egy-egy kapcsolat van, vagyis 1 kölcsönző 1 könyvet kölcsönözhet, valamint 1 adott ID-jú könyv is csak egy kölcsönzőhöz tartozhat. Gyártó és Termék között egy-több kapcsolat van, míg a többi egyed több-több kapcsolattal bír egymáshoz képest.

Fontos lehet még megemlíteni, hogy Kölcsönző és Könyv között a kapcsolat a kölcsönzés, amely pluszban még megadja a kölcsönzés helyszínét és dátumát is.

A Könyv és Könyvtáros egyedek között a kezelés szerepel, amelynek a nevéből adódóan egyértelmű a funkciója. Emellett a könyvtárosok a beszerzés kapcsolat révén beszerezhetnek termékeket a könyvtár számára, amelyek szükségesek az adminisztrációhoz. A gyártás kapcsolat pedig megmondja, hogy az egyes termékeket mely gyártók gyártották és ez visszafelé is igaz.

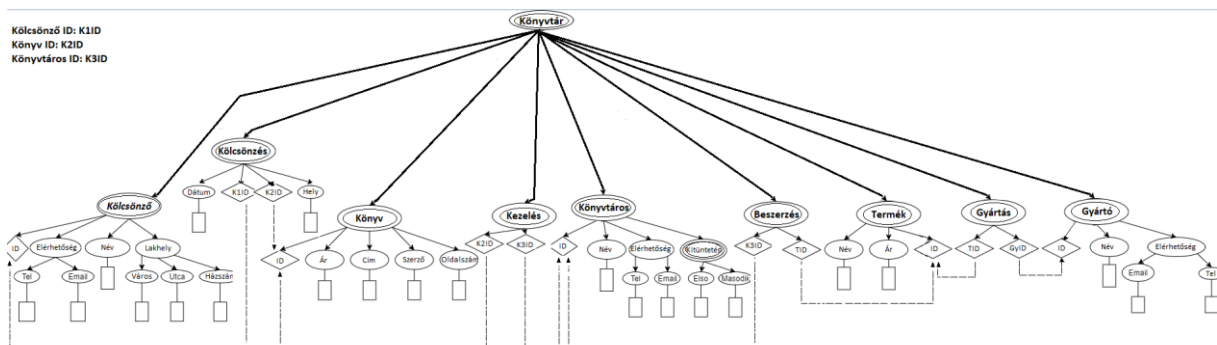
1. feladat

1a) Az adatbázis ER modell



1b) Az adatbázis konvertálása XDM modellre

Az XDM modellen látható a faszzerű kialakítás a gyöker könyvtár elemtől lefelé haladva az egyes szülő, valamint gyerekelemekig. A referencia jelölések megnevezéséhez is látható segédlet a képen.



1c) Az XDM modell alapján az XML dokumentum elkészítése

```

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

<könyvtar xmlns:xsi="https://w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="XMLSchemaWTX6LW.xsd">

  <Kolcsonzo ID="K100">
    <Nev>Molnár Kata</Nev>
    <Lakhely>
      <Varos>Miskolc</Varos>
      <Utca>Kossuth</Utca>
      <Hatszam>1</Hatszam>
    </Lakhely>
    <Elerhetoseg>
      <Tel>06302342211</Tel>
      <Email>molnar@gmail.com</Email>
    </Elerhetoseg>
  </Kolcsonzo>

  <Kolcsonzo ID="K101">
    <Nev>Tóth Andor</Nev>
    <Lakhely>
      <Varos>Budapest</Varos>
      <Utca>Nincs</Utca>
      <Hatszam>30</Hatszam>
    </Lakhely>
    <Elerhetoseg>
      <Tel>06301234567</Tel>
      <Email>toth@gmail.com</Email>
    </Elerhetoseg>
  </Kolcsonzo>

  <Kolcsonzo ID="K102">
    <Nev>Haragos Hugó</Nev>
    <Lakhely>
      <Varos>Eger</Varos>
      <Utca>Egres</Utca>
      <Hatszam>11</Hatszam>
    </Lakhely>
    <Elerhetoseg>
      <Tel>06309876541</Tel>
      <Email>haragos@gmail.com</Email>
    </Elerhetoseg>
  </Kolcsonzo>

  <Kolcsonzes K1ID="K100" K2ID="K200">
    <Datum>2012-07-19</Datum>
    <Helyszin>Miskolc</Helyszin>
  </Kolcsonzes>
  <Kolcsonzes K1ID="K101" K2ID="K201">
    <Datum>2015-05-15</Datum>
    <Helyszin>Budapest</Helyszin>
  </Kolcsonzes>
  <Kolcsonzes K1ID="K102" K2ID="K202">
    <Datum>2010-10-10</Datum>
    <Helyszin>Eger</Helyszin>
  </Kolcsonzes>

  <Konyv ID="K200">
    <Cim>A kis hajó</Cim>
  </Konyv>

