

JEGYZŐKÖNYV  
Adatkezelés XML környezetben  
Féléves feladat  
Horgászbolt

Készítette: Bajzáth Áron  
Neptunkód: ET2PK2

## 1.Feladat leírása:

A választott féléves feladatom egy horgászboltról szól. Alapvetően egy általános horgászbolt működését mutatja be a beadandóm. A horgászbolt egyedből indul ki és épül rá minden további egyed.

Az adatbázis egyedeim:

Horgászbolt egyed:

ID – Horgászbolt Kulcs tulajdonsága

Név – A horgászbolt neve

Telefonszám – A horgászbolt telefonszáma

Cím – Egy összetett tulajdonság ami a bolt címét írja le.

Horgászbolt egyed:

ID – A horgászbolt Kulcs tulajdonsága

Modell – A horgászbolt modell neve

Súly – A horgászbolt súlya

Hossz – A horgászbolt hossza

Minőség – A horgászbolt árát/minőségét jelképezi

Gyártó egyed:

ID – A gyártó Kulcs tulajdonsága

SzármazásiHely – A gyártó gyártási helye

Név – A gyártó neve

Telefonszám – A gyártó telefonszáma

Vásárló egyed:

ID – A vásárló Kulcs tulajdonsága

Név – A vásárló neve

Cím – A vásárló címe ami egy összetett tulajdonság

Email – A vásárló email címe

Vásárlások kapcsoló tábla: Ez kapcsolja össze a Vásárló és a Vásárlás egyedet.

Tulajdonságai:

Darab – A vásárolt termék darabszáma

Dátum – Mikor történt a vásárlás

Vásárlás egyed:

ID – A vásárlás Kulcs tulajdonsága

Jótállás – A termék kap e jótállási jegyet(igen/nem)

Termék – Milyen termék vesz a vásárló

Beszállító – A horgászbotot melyik beszállító szállította a boltba.

Fizetések egyed:

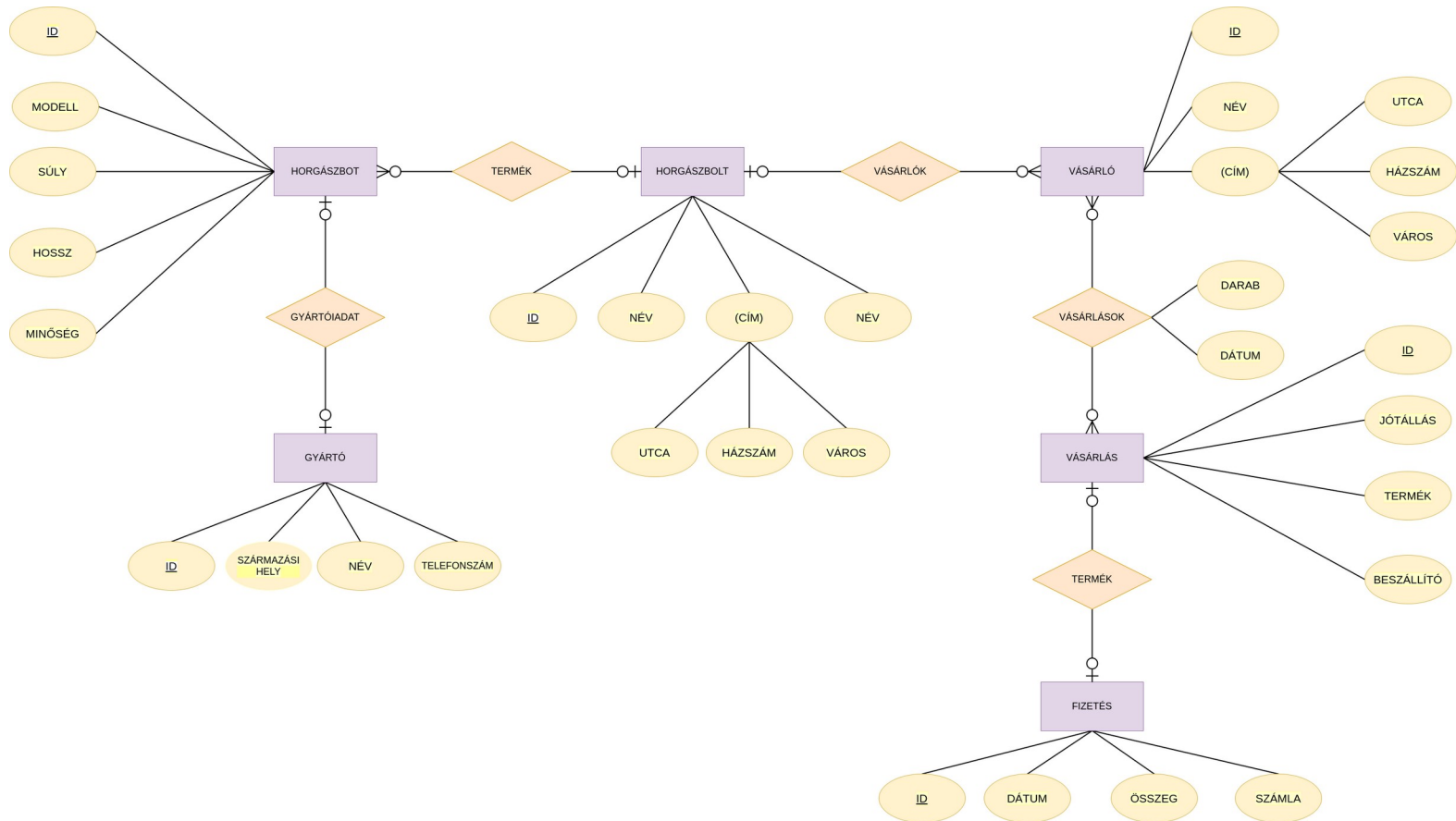
ID – A fizetések Kulcs tulajdonsága

Dátum – Mikor történt a fizetés

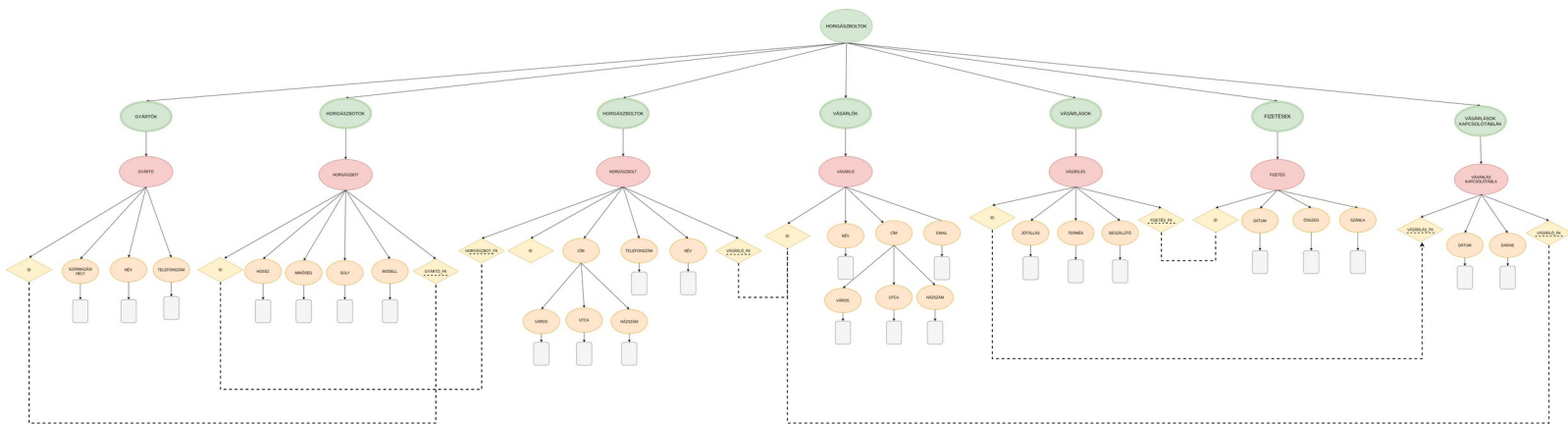
Összeg – Mennyibe került a vásárlás

Számla – Kért-e számlát a vásárló

# ER adatbázis modell:



Xdm modell: Az xdm modellben az elemeket elipszissel jelöljük. Az attribútumokat pedig rombuszsal. A szöveges tartalmakat téglalappal. Az idegen kulcsokat aláhúzással. Az N:M kapcsolatra külön elemet kell létrehozni.



