

JEGYZŐKÖNYV

Adatkezelés XML környezetben

Féléves feladat

Készítette: Váradi Marcell Balázs
Neptunkód: IHDUW3

1.Feladat leírása:

A választott féléves feladatom egy online gitár webshop adatbázis amelyben a rendelő rendeléseit lehet nyomonkövetni.

Egy teljesen átlagos webshop működését próbálom bemutatni. Ebben megtalálható az adott termék ami a gitár. A gitárhoz tartozik egy gyártó ami külön entitás lesz az adatbázisban.

Az adatbázisban lévő egyedek és tulajdonságaik:

Webshop egyed: Ebből indul ki minden egyed.

Webshop attribútumai:

ID – Ez az elsődleges kulcs

Név – Ez a webshop neve

Cím – Webshop telephely címe. Ez egy összetett elem.

Tulajdonos – A Webshop Tulaja

Alapítva – A gitárshop alapítási éve

Gitár egyed: Ez a webshop terméke.

Gitár attribútumai:

ID – Ez a gitár elsődleges kulcsa

Szín – Gitár színe

Kivitel – A gitár kivitele(akusztikus,elektromos, stb...)

Gyártás – A gitár gyártási éve

Típus – A gitár neve

Ár – A gitár ára

Gyártó egyed: A gitár entitáshoz van hozzákapcsolva.

Gyártó attribútumai:

GyártóID – Ez a Gyártó Elsődleges kulcsa

Ország – A gyártó melyik országban gyártotta a gitárt

Név – Gyártó neve

Tulajdonos – A Gyártó tulajdonosa

Rendelő egyed: Ez a ügyfél akinek nyomonkövetjük a rendeléseit.

ID – Ez a Rendelő elsődleges kulcsa

Név – Rendelő neve

Email – Rendelő emailje

Cím – Rendelő címe. Ez egy összetett elem

Rendelések kapcsolótábla: Ez a több-több kapcsolat miatt jött létre. A rendelés egyedet és a rendelő egyedet kapcsolja össze.

Elemei:

Darab – A rendelő rendeléseinek feljegyzésére

FeladásDátuma – A rendelő mikor adta fel a rendelését/rendeléseit

Rendelés egyed: Itt tudjuk megnézni a kiszállítót és a termék aktuális státuszát.

ID – Ez a rendelés elsődleges kulcsa

Státusz – A rendelés hol tart

Kiszállító – A kiszállítást végző cég

Megjegyzés – A csomagról bármilyen megjegyzés

Fizetés egyed: Ez a rendelő Fizetési módját írja le.

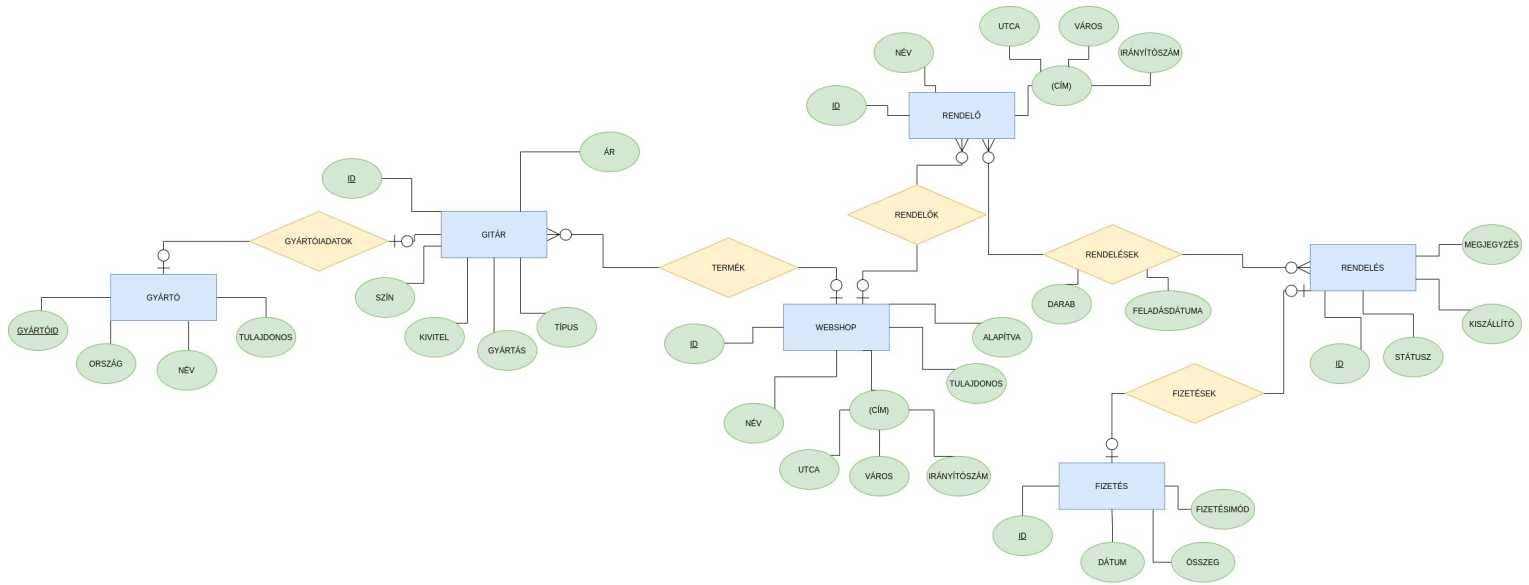
ID – Ez a fizetés elsődleges kulcsa

Dátum – Mikor történt a fizetés

Összeg – Mennyibe került a termék

FizetésiMód – Mivel fizetett a rendelő(készpénz, bankkártya)