```

```

        <Ar>3500</Ar>
        <Szerzo>Kiss Imre</Szerzo>
        <Oldalszam>550</Oldalszam>
    </Konyv>

    <Konyv ID="K201">
        <Cim>A nagy felleg</Cim>
        <Ar>5200</Ar>
        <Szerzo>Nagy Anita</Szerzo>
        <Oldalszam>800</Oldalszam>
    </Konyv>

    <Konyv ID="K202">
        <Cim>A rút kiskacsa</Cim>
        <Ar>2750</Ar>
        <Szerzo>Gellér László</Szerzo>
        <Oldalszam>240</Oldalszam>
    </Konyv>

    <Kezeles K2ID="K200" K3ID="K300"/>
    <Kezeles K2ID="K201" K3ID="K301"/>
    <Kezeles K2ID="K202" K3ID="K302"/>

    <Konyvtaros ID="K300">
        <Nev>Könyvtáros1</Nev>
        <Elerhetoseg>
            <Tel>06301111111</Tel>
            <Email>konyvtaros1@gmail.com</Email>
        </Elerhetoseg>
        <Kituntetes>
            <Elso>Legtöbb kölcsönzés</Elso>
            <Masodik>Év könyvtárosa 2020</Masodik>
        </Kituntetes>
    </Konyvtaros>

    <Konyvtaros ID="K301">
        <Nev>Könyvtáros2</Nev>
        <Elerhetoseg>
            <Tel>06301111112</Tel>
            <Email>konyvtaros2@gmail.com</Email>
        </Elerhetoseg>
        <Kituntetes>
            <Elso>Legkedvesebb könyvtáros</Elso>
            <Masodik>Legnagyobb tudás</Masodik>
        </Kituntetes>
    </Konyvtaros>

    <Konyvtaros ID="K302">
        <Nev>Könyvtáros3</Nev>
        <Elerhetoseg>
            <Tel>06301111113</Tel>
            <Email>konyvtaros3@gmail.com</Email>
        </Elerhetoseg>
        <Kituntetes>
            <Elso>Legtöbb munkaóra</Elso>
            <Masodik>Legtisztább munkakörnyezet</Masodik>
        </Kituntetes>
    </Konyvtaros>

```

```

<Beszerzes K3ID="F300" TID="T400"/>
<Beszerzes K3ID="F301" TID="T401"/>
<Beszerzes K3ID="F302" TID="T402"/>

<Termek ID="T400">
  <Nev>Igénylőlap</Nev>
  <Ar>5555</Ar>
</Termek>

<Termek ID="T401">
  <Nev>Könyvespolc</Nev>
  <Ar>15000</Ar>
</Termek>

<Termek ID="T402">
  <Nev>Írószerek</Nev>
  <Ar>3750</Ar>
</Termek>

<Gyartas TID="T400" GyID="Gy500"/>
<Gyartas TID="T401" GyID="Gy500"/>
<Gyartas TID="T402" GyID="Gy501"/>

<Gyarto ID="Gy500">
  <Nev>Papírbolt Kft.</Nev>
  <Elerhetoseg>
    <Tel>06469347529</Tel>
    <Email>papirbolt.kft@gmail.com</Email>
  </Elerhetoseg>
</Gyarto>
<Gyarto ID="Gy501">
  <Nev>Könyvespolc Kft.</Nev>
  <Elerhetoseg>
    <Tel>06469347530</Tel>
    <Email>konyvespolc.kft@gmail.com</Email>
  </Elerhetoseg>
</Gyarto>
<Gyarto ID="Gy502">
  <Nev>Írkálunk és védünk Kft.</Nev>
  <Elerhetoseg>
    <Tel>06469347531</Tel>
    <Email>irkalunk@gmail.com</Email>
  </Elerhetoseg>
</Gyarto>

</konyvtar>

```

1d) XML dokumentum alapján XMLSchema elkészítése

```

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

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

  <xs:element name="konyvtar" type="konyvtarTipus">
    <xs:key name="K1">
      <xs:selector xpath="kolcsonzo" />
      <xs:field xpath="@id" />
    </xs:key>
  </xs:element>

```

```

</xs:key>
<xs:key name="K2">
  <xs:selector xpath="konyv" />
  <xs:field xpath="@id" />
</xs:key>
<xs:key name="K3">
  <xs:selector xpath="konyvtaros" />
  <xs:field xpath="@id" />
</xs:key>
<xs:key name="K4">
  <xs:selector xpath="termek" />
  <xs:field xpath="@id" />
</xs:key>
<xs:key name="K5">
  <xs:selector xpath="gyarto" />
  <xs:field xpath="@id" />
</xs:key>

<xs:keyref refer="K1" name="refK1">
  <xs:selector xpath="kolcsonzes" />
  <xs:field xpath="@K1ID" />
</xs:keyref>
<xs:keyref refer="K2" name="refK2_1">
  <xs:selector xpath="kolcsonzes" />
  <xs:field xpath="@K2ID" />
</xs:keyref>
<xs:keyref refer="K2" name="refK2_2">
  <xs:selector xpath="kezeles" />
  <xs:field xpath="@K2ID" />
</xs:keyref>
<xs:keyref refer="K3" name="refK3_1">
  <xs:selector xpath="kezeles" />
  <xs:field xpath="@K3ID" />
</xs:keyref>
<xs:keyref refer="K3" name="refK3_2">
  <xs:selector xpath="beszerzes" />
  <xs:field xpath="@K3ID" />
</xs:keyref>
<xs:keyref refer="K4" name="refK4_1">
  <xs:selector xpath="beszerzes" />
  <xs:field xpath="@TID" />
</xs:keyref>
<xs:keyref refer="K4" name="refK4_2">
  <xs:selector xpath="gyartas" />
  <xs:field xpath="@TID" />
</xs:keyref>
<xs:keyref refer="K5" name="refK5_1">
  <xs:selector xpath="gyartas" />
  <xs:field xpath="@GyID" />
</xs:keyref>
</xs:element>

  <xs:complexType name="konyvtarTipus">
    <xs:sequence>
<xs:element name="kolcsonzo" type="kolcsonzoTipus" minOccurs="0" maxOccurs="100"/>
<!-- előfordulás -->
<xs:element name="kolcsonzes" type="kolcsonzesTipus" minOccurs="0"
maxOccurs="100"/>
<xs:element name="konyv" type="konyvTipus" minOccurs="0" maxOccurs="100"/>

