

JEGYZŐKÖNYV

Adatkezelés XML környezetben

Féléves feladat

Pizzaláncok modellezése

Készítette: **Kévés Levente**

Neptunkód: **IG9GGV**

A feladat leírása:

Pizzaláncok adatait és kapcsolatait tároló adatbázis modell. Kitér a pizzaláncokra, a vezetőségükre, a hozzájuk tartozó éttermekre, az ott dolgozó alkalmazottakra, valamint az ott kapható ételekre.

A pizzaláncok rendelkeznek egy névvel és egy alapítási időponttal, valamint feljegyzésre kerül az, hogy az adott pizzalánc tőzsdén jegyzett, vagy sem.

A pizzaláncokat vezeti egy vezérigazgató, akinek neve, születési dátuma, és neve kerül. A pizzaláncok és a vezérigazgatók között 1:1 kapcsolat van. Ehhez a kapcsolathoz tartozik a kinevezés időpontja, mint attribútum.

Egy-egy pizzalánc több étteremből épül fel, amiknek címe (város, utca, házszám) megtalálható az adatbázisban. Egy pizzalánc és az ahhoz tartozó éttermek között 1:N kapcsolat áll fenn.

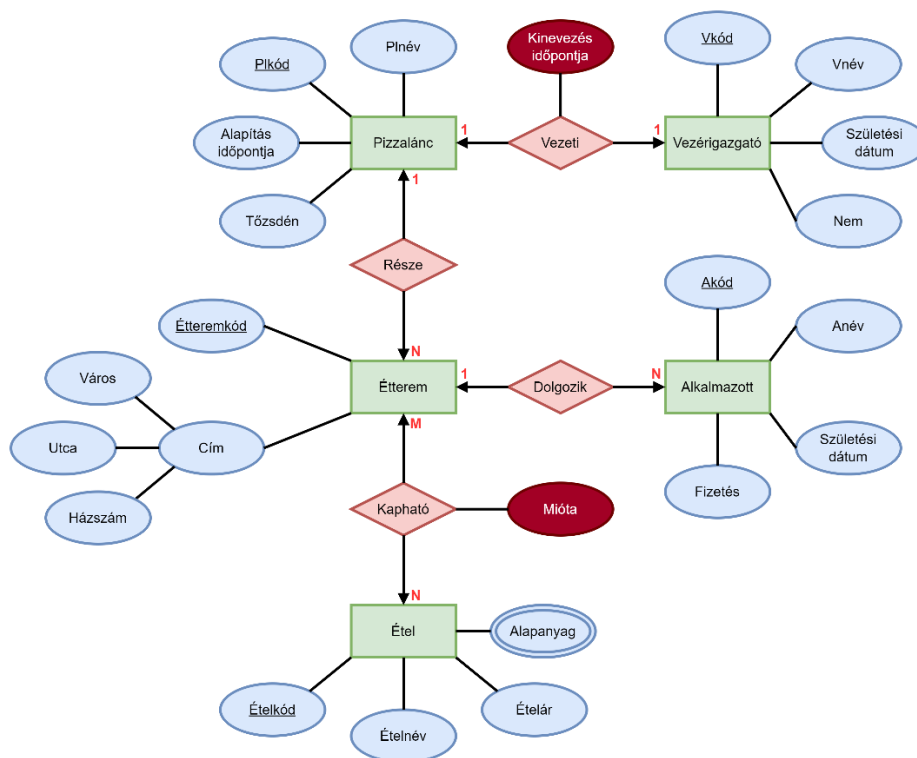
Az éttermekben több étel is kapható. Az ételek neve, ára, és az elkészítéséhez szükséges alapanyagok megtalálhatók az adatbázisban. Az éttermek és az ott kapható ételek között N:M kapcsolat áll fenn, amihez tartozik egy úgynevezett „Mióta” tulajdonság, ami azt írja le, hogy az adott étel az adott étteremben milyen időponttól kezdve volt elérhető

Az éttermekben dolgozó alkalmazottak neve, fizetése, születési dátuma az adatbázis tartalmi része. Egy alkalmazott csak egy étteremben dolgozhat, azaz az éttermek és az alkalmazottak között 1:N kapcsolat áll fenn.

Az éttermekben dolgozó alkalmazottak egy szabályozás miatt egyszerre csak egy pizzalánc éttermében dolgozhatnak. Az alkalmazottak neve, fizetése, születési dátuma az adatbázis tartalmi része.

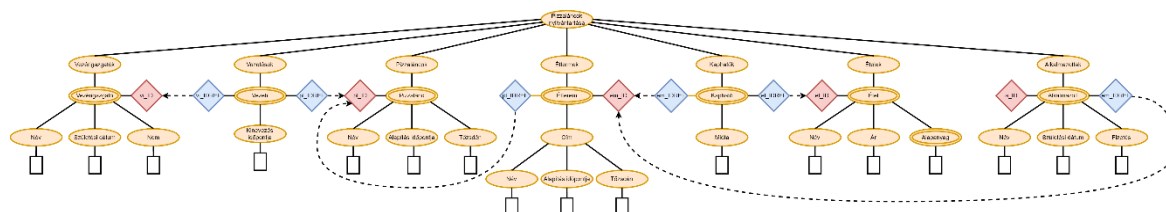
1. feladat

1a) Az adatbázis ER modell:



1b) Az adatbázis konvertálása XDM modellre:

Az ER modell egyedeiből elemek, az egyedek tulajdonságaiból gyerekelemek lesznek. A kulcs tulajdonságok az elemek attribútumaként fog megjelenni. Az ER modellben a Pizzaláncok és a Vezérigazgatók egyedek közötti „Vezeti” kapcsolat külön elemként fog megjelenni az XDM modellben, a hozzá tartozó „Kinevezés időpontja” tulajdonság pedig annak a gyerekeleme lesz. Az Éttermek és az Ételek egyedek közötti „Kapható” kapcsolat ugyanilyen módon kerül megvalósításra.



