

JEGYZŐKÖNYV
Adatkezelés XML környezetben
Féléves feladat
Pékség

Készítette: **Schmidt Bence**

Neptunkód: **D1HKJZ**

A feladat leírása:

Feladatom egy olyan adatbázis ER modelljének a megtervezése, ami megfelel a normálformáknak és nem tartalmaz redundanciát. Az adatbázis megfelelő lesz a pékségek, azokban dolgozó pékek, ott használt eszközök, készített termékek és vevők tárolására, nyilvántartására és az eredmények lekérdezésére.

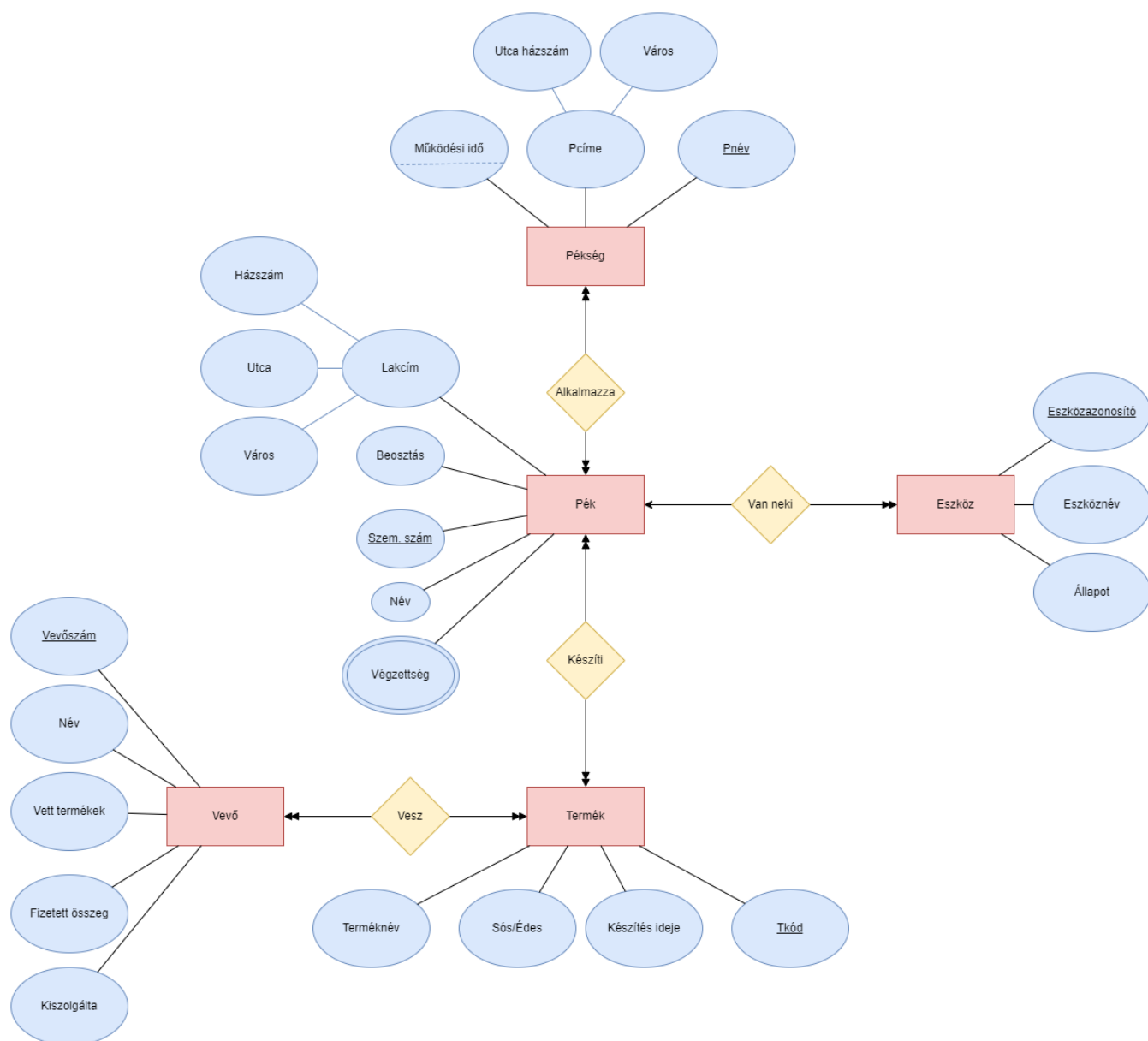
Egyedek:

- A **Pékség** egyed melynek tulajdonságai a Pnév mely primary key, Pcíme ami összetett tulajdonság, ez áll a Városból és Utca házszámból, és a Működési idő, mint származtatott tulajdonság.
-
- **Pék** egyed melynek jellemzői a Szem. szám ami primary key, Név, Beosztás, Lakcím ami összetett és a Végzettség ami többértékű.
- A **Termék** egyedhez tartozó tulajdonságok a Tkód a termékhez tartozó primary key, a Terméknév, Készítési ideje és hogy Sós/Édes-e.
- Létezik egy **Eszköz** egyed is mely a pépek eszközeit írja le, a tulajdonságai a következők: Eszközzazonosító, mint primary key, Eszköznév és Állapot.
- Van **Vevő** egyed is melynek tulajdonságai a Vevőszám ami primary key, Név, Vett termék, Fizetett összeg és aki Kiszolgált.

Kapcsolatok:

- Egy N:M kapcsolat a Pékség és az ott dolgozó Pék között
- N:1 típusú az Eszköz és az azt használó Pék között
- A kapcsolat N:M a Pék és az általa készített Termék között
- N:M kapcsolat van a Vevő és a vásárolt Termék között

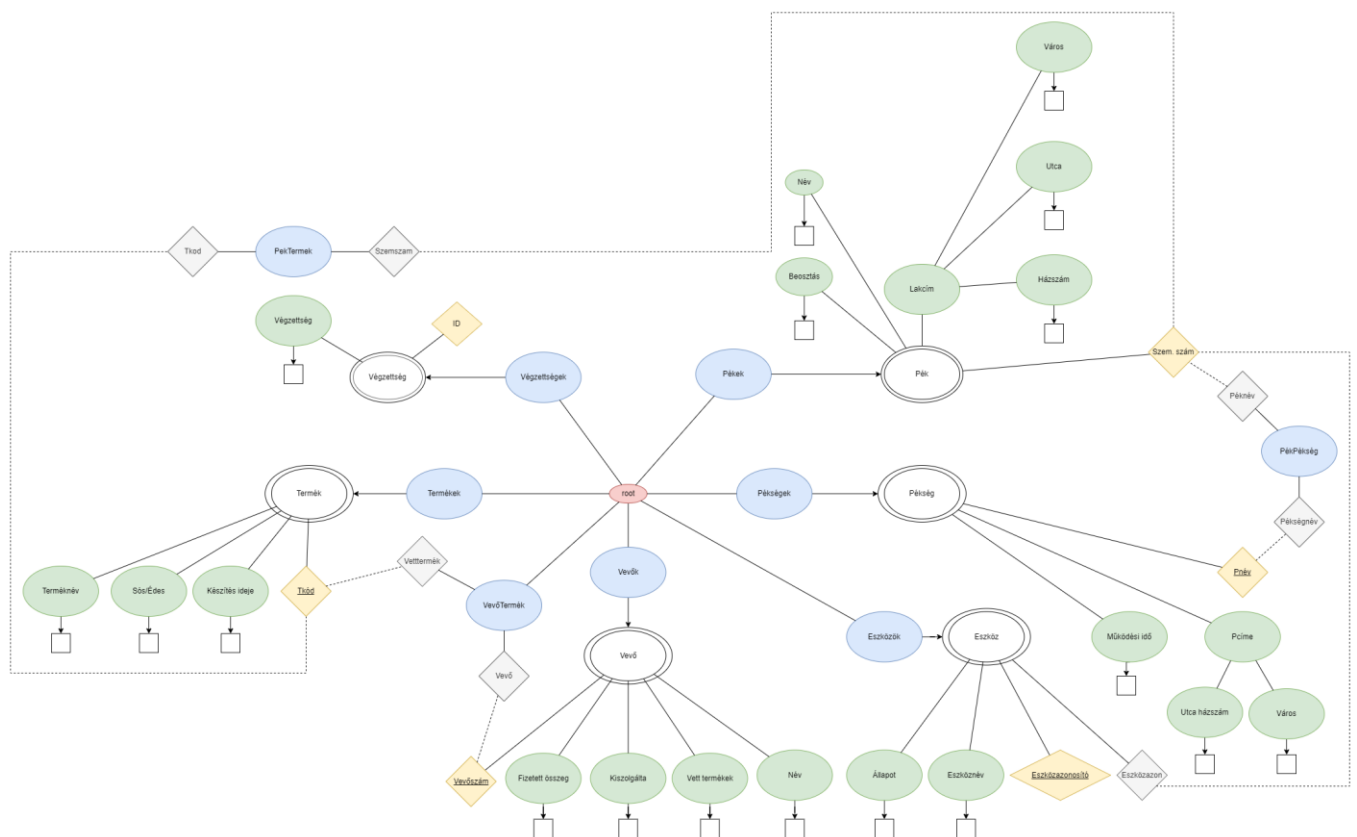
ER modell:



XDM modell:

ER modellt XDM modellre úgy konvertálhatunk, hogy három féle jelölést használunk. Van az elem, amely az XDM modellben ellipszissel ábrázoljuk. Az attribútumot rombuszsal kell jelölni, illetve a szöveget a téglalapban.

Ezekon felül van még a dupla ellipszis, amely a többszörös előfordulási lehetőséget határozza meg. Ezenkívül minden egyedhez és N:M kapcsolathoz felvesszük az egyed többszám alakját is.



Az XDM modell alapján XML dokumentum készítése

```
<?xml version="1.0" encoding="UTF-8"?>
<root xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="B5XVY7.xsd">

  <Pekek>
    <Pek szemszam="12345" eszkoz="00">
      <nev>Amal</nev>
      <beosztas>Főpék</beosztas>
      <lakcim>
        <varos>Miskolc</varos>
        <utca>Eper</utca>
        <hazszam>2</hazszam>
      </lakcim>
    </Pek>
    <Pek szemszam="23456" eszkoz="05">
      <nev>Darius</nev>
      <beosztas>Cukrász</beosztas>
      <lakcim>
        <varos>Tiszaújváros</varos>
        <utca>Kossuth</utca>
        <hazszam>4</hazszam>
      </lakcim>
    </Pek>
    <Pek szemszam="34567" eszkoz="02">
      <nev>Benedict</nev>
      <beosztas>Pék</beosztas>
      <lakcim>
        <varos>Eger</varos>
        <utca>Weöres</utca>
        <hazszam>78</hazszam>
      </lakcim>
    </Pek>
    <Pek szemszam="45678" eszkoz="03">
      <nev>Barbara</nev>
      <beosztas>Cukrász</beosztas>
      <lakcim>
        <varos>Sátoraljaújhely</varos>
        <utca>Deák</utca>
        <hazszam>12</hazszam>
      </lakcim>
    </Pek>
  </Pekek>

  <PekPeksegek>
    <PekPekseg pekszemszam="45678" peksegnev="pekseg3" />
    <PekPekseg pekszemszam="34567" peksegnev="pekseg1" />
  </PekPeksegek>
</root>
```

```
<PekPekseg pekszemszam="23456" peksegnev="pekseg2" />
<PekPekseg pekszemszam="12345" peksegnev="pekseg1" />
<PekPekseg pekszemszam="23456" peksegnev="pekseg4" />

</PekPeksegek>

<Peksegek>
  <Pekseg pnev="pekseg1">
    <pcim>Kenyéten Hold utca 3</pcim>
    <nyitas>2008-02-07 07:56:12</nyitas>
  </Pekseg>
  <Pekseg pnev="pekseg2">
    <pcim>Eger Bartók Béla út 15/B</pcim>
    <nyitas>2018-09-21 08:05:08</nyitas>
  </Pekseg>
  <Pekseg pnev="pekseg3">
    <pcim>Miskolc Eper utca 3</pcim>
    <nyitas>2010-12-17 16:18:34</nyitas>
  </Pekseg>
  <Pekseg pnev="pekseg4">
    <pcim>Miskolc Ifjúság utca 5</pcim>
    <nyitas>2001-09-08 01:51:25</nyitas>
  </Pekseg>
</Peksegek>

<Eszkozok>
  <Eszkoz eszkozazon="00">
    <eszkoznev>Munkaasztallap</eszkoznev>
    <allapot>1</allapot>
  </Eszkoz>
  <Eszkoz eszkozazon="01">
    <eszkoznev>Szakajtó</eszkoznev>
    <allapot>0</allapot>
  </Eszkoz>
  <Eszkoz eszkozazon="02">
    <eszkoznev>Kuglófforma</eszkoznev>
    <allapot>1</allapot>
  </Eszkoz>
  <Eszkoz eszkozazon="03">
    <eszkoznev>Péklapát</eszkoznev>
    <allapot>0</allapot>
  </Eszkoz>