Az adatbázis ER modell:



XDM modell:

Xml:

```
1 <?xml version="1.0" encoding="UTF-8"?><?xml-model href="XMLSchemaIHDUW3.xsd" ?>
2 <gitarWebShops>
3
4   <webshops>
5     <webshop id="web1">
6       <nev>ThomannMusic</nev>
7       <cim>
8         <utca>Hans Thomann utca</utca>
9         <varos>Treppendorf</varos>
10        <iranyitoszam>96138</iranyitoszam>
11      </cim>
12
13      <tulajdonos>Hans Thomann</tulajdonos>
14      <alapitva>1954</alapitva>
15      <idgitar refid="gitar1">gitar1</idgitar>
16      <idrendelo refid="rendelo1">rendelo1</idrendelo>
17    </webshop>
18
19    <webshop id="web2">
20      <nev>Tajti Music Hangszer</nev>
21      <cim>
22        <utca>Rákóczi út</utca>
23        <varos>Pécs</varos>
24        <iranyitoszam>7622</iranyitoszam>
25      </cim>
26
27      <tulajdonos>Monori Tibor</tulajdonos>
28      <alapitva>1990</alapitva>
29      <idgitar refid="gitar1">gitar1</idgitar>
30      <idrendelo refid="rendelo2">rendelo1</idrendelo>
31    </webshop>
32  </webshops>
33
34  <gitarok>
35    <gitar id="gitar1">
36      <tipus>RG370dx</tipus>
37      <kivitel>elektromos</kivitel>
38      <szin>fekete</szin>
39      <ar>140000</ar>
40      <gyartas>2004</gyartas>
41      <idgyarto refid="gy1">gy1</idgyarto>
42    </gitar>
43
44    <gitar id="gitar2">
45      <tipus>AZ3210</tipus>
46      <kivitel>elektromos</kivitel>
47      <szin>fekete</szin>
48      <ar>240000</ar>
49      <gyartas>2019</gyartas>
50      <idgyarto refid="gy1">gy1</idgyarto>
51    </gitar>
52  </gitarok>
53  <gyartok>
54    <gyarto id="gy1">
55      <orszag>Korea</orszag>
56      <nev>Ibanez</nev>
57      <tulajdonos>Hoshino Gakki</tulajdonos>
58    </gyarto>
59  </gyartok>
```

```

60 <rendelok>
61   <rendelo id="rendelo1">
62     <nev>Kiss Bela</nev>
63     <email>bel@gmail.com</email>
64     <cim>
65       <utca>Kossuth utca</utca>
66       <iranyitoszam>3500</iranyitoszam>
67       <varos>Miskolc</varos>
68     </cim>
69   </rendelo>
70
71   <rendelo id="rendelo2">
72     <nev>Nagy Tamas</nev>
73     <email>tamas@gmail.com</email>
74     <cim>
75       <utca>Kovacs utca</utca>
76       <iranyitoszam>3500</iranyitoszam>
77       <varos>Miskolc</varos>
78     </cim>
79   </rendelo>
80 </rendelok>
81 <rendeless>
82   <rendeles id="rendeles1">
83     <statusz>aktív</statusz>
84     <kiszallito>Posta</kiszallito>
85     <megjegyzes>Nincs</megjegyzes>
86     <idfiz refid="fiz1">fiz1</idfiz>
87   </rendeles>
88
89   <rendeles id="rendeles2">
90     <statusz>aktív</statusz>
91     <kiszallito>Posta</kiszallito>
92     <megjegyzes>Nincs</megjegyzes>
93     <idfiz refid="fiz2">fiz1</idfiz>
94   </rendeles>
95 </rendeless>
96
97 <kapcs_rendelesek>
98   <rendelesek id="1">
99     <idrendelo refid="rendelo1">rendelo1</idrendelo>
100     <idrendeles refid="rendeles1">rendeles1</idrendeles>
101     <darab>1</darab>
102     <feladasdatuma>2021</feladasdatuma>
103   </rendelesek>
104
105   <rendelesek id="2">
106     <idrendelo refid="rendelo2">rendelo2</idrendelo>
107     <idrendeles refid="rendeles2">rendeles2</idrendeles>
108     <darab>1</darab>
109     <feladasdatuma>2021</feladasdatuma>
110   </rendelesek>
111 </kapcs_rendelesek>
112