```

```

<xs:element name="kezeles" type="kezelesTipus" minOccurs="0" maxOccurs="100"/>
<xs:element name="konyvtaros" type="konyvtarosTipus" minOccurs="0"
maxOccurs="100"/>
<xs:element name="beszerzes" type="beszerzesTipus" minOccurs="0" maxOccurs="100"/>
<xs:element name="termek" type="termekTipus" minOccurs="0" maxOccurs="100"/>
<xs:element name="gyartas" type="gyartasTipus" minOccurs="0" maxOccurs="100"/>
<xs:element name="gyarto" type="gyartoTipus" minOccurs="0" maxOccurs="100"/>
    </xs:sequence>
</xs:complexType>

<xs:complexType name="kolcsonzoTipus">
    <xs:sequence>
        <xs:element name="nev" type="xs:string"/>
        <xs:element name="lakhely" type="lakhelyTipus"/>
        <xs:element name="elerhetoseg"
type="elerhetosegTipus"/>
    </xs:sequence>
    <xs:attribute name="id" type="xs:string"
use="required"/> <!-- formátuma miatt string -->
</xs:complexType>

<xs:complexType name="lakhelyTipus">
    <xs:sequence>
        <xs:element name="varos" type="xs:string"/>
        <xs:element name="utca" type="xs:string"/>
        <xs:element name="hazszam"
type="xs:unsignedShort"/>
    </xs:sequence>
</xs:complexType>

<xs:complexType name="elerhetosegTipus">
    <xs:sequence>
        <xs:element name="tel" type="xs:unsignedLong"/>
        <xs:element name="email" type="xs:string"/>
    </xs:sequence>
</xs:complexType>

<xs:complexType name="kolcsonzesTipus">
    <xs:sequence>
        <xs:element name="datum" type="xs:date"/>
        <xs:element name="helyszin" type="xs:string"/>
    </xs:sequence>
    <xs:attribute name="K1ID" type="xs:string"
use="required"/>
    <xs:attribute name="K2ID" type="xs:string"
use="required"/>
</xs:complexType>

<xs:complexType name="konyvTipus">
    <xs:sequence>
        <xs:element name="cim" type="xs:string"/>
        <xs:element name="ar" type="xs:unsignedShort"/>
        <xs:element name="szerzo" type="xs:string"/>
        <xs:element name="oldalSzam"
type="xs:unsignedShort"/>
    </xs:sequence>
    <xs:attribute name="id" type="xs:string"
use="required"/> <!-- formátuma miatt string -->

```



```

</xs:complexType>

<xs:complexType name="kezelesTipus">
  <xs:attribute name="K2ID" type="xs:string"
use="required"/>
  <xs:attribute name="K3ID" type="xs:string"
use="required"/>
</xs:complexType>

<xs:complexType name="konyvtarosTipus">
  <xs:sequence>
    <xs:element name="nev" type="xs:string"/>
    <xs:element name="elerhetoseg"
type="elerhetosegTipus"/>
    <xs:element name="kituntetes"
type="kituntetesTipus"/>
  </xs:sequence>
  <xs:attribute name="id" type="xs:string"
use="required"/> <!-- formátuma miatt string -->
</xs:complexType>

<xs:complexType name="kituntetesTipus">
  <xs:sequence>
    <xs:element name="elso" type="xs:string"/>
    <xs:element name="masodik" type="xs:string"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="beszerzesTipus">
  <xs:attribute name="K3ID" type="xs:string"
use="required"/>
  <xs:attribute name="TID" type="xs:string"
use="required"/>
</xs:complexType>

<xs:complexType name="termekTipus">
  <xs:sequence>
    <xs:element name="nev" type="xs:string"/>
    <xs:element name="ar" type="xs:unsignedShort"/>
  </xs:sequence>
  <xs:attribute name="id" type="xs:string"
use="required"/> <!-- formátuma miatt string -->
</xs:complexType>

<xs:complexType name="gyartasTipus">
  <xs:attribute name="TID" type="xs:string"
use="required"/>
  <xs:attribute name="GyID" type="xs:string"
use="required"/>
</xs:complexType>

<xs:complexType name="gyartoTipus">
  <xs:sequence>
    <xs:element name="nev" type="xs:string"/>
    <xs:element name="elerhetoseg"
type="elerhetosegTipus"/>
  </xs:sequence>
  <xs:attribute name="id" type="xs:string"
use="required"/> <!-- formátuma miatt string -->

```

```
</xs:complexType>
```

```
</xs:schema>
```

2. feladat

2a) Adatolvasás (DOMReadWTX6LW)