# XML dokumentum: Az Xdm alapján elkészített xml dokumentum

```
1 <?xml version="1.0" encoding="UTF-8"?><?xml-model href="XMLSchemaET2PK2.xsd" ?>
2 <bolt>
3
4   <horgaszboltok>
5     <horgaszbolt id="h1">
6       <nev>Sporgio Horgaszbolt</nev>
7       <cim>
8         <utca>Szilagyi Dezso utca</utca>
9         <varos>Miskolc</varos>
10        <hazszam>9</hazszam>
11      </cim>
12
13      <telefonszam>06 30 978 2382</telefonszam>
14      <fk_id_horgaszbot refid="bot1">horgaszbot1
15    </fk_id_horgaszbot>
16    <fk_id_vasarlo refid="vasarlo1">vasarlo1</fk_id_vasarlo>
17
18    </horgaszbolt>
19    <horgaszbolt id="h2">
20      <nev>Meduza Horgaszbolt</nev>
21      <cim>
22        <utca>Pesti út</utca>
23        <varos>Miskolc</varos>
24        <hazszam>8</hazszam>
25      </cim>
26
27      <telefonszam>06 20 788 1727</telefonszam>
28      <fk_id_horgaszbot refid="bot2">horgaszbot2
29    </fk_id_horgaszbot>
30    <fk_id_vasarlo refid="vasarlo2">vasarlo2</fk_id_vasarlo>
31    </horgaszbolt>
32  </horgaszboltok>
33
34  <horgaszbotok>
35    <horgaszbot id="bot1">
36      <modell>Black Fox TFD-4</modell>
37      <hossz>3meter</hossz>
38      <minoseg>premium</minoseg>
39      <suly>2kg</suly>
40      <fk_id_gyarto refid="gy1">gy1</fk_id_gyarto>
41    </horgaszbot>
42    <horgaszbot id="bot2">
43      <modell> Black Fox A-2 </modell>
44      <hossz>3.3meter</hossz>
45      <minoseg>premium</minoseg>
46      <suly>4kg</suly>
47      <fk_id_gyarto refid="gy1">gy1</fk_id_gyarto>
48    </horgaszbot>
49  </horgaszbotok>
50  <gyartok>
51    <gyarto id="gy1">
52      <nev>Wind Blade</nev>
53      <szarmazasihely>Canada</szarmazasihely>
54      <telefonszam>45 79 84 00 00</telefonszam>
55    </gyarto>
56  </gyartok>
57
```

```

58<=> <vasarlok>
59<=>   <vasarlo id="vasarlo1">
60       <nev>Kovacs Endre</nev>
61       <email>endre@gmail.com</email>
62<=>       <cim>
63           <utca>Petofi utca</utca>
64           <varos>Miskolc</varos>
65           <hazszam>1</hazszam>
66       </cim>
67   </vasarlo>
68
69<=>   <vasarlo id="vasarlo2">
70       <nev>Pocs Judit</nev>
71       <email>judit@gmail.com</email>
72<=>       <cim>
73           <utca>Dozsa utca</utca>
74           <varos>Miskolc</varos>
75           <hazszam>12</hazszam>
76
77       </cim>
78   </vasarlo>
79 </vasarlok>
80<=> <vasarlasok>
81<=>   <vasarlas id="vasarlas1">
82       <jotallas>igen</jotallas>
83       <termek>horgaszbot</termek>
84       <beszallito>Euro Co.</beszallito>
85       <fk_id_fiz refid="f1">fiz1</fk_id_fiz>
86   </vasarlas>
87
88<=>   <vasarlas id="vasarlas2">
89       <jotallas>igen</jotallas>
90       <termek>horgaszbot</termek>
91       <beszallito>Slovansko Zrt.</beszallito>
92       <fk_id_fiz refid="f2">fiz1</fk_id_fiz>
93   </vasarlas>
94 </vasarlasok>

```

```
96<= <vasarlasok_kapcsolotabla>
97<=   <kapcsvasarlasok id="1">
98       <fk_id_vasarlo refid="vasarlo1">vasarlo1</fk_id_vasarlo>
99       <fk_id_vasarlas refid="vasarlas1">vasarlas1</fk_id_vasarlas>
100       <darab>1</darab>
101       <datum>2021.12.01</datum>
102   </kapcsvasarlasok>
103
104<=   <kapcsvasarlasok id="2">
105       <fk_id_vasarlo refid="vasarlo2">vasarlo2</fk_id_vasarlo>
106       <fk_id_vasarlas refid="vasarlas2">vasarlas2</fk_id_vasarlas>
107       <darab>1</darab>
108       <datum>2021.10.29</datum>
109   </kapcsvasarlasok>
110 </vasarlasok_kapcsolotabla>
111
112<= <fizetesek>
113<=   <fizetes id="f1">
114       <datum>2021.12.01</datum>
115       <osszeg>120000</osszeg>
116       <szamla>igen</szamla>
117   </fizetes>
118
119<=   <fizetes id="f2">
120       <datum>2021.10.29</datum>
121       <osszeg>210000</osszeg>
122       <szamla>igen</szamla>
123   </fizetes>
124 </fizetesek>
125
126 </bolt>
127
128
```