  <Eszkoz eszkozazon="04">
    <eszkoznev>Nyújtófa</eszkoznev>
    <allapot>1</allapot>
  </Eszkoz>
  <Eszkoz eszkozazon="05">
    <eszkoznev>Péklapát</eszkoznev>
```

```

        <allapot>1</allapot>
    </Eszkoz>
</Eszkozok>

<Vevok>
    <Vevo vevoszam="00">
        <fizetett>1200</fizetett>
        <kiszolgalta>Barbara</kiszolgalta>
    </Vevo>
    <Vevo vevoszam="01">
        <fizetett>2400</fizetett>
        <kiszolgalta>Amal</kiszolgalta>
    </Vevo>
    <Vevo vevoszam="02">
        <fizetett>5000</fizetett>
        <kiszolgalta>Darius</kiszolgalta>
    </Vevo>
    <Vevo vevoszam="03">
        <fizetett>200</fizetett>
        <kiszolgalta>Barbara</kiszolgalta>
    </Vevo>
</Vevok>

<VevoTermek>
    <VevoTermek vevoszam="03" tkod="03" />
    <VevoTermek vevoszam="03" tkod="03" />
    <VevoTermek vevoszam="03" tkod="03" />
    <VevoTermek vevoszam="03" tkod="03" />
    <VevoTermek vevoszam="02" tkod="02" />
    <VevoTermek vevoszam="01" tkod="00" />
    <VevoTermek vevoszam="01" tkod="01" />
    <VevoTermek vevoszam="01" tkod="00" />
    <VevoTermek vevoszam="00" tkod="01" />

</VevoTermek>

<Termek>
    <Termek tkod="00">
        <termeknev>Kuglóf</termeknev>
        <sosedes>1</sosedes>
        <kesitesideje>22/Nov/2021</kesitesideje>
    </Termek>
    <Termek tkod="01">
        <termeknev>Kenyér</termeknev>
        <sosedes>0</sosedes>
        <kesitesideje>23/Nov/2021</kesitesideje>
    </Termek>
    <Termek tkod="02">

```

```
<termeknev>Torta</termeknev>
<sosedes>1</sosedes>
<kesitesideje>23/Nov/2021</kesitesideje>
</Termek>
<Termek tkod="03">
  <termeknev>Zsemle</termeknev>
  <sosedes>0</sosedes>
  <kesitesideje>22/Nov/2021</kesitesideje>
</Termek>
</Termekek>

<Vegzettsegek>
  <Vegzettseg ID="veg1">
    <megnevezes>Pék</megnevezes>
  </Vegzettseg>
  <Vegzettseg ID="veg2">
    <megnevezes>Cukrász</megnevezes>
  </Vegzettseg>
  <Vegzettseg ID="veg3">
    <megnevezes>Eladó</megnevezes>
  </Vegzettseg>
  <Vegzettseg ID="veg4">
    <megnevezes>Boltvezető</megnevezes>
  </Vegzettseg>
</Vegzettsegek>

<PekVegzettsegek>
  <PekVegzettse szemszam="12345" vegID="veg1" />
  <PekVegzettse szemszam="12345" vegID="veg1" />
  <PekVegzettse szemszam="12345" vegID="veg4" />
  <PekVegzettse szemszam="23456" vegID="veg2" />
  <PekVegzettse szemszam="23456" vegID="veg3" />
  <PekVegzettse szemszam="34567" vegID="veg1" />
  <PekVegzettse szemszam="45678" vegID="veg2" />
  <PekVegzettse szemszam="45678" vegID="veg3" />

</PekVegzettsegek>

<PekTermekek>

  <PekTermek szemszam="12345" tkod="00" />
  <PekTermek szemszam="23456" tkod="03" />
  <PekTermek szemszam="23456" tkod="02" />
  <PekTermek szemszam="34567" tkod="03" />
  <PekTermek szemszam="45678" tkod="00" />
  <PekTermek szemszam="45678" tkod="01" />

</PekTermekek>
```



```
</root>
```