Forráskód a szükséges kommentekkel együtt:

```
package hu.domparse.WTX6LW;

import java.io.File;
import java.io.IOException;

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

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 DOMReadWTX6LW {

    public static void main(String[] args) {
        //Az esetleges keletkező hibák lekezelése try-catch blokkban
        try {
            //Az olvasandó fájl megadása
            File xmlFile = new File("src/XMLWTX6LW.xml");
```

```

        //Az XML fájl átalakítása DOM objektumokká
        DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance();
        DocumentBuilder dBuilder = factory.newDocumentBuilder();
        Document doc = dBuilder.parse(xmlFile);

        // Node-ok normalizálása
        doc.getDocumentElement().normalize();

        //A gyökér elem nevének kiolvasása és kiírása konzolra
        System.out.println("Gyökér elem: " + doc.getDocumentElement().getNodeName());

        //A gyökér elem alatt fellelhető gyermek/testvér elemek listába illesztése
        NodeList nList1 = doc.getElementsByTagName("Kolcsonzo");
        NodeList nList2 = doc.getElementsByTagName("Kolcsonzes");
        NodeList nList3 = doc.getElementsByTagName("Konyv");
        NodeList nList4 = doc.getElementsByTagName("Kezeles");
        NodeList nList5 = doc.getElementsByTagName("Konyvtaros");
        NodeList nList6 = doc.getElementsByTagName("Beszerzes");
        NodeList nList7 = doc.getElementsByTagName("Termek");
        NodeList nList8 = doc.getElementsByTagName("Gyartas");
        NodeList nList9 = doc.getElementsByTagName("Gyarto");
        System.out.println("-----");

        //Ciklus segítségével végignézzük az aktuális node lista elemeit
        for (int i = 0; i < nList1.getLength(); i++) {
            //Az egyes lista elemeket meghatározzuk nodeként
            Node nNode = nList1.item(i);

            //Ha egyezést talál akkor kiírjuk az adott elemhez tartozó nevet és
            //adatokat is
            if (nNode.getNodeType() == Node.ELEMENT_NODE) {
                //A node-ot elemként definiáljuk
                Element elem = (Element) nNode;

                System.out.println("\nAktuális elem: " + nNode.getNodeName());
                // Az aktuális elemhez tartozó azonosító letárolása egy String adattípusban
                String id = elem.getAttribute("ID");
            }
        }

```

```

// Az adott elem gyermekelem értékének eltárolása String adattípusban
Node node1 = elem.getElementsByTagName("Nev").item(0);
String nev = node1.getTextContent();

Node node2_1 = elem.getElementsByTagName("Varos").item(0);
Node node2_2 = elem.getElementsByTagName("Utca").item(0);
Node node2_3 = elem.getElementsByTagName("Hazzsam").item(0);
String varos = node2_1.getTextContent();
String utca = node2_2.getTextContent();
String hazzsam = node2_3.getTextContent();

Node node3_1 = elem.getElementsByTagName("Tel").item(0);
Node node3_2 = elem.getElementsByTagName("Email").item(0);
String tel = node3_1.getTextContent();
String email = node3_2.getTextContent();

// Az eltárolt értékek kiírása
System.out.println("Kölcsönző ID-ja: " + id);
System.out.println("Név: " + nev);
System.out.println("Lakhely: ");
System.out.println("\tVáros: "+varos);
System.out.println("\tUtca: "+utca);
System.out.println("\tHázzsám: "+hazzsam);
System.out.println("Elérhetőség: ");
System.out.println("\tTel: "+tel);
System.out.println("\tEmail: "+email);

        }

    }

// A továbbiakban minden listához ezt a megoldást alkalmazzuk

for (int i = 0; i < nList2.getLength(); i++) {
    Node nNode = nList2.item(i);

```

```

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

            System.out.println("\nAktuális elem: " + nNode.getNodeName());

            String id1 = elem.getAttribute("K1ID");
            String id2 = elem.getAttribute("K2ID");

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

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

            System.out.println("Kölcsönző ID-ja: " + id1);
            System.out.println("Könyv azonosítója: " + id2);
            System.out.println("Dátum: "+datum);
            System.out.println("Helyszín: "+helyszin);

        }
    }

    for (int i = 0; i < nList3.getLength(); i++) {
        Node nNode = nList3.item(i);

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

            System.out.println("\nAktuális elem: " + nNode.getNodeName());

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

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

```

```

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

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

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

System.out.println("Könyv ID-ja: " + id);
System.out.println("Cím: " + cim);
System.out.println("Ár: "+ar);
System.out.println("Szerző: "+szerzo);
System.out.println("Oldalszám: "+oldalszam);
        }
    }

for (int i = 0; i < nList4.getLength(); i++) {
    Node nNode = nList4.item(i);

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

        System.out.println("\nAktuális elem: " + nNode.getNodeName());

        String id2 = elem.getAttribute("K2ID");
        String id3 = elem.getAttribute("K3ID");

        System.out.println("Könyv ID-ja: " + id2);
        System.out.println("Könyvtáros ID-ja: " + id3);
    }
}

```

```
}
```

```
for (int i = 0; i < nList5.getLength(); i++)
    Node nNode = nList5.item(i);

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

        System.out.println("\nAktuális elem: " + nNode.getNodeName());

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

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

        Node node2_1 = elem.getElementsByTagName("Tel").item(0);
        Node node2_2 = elem.getElementsByTagName("Email").item(0);
        String tel = node2_1.getTextContent();
        String email = node2_2.getTextContent();

        Node node3_1 = elem.getElementsByTagName("Elso").item(0);
        Node node3_2 = elem.getElementsByTagName("Masodik").item(0);
        String elso = node3_1.getTextContent();
        String masodik = node3_2.getTextContent();

        System.out.println("Könyvtáros ID-ja: " + id);
        System.out.println("Név: " + nev);
        System.out.println("Elérhetőség: ");
        System.out.println("\tTel: "+tel);
        System.out.println("\tEmail: "+email);
        System.out.println("Kitüntetések:");
        System.out.println("\tElső: "+elso);
        System.out.println("\tMásodik: "+masodik);
```

```

        }
    }

    for (int i = 0; i < nList6.getLength(); i++) {
        Node nNode = nList6.item(i);

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

            System.out.println("\nAktuális elem: " + nNode.getNodeName());

            String id1 = elem.getAttribute("K3ID");
            String id2 = elem.getAttribute("TID");

            System.out.println("Könyvtáros ID-ja: " + id1);
            System.out.println("Termék ID-ja: " + id2);

        }
    }

    for (int i = 0; i < nList7.getLength(); i++) {
        Node nNode = nList7.item(i);

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

            System.out.println("\nAktuális elem: " + nNode.getNodeName());

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

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

            Node node2 = elem.getElementsByTagName("Ar").item(0);