## XSD: Az xml alapján elkészített xsd dokumentum.

```
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="bolt">
7
8     <xs:complexType>
9       <xs:sequence>
10        <xs:element name="horgaszboltok" type="horgaszbolttype" />
11        <xs:element name="horgaszbotok" type="horgaszbottype" />
12        <xs:element name="gyartok" type="gyartoktype" />
13        <xs:element name="vasarlok" type="vasarloktype" />
14        <xs:element name="vasarlasok" type="vasarlasoktype" />
15        <xs:element name="vasarlasok_kapcsolotabla" type="vasarlasok_kapcsolotablatype" />
16        <xs:element name="fizetesek" type="fizetesektype" />
17      </xs:sequence>
18    </xs:complexType>
19
20
21    <xs:key name="horgaszboltKey">
22      <xs:selector xpath="horgaszboltok/horgaszbolt" />
23      <xs:field xpath="@id" />
24    </xs:key>
25
26
27    <xs:key name="horgaszbotokKey">
28      <xs:selector xpath="horgaszbotok/horgaszbot" />
29      <xs:field xpath="@id" />
30    </xs:key>
31
32    <xs:key name="gyartokKey">
33      <xs:selector xpath="gyartok/gyarto" />
34      <xs:field xpath="@id" />
35    </xs:key>
36
37    <xs:key name="vasarlokKey">
38      <xs:selector xpath="vasarlok/vasarlo" />
39      <xs:field xpath="@id" />
40    </xs:key>
41
42    <xs:key name="vasarlasokKey">
43      <xs:selector xpath="vasarlasok/vasarlas" />
44      <xs:field xpath="@id" />
45    </xs:key>
46
47    <xs:key name="vasarlasok_kapcsolotablaKey">
48      <xs:selector xpath="vasarlasok_kapcsolotabla/vasarlasok" />
49      <xs:field xpath="@id" />
50    </xs:key>
51  </xs:schema>
```

```

52     <xs:key name="fizetesekKey">
53         <xs:selector xpath="fizetesek/fizetes" />
54         <xs:field xpath="@id" />
55     </xs:key>
56
57     <xs:keyref name="horgaszbot-gyarto" refer="gyartokKey">
58         <xs:selector xpath="horgaszbotok/horgaszbot/id" />
59         <xs:field xpath="@id" />
60     </xs:keyref>
61
62     <xs:keyref name="horgaszbolt-horgaszbot" refer="horgaszbotokKey">
63         <xs:selector xpath="horgaszbolt/horgaszbolt/id" />
64         <xs:field xpath="@id" />
65     </xs:keyref>
66
67     <xs:keyref name="horgaszbolt-vasarlo" refer="horgaszboltKey">
68         <xs:selector xpath="horgaszboltok/horgaszbolt/id" />
69         <xs:field xpath="@id" />
70     </xs:keyref>
71
72     <xs:keyref name="vasarlas-fizetes" refer="vasarlasokKey">
73         <xs:selector xpath="fizetesek/fizetes/id" />
74         <xs:field xpath="@id" />
75     </xs:keyref>
76
77 </xs:element>
78
79 <xs:complexType name="horgaszbolttyp">
80     <xs:sequence>
81         <xs:element name="horgaszbolt" maxOccurs="unbounded">
82             <xs:complexType>
83                 <xs:sequence>
84                     <xs:element name="nev" type="xs:string" />
85                     <xs:element name="cim" type="cimtype" />
86                     <xs:element name="telefonszam" type="xs:string" />
87                 </xs:sequence>
88                 <xs:attribute name="id" type="xs:ID" use="required" />
89             </xs:complexType>
90         </xs:element>
91     </xs:sequence>
92 </xs:complexType>
93
94
95 <xs:complexType name="cimtype">
96     <xs:sequence>
97         <xs:element name="utca" type="xs:string" />
98         <xs:element name="varos" type="xs:string" />
99         <xs:element name="hazszam" type="xs:int" />
100     </xs:sequence>
101 </xs:complexType>
102
103 <xs:complexType name="horgaszbottype">
104     <xs:sequence>
105         <xs:element name="horgaszbot" maxOccurs="unbounded">
106             <xs:complexType>
107                 <xs:sequence>
108                     <xs:element name="modell" type="xs:string" />
109                     <xs:element name="minoseg" type="minosegtype" />
110                     <xs:element name="suly" type="xs:int" />
111                     <xs:element name="hossz" type="xs:int" />
112                 </xs:sequence>
113                 <xs:attribute name="id" type="xs:ID" use="required" />
114             </xs:complexType>
115         </xs:element>
116     </xs:sequence>
117 </xs:complexType>

