

# JEGYZŐKÖNYV

Adatkezelés XML környezetben

Féléves feladat

Készítette: **Nemcsik Dániel**

Neptunkód: **I3I4BP**

**Feladat leírás:**

A feladatomban egy könyvtár működését mutatnám be. Egy könyvhöz tartozik szerző, könyvesbolt, kiadó és vásárló, mint ahogy az ER modellben és XDM modellbe is látszik majd. Egy könyvet egy szerzőhöz köthetünk (1:1 kapcsolat), egy könyvet több kiadó is kiadhat (1:n kapcsolat), több könyvet több könyvesbolt is megvehet illetve eladhat (n:n kapcsolat), illetve több könyvesboltból több vásárló is vásárolhat (n:n kapcsolat). Tudom, hogy ebbe a logikába bele lehetne kötni, például egy könyvnek több szerzője is lehet stb..., de annak érdekében lett így elkészítve, hogy mindenféle kapcsolat szerepeljen a feladatban.

Az XDM modellben nem szerepelnek olyan adatok, amiket az ER modellben kapcsolatokkal valósítottam meg (pl.: beszerez, eladás kapcsolat, bÁr, eDarab, stb.), de szerintem ettől függetlenül is sikerült egy értelmes XML file-t létrehoznom.

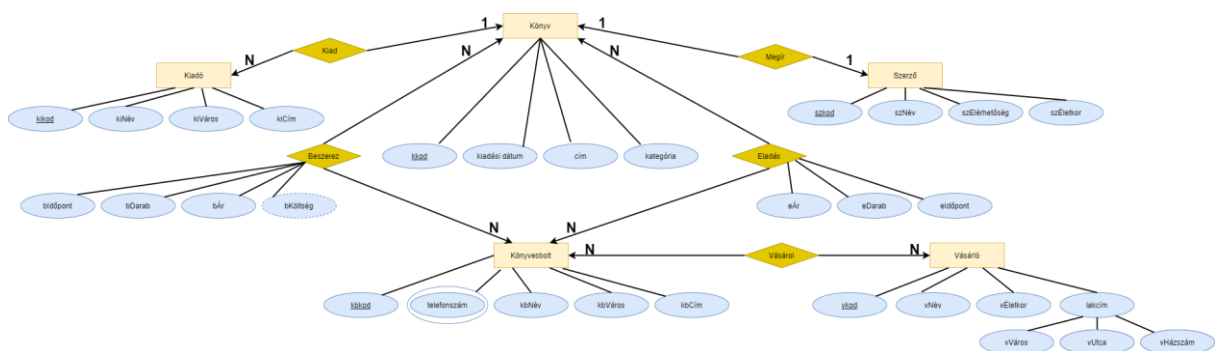
[https://github.com/daninemcsik/I3I4BP\\_XMLGyak](https://github.com/daninemcsik/I3I4BP_XMLGyak)

A feladatomat két külön mappába töltöttem fel, egyik az XMLTaskI3I4BP mappa, amibe az xml, xsd, képek és ez a jegyzőkönyv található a másik pedig a DOMParselI3I4BP, amibe a java dolgok találhatók, illetve az xml-ről egy másolat a könnyebb elérés érdekében.

## 1.feladat

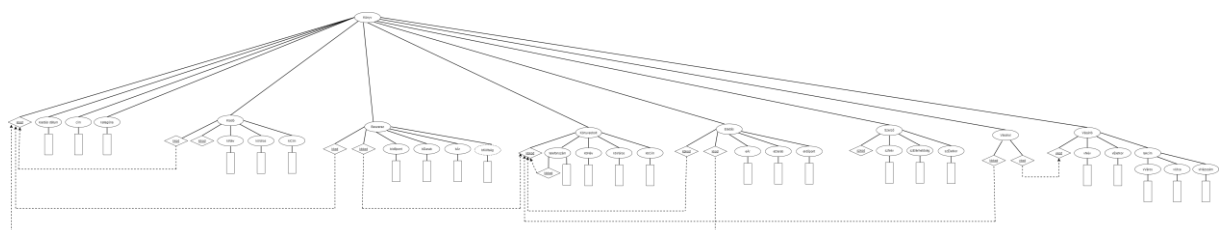
**1a)**

ER modell:



**1 b)**

XDM modell:



(Az XDM és ER modell github-on jobban megtekinthető.)

1c)

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <Könyvek xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="http://www.example.com/XMLSchemaI3I4BP.xsd">
3
4   <Könyv kkod="01">
5     <kiadási_dátum>03/15/2019</kiadási_dátum>
6     <cím>Senki sem az, akinek látszik</cím>
7     <kategória>Irodalom</kategória>
8
9     <Szerző szkod="01">
10      <szElérhetőség>305256314</szElérhetőség>
11      <szNév>Kerry Fisher</szNév>
12      <szÉletkor>60</szÉletkor>
13    </Szerző>
14
15    <Kiadó kikod="01">
16      <kkod>01</kkod>
17      <kiVáros>Szeged</kiVáros>
18      <kiCím>Szegedi utca 3.</kiCím>
19      <kiNév>Szegedi kiadó</kiNév>
20    </Kiadó>
21
22    <Könyvesbolt kbkod="01">
23      <kbVáros>Szeged</kbVáros>
24      <kbCím>Vitéz utca 30.</kbCím>
25      <kbNév>Szögödi könyvök</kbNév>
26    </Könyvesbolt>
27
28    <Beszerez>
29      <kkod>01</kkod>
30      <kbkod>01</kbkod>
31      <bIdőpont>20/02/2020</bIdőpont>
32      <bDarab>2</bDarab>
33      <bÁr>3400</bÁr>
34      <bKöltség>6800</bKöltség> <!-- bDarab * bÁr -->
35    </Beszerez>
36
37    <Telefonszám>
38      <kbkod>01</kbkod>
39      <telefonszam>704813192</telefonszam>
40    </Telefonszám>
41
42    <Eladás>
43      <kkod>01</kkod>
44      <kbkod>01</kbkod>
45      <eIdőpont>21/02/2020</eIdőpont>
46      <eDarab>2</eDarab>
47      <eÁr>4000</eÁr>
48    </Eladás>
49
50    <Vásárol>
51      <kbkod>01</kbkod>
52      <vkod>04</vkod>
53    </Vásárol>
54
55    <Vásárló vkod="04">
56      <vNév>Kovács Tímea</vNév>
57      <vÉletkor>25</vÉletkor>
58      <lakcím>
59        <vVáros>Szeged</vVáros>
60        <vUtca>Zoltán utca</vUtca>
61        <vHázszám>44</vHázszám>
62      </lakcím>
63    </Vásárló>
64  </Könyv>
65
66
67
68   <Könyv kkod="02">
69     <kiadási_dátum>02/10/2018</kiadási_dátum>
70     <cím>Nem titok, amit ketten tudnak</cím>
71     <kategória>Irodalom</kategória>
72
73     <Szerző szkod="02">
74      <szElérhetőség>708312943</szElérhetőség>
75      <szNév>Robin Cuddy</szNév>
76      <szÉletkor>42</szÉletkor>
77    </Szerző>
78
79    <Kiadó kikod="02">
80      <kkod>02</kkod>
81      <kiVáros>Budapest</kiVáros>
82      <kiCím>Pesti út 65.</kiCím>
83      <kiNév>Pesti kiadó</kiNév>
84    </Kiadó>
85
86    <Könyvesbolt kbkod="02">
87      <kbVáros>Budapest</kbVáros>
88      <kbCím>Szugló utca 40.</kbCím>
89      <kbNév>Book-A-Pest</kbNév>
90    </Könyvesbolt>
91
92    <Beszerez>
93      <kkod>02</kkod>
94      <kbkod>02</kbkod>
95      <bIdőpont>11/11/2019</bIdőpont>
96      <bDarab>5</bDarab>
97      <bÁr>2100</bÁr>
98      <bKöltség>10500</bKöltség> <!-- bDarab * bÁr -->
99    </Beszerez>
100
101    <Telefonszám>
102      <kbkod>02</kbkod>
103      <telefonszam>307318491</telefonszam>
104    </Telefonszám>
105
106    <Eladás>
107      <kkod>02</kkod>
108      <kbkod>02</kbkod>
109      <eIdőpont>11/12/2019</eIdőpont>
```

