

Mérési jegyzőkönyv – Adatbázisok Laboratórium

5. mérés: XML

Név:	Szabó Bence Farkas
Neptun kód:	RF57V5
Feladat kódja:	26 - SZALL
Mérésvezető neve:	Bőjti Bence
Mérés időpontja:	2017-04-20 16:15
Mérés helyszíne:	R4K
<i>A működő alkalmazás elérhetősége:</i>	<i>rapid.eik.bme.hu/~rf57v5/xml/</i>
Megoldott feladatok:	1,2,3,4
Elérhető pontszám (plusz pontok nélkül):	28

Mérési feladatok megoldása

1. feladat

Megrendelések kiírása

Magyarázat

Az első feladatban a megrendelések adatait kellett kiírni XSL transzformációval XML formátumba. Ezt a list.xml állományban valósítottam meg. A feladat megoldásához jó alapul szolgált a kiadott példaprogram, melyen nem is kellett sokat módosítani. Eleinte problémák voltak a dátum megjelenítésével, de erre a segédletben találtam megoldást. A dátum két attribútumot tartalmaz, ezek a date és time. Mivel nekünk csak a date-re van szükségünk, a sablonban azt kell kiválsztani. A dátum formátumát nem kellett módosítani, mivel az már helyesen volt tárolva a bemeneti xml-ben. A megoldáshoz a mintaprogramhoz hasonlóan több XSL sablont használtam, melyek rekurzívan hajtják végre a transzformációt. A vezérlő script a kimenetet a list.html állományba teszi. Ez az alábbi kimenetet adja:

Megnevezés	Mennyiség	Határidő
bútor	10000	2017-05-10
építőanyag	8000	2017-04-11
építőanyag	20000	2017-03-21
élelmiszer <tartós>	15000	2017-04-12
bútoralapanyag	30000	2017-02-22
élelmiszer <ital>	20000	2017-05-01
műszaki cikk	20000	2017-02-10
papíráru, <nyers>	34000	2017-03-08
vegyi áru (festék&hígító)	12000	2017-04-10
üzemanyag	16000	2017-03-16
üzemanyag	42000	2017-03-12
baromfi	16000	2017-04-12
versenyló	4000	2017-02-28
sértés	20000	2017-04-13
használt autó	10	2017-04-24
hatóságilag lefoglalt járművek	15	2017-03-12

2. *feladat*

Adatok szűrése

Magyarázat

Ebben a feladatban a 19000-nél nagyobb mennyiségű megrendelések adatait kellett kiírni XML file-ba. Ehhez a példaprogramban található két sablon egészítettem ki úgy, hogy az első sablon kiválasztja a bemeneti XML állományból az orders csomópontokat, majd a második lemásolja azokat és a gyerekeiket. Itt a segédletben is megemlített **xsl:copy-of** elemet felhasználva, annak select attribútumában megadtam a feltételt. Ennek használatát az interneten talált anyagokból fejtettem meg¹. A megoldás a filter.xsl állományban található, a kimenet a fenti link /filter.xml kiegészítésén található, melyből lentebb látható egy részlet.

```
1 <szall element-type="dataset">
2 <orders element-type="recordset">
3 <record element-type="record" order_id="2017/000003">
4 <description is-null="False">építőanyag</description>
5 <vehicle_type is-null="False">D</vehicle_type>
6 <quantity is-null="False">20000</quantity>
7 <origin is-null="False">Milánó (IT)</origin>
8 <destination is-null="False">Szeged (HU)</destination>
9 <order_date is-null="False" date="2017-01-14" time="00:00:00"/>
10 <deadline_date is-null="False" date="2017-03-21" time="12:00:00"/>
11 <comment_text is-null="True"/>
12 </record>
13 <record element-type="record" order_id="2017/000005">
14 <description is-null="False">bútoralapanyag</description>
15 <vehicle_type is-null="False">D</vehicle_type>
16 <quantity is-null="False">30000</quantity>
17 <origin is-null="False">Varsó (PL)</origin>
```

stb...