1c) Az XDM modell alapján XML dokumentum készítése:

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

```
<pizzalanc nyilvantartas xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="XMLSchemaI9GGV.xsd">
  <pizzalancok >
    <pizzalanc pl_ID="01">
      <nev>McPizza</nev>
      <alapitas_idopontja>1987-11-21</alapitas_idopontja>
      <tozsden>true</tozsden>
    </pizzalanc>
    <pizzalanc pl_ID="02">
      <nev>Pizza King</nev>
      <alapitas_idopontja>1975-07-23</alapitas_idopontja>
      <tozsden>false</tozsden>
    </pizzalanc>
  </pizzalancok>
  <vezerigazgatok>
    <vezerigazgato vi_ID="01">
      <nev>Török Vilmos</nev>
      <szulesesi_datum>1984-03-13</szulesesi_datum>
      <nem>Férfi</nem>
    </vezerigazgato>
    <vezerigazgato vi_ID="02">
      <nev>Fekete Ágnes</nev>
      <szulesesi_datum>1974-04-23</szulesesi_datum>
      <nem>Nő</nem>
    </vezerigazgato>
  </vezerigazgatok>
  <etteremek>
    <etterem em_ID="01" pl_IDREF="01">
      <cim>
        <varos>Budapest</varos>
        <utca>Nagy út</utca>
        <hazszam>12</hazszam>
      </cim>
    </etterem>
    <etterem em_ID="02" pl_IDREF="01">
      <cim>
        <varos>Debrecen</varos>
        <utca>Kerek út</utca>
        <hazszam>5</hazszam>
      </cim>
    </etterem>
    <etterem em_ID="03" pl_IDREF="02">
      <cim>
        <varos>Budapest</varos>
        <utca>Kis út</utca>
        <hazszam>7</hazszam>
      </cim>
    </etterem>
    <etterem em_ID="04" pl_IDREF="02">
      <cim>
        <varos>Budapest</varos>
        <utca>Nagy tér</utca>
        <hazszam>21</hazszam>
      </cim>
    </etterem>
  </etteremek>
</pizzalanc nyilvantartas>
```

```

    <etterem em_ID="05" pl_IDREF="02">
      <cim>
        <varos>Debrecen</varos>
        <utca>Kis tér</utca>
        <hazszam>13</hazszam>
      </cim>
    </etterem>
  </etteremek>

  <etelek>
    <etel el_ID="01">
      <nev>Sonkás pizza</nev>
      <ar>1100</ar>
      <alapanyag>Sonka</alapanyag>
      <alapanyag>Sajt</alapanyag>
    </etel>
    <etel el_ID="02">
      <nev>Sonkás-kukoricás pizza</nev>
      <ar>1300</ar>
      <alapanyag>Sonka</alapanyag>
      <alapanyag>Kukorica</alapanyag>
      <alapanyag>Sajt</alapanyag>
    </etel>
    <etel el_ID="03">
      <nev>Szalámis pizza</nev>
      <ar>1600</ar>
      <alapanyag>Szalámi</alapanyag>
      <alapanyag>Sajt</alapanyag>
    </etel>
  </etelek>

  <alkalmazottak>
    <alkalmazott a_ID="01" em_IDREF="05">
      <nev>Fekete Balázs</nev>
      <szulesesi_datum>1988-01-12</szulesesi_datum>
      <fizetes>260000</fizetes>
    </alkalmazott>
    <alkalmazott a_ID="02" em_IDREF="04">
      <nev>Balogh Sára</nev>
      <szulesesi_datum>1977-02-22</szulesesi_datum>
      <fizetes>300000</fizetes>
    </alkalmazott>
    <alkalmazott a_ID="03" em_IDREF="05">
      <nev>Csonka Gábor</nev>
      <szulesesi_datum>2001-12-12</szulesesi_datum>
      <fizetes>460000</fizetes>
    </alkalmazott>
    <alkalmazott a_ID="04" em_IDREF="02">
      <nev>Farkas Anita</nev>
      <szulesesi_datum>1999-03-21</szulesesi_datum>
      <fizetes>257000</fizetes>
    </alkalmazott>
    <alkalmazott a_ID="05" em_IDREF="01">
      <nev>Antal Benjamin</nev>
      <szulesesi_datum>1956-11-30</szulesesi_datum>
      <fizetes>310000</fizetes>
    </alkalmazott>
    <alkalmazott a_ID="06" em_IDREF="03">
      <nev>Csonka Ábel</nev>
      <szulesesi_datum>1976-07-11</szulesesi_datum>
      <fizetes>265000</fizetes>
    </alkalmazott>
    <alkalmazott a_ID="07" em_IDREF="03">
      <nev>Vászoly Martina</nev>
      <szulesesi_datum>1987-12-21</szulesesi_datum>
      <fizetes>280000</fizetes>
    </alkalmazott>
    <alkalmazott a_ID="08" em_IDREF="03">
      <nev>Szabó Soma</nev>
      <szulesesi_datum>1994-05-12</szulesesi_datum>
      <fizetes>400000</fizetes>
    </alkalmazott>
  </alkalmazottak>

  <vezetesek>
    <vezeti pl_IDREF="01" vi_IDREF="01">
      <kinevezes_idopontja>2015-01-12</kinevezes_idopontja>
    </vezeti>
    <vezeti pl_IDREF="02" vi_IDREF="02">
      <kinevezes_idopontja>2020-10-11</kinevezes_idopontja>
    </vezeti>
  </vezetesek>

```

```
<kaphatok>
  <kaphato em_IDREF="01" el_IDREF="01">
    <miota>1987-11-21</miota>
  </kaphato>
  <kaphato em_IDREF="01" el_IDREF="02">
    <miota>1987-11-21</miota>
  </kaphato>
  <kaphato em_IDREF="01" el_IDREF="03">
    <miota>1987-11-21</miota>
  </kaphato>
  <kaphato em_IDREF="02" el_IDREF="02">
    <miota>1997-12-23</miota>
  </kaphato>
  <kaphato em_IDREF="02" el_IDREF="03">
    <miota>1987-12-24</miota>
  </kaphato>
  <kaphato em_IDREF="03" el_IDREF="03">
    <miota>2001-12-21</miota>
  </kaphato>
  <kaphato em_IDREF="04" el_IDREF="01">
    <miota>1987-06-21</miota>
  </kaphato>
  <kaphato em_IDREF="04" el_IDREF="03">
    <miota>1987-12-30</miota>
  </kaphato>
  <kaphato em_IDREF="05" el_IDREF="01">
    <miota>2011-12-21</miota>
  </kaphato>
  <kaphato em_IDREF="05" el_IDREF="02">
    <miota>2020-12-21</miota>
  </kaphato>
</kaphatok>
</pizzalanc_nyilvantartas>
```