```
106 <Eladás>
107   <kkod>02</kkod>
108   <kbkod>02</kbkod>
109   <eIdőpont>11/12/2019</eIdőpont>
110   <eDarab>4</eDarab>
111   <eÁr>3000</eÁr>
112 </Eladás>
113
114 <Vásárol>
115   <kbkod>02</kbkod>
116   <vkod>01</vkod>
117 </Vásárol>
118
119 <Vásárló vkod="01">
120   <vNév>Nagy Ferenc</vNév>
121   <vÉletkor>42</vÉletkor>
122   <lakcím>
123     <vVáros>Budapest</vVáros>
124     <vUtca>Lovarda utca</vUtca>
125     <vHázszám>22</vHázszám>
126   </lakcím>
127 </Vásárló>
128 </Könyv>
129
130
131
132 <Könyv kkod="03">
133   <kiadási_dátum>01/11/2020</kiadási_dátum>
134   <cím>Tanuljunk rajzolni! - Szafari</cím>
135   <kategória>Hobbi</kategória>
136
137   <Szerző szkod="03">
138     <szElérhetőség>207318219</szElérhetőség>
139     <szNév>Ken Follett</szNév>
140     <szÉletkor>72</szÉletkor>
141   </Szerző>
142
143   <Kiadó kikod="03">
144     <kkod>03</kkod>
145     <kiVáros>Miskolc</kiVáros>
146     <kiCím>Miskolci utca 11.</kiCím>
147     <kiNév>Miskolci kiadó</kiNév>
148   </Kiadó>
149
150   <Könyvesbolt kbkod="03">
151     <kbVáros>Miskolci</kbVáros>
152     <kbCím>Széchenyi utca 20.</kbCím>
153     <kbNév>Libri Miskolc</kbNév>
154   </Könyvesbolt>
155
156   <Beszerez>
157     <kkod>03</kkod>
158     <kbkod>03</kbkod>
159     <bIdőpont>21/02/2021</bIdőpont>
160     <bDarab>5</bDarab>
```

```
156 <Beszerez>
157   <kkod>03</kkod>
158   <kbkod>03</kbkod>
159   <bIdőpont>21/02/2021</bIdőpont>
160   <bDarab>5</bDarab>
161   <bÁr>3000</bÁr>
162   <bKöltség>15000</bKöltség> <!-- bDarab * bÁr -->
163 </Beszerez>
164
165 <Telefonszám>
166   <kbkod>03</kbkod>
167   <telefonszam>207431975</telefonszam>
168 </Telefonszám>
169
170 <Eladás>
171   <kkod>03</kkod>
172   <kbkod>03</kbkod>
173   <eIdőpont>21/03/2021</eIdőpont>
174   <eDarab>3</eDarab>
175   <eÁr>4200</eÁr>
176 </Eladás>
177
178 <Vásárol>
179   <kbkod>03</kbkod>
180   <vkod>05</vkod>
181 </Vásárol>
182
183 <Vásárló vkod="05">
184   <vNév>Huszár Zsófia</vNév>
185   <vÉletkor>33</vÉletkor>
186   <lakcím>
187     <vVáros>Miskolc</vVáros>
188     <vUtca>Huba utca</vUtca>
189     <vHázszám>55</vHázszám>
190   </lakcím>
191 </Vásárló>
192 </Könyv>
193
194
195
196 <Könyv kkod="04">
197   <kiadási_dátum>05/03/2015</kiadási_dátum>
198   <cím>Tanuljunk rajzolni! - Házi kedvencek</cím>
199   <kategória>Hobbi</kategória>
200
201   <Szerző szkod="04">
202     <szElérhetőség>304891398</szElérhetőség>
203     <szNév>Fehér Krisztián</szNév>
204     <szÉletkor>45</szÉletkor>
205   </Szerző>
206
207   <Kiadó kikod="04">
208     <kkod>04</kkod>
209     <kiVáros>Eger</kiVáros>
210     <kiCím>Miskolci utca 11.</kiCím>
```

```
207 <Kiadó kikod="04">
208   <kkod>04</kkod>
209   <kiVáros>Eger</kiVáros>
210   <kiCím>Egri út 7.</kiCím>
211   <kiNév>Egri kiadó</kiNév>
212 </Kiadó>
213
214 <Könyvesbolt kbkod="04">
215   <kbVáros>Eger</kbVáros>
216   <kbCím>Sóház utca 11.</kbCím>
217   <kbNév>Lira Eger</kbNév>
218 </Könyvesbolt>
219
220 <Beszerez>
221   <kkod>04</kkod>
222   <kbkod>04</kbkod>
223   <bIdőpont>20/12/2016</bIdőpont>
224   <bDarab>4</bDarab>
225   <bÁr>1500</bÁr>
226   <bKöltség>6000</bKöltség> <!-- bDarab * bÁr -->
227 </Beszerez>
228
229 <Telefonszám>
230   <kbkod>04</kbkod>
231   <telefonszam>309184751</telefonszam>
232 </Telefonszám>
233
234 <Eladás>
235   <kkod>04</kkod>
236   <kbkod>04</kbkod>
237   <eIdőpont>12/12/2017</eIdőpont>
238   <eDarab>2</eDarab>
239   <eÁr>4500</eÁr>
240 </Eladás>
241
242 <Vásárol>
243   <kbkod>04</kbkod>
244   <vkod>03</vkod>
245 </Vásárol>
246
247 <Vásárló vkod="03">
248   <vNév>Fekete Roland</vNév>
249   <vÉletkor>20</vÉletkor>
250   <lakcím>
251     <vVáros>Eger</vVáros>
252     <vUtca>Gáz utca</vUtca>
253     <vHázszám>33</vHázszám>
254   </lakcím>
255 </Vásárló>
256 </Könyv>
257
258
259
260 <Könyv kkod="05">
261   <kiadási_dátum>08/09/2009</kiadási_dátum>
```

Design Source

```
260 <Könyv kkod="05">
261   <kiadási_dátum>08/09/2009</kiadási_dátum>
262   <cím>World Without End</cím>
263   <kategória>Történelem</kategória>
264
265   <Szerző szkod="05">
266     <szElérhetőség>703189418</szElérhetőség>
267     <szNév>Stephen King</szNév>
268     <szÉletkor>74</szÉletkor>
269   </Szerző>
270
271   <Kiadó kikod="05">
272     <kkod>05</kkod>
273     <kiVáros>Szombathely</kiVáros>
274     <kiCím>Szombathely utca 13.</kiCím>
275     <kiNév>Szombathelyi kiadó</kiNév>
276   </Kiadó>
277
278   <Könyvesbolt kbkod="05">
279     <kbVáros>Szombathely</kbVáros>
280     <kbCím>Forró utca 3.</kbCím>
281     <kbNév>Office Könyvesbolt</kbNév>
282   </Könyvesbolt>
283
284   <Beszerez>
285     <kkod>05</kkod>
286     <kbkod>05</kbkod>
287     <bIdőpont>11/11/2011</bIdőpont>
288     <bDarab>10</bDarab>
289     <bÁr>3400</bÁr>
290     <bKöltség>34000</bKöltség> <!-- bDarab * bÁr -->
291   </Beszerez>
292
293   <Telefonszám>
294     <kbkod>05</kbkod>
295     <telefonszam>701831951</telefonszam>
296   </Telefonszám>
297
298   <Eladás>
299     <kkod>05</kkod>
300     <kbkod>05</kbkod>
301     <eIdőpont>12/03/2012</eIdőpont>
302     <eDarab>9</eDarab>
303     <eÁr>3900</eÁr>
304   </Eladás>
305
306   <Vásárol>
307     <kbkod>05</kbkod>
308     <vkod>02</vkod>
309   </Vásárol>
310
311   <Vásárló vkod="02">
312     <vNév>Kis Tibor</vNév>
313     <vÉletkor>30</vÉletkor>
314     <lakcím>
```

Design Source