Az XML dokumentum alapján XMLSchema készítése

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema attributeFormDefault="unqualified"
elementFormDefault="qualified"
xmlns:xs="http://www.w3.org/2001/XMLSchema">

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

  <xs:element name="Pek">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="nev" type="xs:string"/>
        <xs:element name="beosztas" type="xs:string"/>
        <xs:element ref="lakcim"/>
      </xs:sequence>
      <xs:attribute type="xs:int" name="szemszam" use="required"/>
      <xs:attribute type="xs:int" name="eszkoz" use="optional"/>
    </xs:complexType>
  </xs:element>

  <xs:element name="PekPekseg">
    <xs:complexType>
      <xs:simpleContent>
        <xs:extension base="xs:string">
          <xs:attribute type="xs:int" name="pekszemszam"
use="required"/>
          <xs:attribute type="xs:string" name="peksegnev"
use="required"/>
        </xs:extension>
      </xs:simpleContent>
    </xs:complexType>
  </xs:element>

  <xs:element name="Pekseg">
```

```

<xs:complexType>
  <xs:sequence>
    <xs:element name="pcim" type="xs:string"/>
    <xs:element name="nyitas" type="xs:string"/>
  </xs:sequence>
  <xs:attribute type="xs:string" name="pnev" use="required"/>
</xs:complexType>
</xs:element>

<xs:element name="Eszkoz">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="eszkoznev" type="xs:string"/>
      <xs:element name="allapot" type="xs:boolean"/>
    </xs:sequence>
    <xs:attribute type="xs:int" name="eszkoszazon" use="required"/>
  </xs:complexType>
</xs:element>

<xs:element name="Vevo">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="fizetett" type="xs:short"/>
      <xs:element name="kiszolgalta" type="xs:string"/>
    </xs:sequence>
    <xs:attribute type="xs:int" name="vevoszam" use="required"/>
  </xs:complexType>
</xs:element>

<xs:element name="VevoTermek">
  <xs:complexType>
    <xs:simpleContent>
      <xs:extension base="xs:string">
        <xs:attribute type="xs:int" name="vevoszam"
use="required"/>
        <xs:attribute type="xs:int" name="tkod" use="required"/>
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
</xs:element>

<xs:element name="Termek">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="termeknev" type="xs:string"/>
      <xs:element name="sosedes" type="xs:int"/>
      <xs:element name="kesitesideje" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

```

        </xs:sequence>
        <xs:attribute type="xs:int" name="tkod" use="required"/>
    </xs:complexType>
</xs:element>

<xs:element name="Vegzettseg">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="megnevezes" type="xs:string"/>
        </xs:sequence>
        <xs:attribute type="xs:string" name="ID" use="required"/>
    </xs:complexType>
</xs:element>

<xs:element name="PekVegzettse">
    <xs:complexType>
        <xs:simpleContent>
            <xs:extension base="xs:string">
                <xs:attribute type="xs:int" name="szemszam"
use="required"/>
                <xs:attribute type="xs:string" name="vegID"
use="required"/>
            </xs:extension>
        </xs:simpleContent>
    </xs:complexType>
</xs:element>

<xs:element name="PekTermek">
    <xs:complexType>
        <xs:simpleContent>
            <xs:extension base="xs:string">
                <xs:attribute type="xs:int" name="szemszam"
use="required"/>
                <xs:attribute type="xs:string" name="tkod"
use="required"/>
            </xs:extension>
        </xs:simpleContent>
    </xs:complexType>
</xs:element>

<xs:element name="Pekek">
    <xs:complexType>
        <xs:sequence>
            <xs:element ref="Pek" maxOccurs="unbounded" minOccurs="0"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="PekPeksegek">

