

JEGYZŐKÖNYV

Adatkezelés XML környezetben

Féléves feladat

Készítette: Szabó Mátyás

Neptunkód: P3AANP

Feladat leírása:

A beadandó feladatom témája egy gyógyszerértár és annak kapcsolata a raktárakkal, a gyógyszerekkel és a vásárlókkal. Ezt fogom szemléltetni ER modellel, XDM modellel, XML és XMLSchema dokumentumokkal.

A **Gyógyszerértár** mondható a központi elemnek, viszont mind ugyan olyan fontos. Megadtam a gyógyszerértárról az ID-jét, és a dolgozói adatait.

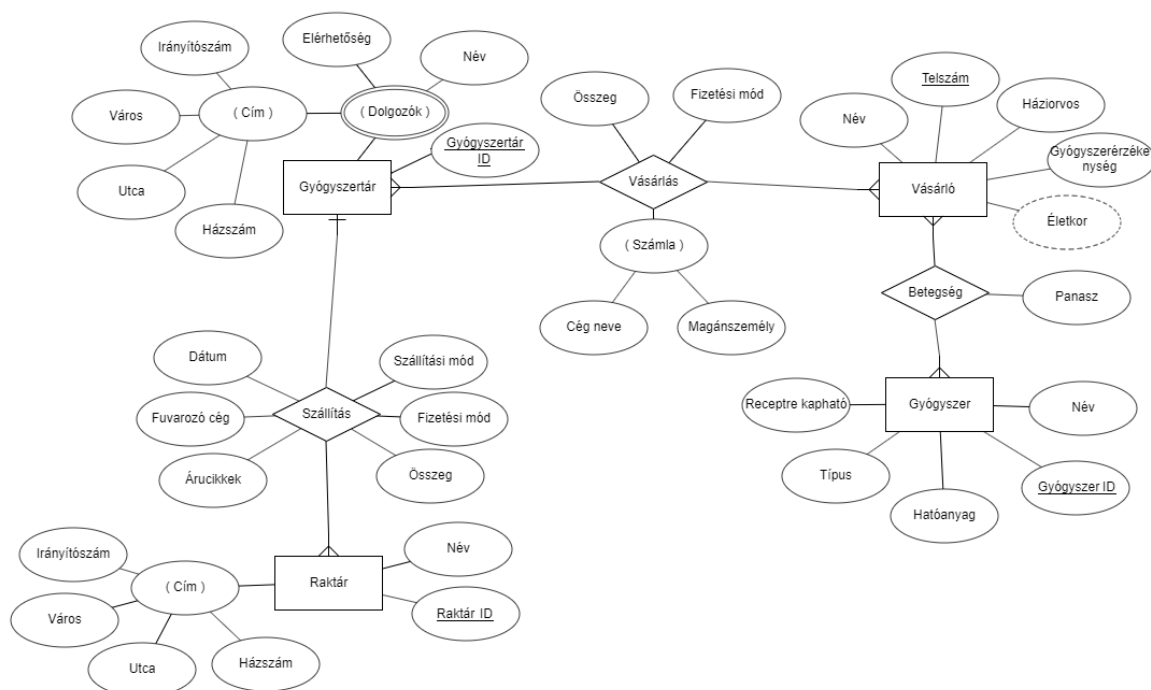
Mindemellett ott van még a **Raktár**, amely fontos szerepet játszik a szállítás terén. A raktárral összeköti a Szállítás kapcsolat, amelynek szintén rengeteg fontos attribútuma is megtalálható.

Mindemellett a **Gyógyszer** sem maradhat ki egy gyógyszerértári nyilvántartásból. Ennek is sok fontos attribútuma van feltüntetve, amelyekről a későbbiekben majd mindent megtudhat.