```
311 <Vásárló vkod="02">
312   <vNév>Kis Tibor</vNév>
313   <vÉletkor>30</vÉletkor>
314   <lakcím>
315     <vVáros>Szombathely</vVáros>
316     <vUtca>Ferencsik utca</vUtca>
317     <vHázszám>11</vHázszám>
318   </lakcím>
319 </Vásárló>
320 </Könyv>
321
322
323
324 <Könyv kkod="06">
325   <kiadási_dátum>10/09/2010</kiadási_dátum>
326   <cím>The Pillars of The Earth</cím>
327   <kategória>Történelem</kategória>
328
329   <Szerző szkod="01">
330     <szElérhetőség>305256314</szElérhetőség>
331     <szNév>Kerry Fisher</szNév>
332     <szÉletkor>60</szÉletkor>
333   </Szerző>
334
335   <Kiadó kikod="01">
336     <kkod>06</kkod>
337     <kiVáros>Szeged</kiVáros>
338     <kiCím>Szegedi utca 3.</kiCím>
339     <kiNév>Szegedi kiadó</kiNév>
340   </Kiadó>
341
342   <Könyvesbolt kbkod="01">
343     <kbVáros>Szeged</kbVáros>
344     <kbCím>Vitéz utca 30.</kbCím>
345     <kbNév>Szögödi könyvök</kbNév>
346   </Könyvesbolt>
347
348   <Beszerez>
349     <kkod>06</kkod>
350     <kbkod>01</kbkod>
351     <bIdőpont>20/09/2012</bIdőpont>
352     <bDarab>3</bDarab>
353     <bÁr>3400</bÁr>
354     <bKöltség>10200</bKöltség> <!-- bDarab * bÁr -->
355   </Beszerez>
356
357   <Telefonszám>
358     <kbkod>01</kbkod>
359     <telefonszam>704813192</telefonszam>
360   </Telefonszám>
361
362   <Eladás>
363     <kkod>06</kkod>
364     <kbkod>01</kbkod>
365     <bIdőpont>21/10/2013</bIdőpont>
```

Design	Source
--------	--------

```
362   </Eladás>
363   <kkod>06</kkod>
364   <kbkod>01</kbkod>
365   <eIdőpont>21/10/2013</eIdőpont>
366   <eDarab>2</eDarab>
367   <eÁr>4000</eÁr>
368 </Eladás>
369
370 <Vásárol>
371   <kbkod>01</kbkod>
372   <vkod>09</vkod>
373 </Vásárol>
374
375 <Vásárló vkod="09">
376   <vNév>Lakatos Noel</vNév>
377   <vÉletkor>50</vÉletkor>
378   <lakcím>
379     <vVáros>Szeged</vVáros>
380     <vUtca>Forró utca</vUtca>
381     <vHázszám>99</vHázszám>
382   </lakcím>
383 </Vásárló>
384 </Könyv>
385
386
387
388 <Könyv kkod="07">
389   <kiadási_dátum>03/11/2020</kiadási_dátum>
390   <cím>Programozz C nyelven</cím>
391   <kategória>Számítástechnika</kategória>
392
393   <Szerző szkod="02">
394     <szElérhetőség>708312943</szElérhetőség>
395     <szNév>Robin Cuddy</szNév>
396     <szÉletkor>42</szÉletkor>
397   </Szerző>
398
399   <Kiadó kikod="02">
400     <kkod>07</kkod>
401     <kiVáros>Budapest</kiVáros>
402     <kiCím>Pesti út 65.</kiCím>
403     <kiNév>Pesti kiadó</kiNév>
404   </Kiadó>
405
406   <Könyvesbolt kbkod="02">
407     <kbVáros>Budapest</kbVáros>
408     <kbCím>Szugló utca 40.</kbCím>
409     <kbNév>Book-A-Pest</kbNév>
410   </Könyvesbolt>
411
412   <Beszerez>
413     <kkod>07</kkod>
414     <kbkod>02</kbkod>
415     <bIdőpont>20/11/2021</bIdőpont>
416     <bDarab>10</bDarab>
```

Design	Source
--------	--------

```
412 <Beszerez>
413   <kkod>07</kkod>
414   <kbkod>02</kbkod>
415   <bIdőpont>20/11/2021</bIdőpont>
416   <bDarab>10</bDarab>
417   <bÁr>2000</bÁr>
418   <bKöltség>20000</bKöltség> <!-- bDarab * bÁr -->
419 </Beszerez>
420
421 <Telefonszám>
422   <kbkod>02</kbkod>
423   <telefonszam>307318491</telefonszam>
424 </Telefonszám>
425
426 <Eladás>
427   <kkod>07</kkod>
428   <kbkod>02</kbkod>
429   <eIdőpont>22/11/2021</eIdőpont>
430   <eDarab>8</eDarab>
431   <eÁr>3100</eÁr>
432 </Eladás>
433
434 <Vásárol>
435   <kbkod>02</kbkod>
436   <vkod>10</vkod>
437 </Vásárol>
438
439 <Vásárló vkod="10">
440   <vNév>Bognár Éva</vNév>
441   <vÉletkor>66</vÉletkor>
442   <lakcím>
443     <vVáros>Budapest</vVáros>
444     <vUtca>Zöld utca</vUtca>
445     <vHázszám>111</vHázszám>
446   </lakcím>
447 </Vásárló>
448 </Könyv>
449
450
451
452 <Könyv kkod="08">
453   <kiadási_dátum>09/11/2020</kiadási_dátum>
454   <cím>Számítógépvásárlási túlélőkönyv</cím>
455   <kategória>Számítástechnika</kategória>
456
457   <Szerző szkod="03">
458     <szElérhetőség>207318219</szElérhetőség>
459     <szNév>Ken Follett</szNév>
460     <szÉletkor>72</szÉletkor>
461   </Szerző>
462
463   <Kiadó kikod="03">
464     <kkod>08</kkod>
465     <kiVáros>Miskolc</kiVáros>
466     <kiCím>Miskolci utca 11.</kiCím>
```

Design	Source
--------	--------

```
463 <Kiadó kikod="03">
464   <kkod>08</kkod>
465   <kiVáros>Miskolc</kiVáros>
466   <kiCím>Miskolci utca 11.</kiCím>
467   <kiNév>Miskolci kiadó</kiNév>
468 </Kiadó>
469
470 <Könyvesbolt kbkod="06">
471   <kbVáros>Miskolc</kbVáros>
472   <kbCím>Iroda utca 4.</kbCím>
473   <kbNév>Office Könyvesbolt</kbNév>
474 </Könyvesbolt>
475
476 <Beszerez>
477   <kkod>08</kkod>
478   <kbkod>06</kbkod>
479   <bIdőpont>11/11/2020</bIdőpont>
480   <bDarab>1</bDarab>
481   <bÁr>5600</bÁr>
482   <bKöltség>5600</bKöltség> <!-- bDarab * bÁr -->
483 </Beszerez>
484
485 <Telefonszám>
486   <kbkod>06</kbkod>
487   <telefonszam>301829231</telefonszam>
488 </Telefonszám>
489
490 <Eladás>
491   <kkod>08</kkod>
492   <kbkod>06</kbkod>
493   <eIdőpont>11/12/2020</eIdőpont>
494   <eDarab>1</eDarab>
495   <eÁr>10000</eÁr>
496 </Eladás>
497
498 <Vásárol>
499   <kbkod>06</kbkod>
500   <vkod>08</vkod>
501 </Vásárol>
502
503 <Vásárló vkod="08">
504   <vNév>Mezei Gábor</vNév>
505   <vÉletkor>39</vÉletkor>
506   <lakcím>
507     <vVáros>Miskolc</vVáros>
508     <vUtca>Gyula utca</vUtca>
509     <vHázszám>88</vHázszám>
510   </lakcím>
511 </Vásárló>
512 </Könyv>
513
514
515
516 <Könyv kkod="09">
517   <kiadási_dátum>11/11/2011</kiadási_dátum>
```

Design	Source
--------	--------