```

```
112
113  <fizetesek>
114    <fizetes id="fiz1">
115      <datum>2021</datum>
116      <ország>Magyarország</ország>
117      <fizetesimod>Bankkártya</fizetesimod>
118    </fizetes>
119
120    <fizetes id="fiz2">
121      <datum>2021</datum>
122      <ország>Magyarország</ország>
123      <fizetesimod>Bankkártya</fizetesimod>
124    </fizetes>
125  </fizetesek>
126
127 </gitarWebShops>
128
129
```


XSD:

```
1 <?xml version="1.0" encoding="UTF-8"?>
2
3 <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
4   elementFormDefault="qualified" attributeFormDefault="qualified">
5
6   <xs:element name="gitarWebShops">
7
8     <xs:complexType>
9       <xs:sequence>
10        <xs:element name="webshops" type="webshopstype" />
11        <xs:element name="gitarok" type="gitaroktype" />
12        <xs:element name="gyartok" type="gyartoktype" />
13        <xs:element name="rendelo" type="rendeloktype" />
14        <xs:element name="rendelessek" type="rendelesektype" />
15        <xs:element name="kapcs_rendelessek" type="kapcs_rendelesektype" />
16        <xs:element name="fizetesek" type="fizetesektype" />
17      </xs:sequence>
18    </xs:complexType>
19
20  </xs:element>
21
22  <xs:complexType name="webshopstype">
23    <xs:sequence>
24      <xs:element name="webshop" maxOccurs="unbounded">
25        <xs:complexType>
26          <xs:sequence>
27            <xs:element name="nev" type="xs:string" />
28            <xs:element name="cim" type="cimtype" />
29            <xs:element name="tulajdonos" type="xs:string" />
30            <xs:element name="alapitva" type="xs:string" />
31            <xs:element name="idgitar">
32              <xs:complexType>
33                <xs:attribute name="refID" type="xs:IDREF" />
34              </xs:complexType>
35            </xs:element>
36            <xs:element name="idrendelo">
37              <xs:complexType>
38                <xs:attribute name="refID" type="xs:IDREF" />
39              </xs:complexType>
40            </xs:element>
41          </xs:sequence>
42          <xs:attribute name="id" type="xs:ID" use="required" />
43        </xs:complexType>
44      </xs:element>
45    </xs:sequence>
46  </xs:complexType>
47
48  <xs:complexType name="cimtype">
49    <xs:sequence>
50      <xs:element name="utca" type="xs:string" />
51      <xs:element name="varos" type="xs:string" />
52      <xs:element name="iranyitoszam" type="xs:string" />
53    </xs:sequence>
54  </xs:complexType>
55
```

```

57<xs:complexType name="gitaroktype">
58  <xs:sequence>
59    <xs:element name="gitarok" maxOccurs="unbounded">
60      <xs:complexType>
61        <xs:sequence>
62          <xs:element name="tipus" type="xs:string" />
63          <xs:element name="kivitel" type="kiviteltype" />
64          <xs:element name="szin" type="xs:string" />
65          <xs:element name="ar" type="xs:string" />
66          <xs:element name="gyartas" type="xs:string" />
67          <xs:element name="idgyarto">
68            <xs:complexType>
69              <xs:attribute name="refID" type="xs:IDREF" />
70            </xs:complexType>
71          </xs:element>
72        </xs:sequence>
73        <xs:attribute name="id" type="xs:ID" use="required" />
74      </xs:complexType>
75    </xs:element>
76  </xs:sequence>
77</xs:complexType>
78
79
80<xs:complexType name="kiviteltype">
81  <xs:sequence>
82    <xs:element name="elektromos" type="xs:string" />
83    <xs:element name="akusztikus" type="xs:string" />
84    <xs:element name="klasszikus" type="xs:string" />
85  </xs:sequence>
86</xs:complexType>
87
88<xs:complexType name="gyartoktype">
89  <xs:sequence>
90    <xs:element name="gyartok" maxOccurs="unbounded">
91      <xs:complexType>
92        <xs:sequence>
93          <xs:element name="ország" type="xs:string" />
94          <xs:element name="nev" type="xs:string" />
95          <xs:element name="tulajdonos" type="xs:string" />
96        </xs:sequence>
97        <xs:attribute name="id" type="xs:ID" use="required" />
98      </xs:complexType>
99    </xs:element>
100  </xs:sequence>
101</xs:complexType>
102
103<xs:complexType name="rendeloktype">
104  <xs:sequence>
105    <xs:element name="rendelo" maxOccurs="unbounded">
106      <xs:complexType>
107        <xs:sequence>
108          <xs:element name="nev" type="xs:string" />
109          <xs:element name="email" type="xs:string" />
110          <xs:element name="cim" type="cimtype" />
111        </xs:sequence>
112        <xs:attribute name="id" type="xs:ID" use="required" />
113      </xs:complexType>
114    </xs:element>
115  </xs:sequence>
116</xs:complexType>