```

```

    <xs:complexType>
      <xs:sequence>
        <xs:element ref="PekPekseg" maxOccurs="unbounded"
minOccurs="0"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="Peksegek">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="Pekseg" maxOccurs="unbounded"
minOccurs="0"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="Eszkozok">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="Eszkoz" maxOccurs="unbounded"
minOccurs="0"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="Vevok">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="Vevo" maxOccurs="unbounded" minOccurs="0"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="VevoTermekek">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="VevoTermek" maxOccurs="unbounded"
minOccurs="0"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="Termekek">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="Termek" maxOccurs="unbounded"
minOccurs="0"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="Vegzettsegek">
    <xs:complexType>

```

```

        <xs:sequence>
            <xs:element ref="Vegzettseg" maxOccurs="unbounded"
minOccurs="0"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="PekVegzettsegek">
    <xs:complexType>
        <xs:sequence>
            <xs:element ref="PekVegzettse" maxOccurs="unbounded"
minOccurs="0"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="PekTermek">
    <xs:complexType>
        <xs:sequence>
            <xs:element ref="PekTermek" maxOccurs="unbounded"
minOccurs="0"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="root">
    <xs:complexType>
        <xs:sequence>
            <xs:element ref="Pekek"/>
            <xs:element ref="PekPeksegek"/>
            <xs:element ref="Peksegek"/>
            <xs:element ref="Eszkozok"/>
            <xs:element ref="Vevok"/>
            <xs:element ref="VevoTermek">
            <xs:element ref="Termek">
            <xs:element ref="Vegzettsegek"/>
            <xs:element ref="PekVegzettsegek"/>
            <xs:element ref="PekTermek"/>
        </xs:sequence>
    </xs:complexType>
    <xs:key name="szemszam">
        <xs:selector xpath="Pek"/>
        <xs:field xpath="@szemszam"/>
    </xs:key>
    <xs:key name="pnev">
        <xs:selector xpath="Pekseg"/>
        <xs:field xpath="@pnev"/>
    </xs:key>
    <xs:key name="eszkozazon">
        <xs:selector xpath="Eszkoz"/>
        <xs:field xpath="@eszkozazon"/>
    </xs:key>

```

```
</xs:key>
<xs:key name="vevoszam">
  <xs:selector xpath="Vevo"/>
  <xs:field xpath="@vevoszam"/>
</xs:key>
<xs:key name="tkod">
  <xs:selector xpath="Termek"/>
  <xs:field xpath="@tkod"/>
</xs:key>
<xs:key name="ID">
  <xs:selector xpath="Vevo"/>
  <xs:field xpath="@ID"/>
</xs:key>
<xs:keyref name="eszkoz" refer="eszkozazon">
  <xs:selector xpath="Pek"/>
  <xs:field xpath="@eszkoz"/>
</xs:keyref>
<xs:keyref name="pekszemszam" refer="szemszam">
  <xs:selector xpath="PekPekseg"/>
  <xs:field xpath="@szemszam"/>
</xs:keyref>
<xs:keyref name="peksegnev" refer="pnev">
  <xs:selector xpath="PekPekseg"/>
  <xs:field xpath="@pnev"/>
</xs:keyref>
<xs:keyref name="fkvevoszam" refer="vevoszam">
  <xs:selector xpath="VevoTermek"/>
  <xs:field xpath="@vevoszam"/>
</xs:keyref>
<xs:keyref name="fktkod" refer="tkod">
  <xs:selector xpath="VevoTermek"/>
  <xs:field xpath="@tkod"/>
</xs:keyref>
<xs:keyref name="fkvegzettsegszemszam" refer="szemszam">
  <xs:selector xpath="PekVegzettse"/>
  <xs:field xpath="@szemszam"/>
</xs:keyref>
<xs:keyref name="vegID" refer="ID">
  <xs:selector xpath="PekVegzettse"/>
  <xs:field xpath="@ID"/>
</xs:keyref>
<xs:keyref name="fkszemszam" refer="szemszam">
  <xs:selector xpath="PekTermek"/>
  <xs:field xpath="@szemszam"/>
</xs:keyref>
<xs:keyref name="fkpektermektkod" refer="tkod">
  <xs:selector xpath="PekTermek"/>
  <xs:field xpath="@tkod"/>
```

```
</xs:keyref>
</xs:element>
</xs:schema>
```