```
516 <Könyv kkod="09">
517   <kiadási_dátum>11/11/2011</kiadási_dátum>
518   <cím>The Shining</cím>
519   <kategória>Horror</kategória>
520
521   <Szerző szkod="04">
522     <szElérhetőség>304891398</szElérhetőség>
523     <szNév>Fehér Krisztián</szNév>
524     <szÉletkor>45</szÉletkor>
525   </Szerző>
526
527   <Kiadó kikod="04">
528     <kkod>09</kkod>
529     <kiVáros>Eger</kiVáros>
530     <kiCím>Egri út 7.</kiCím>
531     <kiNév>Egri kiadó</kiNév>
532   </Kiadó>
533
534   <Könyvesbolt kbkod="04">
535     <kbVáros>Eger</kbVáros>
536     <kbCím>Sóház utca 11.</kbCím>
537     <kbNév>Lira Eger</kbNév>
538   </Könyvesbolt>
539
540   <Beszerez>
541     <kkod>09</kkod>
542     <kbkod>04</kbkod>
543     <bIdőpont>12/12/2012</bIdőpont>
544     <bDarab>3</bDarab>
545     <bÁr>4500</bÁr>
546     <bKöltség>13500</bKöltség> <!-- bDarab * bÁr
547   </Beszerez>
548
549   <Telefonszám>
550     <kbkod>04</kbkod>
551     <telefonszam>309184751</telefonszam>
552   </Telefonszám>
553
554   <Eladás>
555     <kkod>09</kkod>
556     <kbkod>04</kbkod>
557     <eIdőpont>20/12/2013</eIdőpont>
558     <eDarab>3</eDarab>
559     <eÁr>7000</eÁr>
560   </Eladás>
561
562   <Vásárol>
563     <kbkod>04</kbkod>
564     <vkod>06</vkod>
565   </Vásárol>
566
567   <Vásárló vkod="06">
568     <vNév>Kiss Levente</vNév>
569     <vÉletkor>18</vÉletkor>
```

```
567 <Vásárló vkod="06">
568   <vNév>Kiss Levente</vNév>
569   <vÉletkor>18</vÉletkor>
570   <lakcím>
571     <vVáros>Eger</vVáros>
572     <vUtca>Fényes utca</vUtca>
573     <vHázszám>66</vHázszám>
574   </lakcím>
575 </Vásárló>
576 </Könyv>
577
578
579
580 <Könyv kkod="10">
581   <kiadási_dátum>10/01/2018</kiadási_dátum>
582   <cím>It</cím>
583   <kategória>Horror</kategória>
584
585   <Szerző szkod="05">
586     <szElérhetőség>703189418</szElérhetőség>
587     <szNév>Stephen King</szNév>
588     <szÉletkor>74</szÉletkor>
589   </Szerző>
590
591   <Kiadó kikod="05">
592     <kkod>10</kkod>
593     <kiVáros>Szombathely</kiVáros>
594     <kiCím>Szombathely utca 13.</kiCím>
595     <kiNév>Szombathelyi kiadó</kiNév>
596   </Kiadó>
597
598   <Könyvesbolt kbkod="07">
599     <kbVáros>Szombathely</kbVáros>
600     <kbCím>Vezér utca 6.</kbCím>
601     <kbNév>Office Könyvesbolt</kbNév>
602   </Könyvesbolt>
603
604   <Beszerez>
605     <kkod>10</kkod>
606     <kbkod>07</kbkod>
607     <bIdőpont>12/08/2019</bIdőpont>
608     <bDarab>4</bDarab>
609     <bÁr>5500</bÁr>
610     <bKöltség>22000</bKöltség> <!-- bDarab * bÁr -->
611   </Beszerez>
612
613   <Telefonszám>
614     <kbkod>07</kbkod>
615     <telefonszam>208194851</telefonszam>
616   </Telefonszám>
617
618   <Eladás>
619     <kkod>10</kkod>
620     <kbkod>07</kbkod>
621     <eIdőpont>20/09/2019</eIdőpont>
622     <eDarab>4</eDarab>
623     <eÁr>8200</eÁr>
624   </Eladás>
625
626   <Vásárol>
627     <kbkod>07</kbkod>
628     <vkod>07</vkod>
629   </Vásárol>
630
631   <Vásárló vkod="07">
632     <vNév>Barna Virág</vNév>
633     <vÉletkor>55</vÉletkor>
634     <lakcím>
635       <vVáros>Szombathely</vVáros>
636       <vUtca>Szepes utca</vUtca>
637       <vHázszám>77</vHázszám>
638     </lakcím>
639   </Vásárló>
640 </Könyv>
641 </Könyvek>
```



1d)

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" targetNamespace="http://www.example.org/XMLSchemaI3I4BP" xmlns:tns="http://www.example.org/XMLSchemaI3I4BP" elementFormDefault="qualified">
3
4   <xs:element name="Könyvek">
5     <xs:complexType>
6       <xs:sequence>
7         <xs:element name="Könyv" maxOccurs="unbounded">
8           <xs:complexType>
9             <xs:sequence>
10               <xs:element name="kiadási_dátum" type="xs:string" />
11               <xs:element name="cím" type="xs:string" />
12               <xs:element name="kategória" type="xs:string" />
13
14               <xs:element name="Szerző">
15                 <xs:complexType>
16                   <xs:sequence>
17                     <xs:element name="szélerhetőség" type="xs:unsignedInt" />
18                     <xs:element name="sznév" type="xs:string" />
19                     <xs:element name="széletkor" type="xs:unsignedInt" />
20                   </xs:sequence>
21                   <xs:attribute name="szkod" type="xs:unsignedInt" use="required" />
22                 </xs:complexType>
23               </xs:element>
24
25               <xs:element name="Kiadó">
26                 <xs:complexType>
27                   <xs:sequence>
28                     <xs:element name="kkod" type="xs:unsignedInt"/>
29                     <xs:element name="klváros" type="xs:string" />
30                     <xs:element name="klcím" type="xs:string" />
31                     <xs:element name="klnév" type="xs:string" />
32                   </xs:sequence>
33                   <xs:attribute name="kkod" type="xs:unsignedInt" use="required" />
34                 </xs:complexType>
35               </xs:element>
36
37               <xs:element name="Könyvesbolt">
38                 <xs:complexType>
39                   <xs:sequence>
40                     <xs:element name="bbváros" type="xs:string" />
41                     <xs:element name="bbcím" type="xs:string" />