```



```

String ar = node2.getTextContent();

System.out.println("Termék ID-ja: " + id);
System.out.println("Név: "+nev);
System.out.println("Ár: "+ar);

        }
    }

    for (int i = 0; i < nList8.getLength(); i++) {
        Node nNode = nList8.item(i);

        if (nNode.getNodeType() == Node.ELEMENT_NODE) {
            Element elem = (Element) nNode;
            System.out.println("\nAktuális elem: " + nNode.getNodeName());
            String id1 = elem.getAttribute("TID");
            String id2 = elem.getAttribute("GyID");

            System.out.println("Termék ID-ja: " + id1);
            System.out.println("Gyártó ID-ja: " + id2);
        }
    }

    for (int i = 0; i < nList9.getLength(); i++) {
        Node nNode = nList9.item(i);

        if (nNode.getNodeType() == Node.ELEMENT_NODE) {
            Element elem = (Element) nNode;
            System.out.println("\nAktuális elem: " + nNode.getNodeName());
            String id = elem.getAttribute("ID");

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

            Node node2_1 = elem.getElementsByTagName("Tel").item(0);

```

```

Node node2_2 = elem.getElementsByTagName("Email").item(0);
String tel = node2_1.getTextContent();
String email = node2_2.getTextContent();
System.out.println("Gyártó azonosítója: " + id);
System.out.println("Név: " + nev);
System.out.println("Elérhetőség: ");
System.out.println("\tTel: "+tel);
System.out.println("\tEmail: "+email);
        }
    }
}catch(SAXException sxe) {
    sxe.printStackTrace();
}catch(ParserConfigurationException pe) {
    pe.printStackTrace();
}catch(IOException ioe) {
    ioe.printStackTrace();
}
}
}

```

A konzolon megjelenő eredmény:

Gyökér elem: konyvtar

Aktuális elem: Kolcsonzo

Kölcsönző ID-ja: K100

Név: Molnár Kata

Lakhely:

Város: Miskolc

Utca: Kossuth

Házzám: 1

Elérhetőség:

Tel: 06302342211

Email: molnar@gmail.com

Aktuális elem: Kolcsonzo

Kölcsönző ID-ja: K101

Név: Tóth Andor

Lakhely:

Város: Budapest

Utca: Nincs

Házzám: 30

Elérhetőség:

Tel: 06301234567

Email: toth@gmail.com

Aktuális elem: Kolcsonzo

Kölcsönző ID-ja: K102

Név: Haragos Hugó

Lakhely:

Város: Eger

Utca: Egres

Házzám: 11

Elérhetőség:

Tel: 06309876541

Email: haragos@gmail.com

Aktuális elem: Kolcsonzes
Kölcsönző ID-ja: K100
Könyv azonosítója: K200
Dátum: 2012-07-19
Helyszín: Miskolc

Aktuális elem: Kolcsonzes
Kölcsönző ID-ja: K101
Könyv azonosítója: K201
Dátum: 2015-05-15
Helyszín: Budapest

Aktuális elem: Kolcsonzes
Kölcsönző ID-ja: K102
Könyv azonosítója: K202
Dátum: 2010-10-10
Helyszín: Eger

Aktuális elem: Konyv
Könyv ID-ja: K200
Cím: A kis hajó
Ár: 3500
Szerző: Kiss Imre
Oldalszám: 550

Aktuális elem: Konyv
Könyv ID-ja: K201
Cím: A nagy felleg
Ár: 5200
Szerző: Nagy Anita
Oldalszám: 800

Aktuális elem: Konyv
Könyv ID-ja: K202
Cím: A rút kiskacsa
Ár: 2750
Szerző: Gellér László
Oldalszám: 240

Aktuális elem: Kezeles
Könyv ID-ja: K200
Könyvtáros ID-ja: K300

Aktuális elem: Kezeles
Könyv ID-ja: K201
Könyvtáros ID-ja: K301

Aktuális elem: Kezeles
Könyv ID-ja: K202
Könyvtáros ID-ja: K302

Aktuális elem: Konyvtaros
Könyvtáros ID-ja: K300
Név: Könyvtáros1
Elérhetőség:
Tel: 06301111111
Email: konyvtaros1@gmail.com

Kitüntetések:
Első: Legtöbb kölcsönzés
Második: Év könyvtárosa 2020

Aktuális elem: Konyvtaros
Könyvtáros ID-ja: K301
Név: Könyvtáros2
Elérhetőség:
Tel: 06301111112
Email: konyvtaros2@gmail.com

Kitüntetések:
Első: Legkedvesebb könyvtáros
Második: Legnagyobb tudás

Aktuális elem: Konyvtaros
Könyvtáros ID-ja: K302
Név: Könyvtáros3
Elérhetőség:
Tel: 06301111113
Email: konyvtaros3@gmail.com

Kitüntetések:
Első: Legtöbb munkaóra
Második: Legtisztább munkakörnyezet

Aktuális elem: Beszerzes
Könyvtáros ID-ja: F300
Termék ID-ja: T400

Aktuális elem: Beszerzes
Könyvtáros ID-ja: F301
Termék ID-ja: T401

Aktuális elem: Beszerzes
Könyvtáros ID-ja: F302
Termék ID-ja: T402

Aktuális elem: Termek
Termék ID-ja: T400
Név: Igénylőlap
Ár: 5555

Aktuális elem: Termek
Termék ID-ja: T401
Név: Könyvespolc
Ár: 15000

Aktuális elem: Termek
Termék ID-ja: T402
Név: Írószerek
Ár: 3750

Aktuális elem: Gyartas
Termék ID-ja: T400
Gyártó ID-ja: Gy500

Aktuális elem: Gyartas
Termék ID-ja: T401
Gyártó ID-ja: Gy500

Aktuális elem: Gyartas
Termék ID-ja: T402
Gyártó ID-ja: Gy501

Aktuális elem: Gyarto
Gyártó azonosítója: Gy500
Név: Papírbolt Kft.
Elérhetőség:
Tel: 06469347529
Email: papirbolt.kft@gmail.com