Adatolvasás

Dom segítségével listázza ki az összes adatot.

```
public static void main(String[] args) {
    NodeList list;
    try {
        DocumentBuilderFactory
factory=DocumentBuilderFactory.newInstance();
        DocumentBuilder
documentBuilder=factory.newDocumentBuilder();
        Document document=documentBuilder.parse(new
File("B5XVY7.xml"));
        document.getDocumentElement().normalize();
        System.out.println("Root element : " +
document.getDocumentElement().getNodeName());
        System.out.println("-----");
        list=document.getElementsByTagName("Pek");

        for (int i=0;i<list.getLength();i++) {
            Node node=list.item(i);
            System.out.println("\nAktuális elem: " +
node.getNodeName());
            if(node.getNodeType() == Node.ELEMENT_NODE) {
                Element element=(Element) node;
                System.out.println("Személyi szám: " +
element.getAttribute("szemszam"));
                System.out.println("Eszközazonosító id: " +
element.getAttribute("eszkoz"));
                System.out.println("Név: " +
element.getElementsByTagName("nev").item(0).getTextContent());
                System.out.println("Beosztás: " +
element.getElementsByTagName("beosztas").item(0).getTextContent());

                Node nodelakcim=list.item(i);
                if(nodelakcim.getNodeType()==Node.ELEMENT_NODE){
                    Element elementlakcim=(Element) node;
                    System.out.println("Lakcím: "
                                +
elementlakcim.getElementsByTagName("varos").item(0).getTextContent()
                                + " " +
elementlakcim.getElementsByTagName("utca").item(0).getTextContent()
                                + " " +
elementlakcim.getElementsByTagName("hazszam").item(0).getTextContent());
                }
            }
        }
    } catch (Exception e) {
        e.printStackTrace();
    }
}
```

```

+ " " +
elementlalcim.getElementsByTagName("hazszam").item(0).getTextContent
());
    }
}

list=document.getElementsByTagName("Pekseg");

for(int i=0;i<list.getLength();i++){
    Node node=list.item(i);
    System.out.println("\nAktuális elem: " +
node.getNodeName());
    if(node.getNodeType() == Node.ELEMENT_NODE) {
        Element element=(Element) node;
        System.out.println("Pékség név: " +
element.getAttribute("pnev"));
        System.out.println("Cím: " +
element.getElementsByTagName("pcim").item(0).getTextContent());
        System.out.println("Nyitott: " +
element.getElementsByTagName("nyitas").item(0).getTextContent());
    }
}

list=document.getElementsByTagName("Eszkoz");

for(int i=0;i<list.getLength();i++){
    Node node=list.item(i);
    System.out.println("\nAktuális elem: " +
node.getNodeName());
    //Verseny Bírók adatainak kiírása
    if(node.getNodeType() == Node.ELEMENT_NODE) {
        Element element=(Element) node;
        System.out.println("Eszköz id: " +
element.getAttribute("eszkozazon"));
        System.out.println("Eszköznév: " +
element.getElementsByTagName("eszkoznev").item(0).getTextContent());
        System.out.println("Állapot: " +
element.getElementsByTagName("allapot").item(0).getTextContent());
    }
}

list=document.getElementsByTagName("Vevo");

for(int i=0;i<list.getLength();i++){
    Node node=list.item(i);
    System.out.println("\nAktuális elem: " +
node.getNodeName());