```

```

118<xs:complexType name="kapcs_rendelesetype">
119  <xs:sequence>
120    <xs:element name="kapcs_rendeles" maxOccurs="unbounded">
121      <xs:complexType>
122        <xs:sequence>
123          <xs:element name="idrendelo">
124            <xs:complexType>
125              <xs:attribute name="refID" type="xs:IDREF" />
126            </xs:complexType>
127          </xs:element>
128          <xs:element name="idrendeles">
129            <xs:complexType>
130              <xs:attribute name="refID" type="xs:IDREF" />
131            </xs:complexType>
132          </xs:element>
133          <xs:element name="darab" type="xs:integer" />
134          <xs:element name="feladasdatuma" type="xs:string" />
135        </xs:sequence>
136        <xs:attribute name="id" type="xs:ID" use="required" />
137      </xs:complexType>
138    </xs:element>
139  </xs:sequence>
140</xs:complexType>
141</xs:element>
142</xs:sequence>
143</xs:complexType>
144</xs:schema>
145<xs:complexType name="rendelesetype">
146  <xs:sequence>
147    <xs:element name="rendeles" maxOccurs="unbounded">
148      <xs:complexType>
149        <xs:sequence>
150          <xs:element name="statusz" type="xs:string" />
151          <xs:element name="kiszallito" type="cimtype" />
152          <xs:element name="megjegyzes" type="xs:string" />
153          <xs:element name="idfiz">
154            <xs:complexType>
155              <xs:attribute name="refID" type="xs:IDREF" />
156            </xs:complexType>
157          </xs:element>
158        </xs:sequence>
159        <xs:attribute name="id" type="xs:ID" use="required" />
160      </xs:complexType>
161    </xs:element>
162  </xs:sequence>
163</xs:complexType>
164</xs:element>
165</xs:schema>
166<xs:complexType name="fizetesektype">
167  <xs:sequence>
168    <xs:element name="fizetes" maxOccurs="unbounded">
169      <xs:complexType>
170        <xs:sequence>
171          <xs:element name="datum" type="xs:string" />
172          <xs:element name="osszeg" type="cimtype" />
173          <xs:element name="fizetesimod" type="xs:string" />
174        </xs:sequence>
175        <xs:attribute name="id" type="xs:ID" use="required" />
176      </xs:complexType>
177    </xs:element>
178  </xs:sequence>
179</xs:complexType>
180</xs:element>
181</xs:schema>
182