1d) Az XML dokumentum alapján XMLSchema készítése (saját típusok):

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <xs:element name="pizzalanc_nyilvantartas">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="pizzalancok">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="pizzalanc" type="pizzalancTipus" maxOccurs="unbounded"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="vezerigazgatok">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="vezerigazgato" type="vezerigazgatoTipus" maxOccurs="unbounded"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="etteremek">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="etterem" type="etteremTipus" maxOccurs="unbounded"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="etelek">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="etel" type="etelTipus" maxOccurs="unbounded"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="alkalmazottak">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="alkalmazott" type="alkalmazottTipus" maxOccurs="unbounded"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="vezetesek">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="vezeti" type="vezetiTipus" maxOccurs="unbounded"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="kaphatok">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="kaphato" type="kaphatoTipus" maxOccurs="unbounded"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>

    <xs:key name="pizzalancPK">
      <xs:selector xpath="pizzalancok/pizzalanc" />
      <xs:field xpath="@pl_ID" />
    </xs:key>
    <xs:key name="vezerigazgatoPK">
      <xs:selector xpath="vezerigazgatok/vezerigazgato" />
      <xs:field xpath="@vi_ID" />
    </xs:key>
    <xs:key name="etteremPK">
      <xs:selector xpath="etteremek/etterem" />
      <xs:field xpath="@em_ID" />
    </xs:key>
    <xs:key name="etelPK">
      <xs:selector xpath="etelek/etel" />
      <xs:field xpath="@el_ID" />
    </xs:key>
    <xs:key name="alkalmazottPK">
      <xs:selector xpath="alkalmazottak/alkalmazott" />
      <xs:field xpath="@a_ID" />
    </xs:key>
  </xs:element>
</xs:schema>
```

```

<xs:keyref refer="etteremPK" name="etteremFK">
  <xs:selector xpath="kaphatok/kaphato" />
  <xs:field xpath="@em_IDREF" />
</xs:keyref>
<xs:keyref refer="etelPK" name="etelFK">
  <xs:selector xpath="kaphatok/kaphato" />
  <xs:field xpath="@el_IDREF" />
</xs:keyref>
<xs:keyref refer="pizzalancPK" name="pizzalancFK">
  <xs:selector xpath="vezetesekek/vezeti" />
  <xs:field xpath="@pl_IDREF" />
</xs:keyref>
<xs:keyref refer="vezerigazgatoPK" name="vezerigazgatoFK">
  <xs:selector xpath="vezetesekek/vezeti" />
  <xs:field xpath="@vi_IDREF" />
</xs:keyref>

<xs:keyref refer="pizzalancPK" name="etteremPizzalancFK">
  <xs:selector xpath="ettermek/etterem" />
  <xs:field xpath="@pl_IDREF" />
</xs:keyref>
<xs:keyref refer="etteremPK" name="alkalmazottEtteremFK">
  <xs:selector xpath="alkalmazottak/alkalmazott" />
  <xs:field xpath="@em_IDREF" />
</xs:keyref>
<xs:unique name="unique-vezetes-pl">
  <xs:selector xpath="vezetesekek/vezeti" />
  <xs:field xpath="@pl_IDREF" />
</xs:unique>
<xs:unique name="unique-vezetes-vi">
  <xs:selector xpath="vezetesekek/vezeti" />
  <xs:field xpath="@vi_IDREF" />
</xs:unique>
</xs:element>

<xs:complexType name="pizzalancTipus">
  <xs:sequence>
    <xs:element name="nev" type="xs:string" />
    <xs:element name="alapitas_idopontja" type="xs:date" />
    <xs:element name="tozsden" type="xs:boolean" />
  </xs:sequence>
  <xs:attribute name="pl_ID" type="xs:integer" use="required" />
</xs:complexType>

<xs:complexType name="vezerigazgatoTipus">
  <xs:sequence>
    <xs:element name="nev" type="xs:string" />
    <xs:element name="szuletesi_datum" type="xs:date" />
    <xs:element name="nem" type="xs:string" />
  </xs:sequence>
  <xs:attribute name="vi_ID" type="xs:integer" use="required" />
</xs:complexType>

<xs:complexType name="etteremTipus">
  <xs:sequence>
    <xs:element name="cim" type="cimTipus" />
  </xs:sequence>
  <xs:attribute name="em_ID" type="xs:integer" use="required" />
  <xs:attribute name="pl_IDREF" type="xs:integer" use="required" />
</xs:complexType>

<xs:complexType name="etelTipus">
  <xs:sequence>
    <xs:element name="nev" type="xs:string" />
    <xs:element name="ar" type="xs:integer" />
    <xs:element name="alapanyag" type="xs:string" minOccurs="unbounded" />
  </xs:sequence>
  <xs:attribute name="el_ID" type="xs:integer" use="required" />
</xs:complexType>

<xs:complexType name="alkalmazottTipus">
  <xs:sequence>
    <xs:element name="nev" type="xs:string" />
    <xs:element name="szuletesi_datum" type="xs:date" />
    <xs:element name="fizetes" type="xs:integer" />
  </xs:sequence>
  <xs:attribute name="a_ID" type="xs:integer" use="required" />
  <xs:attribute name="em_IDREF" type="xs:integer" use="required" />
</xs:complexType>

<xs:complexType name="vezetiTipus">
  <xs:sequence>
    <xs:element name="kinevezes_idopontja" type="xs:date" />
  </xs:sequence>
  <xs:attribute name="pl_IDREF" type="xs:integer" use="required" />
  <xs:attribute name="vi_IDREF" type="xs:integer" use="required" />
</xs:complexType>

```

```
<xs:complexType name="kaphatoTipus">
  <xs:sequence>
    <xs:element name="miota" type="xs:date" />
  </xs:sequence>
  <xs:attribute name="em_IDREF" type="xs:integer" use="required" />
  <xs:attribute name="el_IDREF" type="xs:integer" use="required" />
</xs:complexType>

<xs:complexType name="cimTipus">
  <xs:sequence>
    <xs:element name="varos" type="xs:string" />
    <xs:element name="utca" type="xs:string" />
    <xs:element name="hazszam" type="xs:integer" />
  </xs:sequence>
</xs:complexType>
</xs:schema>
```


2. feladat

2a) Adatolvasás

Az XML-ben található összes adat kiírása.