Aktuális elem: Gyarto
Gyártó azonosítója: Gy501
Név: Könyvespolc Kft.
Elérhetőség:
Tel: 06469347530
Email: konyvespolc.kft@gmail.com

Aktuális elem: Gyarto
Gyártó azonosítója: Gy502
Név: Írkálunk és védünk Kft.
Elérhetőség:
Tel: 06469347531
Email: irkalunk@gmail.com

2b) Adatlekérdezés (DOMQueryWTX6LW)

```
package hu.domparse.WTX6LW;

import java.io.File;
import java.io.IOException;

import javax.xml.parsers.DocumentBuilder; ; //szükséges csomagok importálása
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 DOMQueryWTX6LW {

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

        File xmlFile = new File("src/XMLWTX6LW.xml"); //felhasznált XML fájl
```

```

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

//A parse() metódus elemzi az XML fájlt
Document doc = dBuilder.parse(xmlFile);
doc.getDocumentElement().normalize();

//Megkapjuk a dokumentum gyökérelemét.
System.out.println("Root element: " + doc.getDocumentElement().getNodeName() +
"\n");

System.out.println("-----");
System.out.println("1. lekérdezés:");
System.out.println("A kölcsönzők adatai:\n");

//Végigiterálás a listán
NodeList kolcsonzoList = doc.getElementsByTagName("Kolcsonzo");
for(int i=0; i<kolcsonzoList.getLength(); i++) {
    Node k = kolcsonzoList.item(i);
    if(k.getNodeType()==Node.ELEMENT_NODE) {
        Element kolcsonzo = (Element) k;
        String ID = kolcsonzo.getAttribute("ID");
        NodeList nevList = kolcsonzo.getChildNodes();
        for(int j=0; j<nevList.getLength(); j++) {
            Node n = nevList.item(j);
            if (n.getNodeType()==Node.ELEMENT_NODE) {
                Element nev = (Element) n;
                System.out.println("Kölcsönző " + ID + ": "
+ nev.getTagName() + "= " + nev.getTextContent());
            }
        }
    }
}

System.out.println("-----");
System.out.println("2. lekérdezés:");
System.out.println("Azon könyvek adatai, amelyeknek ára 5200:\n");

NodeList arList = doc.getElementsByTagName("Konyv");

for(int i = 0; i < arList.getLength(); i++) {

    Node a = arList.item(i);

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

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

        if("5200".equals(ar)) {
            String ID = elem.getAttribute("ID");

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

            Node node2 = elem.getElementsByTagName("Ar").item(0);
            String ar2 = node2.getTextContent();

```

```

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

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

System.out.println("Könyv ID: " + ID);
System.out.println("Cím: " + cim);
System.out.println("Ár: " + ar2);
System.out.println("Szerző: " + szerzo);
System.out.println("Oldalszám: " + oldalszam + "\n");}
    }
}
}

```

A konzolon megjelenő lekérdezések:

1. **lekérdezés:** Az összes kölcsönző adatai (összetett tulajdonságok rövidítve)

```

Root element: konyvtar

-----
1. lekérdezés:
A kölcsönzők adatai:

Kölcsönző K100: Nev= Molnár Kata
Kölcsönző K100: Lakhely=
                Miskolc
                Kossuth
                1

Kölcsönző K100: Elerhetoseg=
                06302342211
                molnar@gmail.com

Kölcsönző K101: Nev= Tóth Andor
Kölcsönző K101: Lakhely=
                Budapest
                Nincs
                30

Kölcsönző K101: Elerhetoseg=
                06301234567
                toth@gmail.com

Kölcsönző K102: Nev= Haragos Hugó
Kölcsönző K102: Lakhely=
                Eger
                Egres
                11

Kölcsönző K102: Elerhetoseg=
                06309876541
                haragos@gmail.com

```


2. lekérdezés: Az 5200-as árral rendelkező könyvek adatai

```
-----  
2. lekérdezés:  
Azon könyvek adatai, amelyeknek ára 5200:  
  
Könyv ID: K201  
Cím: A nagy felleg  
Ár: 5200  
Szerző: Nagy Anita  
Oldalszám: 800
```

2c) Adatmódosítás (DOMModifyWTX6LW)

```
package hu.domparse.WTX6LW;  
  
import java.io.File;  
import javax.xml.parsers.DocumentBuilder;  
import javax.xml.parsers.DocumentBuilderFactory;  
  
import javax.xml.transform.Transformer; //szükséges csomagok importálása  
import javax.xml.transform.TransformerFactory;  
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 DOMModifyWTX6LW {  
  
    public static void main(String[] args){  
  
        try {  
            File xmlFile = new File("src/XMLWTX6LW.xml"); //felhasznált XML fájl  
  
            //DocumentumBuilder létrehozása  
            DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance();  
            DocumentBuilder dBuilder = factory.newDocumentBuilder();  
  
            //A parse() metódus elemzi az XML fájlt  
            Document doc = dBuilder.parse(xmlFile);  
  
            doc.getDocumentElement().normalize();  
  
            //Végig iterálunk a könyv elemeken, a 240-es oldalszámúakat 300-assá módosítjuk,  
            //majd a „A rút kiskacsa” címűt „A rút kiscsibe”-re módosítjuk
```