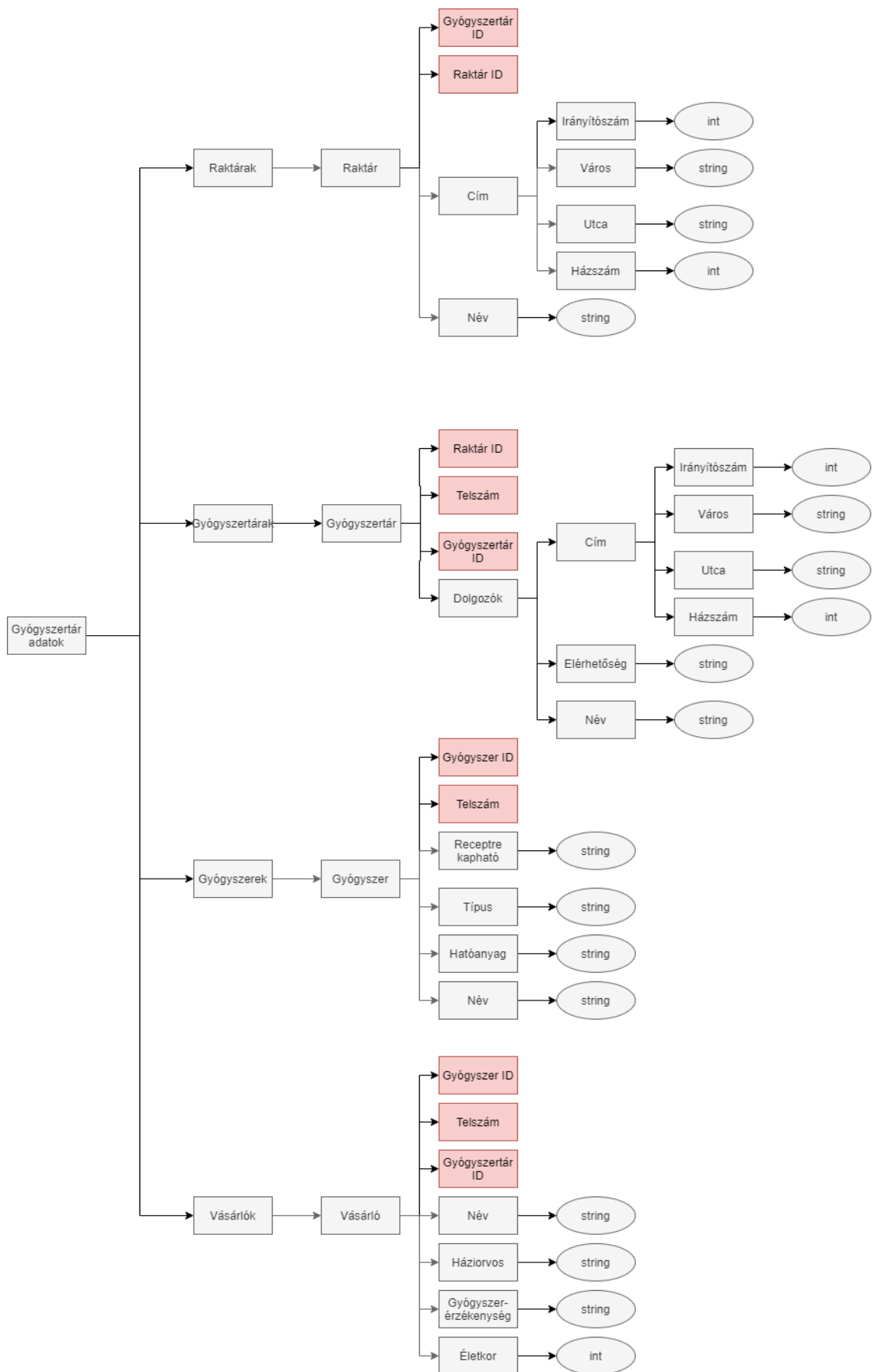
Végül itt van a **Vásárló** is, mivel mind tudjuk, a vásárló elégedettsége fontos nekünk. Így hát róluk is tárolunk el adatokat számlákhoz, különböző fizetési módok miatt.

1.feladat:

1a, ER modell



1b, XDM modell



Az itt látható XDM modell bemutatja az egyedeket és a hozzájuk tartozó attribútumokat a típusokkal karöltve, egy fa struktúrában.

Az egyedek, az attribútumok és a típusok szürkével, az elsődleges és az idegen kulcsok pedig pirossal vannak jelölve.