```

```

120 <xs:complexType name="minosegtype">
121   <xs:sequence>
122     <xs:element name="premium" type="xs:string" />
123     <xs:element name="amateur" type="xs:string" />
124     <xs:element name="medium" type="xs:string" />
125   </xs:sequence>
126 </xs:complexType>
127
128 <xs:complexType name="gyartoktype">
129   <xs:sequence>
130     <xs:element name="gyartok" maxOccurs="unbounded">
131       <xs:complexType>
132         <xs:sequence>
133           <xs:element name="nev" type="xs:string" />
134           <xs:element name="szarmazasihely" type="xs:string" />
135           <xs:element name="telefonszam" type="xs:string" />
136         </xs:sequence>
137         <xs:attribute name="id" type="xs:ID" use="required" />
138       </xs:complexType>
139     </xs:element>
140   </xs:sequence>
141 </xs:complexType>
142
143 <xs:complexType name="vasarloktype">
144   <xs:sequence>
145     <xs:element name="rendelo" maxOccurs="unbounded">
146       <xs:complexType>
147         <xs:sequence>
148           <xs:element name="nev" type="xs:string" />
149           <xs:element name="email" type="xs:string" />
150           <xs:element name="cim" type="cimtype" />
151         </xs:sequence>
152         <xs:attribute name="id" type="xs:ID" use="required" />
153       </xs:complexType>
154     </xs:element>
155   </xs:sequence>
156 </xs:complexType>
157
158 <xs:complexType name="vasarlasok_kapcsolotablatype">
159   <xs:sequence>
160     <xs:element name="kapcsvasarlasok" maxOccurs="unbounded">
161       <xs:complexType>
162         <xs:sequence>
163           <xs:element name="darab" type="xs:integer" />
164           <xs:element name="datum" type="xs:string" />
165         </xs:sequence>
166         <xs:attribute name="id" type="xs:ID" use="required" />
167       </xs:complexType>
168     </xs:element>
169   </xs:sequence>
170 </xs:complexType>
171
172 <xs:complexType name="vasarlasoktype">
173   <xs:sequence>
174     <xs:element name="vasarlas" maxOccurs="unbounded">
175       <xs:complexType>
176         <xs:sequence>
177           <xs:element name="jotallas" type="xs:string" />
178           <xs:element name="termek" type="xs:string" />
179           <xs:element name="beszallito" type="xs:string" />
180         </xs:sequence>
181         <xs:attribute name="id" type="xs:ID" use="required" />
182       </xs:complexType>
183     </xs:element>
184   </xs:sequence>
185 </xs:complexType>
186
187 </xs:sequence>
188 </xs:complexType>
189

```



```
191<xs:complexType name="fizeteseotype">
192  <xs:sequence>
193    <xs:element name="fizetes" maxOccurs="unbounded">
194      <xs:complexType>
195        <xs:sequence>
196          <xs:element name="datum" type="xs:string" />
197          <xs:element name="osszeg" type="cimtype" />
198          <xs:element name="fizetesiid" type="xs:string" />
199        </xs:sequence>
200        <xs:attribute name="id" type="xs:ID" use="required" />
201      </xs:complexType>
202    </xs:element>
203  </xs:sequence>
204</xs:complexType>
205</xs:schema>
206
207
```