```
package hu.domparse.ig9ggv;

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.xml.sax.SAXException;

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

public class DOMReadIG9GGV {

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

        File xmlFile = new File("XMLIG9GGV.xml");

        DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance();
        DocumentBuilder dBuilder = factory.newDocumentBuilder();
        //A DocumentBuilderFactory-ből megkapjuk a DocumentBuildert
        //A DocumentBuilder tartalmazza az API-t a DOM-dokumentum példányok XML-dokumentumból való beszerzéséhez

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

        doc.getDocumentElement().normalize();
        //A dokumentum normalizálása segít a helyes eredmények elérésében.

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

        NodeList nList = doc.getElementsByTagName("pizzalanc");
        //A getElementsByTagName() metódus segítségével megkapjuk a pizzalanc elem NodeListjét

        for(int i = 0; i < nList.getLength(); i++) {
            //A listán for ciklussal megyünk végig

            Node nNode = nList.item(i);
            System.out.println("\nCurrent Element: " + nNode.getNodeName());

            if(nNode.getNodeType() == Node.ELEMENT_NODE) {
                Element elem = (Element) nNode;
                String attribute1 = elem.getAttribute("pl_ID");
                //Az elem attribútumot a getAttribute() segítségével kapjuk meg.

                Node node1 = elem.getElementsByTagName("nev").item(0);
                String element1 = node1.getTextContent();

                Node node2 = elem.getElementsByTagName("alapitas_idopontja").item(0);
                String element2 = node2.getTextContent();

                Node node3 = elem.getElementsByTagName("tozsden").item(0);
                String element3 = node3.getTextContent();

                //Megkapjuk a pizzalanc elem három ajelemének szöveges tartalmát

                System.out.println("Pizzalánc ID: " + attribute1);
                System.out.println("Pizzalánc neve: " + element1);
                System.out.println("Alapítás időpontja: " + element2);
                System.out.println("Tőzsdén jelen van-e: " + element3);
            }
        }

        nList = doc.getElementsByTagName("vezerigazgato");
        //A getElementsByTagName() metódus segítségével megkapjuk a vezerigazgato elem NodeListjét

        for(int i = 0; i < nList.getLength(); i++) {
            //A listán for ciklussal megyünk végig

            Node nNode = nList.item(i);
            System.out.println("\nCurrent Element: " + nNode.getNodeName());

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

                String attribute1 = elem.getAttribute("vi_ID");
                //Az elem attribútumot a getAttribute() segítségével kapjuk meg.
```

```

Node node1 = elem.getElementsByTagName("nev").item(0);
String element1 = node1.getTextContent();

Node node2 = elem.getElementsByTagName("szuletesi_datum").item(0);
String element2 = node2.getTextContent();

Node node3 = elem.getElementsByTagName("nem").item(0);
String element3 = node3.getTextContent();

//Megkapjuk a vezérigazgató elem három alelemének szöveges tartalmát

System.out.println("Vezérigazgató ID: " + attribute1);
System.out.println("Vezérigazgató neve: " + element1);
System.out.println("Születési dátuma: " + element2);
System.out.println("Neme: " + element3);
}
}

nList = doc.getElementsByTagName("etterem");
//A getElementsByTagName() metódus segítségével megkapjuk az etterem elem NodeListjét

for(int i = 0; i < nList.getLength(); i++) {
    //A listán for ciklussal megyünk végig

    Node nNode = nList.item(i);
    System.out.println("\nCurrent Element: " + nNode.getNodeName());

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

        String attribute1 = elem.getAttribute("em_ID");
        String attribute2 = elem.getAttribute("pl_IDREF");
        //Az elem attribútumot a getAttribute() segítségével kapjuk meg.

        Node node1 = elem.getElementsByTagName("varos").item(0);
        String element1 = node1.getTextContent();
        Node node2 = elem.getElementsByTagName("utca").item(0);
        String element2 = node2.getTextContent();
        Node node3 = elem.getElementsByTagName("hazszam").item(0);
        String element3 = node3.getTextContent();

        //Megkapjuk a etterem elem három alelemének szöveges tartalmát

        System.out.println("Étterem ID: " + attribute1);
        System.out.println("Pizzalánc ID: " + attribute2);
        System.out.println("Cím: " + element1 + ", " + element2 + " " + element3);
    }
}

nList = doc.getElementsByTagName("etel");
//A getElementsByTagName() metódus segítségével megkapjuk a etel elem NodeListjét

for(int i = 0; i < nList.getLength(); i++) {
    //A listán for ciklussal megyünk végig

    Node nNode = nList.item(i);
    System.out.println("\nCurrent Element: " + nNode.getNodeName());

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

        String attribute1 = elem.getAttribute("el_ID");
        //Az elem attribútumot a getAttribute() segítségével kapjuk meg.

        Node node1 = elem.getElementsByTagName("nev").item(0);
        String element1 = node1.getTextContent();

        Node node2 = elem.getElementsByTagName("ar").item(0);
        String element2 = node2.getTextContent();

        NodeList alapanyagNodeList = elem.getElementsByTagName("alapanyag");
        String element3 = alapanyagNodeList.item(0).getTextContent();
        for(int j = 1; j < alapanyagNodeList.getLength(); j++) {
            Node alapanyagNode = alapanyagNodeList.item(j);
            element3 = element3 + ", " + alapanyagNode.getTextContent();
        }
        //Végigjárja az alapanyaglistát, és összerakja a tartalmukat őket egy stringbe

        //Megkapjuk a etel elem három alelemének szöveges tartalmát

        System.out.println("Étel ID: " + attribute1);
        System.out.println("Étel név: " + element1);
        System.out.println("Ár: " + element2);
        System.out.println("Alapanyagok: " + element3);
    }
}
}

```

```

nList = doc.getElementsByTagName("alkalmazott");
//A getElementsByTagName() metódus segítségével megkapjuk az alkalmazott elem NodeListjét

for(int i = 0; i < nList.getLength(); i++) {
    //A listán for ciklussal megyünk végig

    Node nNode = nList.item(i);
    System.out.println("\nCurrent Element: " + nNode.getNodeName());

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

        String attribute1 = elem.getAttribute("a_ID");
        String attribute2 = elem.getAttribute("em_IDREF");
        //Az elem attribútumot a getAttribute() segítségével kapjuk meg.