1c, XML dokumentum

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <gyogyszertar_adatok xmlns:xsi="http://www.w3.org/2001/XMLSchema" xsi:noNamespaceSchemaLocation="XMLSchemaP3AANP.xsd">
3
4   <raktarak>
5     <raktar gyogyszertar_id="43" raktar_id="32">
6       <cim>
7         <iranyitoszam>3300</iranyitoszam>
8         <varos>Eger</varos>
9         <utca>Kistályai út</utca>
10        <hazszam>40</hazszam>
11      </cim>
12      <nev>The Storage</nev>
13    </raktar>
14    <raktar gyogyszertar_id="44" raktar_id="21">
15      <cim>
16        <iranyitoszam>3350</iranyitoszam>
17        <varos>Kál</varos>
18        <utca>Fő út</utca>
19        <hazszam>10</hazszam>
20      </cim>
21      <nev>Bosch Storage</nev>
22    </raktar>
23  </raktarak>
24
25  <gyogyszertarak>
26    <gyogyszertar raktar_id="20" gyogyszertar_id="42" telszam="+36302203762">
27      <dolgozok>
28        <cim>
29          <iranyitoszam>3300</iranyitoszam>
30          <varos>Eger</varos>
31          <utca>Hádnagy utca</utca>
32          <hazszam>5</hazszam>
33        </cim>
34        <elerhetoseg>jakabk11@gmail.com</elerhetoseg>
35        <nev>Jakab Kálmán</nev>
36      </dolgozok>
37    </gyogyszertar>
38    <gyogyszertar raktar_id="35" gyogyszertar_id="2" telszam="+36302675762">
39      <dolgozok>
40        <cim>
41          <iranyitoszam>1087</iranyitoszam>
42          <varos>Budapest</varos>
43          <utca>Stróbl Alajos út</utca>
44          <hazszam>12</hazszam>
45        </cim>
46        <elerhetoseg>szimre2@gmail.com</elerhetoseg>
47        <nev>Szabó Imre</nev>
48      </dolgozok>
49    </gyogyszertar>
50  </gyogyszertarak>
51
52  <gyogyszerek>
53    <gyogyszer gyogyszer_id="2113" telszam="+36204567892">
54      <receptre kaphato>Nem</receptre kaphato>
55
56      <hatoanyag>ceclor</hatoanyag>
57      <tipus>Fájdalomcsillapító</tipus>
58      <nev>Algoflex</nev>
59    </gyogyszer>
60    <gyogyszer gyogyszer_id="113" telszam="+36204765432">
61      <receptre kaphato>Igen</receptre kaphato>
62      <hatoanyag>magnézium</hatoanyag>
63      <tipus>Vitamin</tipus>
64      <nev>Magnerot</nev>
65    </gyogyszer>
66  </gyogyszerek>
```