42                     <xs:element name="bbnév" type="xs:string" />
43                   </xs:sequence>
44                   <xs:attribute name="bbkod" type="xs:unsignedInt" use="required"/>
45                 </xs:complexType>
46               </xs:element>
47
48               <xs:element name="Beszerez">
49                 <xs:complexType>
50                   <xs:sequence>
51                     <xs:element name="kkod" type="xs:unsignedInt"/>
52                     <xs:element name="bbkod" type="xs:unsignedInt"/>
53                     <xs:element name="elidőpont" type="xs:string"/>
54                     <xs:element name="bdorab" type="xs:unsignedInt"/>
55                     <xs:element name="bár" type="xs:unsignedInt"/>
56                     <xs:element name="bköltség" type="xs:int"/>
57                   </xs:sequence>
58                 </xs:complexType>
59               </xs:element>
60             </xs:sequence>
61           </xs:complexType>
62         </xs:element>
63       </xs:sequence>
64     </xs:complexType>
65   </xs:element>
66
67   <xs:element name="Telefonszám">
68     <xs:complexType>
69       <xs:sequence>
70         <xs:element name="bbkod" type="xs:unsignedInt"/>
71         <xs:element name="telefonszám" type="xs:unsignedInt"/>
72       </xs:sequence>
73     </xs:complexType>
74   </xs:element>
75
76   <xs:element name="Eladás">
77     <xs:complexType>
78       <xs:sequence>
79         <xs:element name="kkod" type="xs:unsignedInt"/>
80         <xs:element name="bbkod" type="xs:unsignedInt"/>
81         <xs:element name="elidőpont" type="xs:string"/>
82         <xs:element name="bdorab" type="xs:unsignedInt"/>
83         <xs:element name="eár" type="xs:unsignedInt"/>
84       </xs:sequence>
85     </xs:complexType>
86   </xs:element>
87
88   <xs:element name="Vásárol">
89     <xs:complexType>
90       <xs:sequence>
91         <xs:element name="bbkod" type="xs:unsignedInt"/>
92         <xs:element name="vkod" type="xs:unsignedInt"/>
93       </xs:sequence>
94     </xs:complexType>
95   </xs:element>
96
97   <xs:element name="Vásárló">
98     <xs:complexType>
99       <xs:sequence>
100         <xs:element name="vVév" type="xs:string" />
101         <xs:element name="vVéletkor" type="xs:unsignedInt" />
102         <xs:element name="vLokcím">
103           <xs:complexType>
104             <xs:sequence>
105               <xs:element name="vVáros" type="xs:string" />
106               <xs:element name="vUtca" type="xs:string" />
107               <xs:element name="vHázszám" type="xs:unsignedInt" />
108             </xs:sequence>
109           </xs:complexType>
110         </xs:element>
111         <xs:attribute name="vkod" type="xs:unsignedInt" use="required" />
112       </xs:sequence>
113     </xs:complexType>
114   </xs:element>
115
116   <xs:element name="Könyv">
117     <xs:complexType>
118       <xs:sequence>
119         <xs:element name="kiadási_dátum" type="xs:string" />
120         <xs:element name="cím" type="xs:string" />
121         <xs:element name="kategória" type="xs:string" />
122       </xs:sequence>
123     </xs:complexType>
124   </xs:element>
125
126   </xs:schema>
```

```
61 <xs:element name="Telefonszám">
62   <xs:complexType>
63     <xs:sequence>
64       <xs:element name="bbkod" type="xs:unsignedInt"/>
65       <xs:element name="telefonszám" type="xs:unsignedInt"/>
66     </xs:sequence>
67   </xs:complexType>
68 </xs:element>
69
70 <xs:element name="Eladás">
71   <xs:complexType>
72     <xs:sequence>
73       <xs:element name="kkod" type="xs:unsignedInt"/>
74       <xs:element name="bbkod" type="xs:unsignedInt"/>
75       <xs:element name="elidőpont" type="xs:string"/>
76       <xs:element name="bdorab" type="xs:unsignedInt"/>
77       <xs:element name="eár" type="xs:unsignedInt"/>
78     </xs:sequence>
79   </xs:complexType>
80 </xs:element>
81
82 <xs:element name="Vásárol">
83   <xs:complexType>
84     <xs:sequence>
85       <xs:element name="bbkod" type="xs:unsignedInt"/>
86       <xs:element name="vkod" type="xs:unsignedInt"/>
87     </xs:sequence>
88   </xs:complexType>
89 </xs:element>
90
91 <xs:element name="Vásárló">
92   <xs:complexType>
93     <xs:sequence>
94       <xs:element name="vVév" type="xs:string" />
95       <xs:element name="vVéletkor" type="xs:unsignedInt" />
96       <xs:element name="vLokcím">
97         <xs:complexType>
98           <xs:sequence>
99             <xs:element name="vVáros" type="xs:string" />
100             <xs:element name="vUtca" type="xs:string" />
101             <xs:element name="vHázszám" type="xs:unsignedInt" />
102           </xs:sequence>
103         </xs:complexType>
104       </xs:element>
105       <xs:attribute name="vkod" type="xs:unsignedInt" use="required" />
106     </xs:sequence>
107   </xs:complexType>
108 </xs:element>
109
110 <xs:element name="Könyv">
111   <xs:complexType>
112     <xs:sequence>
113       <xs:element name="kiadási_dátum" type="xs:string" />
114       <xs:element name="cím" type="xs:string" />
115       <xs:element name="kategória" type="xs:string" />
116     </xs:sequence>
117   </xs:complexType>
118 </xs:element>
119
120 </xs:schema>
```