## 2. DomRead: Az Xml fájl beolvasása és kiírása

```
1 package hu.domp.parse.ET2PK2;
2
3 import java.io.File;
10
11 public class DomReadET2PK2 {
12
13     public static void main(String[] args) {
14         try {
15             // Fájl elérése
16             File inputFile = new File("src/XmlET2PK2.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("Aktualis elem : " + doc.getDocumentElement().getNodeName());
23             System.out.println("\n");
24
25             // Elemek kiírása metódus hívásokkal
26             NodeList nList = doc.getElementsByTagName("horgaszbolt");
27             horgaszboltRead(nList);
28
29             NodeList nList2 = doc.getElementsByTagName("horgaszbot");
30             horgaszbotokRead(nList2);
31
32             NodeList nList3 = doc.getElementsByTagName("gyarto");
33             gyartoRead(nList3);
34
35             NodeList nList4 = doc.getElementsByTagName("vasarlo");
36             vasarloRead(nList4);
37
38             NodeList nList5 = doc.getElementsByTagName("vasarlas");
39             vasarlasokRead(nList5);
40
41             NodeList nList6 = doc.getElementsByTagName("kapcsvasarlasok");
42             vasarlasok_kapcsolotablaRead(nList6);
43
44             NodeList nList7 = doc.getElementsByTagName("fizetes");
45             fizetesekRead(nList7);
46
47         } catch (Exception e) {
48             e.printStackTrace();
49         }
50
51     }
52
53     // Saját metódus ami egyszerűsíti a kiíratást
54     public static void getElement(Element eElement, String elementOut, String elementName) {
55
56         System.out.println(elementOut + " : " + eElement.getElementsByTagName(elementName).item(0).getTextContent());
57     }
58
59     //Itt implementálom a metódusokat amiket később meghívok
60     //ezek olvassák ki az xml fájlból az adatokat
61     public static void horgaszboltRead(NodeList nList) {
62         for (int temp = 0; temp < nList.getLength(); temp++) {
63             Node nNode = nList.item(temp);
64             System.out.println("\n Aktualis elem : " + nNode.getNodeName());
65
66             if (nNode.getNodeType() == Node.ELEMENT_NODE) {
67                 Element eElement = (Element) nNode;
68                 System.out.println("ID : " + eElement.getAttribute("id"));
69
70                 getElement(eElement, "nev", "nev");
71                 getElement(eElement, "utca", "utca");
72                 getElement(eElement, "varos", "varos");
73                 getElement(eElement, "hazszam", "hazszam");
74                 getElement(eElement, "telefonszam", "telefonszam");
75                 getElement(eElement, "fk_id_horgaszbot", "fk_id_horgaszbot");
76                 getElement(eElement, "fk_id_vasarlo", "fk_id_vasarlo");
77
78             }
79         }
80     }
```

```

82 public static void horgaszbotokRead(NodeList nList) {
83     for (int temp = 0; temp < nList.getLength(); temp++) {
84         Node nNode = nList.item(temp);
85         System.out.println("\n Aktualis elem : " + nNode.getNodeName());
86
87         if (nNode.getNodeType() == Node.ELEMENT_NODE) {
88             Element eElement = (Element) nNode;
89             System.out.println("ID : " + eElement.getAttribute("id"));
90
91             getElement(eElement, "modell", "modell");
92             getElement(eElement, "hossz", "hossz");
93             getElement(eElement, "minoseg", "minoseg");
94             getElement(eElement, "suly", "suly");
95             getElement(eElement, "fk_id_gyarto", "fk_id_gyarto");
96         }
97     }
98 }
99
100
101 public static void gyartoRead(NodeList nList) {
102     for (int temp = 0; temp < nList.getLength(); temp++) {
103         Node nNode = nList.item(temp);
104         System.out.println("\n Aktualis elem : " + nNode.getNodeName());
105
106         if (nNode.getNodeType() == Node.ELEMENT_NODE) {
107             Element eElement = (Element) nNode;
108             System.out.println("ID : " + eElement.getAttribute("id"));
109
110             getElement(eElement, "nev", "nev");
111             getElement(eElement, "szarmazasihely", "szarmazasihely");
112             getElement(eElement, "telefonszam", "telefonszam");
113         }
114     }
115 }
116
117
118 public static void vasarloRead(NodeList nList) {
119     for (int temp = 0; temp < nList.getLength(); temp++) {
120         Node nNode = nList.item(temp);
121         System.out.println("\n Aktualis elem : " + nNode.getNodeName());
122
123         if (nNode.getNodeType() == Node.ELEMENT_NODE) {
124             Element eElement = (Element) nNode;
125             System.out.println("ID : " + eElement.getAttribute("id"));
126
127             getElement(eElement, "nev", "nev");
128             getElement(eElement, "email", "email");
129             getElement(eElement, "utca", "utca");
130             getElement(eElement, "varos", "varos");
131             getElement(eElement, "hazszam", "hazszam");
132         }
133     }
134 }
135
136
137 public static void vasarlasokRead(NodeList nList) {
138     for (int temp = 0; temp < nList.getLength(); temp++) {
139         Node nNode = nList.item(temp);
140         System.out.println("\n Aktualis elem : " + nNode.getNodeName());
141
142         if (nNode.getNodeType() == Node.ELEMENT_NODE) {
143             Element eElement = (Element) nNode;
144             System.out.println("ID : " + eElement.getAttribute("id"));
145
146             getElement(eElement, "jotallas", "jotallas");
147             getElement(eElement, "termek", "termek");
148             getElement(eElement, "beszallito", "beszallito");
149             getElement(eElement, "fk_id_fiz", "fk_id_fiz");
150         }
151     }
152 }
153
154 }

```



```

155 public static void varaslarok_kapcsolotablaRead(NodeList nList) {
156     for (int temp = 0; temp < nList.getLength(); temp++) {
157         Node nNode = nList.item(temp);
158         System.out.println("\n Aktualis elem : " + nNode.getNodeName());
159
160         if (nNode.getNodeType() == Node.ELEMENT_NODE) {
161             Element eElement = (Element) nNode;
162             System.out.println("ID : " + eElement.getAttribute("id"));
163             getElement(eElement, "fk_id_vararlo", "fk_id_vararlo");
164             getElement(eElement, "fk_id_varaslas", "fk_id_varaslas");
165             getElement(eElement, "darab", "darab");
166             getElement(eElement, "datum", "datum");
167
168         }
169     }
170 }
171
172 public static void fizetesekRead(NodeList nList) {
173     for (int temp = 0; temp < nList.getLength(); temp++) {
174         Node nNode = nList.item(temp);
175         System.out.println("\n Aktualis elem : " + nNode.getNodeName());
176
177         if (nNode.getNodeType() == Node.ELEMENT_NODE) {
178             Element eElement = (Element) nNode;
179             System.out.println("ID : " + eElement.getAttribute("id"));
180             getElement(eElement, "datum", "datum");
181             getElement(eElement, "osszeg", "osszeg");
182             getElement(eElement, "szamla", "szamla");
183
184         }
185     }
186 }
187 }
188