//A getElementByTagName() metódus segítségével megkapjuk a könyv elem NodeListjét a dokumentumban.

```
NodeList nList = doc.getElementsByTagName("Könyv");
```

//A listán for ciklussal megyünk végig

```
for(int i = 0; i < nList.getLength(); i++) {
    Node könyv = doc.getElementsByTagName("Könyv").item(i);

    NodeList list = könyv.getChildNodes();

    for (int temp = 0; temp < list.getLength(); temp++) {
        Node node = list.item(temp);
        if (node.getNodeType() == Node.ELEMENT_NODE) {
            Element eElement = (Element) node;
            if ("Oldalszam".equals(eElement.getNodeName())) {
                if("240".equals(eElement.getTextContent())) {
                    eElement.setTextContent("300");
                }
            }
            if ("Cim".equals(eElement.getNodeName())) {
                if("A rút kiskacsa".equals(eElement.getTextContent())) {
                    eElement.setTextContent("A rút kiscsibe");
                }
            }
        }
    }
}
```

//A kölcsönzők közül Haragos Hugó nevét Haragos Hedvig-re változtatjuk meg

```
NodeList nList2 = doc.getElementsByTagName("Kölcsönzo");
```

```
for(int i = 0; i < nList2.getLength(); i++) {
    Node kolcsonzo = doc.getElementsByTagName("Kölcsönzo").item(i);
```

```
NodeList list2 = kolcsonzo.getChildNodes();
```

```
for (int temp = 0; temp < list2.getLength(); temp++) {
    Node node = list2.item(temp);
    if (node.getNodeType() == Node.ELEMENT_NODE) {
        Element eElement = (Element) node;
        if ("Nev".equals(eElement.getNodeName())) {
            if("Haragos Hugó".equals(eElement.getTextContent())) {
                eElement.setTextContent("Haragos Hedvig");
            }
        }
    }
}
```

//Könyvtárosoknak fizetés létrehozása egy alap 220k értékkel

```
NodeList nList3 = doc.getElementsByTagName("Konyvtaros");
```

```
Element Konyvtaros = null;
```

```
for(int temp = 0; temp < nList3.getLength(); temp++) {
    Konyvtaros = (Element) nList3.item(temp);
    Element salaryElement = doc.createElement("Fizetes");
    salaryElement.appendChild(doc.createTextNode("220000"));
    Konyvtaros.appendChild(salaryElement);
}
```

```

//A Termékeknél az ár mező törlése
NodeList nList4 = doc.getElementsByTagName("Termek");
Element Termek = null;

for(int temp = 0; temp < nList4.getLength(); temp++) {
    Termek = (Element) nList4.item(temp);
    Node priceNode = Termek.getElementsByTagName("Ar").item(0);
    Termek.removeChild(priceNode);
}

System.out.println("Az adott kategóriákban módosított adatok:");
System.out.println("\n1. módosítás: [Oldalszám: 240 -> Oldalszám: 300]");
System.out.println("2. módosítás: [Cím: Rút kiskacsa -> Cím: Rút kiscsibe]");

    for (int i2 = 0; i2 < nList.getLength(); i2++) {
        Node nNode = nList.item(i2);

        System.out.println("\nAktuális elem: " + nNode.getNodeName());

        if (nNode.getNodeType() == Node.ELEMENT_NODE) {
            Element eElement2 = (Element) nNode;

            System.out.println("Könyv ID-ja: " + eElement2.getAttribute("ID"));

            System.out.println("Cím: " +
                eElement2.getElementsByTagName("Cim").item(0).getTextContent());

            System.out.println("Ár: " +
                eElement2.getElementsByTagName("Ar").item(0).getTextContent());

            System.out.println("Szerző: " +
                eElement2.getElementsByTagName("Szerzo").item(0).getTextContent());

            System.out.println("Oldalszám: " +
                eElement2.getElementsByTagName("Oldalszam").item(0).getTextContent());
        }
    }

System.out.println("\n3. módosítás: [Név: Haragos Hugó -> Név: Haragos Hedvig]");

    for (int i = 0; i < nList2.getLength(); i++) {
        Node nNode = nList2.item(i);

        System.out.println("\nAktuális elem: " + nNode.getNodeName());

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

            System.out.println("Kölcsönző ID-ja: " + eElement.getAttribute("ID"));

            System.out.println("Név: " +
                eElement.getElementsByTagName("Nev").item(0).getTextContent());

            System.out.println("Cím: "+"\\n\\tVáros: " +
                eElement.getElementsByTagName("Varos").item(0).getTextContent()+"\\n\\tUtca
                : "