## 2.feladat

A kommentek a kódon belül találhatóak

2a)

Adatolvasás: - DOMReadI3I4BP.java

```
1 package hu.domparse.i3i4bp;
2
3 import org.w3c.dom.Document;
4
5
6
7
8
9
10
11
12
13
14
15
16
17 public class DOMReadI3I4BP {
18
19     public static void main(String[] args) throws IOException, ParserConfigurationException, SAXException{
20         DocumentBuilderFactory dbf = DocumentBuilderFactory.newInstance();
21
22         try {
23
24             DocumentBuilder db = dbf.newDocumentBuilder();
25             Document doc = (Document) db.parse(new File("src\\XMLI3I4BP.xml"));
26             doc.getDocumentElement().normalize();
27
28             System.out.println("Gyökérelem : " + doc.getDocumentElement().getNodeName()); //gyökérelem meghatározása
29             System.out.println();
30
31             NodeList könyv = doc.getElementsByTagName("Könyv"); //Könyv elemek elérése
32             for(int i = 0; i < könyv.getLength(); i++) {
33                 Node node = könyv.item(i);
34                 if(node.getNodeType() == Node.ELEMENT_NODE) {
35                     Element element = (Element) node;
36
37                     String id = element.getAttribute("kkod");
38
39                     String datum = element.getElementsByTagName("kiadási dátum").item(0).getTextContent(); //Könyv elem belső tulajdonságainak elérése
40                     String cim = element.getElementsByTagName("cim").item(0).getTextContent();
41                     String kategoria = element.getElementsByTagName("kategória").item(0).getTextContent();
42
43                     System.out.println("Jelenlegi elem : " + node.getNodeName()); //Ezek kiírása
44                     System.out.println("Könyv kód : " + id);
45                     System.out.println("Dátum : " + datum);
46                     System.out.println("Cím : " + cim);
47                     System.out.println("Kategória : " + kategoria);
48                     System.out.println();
49
50                     NodeList szerzo = doc.getElementsByTagName("Szerző"); //Szerző elem elérése
51                     Node Sznod = szerzo.item(i);
52                     if(Sznod.getNodeType() == Node.ELEMENT_NODE) {
53                         Element Szelement = (Element) Sznod;
54
55                         String sid = Szelement.getAttribute("szkod");
56                         String szElerhetoseg = Szelement.getElementsByTagName("szElérhetőség").item(0).getTextContent(); //Szerző elem belső
57                         String szNev = Szelement.getElementsByTagName("szNév").item(0).getTextContent(); //tulajdonságainak elérése
58                         String szEletkor = Szelement.getElementsByTagName("szEletkor").item(0).getTextContent();
59
60                         System.out.println("Szerző kódja : " + sid);
61                         System.out.println("Szerző elérhetősége : " + szElerhetoseg); //Ezek kiírása
62                         System.out.println("Szerző neve : " + szNev);
63                         System.out.println("Szerző életkora : " + szEletkor);
64                         System.out.println();
65
66                     }
67
68                     NodeList kiado = doc.getElementsByTagName("Kiadó"); //Kiadó elem elérése
69                     Node Knode = kiado.item(i);
70                     if(Knode.getNodeType() == Node.ELEMENT_NODE) {
71                         Element Kelement = (Element) Knode;
72
73                         String kikod = Kelement.getAttribute("kikod");
74                         String kkod = Kelement.getElementsByTagName("kkod").item(0).getTextContent();
75                         String kiVaros = Kelement.getElementsByTagName("kiVáros").item(0).getTextContent(); //Kiadó elem belső tulajdonságainak elérése
76                         String kiCim = Kelement.getElementsByTagName("kiCím").item(0).getTextContent();
77                         String kiNev = Kelement.getElementsByTagName("kiNév").item(0).getTextContent();
78
79                         System.out.println("Kiadó kódja : " + kikod);
80                         System.out.println("Kiadott könyv kódja : " + kkod);
81                         System.out.println("Kiadó városa : " + kiVaros); //Ezek kiírása
82                         System.out.println("Kiadó címe : " + kiCim);
83                         System.out.println("Kiadó neve : " + kiNev);
84                         System.out.println();
85
86                     }
87
88                     NodeList kbolt = doc.getElementsByTagName("Könyvesbolt"); //Könyvesbolt elem elérése
89                     Node Kbnod = kbolt.item(i);
90                     if(Kbnod.getNodeType() == Node.ELEMENT_NODE) {
91                         Element KBelement = (Element) Kbnod;
92
93                         String kbkod = KBelement.getAttribute("kbkod");
94                         String kbVaros = KBelement.getElementsByTagName("kbVáros").item(0).getTextContent(); //Könyvesbolt elem belső
95                         String kbCim = KBelement.getElementsByTagName("kbCím").item(0).getTextContent(); //tulajdonságainak elérése
96                         String kbNev = KBelement.getElementsByTagName("kbNév").item(0).getTextContent();
97
98                         System.out.println("Könyvesbolt kódja : " + kbkod);
99                         System.out.println("Könyvesbolt városa : " + kbVaros);
100                         System.out.println("Könyvesbolt címe : " + kbCim); //Ezek kiírása
101                         System.out.println("Könyvesbolt neve : " + kbNev);
102                         System.out.println();
103
104                     }
105
106                 }
107             }
108         }
109     }
110 }
```

```

105 NodeList tszam = doc.getElementsByTagName("Telefonszám"); //Telefonszám elem elérése
106 Node tsznode = tszam.item(i);
107 if(tsznode.getNodeType() == Node.ELEMENT_NODE) {
108     Element TSzelement = (Element) tsznode;
109
110     String kbkod = TSzelement.getElementsByTagName("kbkod").item(0).getTextContent(); //Telefonszám elem belső
111     String telefonszam = TSzelement.getElementsByTagName("telefonszam").item(0).getTextContent(); //tulajdonságainak elérése
112
113     System.out.println("Könyvesbolt kódja : " + kbkod); //Ezek kiírása
114     System.out.println("Könyvesbolt telefonszám : " + telefonszam);
115     System.out.println();
116 }
117
118
119 NodeList beszerez = doc.getElementsByTagName("Beszerez"); //Beszerez kapcsolat-elem elérése
120 Node Besznode = beszerez.item(i);
121 if(Besznode.getNodeType() == Node.ELEMENT_NODE) {
122     Element Beszelement = (Element) Besznode;
123
124     String kkod = Beszelement.getElementsByTagName("kkod").item(0).getTextContent();
125     String kbkod = Beszelement.getElementsByTagName("kbkod").item(0).getTextContent();
126     String bIdőpont = Beszelement.getElementsByTagName("bIdőpont").item(0).getTextContent(); //Beszerez kapcsolat-elem belső
127     String bDarab = Beszelement.getElementsByTagName("bDarab").item(0).getTextContent(); //tulajdonságainak elérése
128     String bár = Beszelement.getElementsByTagName("bár").item(0).getTextContent();
129     String bKöltség = Beszelement.getElementsByTagName("bKöltség").item(0).getTextContent();
130
131     System.out.println("Beszerzett könyv kódja : " + kkod);
132     System.out.println("Beszerző könyvesbolt kódja : " + kbkod);
133     System.out.println("Beszerzés időpontja : " + bIdőpont); //Ezek kiírása
134     System.out.println("Beszerzett darabszám : " + bDarab);
135     System.out.println("Beszerzési ár : " + bár);
136     System.out.println("Beszerzési össz költség : " + bKöltség);
137     System.out.println();
138 }
139
140 NodeList eladás = doc.getElementsByTagName("Eladás"); //Eladás kapcsolat-elem elérése
141 Node Eladnode = eladás.item(i);
142 if(Eladnode.getNodeType() == Node.ELEMENT_NODE) {
143     Element Eladelement = (Element) Eladnode;
144
145     String kkod = Eladelement.getElementsByTagName("kkod").item(0).getTextContent();
146     String kbkod = Eladelement.getElementsByTagName("kbkod").item(0).getTextContent();
147     String eIdőpont = Eladelement.getElementsByTagName("eIdőpont").item(0).getTextContent(); //Eladás kapcsolat-elem belső
148     String eDarab = Eladelement.getElementsByTagName("eDarab").item(0).getTextContent(); //tulajdonságainak elérése
149     String eÁr = Eladelement.getElementsByTagName("eÁr").item(0).getTextContent();
150
151     System.out.println("Eladott könyv kódja : " + kkod);
152     System.out.println("Eladó könyvesbolt kódja : " + kbkod);
153     System.out.println("Eladás időpontja : " + eIdőpont); //Ezek kiírása
154     System.out.println("Eladott darabszám : " + eDarab);
155     System.out.println("Eladási ár : " + eÁr);
156     System.out.println();
157 }
158
159 NodeList vasarol = doc.getElementsByTagName("Vásárol"); //Vásárol kapcsolat-elem elérése
160 Node vsnode = vasarol.item(i);
161 if(vsnode.getNodeType() == Node.ELEMENT_NODE) {
162     Element VSelement = (Element) vsnode;
163
164     String kbkod = VSelement.getElementsByTagName("kbkod").item(0).getTextContent(); //Vásárol kapcsolat-elem belső
165     String vkod = VSelement.getElementsByTagName("vkod").item(0).getTextContent(); //tulajdonságainak elérése
166
167     System.out.println("Könyvesbolt kódja : " + kbkod); //Ezek kiírása
168     System.out.println("Vásárló kódja : " + vkod);
169     System.out.println();
170 }
171
172
173
174 NodeList vasarlo = doc.getElementsByTagName("Vásárló"); //Vásárló elem elérése
175 Node Vnode = vasarlo.item(i);
176 if(Vnode.getNodeType() == Node.ELEMENT_NODE) {
177     Element Velement = (Element) Vnode;
178
179     String vNév = Velement.getElementsByTagName("vNév").item(0).getTextContent(); //Vásárló elem belső tulajdonságainak elérése
180     String vÉletkor = Velement.getElementsByTagName("vÉletkor").item(0).getTextContent();
181
182     System.out.println("Vásárló neve : " + vNév);
183     System.out.println("Vásárló életkora : " + vÉletkor); //Vásárló kiírása
184
185     NodeList lakcim = doc.getElementsByTagName("lakcim"); //Vásárló elem lakcím összetett elemének elérése
186     Node lnode = lakcim.item(i);
187     if(lnode.getNodeType() == Node.ELEMENT_NODE) {
188         Element lelement = (Element) lnode;
189
190         String vVaros = Velement.getElementsByTagName("vVáros").item(0).getTextContent(); //lakcím elem belső tulajdonságainak elérése
191         String vUtca = Velement.getElementsByTagName("vUtca").item(0).getTextContent();
192         String vHazszam = Velement.getElementsByTagName("vHázzsám").item(0).getTextContent();
193
194         System.out.println("Vásárló városa : " + vVaros);
195         System.out.println("Vásárló utcája : " + vUtca);
196         System.out.println("Vásárló házzsáma : " + vHazzsam); //Ezek kiírása
197         System.out.println();
198     }
199 }
200
201 System.out.println("-----");
202 }
203 }
204 }
205
206 }catch(Exception e) {
207     e.printStackTrace();
208 }
209 }
210 }
211