        Node node1 = elem.getElementsByTagName("nev").item(0);
        String element1 = node1.getTextContent();
        Node node2 = elem.getElementsByTagName("szuletesi_datum").item(0);
        String element2 = node2.getTextContent();
        Node node3 = elem.getElementsByTagName("fizetes").item(0);
        String element3 = node3.getTextContent();

        //Megkapjuk a etterem elem három alelemének szöveges tartalmát

        System.out.println("Alkalmazott ID: " + attribute1);
        System.out.println("Étterem ID: " + attribute2);
        System.out.println("Alkalmazott név: " + element1);
        System.out.println("Születési dátum: " + element2);
        System.out.println("Fizetés: " + element3);
    }
}

nList = doc.getElementsByTagName("vezeti");
//A getElementsByTagName() metódus segítségével megkapjuk az vezeti elem NodeListjét

for(int i = 0; i < nList.getLength(); i++) {
    //A listán for ciklussal megyünk végig

    Node nNode = nList.item(i);
    System.out.println("\nCurrent Element: " + nNode.getNodeName());

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

        String attribute1 = elem.getAttribute("pl_IDREF");
        String attribute2 = elem.getAttribute("vi_IDREF");
        //Az elem attribútumot a getAttribute() segítségével kapjuk meg.

        Node node1 = elem.getElementsByTagName("kinevezes_idopontja").item(0);
        String element1 = node1.getTextContent();
        //Megkapjuk a vezeti elem alelemének szöveges tartalmát

        System.out.format("A %s ID-jű vezérigazgató a %s ID-jű pizzaláncot vezeti %s óta", attribute1, attribute2, element1);
    }
}

nList = doc.getElementsByTagName("kaphato");
//A getElementsByTagName() metódus segítségével megkapjuk az kaphato elem NodeListjét

for(int i = 0; i < nList.getLength(); i++) {
    //A listán for ciklussal megyünk végig

    Node nNode = nList.item(i);
    System.out.println("\nCurrent Element: " + nNode.getNodeName());

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

        String attribute1 = elem.getAttribute("em_IDREF");
        String attribute2 = elem.getAttribute("el_IDREF");
        //Az elem attribútumot a getAttribute() segítségével kapjuk meg.

        Node node1 = elem.getElementsByTagName("miota").item(0);
        String element1 = node1.getTextContent();
        //Megkapjuk a kaphato elem alelemének szöveges tartalmát

        System.out.format("A %s ID-jű étteremben kapható a %s ID-jű étel %s óta", attribute1, attribute2, element1);
    }
}
}
}

```

Futtatási eredmény:

Root element: pizzalanc_nyilvantartas

Current Element: pizzalanc
Pizzalánc ID: 01
Pizzalánc neve: McPizza
Alapítás időpontja: 1987-11-21
Tőzsdén jelen van-e: true

Current Element: pizzalanc
Pizzalánc ID: 02
Pizzalánc neve: Pizza King
Alapítás időpontja: 1975-07-23
Tőzsdén jelen van-e: false

Current Element: vezérigazgato
Vezérigazgató ID: 01
Vezérigazgató neve: Török Vilmos
Születési dátuma: 1984-03-13
Neme: Férfi

Current Element: vezérigazgato
Vezérigazgató ID: 02
Vezérigazgató neve: Fekete Ágnes
Születési dátuma: 1974-04-23
Neme: Nő

Current Element: etterem
Étterem ID: 01
Pizzalánc ID: 01
Cím: Budapest, Nagy út 12

Current Element: etterem
Étterem ID: 02
Pizzalánc ID: 01
Cím: Debrecen, Kerek út 5

Current Element: etterem
Étterem ID: 03
Pizzalánc ID: 02
Cím: Budapest, Kis út 7

Current Element: etterem
Étterem ID: 04
Pizzalánc ID: 02
Cím: Budapest, Nagy tér 21

Current Element: etterem
Étterem ID: 05
Pizzalánc ID: 02
Cím: Debrecen, Kis tér 13

Current Element: etel
Étel ID: 01
Étel név: Sonkás pizza
Ár: 1100
Alapanyagok: Sonka, Sajt

Current Element: etel
Étel ID: 02
Étel név: Sonkás-kukoricás pizza
Ár: 1300
Alapanyagok: Sonka, Kukorica, Sajt

Current Element: etel
Étel ID: 03
Étel név: Szalámis pizza
Ár: 1600
Alapanyagok: Szalámi, Sajt

Current Element: alkalmazott
Alkalmazott ID: 01
Étterem ID: 05
Alkalmazott név: Fekete Balázs
Születési dátum: 1988-01-12
Fizetés: 260000

Current Element: alkalmazott
Alkalmazott ID: 02
Étterem ID: 04
Alkalmazott név: Balogh Sára
Születési dátum: 1977-02-22
Fizetés: 300000

Current Element: alkalmazott
Alkalmazott ID: 03
Étterem ID: 05
Alkalmazott név: Csonka Gábor
Születési dátum: 2001-12-12
Fizetés: 460000

Current Element: alkalmazott
Alkalmazott ID: 04
Étterem ID: 02
Alkalmazott név: Farkas Anita
Születési dátum: 1999-03-21
Fizetés: 257000

Current Element: alkalmazott
Alkalmazott ID: 05
Étterem ID: 01
Alkalmazott név: Antal Benjamin
Születési dátum: 1956-11-30
Fizetés: 310000

Current Element: alkalmazott
Alkalmazott ID: 06
Étterem ID: 03
Alkalmazott név: Csonka Ábel
Születési dátum: 1976-07-11
Fizetés: 265000

Current Element: alkalmazott
Alkalmazott ID: 07
Étterem ID: 03
Alkalmazott név: Vászoly Martina
Születési dátum: 1987-12-21
Fizetés: 280000

Current Element: alkalmazott
Alkalmazott ID: 08
Étterem ID: 03
Alkalmazott név: Szabó Soma
Születési dátum: 1994-05-12
Fizetés: 400000

Current Element: vezeti
A 01 ID-jű vezérigazgató a 01 ID-jű pizzaláncot vezeti 2015-01-12 óta
Current Element: vezeti
A 02 ID-jű vezérigazgató a 02 ID-jű pizzaláncot vezeti 2020-10-11 óta
Current Element: kaphato
A 01 ID-jű étteremben kapható a 01 ID-jű étel 1987-11-21 óta
Current Element: kaphato
A 01 ID-jű étteremben kapható a 02 ID-jű étel 1987-11-21 óta
Current Element: kaphato
A 01 ID-jű étteremben kapható a 03 ID-jű étel 1987-11-21 óta
Current Element: kaphato
A 02 ID-jű étteremben kapható a 02 ID-jű étel 1997-12-23 óta
Current Element: kaphato
A 02 ID-jű étteremben kapható a 03 ID-jű étel 1987-12-24 óta
Current Element: kaphato
A 03 ID-jű étteremben kapható a 03 ID-jű étel 2001-12-21 óta
Current Element: kaphato
A 04 ID-jű étteremben kapható a 01 ID-jű étel 1987-06-21 óta
Current Element: kaphato
A 04 ID-jű étteremben kapható a 03 ID-jű étel 1987-12-30 óta
Current Element: kaphato
A 05 ID-jű étteremben kapható a 01 ID-jű étel 2011-12-21 óta
Current Element: kaphato
A 05 ID-jű étteremben kapható a 02 ID-jű étel 2020-12-21 óta

2b) Adatmódosítás

- A 2. vezérigazgató ID-nek megváltoztatása 02-ről 03-ra, majd a vezetési kapcsolat frissítése
- A 03 ID-jű étteremben dolgozó alkalmazottak fizetésének növelése 10000-el
- A 08 ID-jű alkalmazott eltávolítása