```

67 <vasarlok>
68   <vasarlo gyogyszer_id="221" telszam="+36313425678" gyogyszertar_id="30">
69     <nev>Kiss Gergő</nev>
70     <haziorvos>Dr.Lakatos László</haziorvos>
71     <gyogyszererzekenyseg>nincs</gyogyszererzekenyseg>
72     <eletkor>44</eletkor>
73   </vasarlo>
74   <vasarlo gyogyszer_id="245" telszam="+3631987678" gyogyszertar_id="10">
75     <nev>Szabó Mátyás</nev>
76     <haziorvos>Dr.Mann László</haziorvos>
77     <gyogyszererzekenyseg>ceclor</gyogyszererzekenyseg>
78     <eletkor>22</eletkor>
79   </vasarlo>
80 </vasarlok>
81
82 <gyogyszertar-raktar>
83   <szallitas gyogyszertar_id="44" raktar_id="35">
84     <datum>2020.12.03.</datum>
85     <fuvarozo_ceg>FastAndFurious Kft.</fuvarozo_ceg>
86     <arucikkek>Gyógyszerek</arucikkek>
87     <szallitasi_mod>Futárszolgálat</szallitasi_mod>
88     <fizetesi_mod>Előre utalás</fizetesi_mod>
89     <osszeg>456.300</osszeg>
90   </szallitas>
91   <szallitas gyogyszertar_id="42" raktar_id="10">
92     <datum>2020.12.10.</datum>
93     <fuvarozo_ceg>GLS</fuvarozo_ceg>
94     <arucikkek>Fecskendők</arucikkek>
95     <szallitasi_mod>Futárszolgálat</szallitasi_mod>
96     <fizetesi_mod>Készpénz, Utánvét</fizetesi_mod>
97     <osszeg>118.500</osszeg>
98   </szallitas>
99 </gyogyszertar-raktar>
100
101 <gyogyszertar-vasarlo>
102   <vasarlas gyogyszertar_id="2" telszam="+3631987678">
103     <osszeg>5300</osszeg>
104     <fizetesi_mod>Bankkártya</fizetesi_mod>
105     <szamla>
106       <ceg_neve>nincs</ceg_neve>
107       <maganszemely>Kolozsvári Anna</maganszemely>
108     </szamla>
109   </vasarlas>
110   <vasarlas gyogyszertar_id="43" telszam="+36204765432">
111     <osszeg>12340</osszeg>
112     <fizetesi_mod>Készpénz</fizetesi_mod>
113     <szamla>
114       <ceg_neve>Bosch Automotive Steering Kft.</ceg_neve>
115       <maganszemely>nincs</maganszemely>
116     </szamla>
117   </vasarlas>
118 </gyogyszertar-vasarlo>
119
120 <vasarlo-gyogyszer>
121   <betegseg telszam="+36204765432" gyogyszer_id="321">
122     <panasz>hányás, hasmenés</panasz>
123   </betegseg>
124   <betegseg telszam="+36204341675" gyogyszer_id="2230">
125     <panasz>fejfájás, szédülés</panasz>
126   </betegseg>
127 </vasarlo-gyogyszer>
128
129 </gyogyszertar_adatok>

```

1d, XMLSchema

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified" xmlns:xs="http://www.w3.org/2001/XMLSchema">
3   <xs:element name="gyogyszertar_adatok">
4     <xs:complexType>
5       <xs:sequence>
6
7         <xs:element name="raktarak">
8           <xs:complexType>
9             <xs:sequence>
10
11               <xs:element name="raktar" maxOccurs="unbounded" minOccurs="0">
12                 <xs:complexType>
13                   <xs:sequence>
14                     <xs:element name="cim" maxOccurs="unbounded" minOccurs="1">
15                       <xs:complexType>
16                         <xs:sequence>
17                           <xs:element type="xs:short" name="iranyitoszam"/>
18                           <xs:element type="xs:string" name="varos"/>
19                           <xs:element type="xs:string" name="utoa"/>
20                           <xs:element type="xs:short" name="hazszam"/>
21                         </xs:sequence>
22                       </xs:complexType>
23                     </xs:element>
24                     <xs:element type="xs:string" name="nev" />
25                   </xs:sequence>
26                   <xs:attribute type="xs:short" name="gyogyszertar_id"/>
27                   <xs:attribute type="xs:short" name="raktar_id"/>
28                 </xs:complexType>
29               </xs:element>
30             </xs:sequence>
31           </xs:complexType>
32         </xs:element>
33
34         <xs:element name="gyogyszertarak">
35           <xs:complexType>
36             <xs:sequence>
37               <xs:element name="gyogyszertar" maxOccurs="unbounded" minOccurs="0">
38                 <xs:complexType>
39                   <xs:sequence>
40                     <xs:element name="dolgozok" maxOccurs="unbounded" minOccurs="1">
41                       <xs:complexType>
42                         <xs:sequence>
43                           <xs:element name="cim" maxOccurs="unbounded" minOccurs="1">
44                             <xs:complexType>
45                               <xs:sequence>
46                                 <xs:element type="xs:short" name="iranyitoszam"/>
47                                 <xs:element type="xs:string" name="varos"/>
48                                 <xs:element type="xs:string" name="utoa"/>
49                                 <xs:element type="xs:short" name="hazszam"/>
50                               </xs:sequence>
51                             </xs:complexType>
52                           </xs:element>
53                           <xs:element type="xs:string" name="elerhetoseg" />
54                           <xs:element type="xs:string" name="nev" />
55                         </xs:sequence>
56                       </xs:complexType>
57                     </xs:element>
58                     <xs:attribute type="xs:short" name="raktar_id"/>
59                     <xs:attribute type="xs:short" name="gyogyszertar_id"/>
60                     <xs:attribute type="xs:long" name="telszam"/>