```

Eredmény:

Mind a tíz könyv adatait így írja ki.

```
Jelenlegi elem :Könyv
Könyv kód : 10
Dátum : 10/01/2018
Cím : It
Kategória : Horror

Szerző kódja : 05
Szerző elérhetősége : 703189418
Szerző neve : Stephen King
Szerző életkora : 74

Kiadó kódja : 05
Kiadott könyv kódja : 10
Kiadó városa : Szombathely
Kiadó címe : Szombathely utca 13.
Kiadó neve : Szombathelyi kiadó

Könyvesbolt kódja : 07
Könyvesbolt városa : Szombathely
Könyvesbolt címe : Vezér utca 6.
Könyvesbolt neve : Office Könyvesbolt

Könyvesbolt kódja : 07
Könyvesbolt telefonszám : 208194851

Beszerzett könyv kódja : 10
Beszerző könyvesbolt kódja : 07
Beszerzés időpontja : 12/08/2019
Beszerzett darabszám : 4
Beszerzési ár : 5500
Beszerzési össz költség : 22000

Eladott könyv kódja : 10
Eladó könyvesbolt kódja : 07
Eladás időpontja : 20/09/2019
Eladott darabszám : 4
Eladási ár : 8200

Könyvesbolt kódja : 07
Vásárló kódja : 07

Vásárló neve : Barna Virág
Vásárló életkora : 55
Vásárló városa : Szombathely
Vásárló utcája : Szepes utca
Vásárló házszáma : 77
```

2b)

## Adatmódosítás: - DOMModifyI3I4BP.java

```
1 package hu.domparsing.i3i4bp;
2
3 import java.io.File;
4
5 public class DOMModifyI3I4BP {
6
7     public static void main(String[] args) throws IOException, ParserConfigurationException, SAXException {
8         DocumentBuilderFactory dbf = DocumentBuilderFactory.newInstance();
9
10        try {
11            DocumentBuilder db = dbf.newDocumentBuilder();
12            Document doc = (Document) db.parse(new File("src\\XML\\I3I4BP.xml"));
13            Node könyv = doc.getElementsByTagName("Könyv").item(0); //első könyv elem elérése
14
15            NamedNodeMap attr = könyv.getAttributes(); //első könyv elem tulajdonságának lekérdezése
16            Node nodeAttr = attr.getNamedItem("kkod"); //kkod tulajdonság megkeresése
17            nodeAttr.setTextContent("444"); //tulajdonság értékének megváltoztatása
18
19            NodeList list = könyv.getChildNodes(); //könyv elem gyerekelemeinek elérése
20
21            for(int i = 0; i < list.getLength(); i++) { //iteráció elemek között
22                Node node = list.item(i);
23                if(node.getNodeType() == Node.ELEMENT_NODE) {
24                    Element eElement = (Element) node;
25                    if("cím".equals(eElement.getNodeName())) { //cím megegyezik-e az éppen vizsgált elem névvel? Ha igen akkor...
26                        if("Senki sem az, akinek látszik".equals(eElement.getTextContent())) { //...megnézi, hogy a cím elem tartalma is
27                            //megegyezik-e az éppen vizsgált elem tartalmával? Ha igen...
28                            eElement.setTextContent("Mindenki az, akinek látszik"); //...akkor megváltoztatja erre.
29                        }
30                    }
31                    if("kategória".equals(eElement.getNodeName())) { //kategória megegyezik-e az éppen vizsgált elem névvel? Ha igen akkor...
32                        if("Irodalom".equals(eElement.getTextContent())) { //...megnézi, hogy a kategória elem tartalma is
33                            //megegyezik-e az éppen vizsgált elem tartalmával? Ha igen...
34                            eElement.setTextContent("Történelem"); //...akkor megváltoztatja erre.
35                        }
36                    }
37                }
38            }
39
40            TransformerFactory transformerFactory = TransformerFactory.newInstance();
41            Transformer transformer = transformerFactory.newTransformer(); //transformer létrehozása, ami lehetővé teszi majd az eredmény különféle kiíratását
42            DOMSource source = new DOMSource(doc); //transzformációs 'source tree' számára egy tároló DOM fáként
43            transformer.setOutputProperty(OutputKeys.ENCODING, "UTF-8"); //elvitelnek a két sornak kéne a kimenetet utf-8 karakterkészlettel
44            transformer.setOutputProperty(OutputKeys.OMIT_XML_DECLARATION, "yes"); //de egyébként ki lehet olvasni, hogy a megfelelő dolgok kerültek változtatásra,
45            //csak az ékezetes betűk helyett random karakterek szerepelnek
46
47            StreamResult consoleResult = new StreamResult(System.out); //transzformáció eredményének tárolója, xml, html, sima szöveg, vagy egyéb módon
48            transformer.transform(source, consoleResult); //a forrást (source) belerakja az eredmény tárolóba (consoleResult)
49
50        } catch (Exception e) { //hibakezelés, ha csak Exception-t adunk meg, akkor minden hibát kezel, ami nem saját hiba, mivel ez az összes hiba ebből ered.
51            e.printStackTrace();
52        }
53    }
54 }
55
56 }
```

Kiíráskor UTF-8-at nem támogat, pedig a két sor transformer kód arra lenne elvileg.

Eredmény:

```
<Könyvek xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xs:noNamespaceSchemaLocation="...">
  <Könyv kkod="444">
    <kiadás dátum>03/15/2019</kiadás dátum>
    <cím>Mindenki az, akinek látszik</cím>
    <kategória>Történelem</kategória>
    <Szerző szkod="01">
      <szelőorhető száma>305256314</szelőorhető száma>
      <szelőnév>Kerry Fisher</szelőnév>
      <szelőletkor>60</szelőletkor>
    </Szerző>
    <kiadás kód>01</kiadás kód>
    <kiadás>Szeged</kiadás>
    <kiadás>Szegedi utca 3.</kiadás>
    <kiadás>Szegedi kiadás</kiadás>
  </Könyv>
  <Könyvesbolt kbkod="01">
    <kbvásáros>Szeged</kbvásáros>
    <kbvásáros>Vita utca 30.</kbvásáros>
    <kbvásáros>Szállásdi könyv</kbvásáros>
  </Könyvesbolt>
</Könyvek>
```

Ennek ellenére a változtatások látszanak, csak nem olvashatóak annyira, de a program azt csinálja, amit kell neki.

A könyvkód 444 lett, a cím "Mindenki az, akinek látszik" és a kategória irodalomról történelemre változott.

2c)