```
package hu.dompars.ig9ggv;

import java.io.File;

import javax.xml.parsers.DocumentBuilder;
import javax.xml.parsers.DocumentBuilderFactory;
import javax.xml.transform.Transformer;
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.NamedNodeMap;
import org.w3c.dom.Node;
import org.w3c.dom.NodeList;

public class DOMModifyIG9GGV {

    public static void main(String argv[]) throws Exception{

        File xmlFile = new File("XMLIG9GGV.xml");
        DocumentBuilderFactory docFactory = DocumentBuilderFactory.newInstance();
        DocumentBuilder docBuilder = docFactory.newDocumentBuilder();
        Document doc = docBuilder.parse(xmlFile);

        // A 2. vezérigazgató ID-jnek megváltoztatása 02-ről 03-ra, majd a vezetési kapcsolat frissítése
        Node vezérigazgato = doc.getElementsByTagName("vezérigazgato").item(1);
        Node vezeti = doc.getElementsByTagName("vezeti").item(1);

        NamedNodeMap viAttr = vezérigazgato.getAttributes();
        Node nodeViAttr = viAttr.getNamedItem("vi_ID");
        nodeViAttr.setTextContent("03");

        NamedNodeMap vezetesAttr = vezeti.getAttributes();
        Node nodeVezetesAttr = vezetesAttr.getNamedItem("vi_IDREF");
        nodeVezetesAttr.setTextContent("03");

        //A 03 ID-jű étteremben dolgozó alkalmazottak fizetésének növelése 10000-el
        NodeList list = doc.getElementsByTagName("alkalmazott");

        for (int i = 0; i < list.getLength(); i++) {
            Node node = list.item(i);
            if (node.getNodeType() == Node.ELEMENT_NODE) {

                Element elem = (Element) node;
                String etteremID = elem.getAttribute("em_IDREF");
                if(etteremID.equals("03")) {
                    Node nNode = elem.getElementsByTagName("fizetes").item(0);
                    int ujFizetes = Integer.parseInt(nNode.getTextContent()) + 10000;
                    nNode.setTextContent(String.valueOf(ujFizetes));
                }
            }
        }

        // A 08 ID-jű alkalmazott eltávolítása
        Node nNode = doc.getFirstChild();
        NodeList childNodes = nNode.getChildNodes();

        for(int i = 0; i < childNodes.getLength(); i++) {
            Node node = childNodes.item(i);
            NodeList nodeList = node.getChildNodes();

            for(int j = 0; j < nodeList.getLength(); j++) {
                Node childNode = nodeList.item(j);

                if("alkalmazott".equals(childNode.getNodeName()) && "08".equals(childNode.getAttributes()
                    .getNamedItem("a_ID").getNodeValue()) ) {
                    childNode.getParentNode().removeChild(childNode);
                }
            }
        }

        // A módosított XML kiírása konzolra
        TransformerFactory transformerFactory = TransformerFactory.newInstance();
        Transformer transformer = transformerFactory.newTransformer();
        DOMSource source = new DOMSource(doc);
        System.out.println("-----Modified File-----");
        StreamResult consoleResult = new StreamResult(System.out);
        transformer.transform(source, consoleResult);
    }
}
```

Futtatási eredmény:

```
-----Modified File-----
<?xml version="1.0" encoding="UTF-8" standalone="no"?><pizzalanc_nyilvantartas
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="XMLSchemaIG9GGV.xsd">
  <pizzalancok>
    <pizzalanc pl_ID="01">
      <nev>McPizza</nev>
      <alapitas_idopontja>1987-11-21</alapitas_idopontja>
      <tozsden>true</tozsden>
    </pizzalanc>
    <pizzalanc pl_ID="02">
      <nev>Pizza King</nev>
      <alapitas_idopontja>1975-07-23</alapitas_idopontja>
      <tozsden>false</tozsden>
    </pizzalanc>
  </pizzalancok>

  <vezerigazgato>
    <vezerigazgato vi_ID="01">
      <nev>TÁrÁjk Vilmos</nev>
      <szuletesi_datum>1984-03-13</szuletesi_datum>
      <nem>FÁrfi</nem>
    </vezerigazgato>
    <vezerigazgato vi_ID="03">
      <nev>Fekete Ágnes</nev>
      <szuletesi_datum>1974-04-23</szuletesi_datum>
      <nem>NLÉ</nem>
    </vezerigazgato>
  </vezerigazgato>