61                   </xs:complexType>
62                 </xs:element>
63               </xs:sequence>
64             </xs:complexType>
65           </xs:element>
66
67         <xs:element name="gyogyszerek">
68           <xs:complexType>
69             <xs:sequence>
70               <xs:element name="gyogyszer" maxOccurs="unbounded" minOccurs="0">
71                 <xs:complexType>
72                   <xs:sequence>
73                     <xs:element type="xs:string" name="receptre_kaphato"/>
74                     <xs:element type="xs:string" name="hatoanyag"/>
75                     <xs:element type="xs:string" name="tipus"/>
76                     <xs:element type="xs:string" name="nev"/>
77                   </xs:sequence>
78                 </xs:complexType>
79               </xs:element>
80             </xs:sequence>
81             <xs:attribute type="xs:short" name="gyogyszer_id"/>
82             <xs:attribute type="xs:long" name="telszam"/>
83           </xs:complexType>
84         </xs:element>
85
```

```

85
86     <xs:element name="vasarlok">
87         <xs:complexType>
88             <xs:sequence>
89                 <xs:element name="vasarlo" maxOccurs="unbounded" minOccurs="0">
90                     <xs:complexType>
91                         <xs:sequence>
92                             <xs:element type="xs:string" name="nev"/>
93                             <xs:element type="xs:string" name="hazi orvos"/>
94                             <xs:element type="xs:string" name="gyogyszererzekenyseg"/>
95                             <xs:element type="xs:short" name="eletkor"/>
96                         </xs:sequence>
97                     </xs:complexType>
98                 </xs:element>
99             </xs:sequence>
100             <xs:attribute type="xs:short" name="gyogyszer_id"/>
101             <xs:attribute type="xs:long" name="telszam"/>
102             <xs:attribute type="xs:short" name="gyogyszertar_id"/>
103         </xs:complexType>
104     </xs:element>
105
106     <xs:element name="gyogyszertar-raktar">
107         <xs:complexType>
108             <xs:sequence>
109                 <xs:element name="szallitas" maxOccurs="unbounded" minOccurs="0">
110                     <xs:complexType>
111                         <xs:sequence>
112                             <xs:element type="xs:date" name="datum" />
113                             <xs:element type="xs:string" name="fuvarozo_ceg" />
114                             <xs:element type="xs:string" name="arucikkek" />
115                             <xs:element type="xs:string" name="szallitasi_mod" />
116                             <xs:element type="xs:string" name="fizetesi_mod" />
117                             <xs:element type="xs:int" name="osszeg" />
118                         </xs:sequence>
119                         <xs:attribute type="xs:short" name="gyogyszertar_id" />
120                         <xs:attribute type="xs:short" name="raktar_id" />
121                     </xs:complexType>
122                 </xs:element>
123             </xs:sequence>
124         </xs:complexType>
125     </xs:element>
126
127     <xs:element name="gyogyszertar-vasarlo">
128         <xs:complexType>
129             <xs:sequence>
130                 <xs:element name="vasarlas" maxOccurs="unbounded" minOccurs="0">
131                     <xs:complexType>
132                         <xs:sequence>
133                             <xs:element type="xs:int" name="osszeg" />
134                             <xs:element type="xs:string" name="fizetesi_mod" />
135                             <xs:element name="szamla" maxOccurs="unbounded" minOccurs="1">
136                                 <xs:complexType>
137                                     <xs:sequence>
138                                         <xs:element type="xs:string" name="ceg neve" />
139                                         <xs:element type="xs:string" name="maganszemely" />
140                                     </xs:sequence>
141                                 </xs:complexType>
142                             </xs:element>
143                         </xs:sequence>
144                         <xs:attribute type="xs:short" name="gyogyszertar_id" />
145                         <xs:attribute type="xs:long" name="telszam" />
146                     </xs:complexType>
147                 </xs:element>
148             </xs:sequence>
149         </xs:complexType>
150     </xs:element>
151
152     <xs:element name="vasarlo-gyogyszer">
153         <xs:complexType>
154             <xs:sequence>
155                 <xs:element name="betegseg" maxOccurs="unbounded" minOccurs="0">
156                     <xs:complexType>
157                         <xs:sequence>
158                             <xs:element type="xs:string" name="panasz" />
159                         </xs:sequence>
160                         <xs:attribute type="xs:long" name="telszam" />
161                         <xs:attribute type="xs:short" name="gyogyszer_id" />
162                     </xs:complexType>
163                 </xs:element>
164             </xs:sequence>
165         </xs:complexType>
166     </xs:element>
167
168 </xs:sequence>
169 </xs:complexType>
170 </xs:element>
171 </xs:schema>