## Adatlekérdezés: - DOMQueryI3I4BP.java

```
1 package hu.domparsa.i3i4bp;
2
3 import java.io.File;
4
5 public class DOMQueryI3I4BP {
6     public static void main(String[] args) throws IOException, ParserConfigurationException, SAXException{
7         DocumentBuilderFactory dbf = DocumentBuilderFactory.newInstance();
8
9         try {
10             DocumentBuilder db = dbf.newDocumentBuilder();
11             Document doc = (Document) db.parse(new File("src\\XMLI3I4BP.xml"));
12             doc.getDocumentElement().normalize();
13
14             System.out.println("Könyvek címe, amik irodalom kategóriába sorolhatók:");
15             ListLiteratureBookNames(doc);
16             System.out.println("-----\n"); //kiírás
17             ListKerryFisherBookBuyersNamesAndAges(doc);
18             System.out.println("-----\n");
19             ListShopPhonenumberAddressAndName(doc);
20             System.out.println("-----\n");
21             ListPublisherAndWriterNames(doc);
22             System.out.println("-----\n");
23         } catch (Exception e) {
24             e.printStackTrace();
25         }
26     }
27
28     //40-83 sor Szerző, Kiadó, Könyvesbolt, Vásárló elem elérése kódismétlés elkerülése érdekében.
29     static Element getSzerzoElement(Document doc, int i) {
30         NodeList nodeList = doc.getElementsByTagName("Szerző");
31         Node node = nodeList.item(i);
32         if (node.getNodeType() == Node.ELEMENT_NODE) {
33             Element element = (Element) node;
34             return element;
35         } else {
36             return null;
37         }
38     }
39
40     static Element getKiadoElement(Document doc, int i) {
41         NodeList nodeList = doc.getElementsByTagName("Kiadó");
42         Node node = nodeList.item(i);
43         if (node.getNodeType() == Node.ELEMENT_NODE) {
44             Element element = (Element) node;
45             return element;
46         } else {
47             return null;
48         }
49     }
50
51     static Element getKonyvesboltElement(Document doc, int i) {
52         NodeList nodeList = doc.getElementsByTagName("Könyvesbolt");
53         Node node = nodeList.item(i);
54         if (node.getNodeType() == Node.ELEMENT_NODE) {
55             Element element = (Element) node;
56             return element;
57         } else {
58             return null;
59         }
60     }
61
62     static Element getVasarloElement(Document doc, int i) {
63         NodeList nodeList = doc.getElementsByTagName("Vásárló");
64         Node node = nodeList.item(i);
65         if (node.getNodeType() == Node.ELEMENT_NODE) {
66             Element element = (Element) node;
67             return element;
68         } else {
69             return null;
70         }
71     }
72
73     static void listLiteratureBookNames(Document doc) { //Irodalom kategóriájú könyvek nevének lekérdezése.
74         NodeList konyv = doc.getElementsByTagName("Könyv");
75         for (int i = 0; i < konyv.getLength(); i++) {
76             Node node = konyv.item(i);
77             if (node.getNodeType() == Node.ELEMENT_NODE) {
78                 Element element = (Element) node;
79                 //kiírás
80                 if ((element.getElementsByTagName("kategória").item(0).getTextContent()).equalsIgnoreCase("Irodalom")) {
81                     System.out.println(" " + element.getElementsByTagName("cím").item(0).getTextContent());
82                 }
83             }
84         }
85     }
86
87     static void listKerryFisherBookBuyersNamesAndAges(Document doc) { //Kerry Fisher könyveket vásárló emberek nevének és korának lekérdezése.
88         NodeList konyv = doc.getElementsByTagName("Könyv");
89         System.out.println("Kerry Fisher könyveit ők vették meg:");
90         for (int i = 0; i < konyv.getLength(); i++) {
91             Node node = konyv.item(i);
92             if (node.getNodeType() == Node.ELEMENT_NODE) {
93                 Element szerzo = getSzerzoElement(doc, i);
94                 Element vasarlo = getVasarloElement(doc, i);
95                 //kiírás
96                 if ((szerzo.getElementsByTagName("szNév").item(0).getTextContent()).equalsIgnoreCase("Kerry Fisher")) {
97                     System.out.println(" Vásárló neve: " + vasarlo.getElementsByTagName("vNév").item(0).getTextContent()
98                         + ", " + vasarlo.getElementsByTagName("vÉletkor").item(0).getTextContent() + " éves.");
99                 }
100             }
101         }
102     }
103
104 }
```



```

118 static void listShopPhonenumberAddressAndName(Document doc) { //Lekérdezés, hogy Ken Follett melyik könyveit hol adták el, ezek a helyek adatai.
119     NodeList könyv = doc.getElementsByTagName("Könyv");
120     System.out.println("Ken Follett alábbi könyveit adták el: ");
121     for(int i = 0; i < könyv.getLength(); i++) {
122         Node node = könyv.item(i);
123         if(node.getNodeType() == Node.ELEMENT_NODE) {
124             Element kbolt = getKönyvesboltElement(doc, i);
125             Element szerzo = getSzerzoElement(doc,i);
126             Element element =(Element) node;
127             //kiíratás
128             if((szerzo.getElementsByTagName("szNév").item(0).getTextContent()).equalsIgnoreCase("Ken Follett")){
129                 System.out.println("      Könyv címe: " + element.getElementsByTagName("cím").item(0).getTextContent() + "\nAz alábbi könyvesboltba:");
130                 System.out.println("      Neve: " + kbolt.getElementsByTagName("kbNév").item(0).getTextContent() + "\n      Címe: " +
131                     kbolt.getElementsByTagName("kbVáros").item(0).getTextContent() + ", " + kbolt.getElementsByTagName("kbCím").item(0).getTextContent()
132                     + "\n      Telefonszáma: " + kbolt.getAttribute("telefonszam"));
133                 System.out.println();
134             }
135         }
136     }
137 }
138 }
139 }
140 }
141 }
142 static boolean isInList(ArrayList<String> a, String str) { //Megnézi, hogy egy név benne van-e már az arraylist-ben, ismétlődés elkerülése érdekében.
143     boolean state = false; //Ez csak azért szükséges bele, mert úgy írtam meg az xml dokumentumot, hogy egy szerzőhöz egy
144                               //kiadó tartozik, tehát nem lehet 2 különböző kiadó egy szerzőhöz.
145     for(int i = 0; i < a.size(); i++) { //Így 5 sort ír ki, szerző neve és a hozzá vett kiadó. Ha nem lenne ez benne, akkor mindent 2x írna ki.
146         if(a.get(i).equalsIgnoreCase(str)) { //Ha egy szerző több kiadóhoz is tartozna, akkor másképp kéne csinálni.
147             state = true;
148             break;
149         }else {
150             state = false;
151         }
152     }
153     return state;
154 }
155 }
156 static void listPublisherAndWriterNames(Document doc) { //Lekérdezés, ami megadja, hogy melyik író melyik kiadónál publikálta könyveit.
157     NodeList könyv = doc.getElementsByTagName("Könyv");
158     System.out.println("Melyik író melyik kiadónál publikálta könyveit:");
159     ArrayList<String> publisherList = new ArrayList<String>();
160     for(int i = 0; i < könyv.getLength(); i++) {
161         Node node = könyv.item(i);
162         if(node.getNodeType() == Node.ELEMENT_NODE) {
163             Element kiado = getKiadoElement(doc, i);
164             Element szerzo = getSzerzoElement(doc,i);
165             String sznev = szerzo.getElementsByTagName("szNév").item(0).getTextContent();
166             if(isInList(publisherList, sznev) == false) {
167                 publisherList.add(sznev);
168                 System.out.println("      " + sznev + " a " +
169                     kiado.getElementsByTagName("kiNév").item(0).getTextContent() + " kiadónál publikálta könyveit."); //kiíratás
170             }
171         }
172     }
173 }
174 }
175 }
176 }
177 }

```

Eredmény:

```

Könyvek címe, amik irodalm kategóriába sorolhatók:
    Senki sem az, akinek látszik
    Nem titok, amit ketten tudnak
-----

Kerry Fisher könyveit ők vették meg:
    Vásárló neve: Kovács Tímea, 25 éves.
    Vásárló neve: Lakatos Noel, 50 éves.
-----

Ken Follett alábbi könyveit adták el:
    Könyv címe: Tanuljunk rajzolni! - Szafari
Az alábbi könyvesboltba:
    Neve: Libri Miskolc
    Címe: Miskolci, Széchenyi utca 20.
    Telefonszáma:

    Könyv címe: Számítógépvásárlási túlélőkönyv
Az alábbi könyvesboltba:
    Neve: Office Könyvesbolt
    Címe: Miskolc, Iroda utca 4.
    Telefonszáma:
-----

Melyik író melyik kiadónál publikálta könyveit:
    Kerry Fisher a Szegedi kiadó kiadónál publikálta könyveit.
    Robin Cuddy a Pesti kiadó kiadónál publikálta könyveit.
    Ken Follett a Miskolci kiadó kiadónál publikálta könyveit.
    Fehér Krisztián a Egri kiadó kiadónál publikálta könyveit.
    Stephen King a Szombathelyi kiadó kiadónál publikálta könyveit.
-----

```