```


2. DOMReader

```
1 package hu.domp.parse.IHDUW3;
2
3 import java.io.File;
4 import javax.xml.parsers.DocumentBuilderFactory;
5 import javax.xml.parsers.DocumentBuilder;
6 import org.w3c.dom.Document;
7 import org.w3c.dom.NodeList;
8 import org.w3c.dom.Node;
9 import org.w3c.dom.Element;
10
11 public class DomReadIHDUW3 {
12
13     public static void main(String[] args) {
14         try {
15             //Fájl elérése
16             File inputFile = new File("/home/marci/eclipse-workspace/DOMParseIHDUW3/src/XmlIHDUW3.xml");
17             //Dom-dokumentum létrehozása az XML dokumentum eléréséhez
18             DocumentBuilderFactory dbFactory = DocumentBuilderFactory.newInstance();
19             DocumentBuilder dBuilder = dbFactory.newDocumentBuilder();
20             Document doc = dBuilder.parse(inputFile);
21             doc.getDocumentElement().normalize();
22             System.out.println("Gyoker elem : " + doc.getDocumentElement().getNodeName());
23             System.out.println("-----");
24
25             //Elemek kiiratasa metodus hivasokkal
26             NodeList nList = doc.getElementsByTagName("webshop");
27             webshopRead(nList);
28             System.out.println("-----");
29
30             NodeList nList2 = doc.getElementsByTagName("gitar");
31             guitarsRead(nList2);
32             System.out.println("-----");
33
34             NodeList nList3 = doc.getElementsByTagName("gyarto");
35             gyartoRead(nList3);
36             System.out.println("-----");
37
38             NodeList nList4 = doc.getElementsByTagName("rendelo");
39             rendeloRead(nList4);
40             System.out.println("-----");
41
42             NodeList nList5 = doc.getElementsByTagName("rendeles");
43             rendelessRead(nList5);
44             System.out.println("-----");
45
46             NodeList nList6 = doc.getElementsByTagName("rendelessek");
47             rendelessekRead(nList6);
48             System.out.println("-----");
49
50             NodeList nList7 = doc.getElementsByTagName("fizetes");
51             fizetesekRead(nList7);
52
53             System.out.println("-----");
54
55         } catch (Exception e) {
56             e.printStackTrace();
57         }
58     }
59
60     //Külön metódus a gyerek elemek kiirására
61     public static void getElement(Element eElement, String elementOut, String elementName) {
62
63         System.out.println(elementOut + " : " + eElement.getElementsByTagName(elementName).item(0).getTextContent());
64     }
65
66     //Innentől minden Elem kiirására külön metódus
67     public static void webshopRead(NodeList nList) {
68         for (int temp = 0; temp < nList.getLength(); temp++) {
69             Node nNode = nList.item(temp);
70             System.out.println("\n Gyoker elem : " + nNode.getNodeName());
71
72             if (nNode.getNodeType() == Node.ELEMENT_NODE) {
73                 Element eElement = (Element) nNode;
74                 System.out.println("ID : " + eElement.getAttribute("id"));
75
76                 getElement(eElement, "webshopnev", "nev");
77                 getElement(eElement, "utca", "utca");
78                 getElement(eElement, "varos", "varos");
79                 getElement(eElement, "iranyitoszam", "iranyitoszam");
80                 getElement(eElement, "tulajdonos", "tulajdonos");
81                 getElement(eElement, "alapitva", "alapitva");
82                 getElement(eElement, "idgitar", "idgitar");
83             }
84         }
85     }
86
87     public static void guitarsRead(NodeList nList) {
88         for (int temp = 0; temp < nList.getLength(); temp++) {
89             Node nNode = nList.item(temp);
90             System.out.println("\n Gyoker elem : " + nNode.getNodeName());
91
92             if (nNode.getNodeType() == Node.ELEMENT_NODE) {
93                 Element eElement = (Element) nNode;
94                 System.out.println("ID : " + eElement.getAttribute("id"));
95
96                 getElement(eElement, "gitarnev", "nev");
97                 getElement(eElement, "gyarto", "gyarto");
98                 getElement(eElement, "idgyarto", "idgyarto");
99                 getElement(eElement, "idgitar", "idgitar");
100             }
101         }
102     }
103
104     public static void gyartoRead(NodeList nList) {
105         for (int temp = 0; temp < nList.getLength(); temp++) {
106             Node nNode = nList.item(temp);
107             System.out.println("\n Gyoker elem : " + nNode.getNodeName());
108
109             if (nNode.getNodeType() == Node.ELEMENT_NODE) {
110                 Element eElement = (Element) nNode;
111                 System.out.println("ID : " + eElement.getAttribute("id"));
112
113                 getElement(eElement, "gyarto", "gyarto");
114                 getElement(eElement, "idgyarto", "idgyarto");
115             }
116         }
117     }
118
119     public static void rendeloRead(NodeList nList) {
120         for (int temp = 0; temp < nList.getLength(); temp++) {
121             Node nNode = nList.item(temp);
122             System.out.println("\n Gyoker elem : " + nNode.getNodeName());
123
124             if (nNode.getNodeType() == Node.ELEMENT_NODE) {
125                 Element eElement = (Element) nNode;
126                 System.out.println("ID : " + eElement.getAttribute("id"));
127
128                 getElement(eElement, "rendelo", "rendelo");
129                 getElement(eElement, "idrendelo", "idrendelo");
130             }
131         }
132     }
133
134     public static void rendelessRead(NodeList nList) {
135         for (int temp = 0; temp < nList.getLength(); temp++) {
136             Node nNode = nList.item(temp);
137             System.out.println("\n Gyoker elem : " + nNode.getNodeName());
138
139             if (nNode.getNodeType() == Node.ELEMENT_NODE) {
140                 Element eElement = (Element) nNode;
141                 System.out.println("ID : " + eElement.getAttribute("id"));
142
143                 getElement(eElement, "rendeless", "rendeless");
144                 getElement(eElement, "idrendeless", "idrendeless");
145             }
146         }
147     }
148
149     public static void rendelessekRead(NodeList nList) {
150         for (int temp = 0; temp < nList.getLength(); temp++) {
151             Node nNode = nList.item(temp);
152             System.out.println("\n Gyoker elem : " + nNode.getNodeName());
153
154             if (nNode.getNodeType() == Node.ELEMENT_NODE) {
155                 Element eElement = (Element) nNode;
156                 System.out.println("ID : " + eElement.getAttribute("id"));
157
158                 getElement(eElement, "rendelessek", "rendelessek");
159                 getElement(eElement, "idrendelessek", "idrendelessek");
160             }
161         }
162     }
163
164     public static void fizetesekRead(NodeList nList) {
165         for (int temp = 0; temp < nList.getLength(); temp++) {
166             Node nNode = nList.item(temp);
167             System.out.println("\n Gyoker elem : " + nNode.getNodeName());
168
169             if (nNode.getNodeType() == Node.ELEMENT_NODE) {
170                 Element eElement = (Element) nNode;
171                 System.out.println("ID : " + eElement.getAttribute("id"));
172
173                 getElement(eElement, "fizetesek", "fizetesek");
174                 getElement(eElement, "idfizetesek", "idfizetesek");
175             }
176         }
177     }
178 }
```

```

82         getElement(eElement, "idrendelo", "idrendelo");
83
84     }
85 }
86
87 }
88
89 public static void guitarsRead(NodeList nList) {
90     for (int temp = 0; temp < nList.getLength(); temp++) {
91         Node nNode = nList.item(temp);
92         System.out.println("\n Gyoker elem :" + nNode.getNodeName());
93
94         if (nNode.getNodeType() == Node.ELEMENT_NODE) {
95             Element eElement = (Element) nNode;
96             System.out.println("ID : " + eElement.getAttribute("id"));
97
98             getElement(eElement, "tipus", "tipus");
99             getElement(eElement, "kivitel", "kivitel");
100             getElement(eElement, "szin", "szin");
101             getElement(eElement, "ar", "ar");
102             getElement(eElement, "gyartas", "gyartas");
103             getElement(eElement, "idgyarto", "idgyarto");
104
105         }
106     }
107 }
108
109 public static void gyartoRead(NodeList nList) {
110     for (int temp = 0; temp < nList.getLength(); temp++) {
111         Node nNode = nList.item(temp);
112         System.out.println("\n Gyoker elem :" + nNode.getNodeName());
113
114         if (nNode.getNodeType() == Node.ELEMENT_NODE) {
115             Element eElement = (Element) nNode;
116             System.out.println("ID : " + eElement.getAttribute("id"));
117
118             getElement(eElement, "orszag", "orszag");
119             getElement(eElement, "nev", "nev");
120             getElement(eElement, "tulajdonos", "tulajdonos");
121
122         }
123     }
124 }
125
126 public static void rendeloRead(NodeList nList) {
127     for (int temp = 0; temp < nList.getLength(); temp++) {
128         Node nNode = nList.item(temp);
129         System.out.println("\n Gyoker elem :" + nNode.getNodeName());
130
131         if (nNode.getNodeType() == Node.ELEMENT_NODE) {
132             Element eElement = (Element) nNode;
133             System.out.println("ID : " + eElement.getAttribute("id"));
134
135             getElement(eElement, "nev", "nev");
136             getElement(eElement, "email", "email");
137             getElement(eElement, "utca", "utca");
138             getElement(eElement, "varos", "varos");
139             getElement(eElement, "iranyitoszam", "iranyitoszam");
140
141         }
142     }
143 }
144
145 public static void rendelessRead(NodeList nList) {
146     for (int temp = 0; temp < nList.getLength(); temp++) {
147         Node nNode = nList.item(temp);
148         System.out.println("\n Gyoker elem :" + nNode.getNodeName());
149
150         if (nNode.getNodeType() == Node.ELEMENT_NODE) {
151             Element eElement = (Element) nNode;
152             System.out.println("ID : " + eElement.getAttribute("id"));
153
154             getElement(eElement, "statusz", "statusz");
155             getElement(eElement, "kiszallito", "kiszallito");
156             getElement(eElement, "megjegyzes", "megjegyzes");
157             getElement(eElement, "idfiz", "idfiz");