```

2.feladat:

2a, DOM adatolvasás

```

1 package hu.domparsa.p3aanp;
2
3 import org.w3c.dom.Document;
4 import org.w3c.dom.Element;
5 import org.w3c.dom.Node;
6 import org.w3c.dom.NodeList;
7 import org.xml.sax.SAXException;
8
9 import javax.xml.parsers.DocumentBuilder;
10 import javax.xml.parsers.DocumentBuilderFactory;
11 import javax.xml.parsers.ParserConfigurationException;
12
13 import java.io.File;
14 import java.io.IOException;
15
16 public class DOMReadP3AANP {
17
18     public static void main(String[] args)
19
20         throws ParserConfigurationException, IOException, SAXException {
21
22         File xml = new File("XMLP3AANP.xml"); //Az XML file beolvasása
23         DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance(); //DocumentBuilderFactory inicializálása
24
25         DocumentBuilder docBuilder = factory.newDocumentBuilder(); // A builder példányosítása
26         Document document = docBuilder.parse(xml); //A dokumentum konvertálása hogy értse az xml-t
27         document.getDocumentElement().normalize();
28
29         ReadFile(document); //Kiíratjuk az összes adatot
30
31     }
32
33     public static void ReadFile(Document document) { //Az adatok beolvasása és kiírása
34
35         NodeList nodeList1 = document.getElementsByTagName("raktar"); //Lista létrehozása a gyermekelemeknek a raktár kulcsszóra
36         for (int i = 0; i < nodeList1.getLength(); i++) {
37             Node rNode = nodeList1.item(i);
38
39             if (rNode.getNodeType() == Node.ELEMENT_NODE) { //Elemek id-hez adása
40                 Element element1 = (Element) rNode;
41                 String gyogyszertar_id = element1.getAttribute("gyogyszertar_id"); //Attribútumok eltárolása
42                 String raktar_id = element1.getAttribute("raktar_id");
43
44                 Node rNode1 = element1.getElementsByTagName("iranyitoszam").item(0); //Elemek eltárolása
45                 String irányitoszam = rNode1.getTextContent();
46
47                 Node rNode2 = element1.getElementsByTagName("varos").item(0);
48                 String varos = rNode2.getTextContent();
49
50                 Node rNode3 = element1.getElementsByTagName("utca").item(0);
51                 String utca = rNode3.getTextContent();
52
53                 Node rNode4 = element1.getElementsByTagName("hazszam").item(0);
54                 String hazszam = rNode4.getTextContent();
55
56                 System.out.println("Raktár adatok: Gyógyszertár ID: " + gyogyszertar_id + ", Raktár ID: " + raktar_id + ", Irányítószám: "
57                     + irányitoszam + ", Város: " + varos + ", Utca: " + utca + ", Házzám: " + hazszam + "."); //Az eltárolt adatok kiírása
58             }
59         }
60
61         NodeList nodeList2 = document.getElementsByTagName("gyogyszertar");
62         for (int i = 0; i < nodeList2.getLength(); i++) {
63             Node gytNode = nodeList2.item(i);
64
65             if (gytNode.getNodeType() == Node.ELEMENT_NODE) {
66                 Element element2 = (Element) gytNode;
67                 String raktar_id = element2.getAttribute("raktar_id");
68                 String gyogyszertar_id = element2.getAttribute("gyogyszertar_id");
69                 String telszam = element2.getAttribute("telszam");
70
71                 Node gytNode1 = element2.getElementsByTagName("iranyitoszam").item(0);
72                 String irányitoszam = gytNode1.getTextContent();
73
74                 Node gytNode2 = element2.getElementsByTagName("varos").item(0);
75                 String varos = gytNode2.getTextContent();
76
77                 Node gytNode3 = element2.getElementsByTagName("utca").item(0);
78                 String utca = gytNode3.getTextContent();
79
80                 Node gytNode4 = element2.getElementsByTagName("hazszam").item(0);
81                 String hazszam = gytNode4.getTextContent();
82
83                 Node gytNode5 = element2.getElementsByTagName("elerhetoseg").item(0);
84                 String elerhetoseg = gytNode5.getTextContent();
85
86                 Node gytNode6 = element2.getElementsByTagName("nev").item(0);
87                 String nev = gytNode6.getTextContent();
88
89                 System.out.println("Gyógyszertár adatok: Raktár ID: " + raktar_id + ", Gyógyszertár ID: " + gyogyszertar_id + ", Telefonszám: "
90                     + telszam + ", Dolgozó adatok: (Név: " + nev + ", Irányítószám: " + irányitoszam + ", Város: " + varos + ", Utca: " + utca + ", Házzám: "
91                     + hazszam + ", Elérhetőség: " + elerhetoseg + ".).");
92             }
93         }
94
95         NodeList nodeList3 = document.getElementsByTagName("gyogyszer");
96         for (int i = 0; i < nodeList3.getLength(); i++) {
97             Node gytNode = nodeList3.item(i);
98
99             if (gytNode.getNodeType() == Node.ELEMENT_NODE) {
100                 Element element3 = (Element) gytNode;
101                 String nev = element3.getAttribute("nev");
102                 String dozis = element3.getAttribute("dozis");
103                 String eljaras = element3.getAttribute("eljaras");
104                 String elerhetoseg = element3.getAttribute("elerhetoseg");
105                 String hazszam = element3.getAttribute("hazszam");
106                 String utca = element3.getAttribute("utca");
107                 String varos = element3.getAttribute("varos");
108                 String irányitoszam = element3.getAttribute("iranyitoszam");
109                 String raktar_id = element3.getAttribute("raktar_id");
110                 String gyogyszertar_id = element3.getAttribute("gyogyszertar_id");
111                 String telszam = element3.getAttribute("telszam");
112
113                 System.out.println("Gyógyszer adatok: Raktár ID: " + raktar_id + ", Gyógyszertár ID: " + gyogyszertar_id + ", Telefonszám: "
114                     + telszam + ", Dózis: " + dozis + ", Eljárás: " + eljaras + ", Elérhetőség: " + elerhetoseg + ", Házzám: " + hazszam + ", Utca: "
115                     + utca + ", Város: " + varos + ", Irányítószám: " + irányitoszam + ".");
116             }
117         }
118     }
119 }