```

```

eElement.getElementsByTagName("Utca").item(0).getTextContent()+"\n\tHázzszám: "
+
eElement.getElementsByTagName("Hazzsam").item(0).getTextContent());

        System.out.println("Elérhetőség: "+"\\n\\tTel: " +
eElement.getElementsByTagName("Tel").item(0).getTextContent()+"\\n\\tEmail: "
+
eElement.getElementsByTagName("Email").item(0).getTextContent());
    }
}

System.out.println("\\n4. módosítás: [Alap fizetés beállítása a könyvtárosoknak]");

    for (int i = 0; i < nList3.getLength(); i++) {
        Node nNode = nList3.item(i);

        System.out.println("\\nAktuális elem: " + nNode.getNodeName());

        if (nNode.getNodeType() == Node.ELEMENT_NODE) {
            Element eElement3 = (Element) nNode;

            System.out.println("Könyvtáros ID-ja: " + eElement3.getAttribute("ID"));

            System.out.println("Név: " +
eElement3.getElementsByTagName("Nev").item(0).getTextContent());

            System.out.println("Elérhetőség: "+"\\n\\tTel: " +
eElement3.getElementsByTagName("Tel").item(0).getTextContent()+"\\n\\tEmail: "
+
eElement3.getElementsByTagName("Email").item(0).getTextContent());

            System.out.println("Kituntetes: "+"\\n\\tElső: " +
eElement3.getElementsByTagName("Elso").item(0).getTextContent()+"\\n\\tMásodik
: "
+
eElement3.getElementsByTagName("Masodik").item(0).getTextContent());

            System.out.println("Fizetés: " +
eElement3.getElementsByTagName("Fizetes").item(0).getTextContent());
        }
    }

System.out.println("\\n5. módosítás: [Termék gyerekelemből Ár törlése]");

    for (int i2 = 0; i2 < nList4.getLength(); i2++) {
        Node nNode = nList4.item(i2);

        System.out.println("\\nAktuális elem: " + nNode.getNodeName());

        if (nNode.getNodeType() == Node.ELEMENT_NODE) {
            Element eElement4 = (Element) nNode;

            System.out.println("Termék ID-ja: " + eElement4.getAttribute("ID"));

            System.out.println("Név: " +
eElement4.getElementsByTagName("Nev").item(0).getTextContent());

        }
    }
}

```

```

    }

    //Módosított XML kiírása a konzolra
    TransformerFactory transformerFactory = TransformerFactory.newInstance();
    Transformer transformer = transformerFactory.newTransformer();
    DOMSource source = new DOMSource(doc);
    System.out.println("\n-----A teljes módosított XML dokumentum-----");
    StreamResult consoleResult = new StreamResult(System.out);
    transformer.transform(source, consoleResult);

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

```

A konzolon megjelenő módosítások:

1. módosítás: A könyv adatai közül az oldalszám és cím módosítása

Régi oldalszám: 240

Új oldalszám: 300

2. módosítás:

Régi cím: A rút kiskacsa

Új cím: A rút kiscsibe

Az adott kategóriákban módosított adatok:

1. módosítás: [Oldalszám: 240 -> Oldalszám: 300]
2. módosítás: [Cím: Rút kiskacsa -> Cím: Rút kiscsibe]

Aktuális elem: Könyv
 Könyv ID-ja: K200
 Cím: A kis hajó
 Ár: 3500
 Szerző: Kiss Imre
 Oldalszám: 550

Aktuális elem: Könyv
 Könyv ID-ja: K201
 Cím: A nagy felleg
 Ár: 5200
 Szerző: Nagy Anita
 Oldalszám: 800

Aktuális elem: Könyv
 Könyv ID-ja: K202
 Cím: A rút kiscsibe
 Ár: 2750
 Szerző: Gellér László
 Oldalszám: 300

3. **módosítás:** Adott kölcsönző nevének módosítása

Régi név: Haragos Hugó

Új név: Haragos Hedvig

3. módosítás: [Név: Haragos Hugó -> Név: Haragos Hedvig]

Aktuális elem: Kolcsonzo

Kölcsönző ID-ja: K100

Név: Molnár Kata

Cím:

Város: Miskolc

Utca: Kossuth

Házzám: 1

Elérhetőség:

Tel: 06302342211

Email: molnar@gmail.com

Aktuális elem: Kolcsonzo

Kölcsönző ID-ja: K101

Név: Tóth Andor

Cím:

Város: Budapest

Utca: Nincs

Házzám: 30

Elérhetőség:

Tel: 06301234567

Email: toth@gmail.com

Aktuális elem: Kolcsonzo

Kölcsönző ID-ja: K102

Név: Haragos Hedvig

Cím:

Város: Eger

Utca: Egres

Házzám: 11

Elérhetőség:

Tel: 06309876541

Email: haragos@gmail.com

4. **módosítás:** Új, fizetés mező beállítása a könyvtárosoknak egy alap
220k-s értékkel

Régi mező: -

Új mező: Fizetés (220k)

4. módosítás: [Alap fizetés beállítása a könyvtárosoknak]

Aktuális elem: Konyvtaros
Könyvtáros ID-ja: K300
Név: Könyvtáros1
Elérhetőség:
Tel: 06301111111
Email: konyvtaros1@gmail.com
Kituntetes:
Első: Legtöbb kölcsönzés
Második: Év könyvtárosa 2020
Fizetés: 220000

Aktuális elem: Konyvtaros
Könyvtáros ID-ja: K301
Név: Könyvtáros2
Elérhetőség:
Tel: 06301111112
Email: konyvtaros2@gmail.com
Kituntetes:
Első: Legkedvesebb könyvtáros
Második: Legnagyobb tudás
Fizetés: 220000

Aktuális elem: Konyvtaros
Könyvtáros ID-ja: K302
Név: Könyvtáros3
Elérhetőség:
Tel: 06301111113
Email: konyvtaros3@gmail.com
Kituntetes:
Első: Legtöbb munkaóra
Második: Legtisztább munkakörnyezet
Fizetés: 220000

5. módosítás: Termékek kategóriából az ár mező törlése

Régi mező: Ár

Új mező: -

5. módosítás: [Termék gyerekelemből Ár törlése]

Aktuális elem: Termek
Termék ID-ja: T400
Név: Igénylőlap

Aktuális elem: Termek
Termék ID-ja: T401
Név: Könyvespolc

Aktuális elem: Termek
Termék ID-ja: T402
Név: Írószerek