```

```

157         getElement(eElement, "idrendelo", "idrendelo");
158
159     }
160 }
161 }
162 }
163
164 public static void rendelesekRead(NodeList nList) {
165     for (int temp = 0; temp < nList.getLength(); temp++) {
166         Node nNode = nList.item(temp);
167         System.out.println("\n Gyoker elem : " + nNode.getNodeName());
168
169         if (nNode.getNodeType() == Node.ELEMENT_NODE) {
170             Element eElement = (Element) nNode;
171             System.out.println("ID : " + eElement.getAttribute("id"));
172             getElement(eElement, "idrendelo", "idrendelo");
173             getElement(eElement, "idrendeles", "idrendeles");
174             getElement(eElement, "darab", "darab");
175             getElement(eElement, "feladasdatuma", "feladasdatuma");
176
177         }
178     }
179 }
180
181 public static void fizetesekRead(NodeList nList) {
182     for (int temp = 0; temp < nList.getLength(); temp++) {
183         Node nNode = nList.item(temp);
184         System.out.println("\n Gyoker elem: " + nNode.getNodeName());
185
186         if (nNode.getNodeType() == Node.ELEMENT_NODE) {
187             Element eElement = (Element) nNode;
188             System.out.println("ID : " + eElement.getAttribute("id"));
189             getElement(eElement, "datum", "datum");
190             getElement(eElement, "orszag", "orszag");
191             getElement(eElement, "fizetesimod", "fizetesimod");
192
193         }
194     }
195 }
196 }
197

```

Output:

```

<terminated> DomReadH3DUW3 [Java Application] /usr/lib/jvm/java-11-open
Gyoker elem :gitarWebShops
-----

```

```

Gyoker elem :webshop
ID : web1
webshopnev : ThomannMusic
utca : Hans Thomann utca
varos : Treppendorf
iranyitoszam : 96138
tulajdonos : Hans Thomann
alapitva : 1954
idgitar : gitar1
idrendelo : rendelo1
-----

```

```

Gyoker elem :webshop
ID : web2
webshopnev : Tajti Music Hangszer
utca : Rákóczi út
varos : Pécs
iranyitoszam : 7622
tulajdonos : Monori Tibor
alapitva : 1990
idgitar : gitar1
idrendelo : rendelo1
-----

```

```

Gyoker elem :gitar
ID : gitar1
tipus : RG370dx
kivitel : elektromos
szin : fekete
ar : 140000
gyartas : 2004
idgyarto : gyl
-----

```

```

Gyoker elem :gitar
ID : gitar2
tipus : AZ3210
kivitel : elektromos
szin : fekete
ar : 240000
gyartas : 2019
idgyarto : gyl
-----

```

```

Gyoker elem :gyarto
ID : gyl
orszag : Korea
nev : Ibanez
tulajdonos : Hoshino Gakki
-----

```

```

Gyoker elem :rendelo
ID : rendelo1
nev : Kiss Bela
email : bela@gmail.com
utca : Kossuth utca
varos : Miskolc
iranyitoszam : 3500
-----