```

```

sterminated> D:\k\ad\12\K2 java Application\7\as\107\jvm\java
Aktualis elem :bolt

Aktualis elem :horgaszbolt
ID : h1
nev : Sporgio Horgaszbolt
utca : Szilagyi Dezso utca
varos : Miskolc
hazszam : 9
telefonszam : 06 30 978 2382
fk_id_horgaszbot : horgaszbot1

fk_id_vararlo : vararlo1

Aktualis elem :horgaszbolt
ID : h2
nev : Meduza Horgaszbolt
utca : Pesti út
varos : Miskolc
hazszam : 8
telefonszam : 06 20 788 1727
fk_id_horgaszbot : horgaszbot2

fk_id_vararlo : vararlo2

Aktualis elem :horgaszbot
ID : bot1
modell : Black Fox TFD-4
hossz : 3meter
minoseg : premium
suly : 2kg
fk_id_gyarto : gyl

Aktualis elem :horgaszbot
ID : bot2
modell : Black Fox A-2
hossz : 3.3meter
minoseg : premium
suly : 4kg
fk_id_gyarto : gyl

```

```

Aktualis elem :gyarto
ID : gyl
nev : Wind Blade
szarmazasihely : Canada
telefonszam : 45 79 84 00 00

Aktualis elem :vararlo
ID : vararlo1
nev : Kovacs Endre
email : endre@gmail.com
utca : Petofi utca
varos : Miskolc
hazszam : 1

Aktualis elem :vararlo
ID : vararlo2
nev : Pocs Judit
email : judit@gmail.com
utca : Dozsa utca
varos : Miskolc
hazszam : 12

Aktualis elem :varaslas
ID : varaslas1
jotallas : igen
termek : horgaszbot
beszallito : Euro Co.
fk_id_fiz : fiz1

Aktualis elem :varaslas
ID : varaslas2
jotallas : igen
termek : horgaszbot
beszallito : Slovansko Zrt.
fk_id_fiz : fiz1

Aktualis elem :kapcsvaraslarok
ID : 1
fk_id_vararlo : vararlo1
fk_id_varaslas : varaslas1
darab : 1
datum : 2021.12.01

```

## XDM Modify: Itt módosítom az xml fájlt

```
1 package hu.domparse.ET2PK2;
2
3 import java.io.FileInputStream;
4
21
22 public class DomModifyET2PK2{
23
24     public static void main(String[] args) throws TransformerException {
25         // Dom-dokumentum létrehozása az XML dokumentum eléréséhez
26         DocumentBuilderFactory dbf = DocumentBuilderFactory.newInstance();
27
28         try (InputStream is = new FileInputStream("src/XmlET2PK2.xml")) {
29
30             DocumentBuilder db = dbf.newDocumentBuilder();
31
32             Document doc = db.parse(is);
33             // Itt bescanneljük az elemeket
34             Scanner rootElem = new Scanner(System.in);
35             System.out.println("Módosítani kívánt elem?");
36             String element = rootElem.nextLine();
37
38             Scanner elemID = new Scanner(System.in);
39             System.out.println("Add meg az ID-t!");
40             String childElementID = elemID.nextLine();
41
42             NodeList listOffishshop = doc.getElementsByTagName(element);
43
44             Scanner elem = new Scanner(System.in);
45             System.out.println("Melyik gyerek elemet módosítsuk?");
46             String childElement = elem.nextLine();
47
48             System.out.println("Mi legyen az új elem értéke?");
49             Scanner ujElem = new Scanner(System.in);
50             String childElementElementNew = ujElem.nextLine();
51
52             //Végigmegyünk az xml dokumentumon
53             for (int i = 0; i < listOffishshop.getLength(); i++) {
54                 //a temp megkapja a kiválasztott elemeket
55                 Node temp = listOffishshop.item(i);
56                 if (temp.getNodeType() == Node.ELEMENT_NODE) {
57                     //Megadjuk melyik az az ID-jú elem amit módosítunk
58                     String id = temp.getAttributes().getNamedItem("id").getTextContent();
59                     //ha egyezik valamelyik id-vel akkor tovább hajtódik az utasítás
60                     if (childElementID.equals(id.trim())) {
61                         //A childNodes-ba beolvassuk a Root elementnek a gyerekeit
62                         NodeList childNodes = temp.getChildNodes();
63                         //Itt történik utána a módosítás
64                         for (int j = 0; j < childNodes.getLength(); j++) {
65                             Node item = childNodes.item(j);
66                             if (item.getNodeType() == Node.ELEMENT_NODE) {
67
68                                 if (childElement.equalsIgnoreCase(item.getNodeName())) {
69                                     item.setTextContent(childElementElementNew);
70                                 }
71                             }
72                         }
73                     }
74                 }
75             }
76         }
77     }
78 }
79
80 }
81
```



```

82
83 // Létrehozunk egy TransformerFactory és egy DOMSource objektumot ezzel leegyszerűsítve a kiíratást
84 //A transformerfactory-val bármilyen szöveges tartalmat egyszerűen kiíráthatunk
85
86 TransformerFactory transformerFactory = TransformerFactory.newInstance();
87 Transformer transformer = transformerFactory.newTransformer();
88 // itt megadjuk a saját xmlünket amit már az elején deklaráltunk
89 DOMSource source = new DOMSource(doc);
90 //Itt a streamet adjuk meg jelen esetben a console-ra fogunk kiíratni
91 StreamResult consoleResult = new StreamResult(System.out);
92 transformer.transform(source, consoleResult);
93
94 } catch (ParserConfigurationException | SAXException | IOException e) {
95     e.printStackTrace();
96 }
97
98 }
99
100 }
101