```



```

        if(node.getNodeType() == Node.ELEMENT_NODE) {
            Element element=(Element) node;
            System.out.println("Vevőszám: " +
element.getAttribute("vevoszam"));
            System.out.println("Fizetett összeg: " +
element.getElementsByTagName("fizetett").item(0).getTextContent());
            System.out.println("Kiszolgálta: " +
element.getElementsByTagName("kiszolgalta").item(0).getTextContent()
);
        }
    }

    list=document.getElementsByTagName("Termek");

    for(int i=0;i<list.getLength();i++){
        Node node=list.item(i);
        System.out.println("\nAktuális elem: " +
node.getNodeName());
        if(node.getNodeType() == Node.ELEMENT_NODE) {
            Element element=(Element) node;
            System.out.println("Termek kód: " +
element.getAttribute("tkod"));
            System.out.println("Név: " +
element.getElementsByTagName("termeknev").item(0).getTextContent());
            System.out.println("Sós/Édes: " +
element.getElementsByTagName("sosedes").item(0).getTextContent());
            System.out.println("Készítés ideje: " +
element.getElementsByTagName("kesitesideje").item(0).getTextContent(
));
        }
    }

    list=document.getElementsByTagName("Vegzettseg");

    for(int i=0;i<list.getLength();i++){
        Node node=list.item(i);
        System.out.println("\nAktuális elem: " +
node.getNodeName());
        if(node.getNodeType() == Node.ELEMENT_NODE) {
            Element element=(Element) node;
            System.out.println("Végzettség id: " +
element.getAttribute("ID"));
            System.out.println("Neve: " +
element.getElementsByTagName("megnevezes").item(0).getTextContent()
);
        }
    }

```

```

    }

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

```

Adatmódosítás:

Az eszközök elementek közül kikeresi azokat az eszközöket amelyek még nincsenek elhasználva (azaz az állapotuk 1) majd ezek állapotát 0-ra állítja és egy új XML fájlba kiírja a módosított verziót.

```

public static void main(String[] args) {
    NodeList nodeList;
    try{
        DocumentBuilderFactory factory
=DocumentBuilderFactory.newInstance();
        DocumentBuilder builder=factory.newDocumentBuilder();
        Document document=builder.parse(new File("B5XVY7.xml"));
        document.getDocumentElement().normalize();

        System.out.println("Minden eszköz állapot 0-ra
állítása:");
        nodeList=document.getElementsByTagName("Eszkoz");
        for(int i =0;i<nodeList.getLength();i++){
            Node node=nodeList.item(i);
            System.out.println("\nAktuális elem: " +
node.getNodeName());
            if(node.getNodeType()==Node.ELEMENT_NODE){
                Element element=(Element) node;
                element.getElementsByTagName("allapot").item(0).
setTextContent("0");
                System.out.println("Eszköz id: " +
element.getAttribute("eszkoszazon"));
                System.out.println("Eszköznév: " +
element.getElementsByTagName("eszkoznev").item(0).getTextContent());
                System.out.println("Állapot: " +
element.getElementsByTagName("allapot").item(0).getTextContent());
            }
        }
    }
}

```

```

        //létrehozza az xml fájlt módosítva
        Transformer transformer=
TransformerFactory.newInstance().newTransformer();
        Source input=new DOMSource(document);
        Result output=new StreamResult(new
File("Modded_B5XVY.xml"));
        transformer.transform(input,output);
    }catch (ParserConfigurationException e){
        e.printStackTrace();
    } catch (Exception e){
        e.printStackTrace();
    }
}

```

Adatlekérdezés

A működő eszközöket gyűjti össze, és kiírja a console-ra.

```

    try {
        NodeList nodeList;

        DocumentBuilderFactory factory
=DocumentBuilderFactory.newInstance();
        DocumentBuilder builder=factory.newDocumentBuilder();
        Document document=builder.parse(new File("B5XVY7.xml"));
        document.getDocumentElement().normalize();

        nodeList=document.getElementsByTagName("Eszkoz");

        System.out.println("Működő eszközök listája(állapot ==
1): ");
        for(int i =0;i<nodeList.getLength();i++){
            Node node=nodeList.item(i);

            if(node.getNodeType()==Node.ELEMENT_NODE){
                Element element=(Element) node;
                String allapot=
element.getElementsByTagName("allapot").item(0).getTextContent();

                if(allapot.equals("1"))
                {
                    System.out.println("\nAktuális elem: " +
node.getNodeName());

                    System.out.println("-----");
                    System.out.println("");

```

```
        System.out.println("Eszköz id: " +  
element.getAttribute("eszkoszazon"));  
        System.out.println("Eszköznév: " +  
element.getElementsByTagName("eszkoszev").item(0).getTextContent());  
        System.out.println("Állapot: " +  
element.getElementsByTagName("allapot").item(0).getTextContent());  
    }  
}  
} catch (ParserConfigurationException | IOException |  
SAXException e){  
    e.printStackTrace();  
}
```