```

```

Gyoker elem :rendelo
ID : rendelo2
nev : Nagy Tamas
email : tamas@gmail.com
utca : Kovacs utca
varos : Miskolc
iranyitoszam : 3500
-----

```

```

Gyoker elem :rendeles
ID : rendeles1
statusz : aktiv
kiszallito : Posta
megjegyzes : Nincs
idfiz : fizl
-----

```

```

Gyoker elem :rendeles
ID : rendeles2
statusz : aktiv
kiszallito : Posta
megjegyzes : Nincs
idfiz : fizl
-----

```

```

Gyoker elem :rendelesek
ID : 1
idrendelo : rendelo1
idrendeles : rendeles1
darab : 1
feladasdatuma : 2021
-----

```

```

Gyoker elem :rendelesek
ID : 2
idrendelo : rendelo2
idrendeles : rendeles2
darab : 1
feladasdatuma : 2021
-----

```

```

Gyoker elem:fizetes
ID : fizl
datum : 2021
orszag : Magyarország
fizetesimod : Bankkártya
-----

```

```

Gyoker elem:fizetes
ID : fiz2
datum : 2021
orszag : Magyarország
fizetesimod : Bankkártya
-----

```


DOMModify

```
1 package hu.domparse.IHDUW3;
2
3 import java.io.FileInputStream;
4 import java.io.IOException;
5 import java.io.InputStream;
6 import java.util.Scanner;
7
8 import javax.xml.parsers.DocumentBuilder;
9 import javax.xml.parsers.DocumentBuilderFactory;
10 import javax.xml.parsers.ParserConfigurationException;
11 import javax.xml.transform.Transformer;
12 import javax.xml.transform.TransformerException;
13 import javax.xml.transform.TransformerFactory;
14 import javax.xml.transform.dom.DOMSource;
15 import javax.xml.transform.stream.StreamResult;
16
17 import org.w3c.dom.Document;
18 import org.w3c.dom.Node;
19 import org.w3c.dom.NodeList;
20 import org.xml.sax.SAXException;
21
22 public class DomModifyIHDUW3 {
23
24
25     public static void main(String[] args) throws TransformerException {
26         // Dom-dokumentum létrehozása az XML dokumentum eléréséhez
27         DocumentBuilderFactory dbf = DocumentBuilderFactory.newInstance();
28
29         try (InputStream is = new FileInputStream("/home/marci/eclipse-workspace/DOMParseIHDUW3/src/XmlIHDUW3.xml")) {
30
31             DocumentBuilder db = dbf.newDocumentBuilder();
32
33             Document doc = db.parse(is);
34             // Bekerjük a módosítani kívánt elemet
35             Scanner rootElem = new Scanner(System.in);
36             System.out.println("Melyik elemet módosítsuk?");
37             String element = rootElem.nextLine();
38
39             Scanner elemID = new Scanner(System.in);
40             System.out.println("Add meg az ID-t?");
41             String childElementID = elemID.nextLine();
42
43             NodeList listOfWebshop = doc.getElementsByTagName(element);
44
45             Scanner elem = new Scanner(System.in);
46             System.out.println("Melyik gyerek elemet módosítsuk?");
47             System.out.println("Melyik gyerek elemet módosítsuk?");
48             String childElement = elem.nextLine();
49
50             System.out.println("Mi legyen az új elem értéke?");
51             Scanner ujElem = new Scanner(System.in);
52             String childElementElementNew = ujElem.nextLine();
53
54             for (int i = 0; i < listOfWebshop.getLength(); i++) {
55                 Node staff = listOfWebshop.item(i);
56                 if (staff.getNodeType() == Node.ELEMENT_NODE) {
57                     String id = staff.getAttribute().getNamedItem("id").getTextContent();
58                     if (childElementID.equals(id.trim())) {
59
60                         NodeList childNodes = staff.getChildNodes();
61
62                         for (int j = 0; j < childNodes.getLength(); j++) {
63                             Node item = childNodes.item(j);
64                             if (item.getNodeType() == Node.ELEMENT_NODE) {
65
66                                 if (childElement.equalsIgnoreCase(item.getNodeName())) {
67                                     item.setTextContent(childElementElementNew);
68                                 }
69                             }
70                         }
71                     }
72                 }
73             }
74
75         }
76
77     }
78
79 }
80
81 // Létrehozunk egy TransformerFactory és egy DOMSource objektumot ezzel leegyszerűsítve a kiíratást
82 TransformerFactory transformerFactory = TransformerFactory.newInstance();
83 Transformer transformer = transformerFactory.newTransformer();
84 DOMSource source = new DOMSource(doc);
85 System.out.println("-----Módosítás után-----");
86 StreamResult consoleResult = new StreamResult(System.out);
87 transformer.transform(source, consoleResult);
88
89 } catch (ParserConfigurationException | SAXException | IOException e) {
90     e.printStackTrace();
91 }
```

```

81 // Előrehozunk egy TransformerFactory és egy DOMSource objektumot ezzel leegyszerűsítve a kódat
82 TransformerFactory transformerFactory = TransformerFactory.newInstance();
83 Transformer transformer = transformerFactory.newTransformer();
84 DOMSource source = new DOMSource(doc);
85 System.out.println("-----Modositas utan-----");
86 StreamResult consoleResult = new StreamResult(System.out);
87 transformer.transform(source, consoleResult);
88
89 } catch (ParserConfigurationException | SAXException | IOException e) {
90     e.printStackTrace();
91 }
92
93 }
94
95
96 }
97