```



```

98
99
100     if (gyNode.getNodeType() == Node.ELEMENT_NODE) {
101         Element element3 = (Element) gyNode;
102         String gyogyszer_id = element3.getAttribute("gyogyszer_id");
103         String telaszam = element3.getAttribute("telaszam");
104
105         Node gyNode1 = element3.getElementsByTagName("receptre_kaphato").item(0);
106         String receptre_kaphato = gyNode1.getTextContent();
107
108         Node gyNode2 = element3.getElementsByTagName("hatoanyag").item(0);
109         String hatoanyag = gyNode2.getTextContent();
110
111         Node gyNode3 = element3.getElementsByTagName("tipus").item(0);
112         String tipus = gyNode3.getTextContent();
113
114         Node gyNode4 = element3.getElementsByTagName("nev").item(0);
115         String nev = gyNode4.getTextContent();
116
117         System.out.println("Gyógyszer adatok: Gyógyszer ID: " + gyogyszer_id + ", Telefonszám: " + telaszam + ", Név: " + nev +
118             ", Receptre kapható: " + receptre_kaphato + ", Hatóanyag: " + hatoanyag + ", Típus: " + tipus + ".");
119     }
120 }
121
122 NodeList nodeList4 = document.getElementsByTagName("vasarolo");
123 for (int i = 0; i < nodeList4.getLength(); i++) {
124     Node vNode = nodeList4.item(i);
125
126     if (vNode.getNodeType() == Node.ELEMENT_NODE) {
127         Element element4 = (Element) vNode;
128         String gyogyszer_id = element4.getAttribute("gyogyszer_id");
129         String telaszam = element4.getAttribute("telaszam");
130         String gyogyszertar_id = element4.getAttribute("gyogyszertar_id");
131
132         Node vNode1 = element4.getElementsByTagName("nev").item(0);
133         String nev = vNode1.getTextContent();
134
135         Node vNode2 = element4.getElementsByTagName("haziorvos").item(0);
136         String haziorvos = vNode2.getTextContent();
137
138         Node vNode3 = element4.getElementsByTagName("gyogyszererzekenyseg").item(0);
139         String gyogyszererzekenyseg = vNode3.getTextContent();
140
141         Node vNode4 = element4.getElementsByTagName("eletkor").item(0);
142         String eletkor = vNode4.getTextContent();
143
144         System.out.println("Vásárló adatok: Gyógyszer ID: " + gyogyszer_id + ", Telefonszám: " + telaszam + ", Gyógyszertár ID: "
145             + gyogyszertar_id + ", Név: " + nev + ", Házi orvos: " + haziorvos + ", Gyógyszerérzékenység: " + gyogyszererzekenyseg +
146             ", Életkor: " + eletkor + ".");
147     }
148 }
149
150
151 NodeList nodeList5 = document.getElementsByTagName("szallitas");
152 for (int i = 0; i < nodeList5.getLength(); i++) {
153     Node szNode = nodeList5.item(i);

```

```

154
155     if (szNode.getNodeType() == Node.ELEMENT_NODE) {
156         Element element5 = (Element) szNode;
157         String gyogyszertar_id = element5.getAttribute("gyogyszertar_id");
158         String raktar_id = element5.getAttribute("raktar_id");
159
160         Node szNode1 = element5.getElementsByTagName("datum").item(0);
161         String datum = szNode1.getTextContent();
162
163         Node szNode2 = element5.getElementsByTagName("fuvarozo_ceg").item(0);
164         String fuvarozo_ceg = szNode2.getTextContent();
165
166         Node szNode3 = element5.getElementsByTagName("arucikkek").item(0);
167         String arucikkek = szNode3.getTextContent();
168
169         Node szNode4 = element5.getElementsByTagName("szallitasi_mod").item(0);
170         String szallitasi_mod = szNode4.getTextContent();
171
172         Node szNode5 = element5.getElementsByTagName("fizetesi_mod").item(0);
173         String fizetesi_mod = szNode5.getTextContent();
174
175         Node szNode6 = element5.getElementsByTagName("osszeg").item(0);
176         String osszeg = szNode6.getTextContent();
177
178         System.out.println("Szállítási adatok: Gyógyszertár ID: " + gyogyszertar_id + ", Raktár ID: " + raktar_id + ", Dátum: " + datum
179             + ", Fuvarozó cég: " + fuvarozo_ceg + ", Árucikkek: " + arucikkek + ", Szállítási mód: " + szallitasi_mod + ", Fizetési mód: "
180             + fizetesi_mod + ", Összeg: " + osszeg + ".");
181     }
182 }
183
184 NodeList nodeList6 = document.getElementsByTagName("vasarlas");
185 for (int i = 0; i < nodeList6.getLength(); i++) {
186     Node vNode = nodeList6.item(i);
187
188     if (vNode.getNodeType() == Node.ELEMENT_NODE) {
189         Element element6 = (Element) vNode;
190         String gyogyszertar_id = element6.getAttribute("gyogyszertar_id");
191         String telaszam = element6.getAttribute("telaszam");
192
193         Node vNode1 = element6.getElementsByTagName("osszeg").item(0);
194         String osszeg = vNode1.getTextContent();
195
196         Node vNode2 = element6.getElementsByTagName("fizetesi_mod").item(0);
197         String fizetesi_mod = vNode2.getTextContent();
198
199         Node vNode3 = element6.getElementsByTagName("ceg_neve").item(0);
200         String ceg_neve = vNode3.getTextContent();
201
202         Node vNode4 = element6.getElementsByTagName("maganszemely").item(0);
203         String maganszemely = vNode4.getTextContent();
204
205         System.out.println("Vásárlási adatok: Gyógyszertár ID: " + gyogyszertar_id + ", Telefonszám: " + telaszam + ", Összeg: " + osszeg
206             + ", Fizetési mód: " + fizetesi_mod + ", Számlázás: (Cég neve: " + ceg_neve + ", Magánszemély: " + maganszemely + ").");
207     }
208 }
209