```

Modosítani kívánt elem?

horgaszbolt

Add meg az ID-t!

h1

Melyik gyerek elemet módosítsuk?

nev

Mi legyen az új elem értéke?

ujtertek

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?><?xml-model href="XMLSchemaET2PK2.xsd" ?><bolt>
```

```
    <horgaszboltok>
```

```
        <horgaszbolt id="h1">
```

```
            <nev>ujtertek</nev>
```

```
            <cim>
```

```
                <utca>Szilagyi Dezső utca</utca>
```

```
                <varos>Miskolc</varos>
```

```
                <hazszam>9</hazszam>
```

```
            </cim>
```

```
            <telefonszam>06 30 978 2382</telefonszam>
```

```
            <fk_id_horgaszbot refid="bot1">horgaszbot1
```

```
            </fk_id_horgaszbot>
```

```
            <fk_id_vasarlo refid="vasarlo1">vasarlo1</fk_id_vasarlo>
```

```
        </horgaszbolt>
```

DomQuery: Itt lehet lekérdezni az elemeket.

```
package hu.domparse.ET2PK2;

import java.io.File;
//Példányosítom a readosztályt és felhasználom a metódusait
public class DomQueryET2PK2 extends DomReadET2PK2 {

    public static void main(String[] args) {
        try {
            // Fájl Elérése
            File inputFile = new File("src/XmlET2PK2.xml");
            // Dom-dokumentum létrehozása az XML dokumentum eléréséhez
            DocumentBuilderFactory dbFactory = DocumentBuilderFactory.newInstance();
            DocumentBuilder dBuilder = dbFactory.newDocumentBuilder();
            Document doc = dBuilder.parse(inputFile);
            doc.getDocumentElement().normalize();
            System.out.println("Gyoker elem : " + doc.getDocumentElement().getNodeName());
            //Itt lehet kiválasztani melyik elemet irassuk ki
            //horgaszbolt
            //horgaszbot
            //gyarto
            //vasarlo
            //vasarlas
            //kapcsvasarlasok
            //fizetes
            String element = "horgaszbolt";
            //itt adjuk meg melyik elemet hívjuk meg
            NodeList nList = doc.getElementsByTagName(element);
            //itt hívjuk meg a metódust
            horgaszboltRead(nList);
            //horgaszbotokRead(nList);
            //gyartoRead(nList);
            //vasarloRead(nList);
            //vasarlasokRead(nList);
            //vasarlasok_kapcsolotablaRead(nList);
            //fizetesekRead(nList);

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

<terminated> DomQueryET2PK2 [Java Applicati

Gyoker elem :bolt

Aktualis elem :horgaszbolt

ID : h1

nev : Sporgio Horgaszbolt

utca : Szilagyi Dezso utca

varos : Miskolc

hazszam : 9

telefonszam : 06 30 978 2382

fk\_id\_horgaszbot : horgaszbot1

fk\_id\_vasarlo : vasarlo1

Aktualis elem :horgaszbolt

ID : h2

nev : Meduza Horgaszbolt

utca : Pesti út

varos : Miskolc

hazszam : 8

telefonszam : 06 20 788 1727

fk\_id\_horgaszbot : horgaszbot2

fk\_id\_vasarlo : vasarlo2