```

Output:

```

C:\Program Files\Java\jdk-11\bin>java -Duser.dir=C:\Program Files\Java\jdk-11\bin\java -Duser.dir=C:\Program Files\Java\jdk-11\bin\java -Duser.dir=C:\Program Files\Java\jdk-11\bin\java
Melyik elemet modosituk
webshop
Add meg az ID-t?
web1
Melyik gyerek elemet modosituk?
nev
Mi legyen az uj elem erteke?
Ujertek
-----Modositas utan-----
<?xml version="1.0" encoding="UTF-8" standalone="no"?><?xml-model href="XMLSchemaIHDUW3.xsd" ?><gitarWebShops>

    <webshops>
        <webshop id="web1">
            <nev>Ujertek</nev>
            <cim>
                <utca>Hans Thomann utca</utca>
                <varos>Treppendorf</varos>
                <iranyitoszam>96138</iranyitoszam>
            </cim>

            <tulajdonos>Hans Thomann</tulajdonos>

```


DomQuery:

```
1 package hu.domparse.IHDUW3;
2
3 import java.io.File;
4 import javax.xml.parsers.DocumentBuilderFactory;
5 import javax.xml.parsers.DocumentBuilder;
6 import org.w3c.dom.Document;
7 import org.w3c.dom.NodeList;
8 import java.util.Scanner;
9
10 //Megadom a DomRead osztályt szülőnek mivel már ott implementálva vannak a read metódusok ezért itt már csak megkell őket hívni
11 public class DomQueryIHDUW3 extends DomReadIHDUW3 {
12
13     public static void main(String[] args) {
14         try {
15             //Fájl Elérése
16             File inputFile = new File("/home/marci/eclipse-workspace/DOMParseIHDUW3/src/XmlIHDUW3.xml");
17             //Dom-dokumentum létrehozása az XML dokumentum eléréséhez
18             DocumentBuilderFactory dbFactory = DocumentBuilderFactory.newInstance();
19             DocumentBuilder dBuilder = dbFactory.newDocumentBuilder();
20             Document doc = dBuilder.parse(inputFile);
21             doc.getDocumentElement().normalize();
22             System.out.println("Gyoker elem : " + doc.getDocumentElement().getNodeName());
23             System.out.println("-----");
24
25             //Beolvasas
26             Scanner myObj = new Scanner(System.in);
27             System.out.println("Melyik elemet irassuk ki?");
28             String element = myObj.nextLine();
29
30             NodeList nList = doc.getElementsByTagName(element);
31             switch(element) {
32                 case "webshop":
33                     webshopRead(nList);
34                     break;
35                 case "gitar":
36                     guitarsRead(nList);
37                     break;
38                 case "gyarto":
39                     gyartoRead(nList);
40                     break;
41                 case "rendelo":
42                     rendeloRead(nList);
43                     break;
44                 case "rendeles":
45
46                     //Beolvasas
47                     Scanner myObj2 = new Scanner(System.in);
48                     System.out.println("Melyik elemet irassuk ki?");
49                     String element2 = myObj2.nextLine();
50
51                     NodeList nList2 = doc.getElementsByTagName(element2);
52                     switch(element2) {
53                         case "webshop":
54                             webshopRead(nList2);
55                             break;
56                         case "gitar":
57                             guitarsRead(nList2);
58                             break;
59                         case "gyarto":
60                             gyartoRead(nList2);
61                             break;
62                         case "rendelo":
63                             rendeloRead(nList2);
64                             break;
65                         case "rendeles":
66                             rendelessRead(nList2);
67                             break;
68                         case "rendelesek":
69                             rendelesekRead(nList2);
70                             break;
71                         case "fizetes":
72                             fizetesekRead(nList2);
73                             break;
74                         default:
75                             System.out.println("nincs ilyen Elem!");
76                     }
77                     myObj2.close();
78                 } catch (Exception e) {
79                     e.printStackTrace();
80                 }
81             }
82         }
83     }
84 }
```

Output:

```
Gyoker elem :gitarWebShops
-----
Melyik elemet irassuk ki?
webshop

  Gyoker elem :webshop
  ID : web1
  webshopnev : ThomannMusic
  utca : Hans Thomann utca
  varos : Treppendorf
  iranyitoszam : 96138
  tulajdonos : Hans Thomann
  alapitva : 1954
  idgitar : gitar1
  idrendelo : rendelo1

  Gyoker elem :webshop
  ID : web2
  webshopnev : Tajti Music Hangszer
  utca : Rákóczi út
  varos : Pécs
  iranyitoszam : 7622
  tulajdonos : Monori Tibor
  alapitva : 1990
  idgitar : gitar1
  idrendelo : rendelo1
```