```

```

210     NodeList nodeList7 = document.getElementsByTagName("betegseg");
211     for (int i = 0; i < nodeList7.getLength(); i++) {
212         Node bNode = nodeList7.item(i);
213
214         if (bNode.getNodeType() == Node.ELEMENT_NODE) {
215             Element element7 = (Element) bNode;
216             String telszam = element7.getAttribute("telszam");
217             String gyogyszer_id = element7.getAttribute("gyogyszer_id");
218
219             Node bNode1 = element7.getElementsByTagName("panasz").item(0);
220             String panasz = bNode1.getTextContent();
221
222             System.out.println("Betegség adatok: Telefonszám: " + telszam + ", Gyógyszer ID: " + gyogyszer_id + ", Panasz: " + panasz + ".");
223         }
224     }
225 }
226 }
227

```

2b, DOM módosítás

```

1  package hu.dompars.p3aanp;
2
3  import java.io.*;
4
5  import javax.xml.transform.dom.DOMSource;
6  import javax.xml.transform.stream.StreamResult;
7  import javax.xml.parsers.*;
8  import javax.xml.xpath.*;
9  import javax.xml.transform.Transformer;
10 import javax.xml.transform.TransformerFactory;
11 import javax.xml.transform.TransformerException;
12
13 import org.w3c.dom.*;
14 import org.xml.sax.*;
15
16 public class DOMModifyP3AANP {
17
18     public static void main(String[] args) throws ParserConfigurationException, IOException, SAXException, XPathExpressionException, TransformerException { (
19
20         String xmlFilePath = "XMLP3AANP.xml"; //File beolvasása
21         DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance();
22         DocumentBuilder builder = factory.newDocumentBuilder();
23         Document document = builder.parse(new File(xmlFilePath));
24
25         Node masodikGyogyszer = document.getElementsByTagName("gyogyszer").item(1); //A 2. gyógyszer elem gyerekelemeinek megkeresése
26         NodeList masodikGyogyszerNodeList = masodikGyogyszer.getChildNodes();
27
28         for (int i = 0; i < masodikGyogyszerNodeList.getLength(); i++) { //A 2. gyógyszer elem hatóanyagának megváltoztatása magnéziumra
29             Node element = masodikGyogyszerNodeList.item(i);
30             if ("hatóanyag".equals(element.getNodeName())) {
31                 element.setTextContent("magnézium");
32             }
33         }
34
35         Element ar = document.createElement("ár"); //A 2. gyógyszer elemhez hozzáadunk egy ár elemet is.
36         ar.appendChild(document.createTextNode("1750"));
37         masodikGyogyszer.appendChild(ar);
38
39         TransformerFactory transformerFactory = TransformerFactory.newInstance(); //A módosításokat feltöltjük az XML file-ba
40         Transformer transformer = transformerFactory.newTransformer();
41         DOMSource source = new DOMSource(document);
42         StreamResult result = new StreamResult(new File(xmlFilePath));
43         transformer.transform(source, result);
44
45         System.out.println("Változtatások sikeresen eszközölve az XML file-on!!!"); //Nyugtázzuk az elvégzett módosításokat
46     }
47 }
48

```