  <etteremek>
    <etterem em_ID="01" pl_IDREF="01">
      <cim>
        <varos>Budapest</varos>
        <utca>Nagy Ást</utca>
        <hazszam>12</hazszam>
      </cim>
    </etterem>
    <etterem em_ID="02" pl_IDREF="01">
      <cim>
        <varos>Debrecen</varos>
        <utca>Kerek Ást</utca>
        <hazszam>5</hazszam>
      </cim>
    </etterem>
    <etterem em_ID="03" pl_IDREF="02">
      <cim>
        <varos>Budapest</varos>
        <utca>Kis Ást</utca>
        <hazszam>7</hazszam>
      </cim>
    </etterem>
    <etterem em_ID="04" pl_IDREF="02">
      <cim>
        <varos>Budapest</varos>
        <utca>Nagy tÁr</utca>
        <hazszam>21</hazszam>
      </cim>
    </etterem>
    <etterem em_ID="05" pl_IDREF="02">
      <cim>
        <varos>Debrecen</varos>
        <utca>Kis tÁr</utca>
        <hazszam>13</hazszam>
      </cim>
    </etterem>
  </etteremek>

  <etelek>
    <etel el_ID="01">
      <nev>SonkÁs pizza</nev>
      <ar>1100</ar>
      <alapanyag>Sonka</alapanyag>
      <alapanyag>Sajt</alapanyag>
    </etel>
    <etel el_ID="02">
      <nev>SonkÁs-kukoricÁs pizza</nev>
      <ar>1300</ar>
      <alapanyag>Sonka</alapanyag>
      <alapanyag>Kukorica</alapanyag>
      <alapanyag>Sajt</alapanyag>
    </etel>
    <etel el_ID="03">
      <nev>SzalÁmis pizza</nev>
      <ar>1600</ar>
      <alapanyag>SzalÁmi</alapanyag>
      <alapanyag>Sajt</alapanyag>
    </etel>
  </etelek>
</pizzalanc_nyilvantartas>
```

```

    </etel>
</etelek>

<alkalmazottak>
  <alkalmazott a_ID="01" em_IDREF="05">
    <nev>Fekete Balázs</nev>
    <szuletesi_datum>1988-01-12</szuletesi_datum>
    <fizetes>260000</fizetes>
  </alkalmazott>
  <alkalmazott a_ID="02" em_IDREF="04">
    <nev>Balogh Sára</nev>
    <szuletesi_datum>1977-02-22</szuletesi_datum>
    <fizetes>300000</fizetes>
  </alkalmazott>
  <alkalmazott a_ID="03" em_IDREF="05">
    <nev>Csonka Gábor</nev>
    <szuletesi_datum>2001-12-12</szuletesi_datum>
    <fizetes>460000</fizetes>
  </alkalmazott>
  <alkalmazott a_ID="04" em_IDREF="02">
    <nev>Farkas Anita</nev>
    <szuletesi_datum>1999-03-21</szuletesi_datum>
    <fizetes>257000</fizetes>
  </alkalmazott>
  <alkalmazott a_ID="05" em_IDREF="01">
    <nev>Antal Benjamin</nev>
    <szuletesi_datum>1956-11-30</szuletesi_datum>
    <fizetes>310000</fizetes>
  </alkalmazott>
  <alkalmazott a_ID="06" em_IDREF="03">
    <nev>Csonka Ábel</nev>
    <szuletesi_datum>1976-07-11</szuletesi_datum>
    <fizetes>275000</fizetes>
  </alkalmazott>
  <alkalmazott a_ID="07" em_IDREF="03">
    <nev>Vászlóy Martina</nev>
    <szuletesi_datum>1987-12-21</szuletesi_datum>
    <fizetes>290000</fizetes>
  </alkalmazott>

</alkalmazottak>

<vezeteselek>
  <vezeti pl_IDREF="01" vi_IDREF="01">
    <kinevezes_idopontja>2015-01-12</kinevezes_idopontja>
  </vezeti>
  <vezeti pl_IDREF="02" vi_IDREF="03">
    <kinevezes_idopontja>2020-10-11</kinevezes_idopontja>
  </vezeti>
</vezeteselek>

<kaphatok>
  <kaphato el_IDREF="01" em_IDREF="01">
    <miota>1987-11-21</miota>
  </kaphato>
  <kaphato el_IDREF="02" em_IDREF="01">
    <miota>1987-11-21</miota>
  </kaphato>
  <kaphato el_IDREF="03" em_IDREF="01">
    <miota>1987-11-21</miota>
  </kaphato>
  <kaphato el_IDREF="02" em_IDREF="02">
    <miota>1997-12-23</miota>
  </kaphato>
  <kaphato el_IDREF="03" em_IDREF="02">
    <miota>1987-12-24</miota>
  </kaphato>
  <kaphato el_IDREF="03" em_IDREF="03">
    <miota>2001-12-21</miota>
  </kaphato>
  <kaphato el_IDREF="01" em_IDREF="04">
    <miota>1987-06-21</miota>
  </kaphato>
  <kaphato el_IDREF="03" em_IDREF="04">
    <miota>1987-12-30</miota>
  </kaphato>
  <kaphato el_IDREF="01" em_IDREF="05">
    <miota>2011-12-21</miota>
  </kaphato>
  <kaphato el_IDREF="02" em_IDREF="05">
    <miota>2020-12-21</miota>
  </kaphato>
</kaphatok>
</pizzalanc_nyilvantartas>

```


2c) Adatlekérdezés

- 1. lekérdezés: 300000-nél kevesebbet kereső alkalmazottak kilistázása
- 2. lekérdezés: Budapesti éttermek kilistázása
- 3. lekérdezés: 1200-nál drágább pizzák kilistázása

```
package hu.domparsa.ig9ggv;

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

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 DOMQueryIG9GGV {

    public static void main(String[] args)
        throws ParserConfigurationException, IOException, SAXException, TransformerException {
        File xmlFile = new File("XMLIG9GGV.xml");

        DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance();
        DocumentBuilder dBuilder = factory.newDocumentBuilder();
        // A DocumentBuilderFactory-ből megkapjuk a DocumentBuildert
        // A DocumentBuilder tartalmazza az API-t a DOM-dokumentum példányok XML-dokumentumból való beszerzéséhez
        Document doc = dBuilder.parse(xmlFile);
        // A parse() metódus elemzi az XML fájlt

        doc.getDocumentElement().normalize();
        // A dokumentum normalizálása segít a helyes eredmények elérésében.

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

        // 1. lekérdezés: 300000-nél kevesebbet kereső alkalmazottak kilistázása
        System.out.println("-----");
        System.out.println("300000-nél kevesebbet kereső alkalmazottak:");
        NodeList nodeList = doc.getElementsByTagName("alkalmazott");
        for (int i = 0; i < nodeList.getLength(); i++) {
            Node nNode = nodeList.item(i);
            Element elem = (Element) nNode;
            if (nNode.getNodeType() == Node.ELEMENT_NODE) {

                int fizetesHatar = 300000;
                int fizetes = Integer.parseInt(elem.getElementsByTagName("fizetes").item(0).getTextContent());
                if (fizetesHatar > fizetes) {
                    System.out.println("\nCurrent Element: " + nNode.getNodeName());

                    String attribute1 = elem.getAttribute("a_ID");
                    String attribute2 = elem.getAttribute("em_IDREF");
                    // Az elem attribútumot a getAttribute() segítségével kapjuk meg.
                    Node node1 = elem.getElementsByTagName("nev").item(0);
                    String element1 = node1.getTextContent();
                    Node node2 = elem.getElementsByTagName("szuletesi_datum").item(0);
                    String element2 = node2.getTextContent();
                    Node node3 = elem.getElementsByTagName("fizetes").item(0);
                    String element3 = node3.getTextContent();

                    // Megkapjuk a etterem elem három aelemének szöveges tartalmát

                    System.out.println("Alkalmazott ID: " + attribute1);
                    System.out.println("Étterem ID: " + attribute2);
                    System.out.println("Alkalmazott név: " + element1);
                    System.out.println("Születési dátum: " + element2);
                    System.out.println("Fizetés: " + element3);

                }
            }
        }

        // 2. lekérdezés: Budapesti éttermek kilistázása
        System.out.println("-----");
        System.out.println("Budapesti éttermek:");
        nodeList = doc.getElementsByTagName("etterem");
        for (int i = 0; i < nodeList.getLength(); i++) {
            Node nNode = nodeList.item(i);
            Element elem = (Element) nNode;
            if (nNode.getNodeType() == Node.ELEMENT_NODE) {
```

```

        String attribute1 = elem.getAttribute("em_ID");
        String attribute2 = elem.getAttribute("pl_IDREF");
        // Az elem attribútumot a getAttribute() segítségével kapjuk meg.

        Node node1 = elem.getElementsByTagName("varos").item(0);
        String element1 = node1.getTextContent();
        Node node2 = elem.getElementsByTagName("utca").item(0);
        String element2 = node2.getTextContent();
        Node node3 = elem.getElementsByTagName("hazszam").item(0);
        String element3 = node3.getTextContent();

        // Megkapjuk a etterem elem három alevelének szöveges tartalmát

        System.out.println("Étterem ID: " + attribute1);
        System.out.println("Pizzalánc ID: " + attribute2);
        System.out.println("Cím: " + element1 + ", " + element2 + " " + element3);
    }
}

// 3. lekérdezés: 1200-nál drágább pizzák kilistázása
System.out.println("-----");
System.out.println("1200-nál drágább pizzák:");
nodeList = doc.getElementsByTagName("etel");

for (int i = 0; i < nodeList.getLength(); i++) {
    Node nNode = nodeList.item(i);
    Element elem = (Element) nNode;
    if (nNode.getNodeType() == Node.ELEMENT_NODE) {
        int arHatar = 1200;
        int ar = Integer.parseInt(elem.getElementsByTagName("ar").item(0).getTextContent());
        if (ar > arHatar) {
            System.out.println("\nCurrent Element: " + nNode.getNodeName());

            String attribute1 = elem.getAttribute("el_ID");
            // Az elem attribútumot a getAttribute() segítségével kapjuk meg.

            Node node1 = elem.getElementsByTagName("nev").item(0);
            String element1 = node1.getTextContent();

            Node node2 = elem.getElementsByTagName("ar").item(0);
            String element2 = node2.getTextContent();

            NodeList alapanyagNodeList = elem.getElementsByTagName("alapanyag");
            String element3 = alapanyagNodeList.item(0).getTextContent();
            for (int j = 1; j < alapanyagNodeList.getLength(); j++) {
                Node alapanyagNode = alapanyagNodeList.item(j);
                element3 = element3 + ", " + alapanyagNode.getTextContent();
            }
            // Végigjárja az alapanyaglistát, és összerakja a tartalmukat őket egy stringbe

            // Megkapjuk a etel elem három alevelének szöveges tartalmát

            System.out.println("Étel ID: " + attribute1);
            System.out.println("Étel név: " + element1);
            System.out.println("Ár: " + element2);
            System.out.println("Alapanyagok: " + element3);
        }
    }
}
}
}

```

Futtatási eredmény:

Root element: pizzalanc_nyilvantartas

300000-nél kevesebbet kereső alkalmazottak:

Current Element: alkalmazott
Alkalmazott ID: 01
Étterem ID: 05
Alkalmazott név: Fekete Balázs
Születési dátum: 1988-01-12
Fizetés: 260000

Current Element: alkalmazott
Alkalmazott ID: 04
Étterem ID: 02
Alkalmazott név: Farkas Anita
Születési dátum: 1999-03-21
Fizetés: 257000

Current Element: alkalmazott
Alkalmazott ID: 06
Étterem ID: 03
Alkalmazott név: Csonka Ábel
Születési dátum: 1976-07-11
Fizetés: 265000

Current Element: alkalmazott
Alkalmazott ID: 07
Étterem ID: 03
Alkalmazott név: Vászoly Martina
Születési dátum: 1987-12-21
Fizetés: 280000

Budapesti éttermek:

Current Element: etterem
Étterem ID: 01
Pizzalánc ID: 01
Cím: Budapest, Nagy út 12

Current Element: etterem
Étterem ID: 03
Pizzalánc ID: 02
Cím: Budapest, Kis út 7

Current Element: etterem
Étterem ID: 04
Pizzalánc ID: 02
Cím: Budapest, Nagy tér 21

1200-nál drágább pizzák:

Current Element: etel
Étel ID: 02
Étel név: Sonkás-kukoricás pizza
Ár: 1300
Alapanyagok: Sonka, Kukorica, Sajt

Current Element: etel
Étel ID: 03
Étel név: Szalámis pizza
Ár: 1600
Alapanyagok: Szalámi, Sajt