¹ https://www.w3schools.com/xml/ref_xsl_el_copy-of.asp

3. feladat

Az index.html lap módosítása

Magyarázat

A feladat során az index.html oldal tartalmát módosítjuk a megrendelések adataival a list.xsl állományt felhasználva. Ebben a feladatban is a példaprogramot írtam át illetve egészítettem ki. Ahhoz, hogy a megfelelő adatokat kinyerhessük, a sablon gyökérelemének az „//orders” Xpath kifejezést használtam, mely minden orders csomópontra illeszkedik, így minden megrendelés adatait kinyerhetjük. Ezek után már csak egy táblázatot és annak fejlécét kellett kialakítani, illetve meghívni a list.xsl sablonjait, melyek feltöltik a táblázat sorait. A táblázatnak a szebb és áttekinthetőbb megjelenés érdekében egy szegélyt is adtam. A stíluslap a feldolgozott adatokat html formátumban írja ki a kimenetre, melyet a vezérlés az index.html állományba tesz. Ez a fenti link megnyitásával tekinthető meg (illetve a lenti ábrán is látható a kimenet).

RENDELÉSEK

Megnevezés	Mennyiség	Határidő
bútor	10000	2017-05-10
építőanyag	8000	2017-04-11
építőanyag	20000	2017-03-21
élelmiszer <tartós>	15000	2017-04-12
bútoralapanyag	30000	2017-02-22
élelmiszer <ital>	20000	2017-05-01
műszaki cikk	20000	2017-02-10
papíráru, <nyers>	34000	2017-03-08
vegyi áru (festék&hígító)	12000	2017-04-10
üzemanyag	16000	2017-03-16
üzemanyag	42000	2017-03-12
baromfi	16000	2017-04-12
versenyló	4000	2017-02-28
sertés	20000	2017-04-13
használt autó	10	2017-04-24
hatóságilag lefoglalt járművek	15	2017-03-12

4. feladat

Táblák összekapcsolása.

Magyarázat

A feladatban a megrendelések listázásában hozzá kellett csatolni a megrendeléshez tartozó fuvarokat. A megoldáshoz szinté jó alapul szolgált a kiadott példaprogram, melyet átalakítottam a feladatnak megfelelően. Először az első sablonban változtattam meg a sablon paramétereit, hogy az orders tábla adatait dolgozza fel. Ezek után a megrendelés adatainak kiírása következett. Itt a legnagyobb nehézséget a shipments tábla adatainak elérése jelentette. Ezt a laborvezetőtől kapott útmutatás segítségével tudtam megoldani, miszerint egy változóban tároljuk az aktuális megrendelés azonosítóját, majd ez alapján alkalmazzuk a következő sablont.

```
<xsl:variable name="my_id" select="@order_id" />
<xsl:apply-templates select="//shipments/record[order_id=$my_id]" />
```

Itt a shipments record elemeit akkor dolgozzuk fel, ha a shipment-ben tárolt order_id azonosító megegyezik az változónkban tárolt értékkel.

Ezek után a szállítmány adatainak listázása nem volt nehéz, itt egy harmadik sablont kellett megírni, hasonlóan az előzőhöz.

A vezérlő skript a kimenetet a join.xml állományba teszi, mely a fenti link /join.xml kiegészítésén található, illetve lentebb látható belőle egy részlet.

```
<orders xmlns:functx="http://www.functx.com"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <order id="2017/000001">
    <description>bútor</description>
    <quantity>10000</quantity>
    <deadline_date>2017-05-10</deadline_date>
    <shipments>
      <shipment vehicle_id="0" arrival_date="2017-01-06"/>
      <shipment vehicle_id="0" arrival_date="2017-01-07"/>
    </shipments>
  </order>
  <order id="2017/000002">
    <description>építőanyag</description>
    <quantity>8000</quantity>
    <deadline_date>2017-04-11</deadline_date>
    <shipments>
      <shipment vehicle_id="0" arrival_date="2017-01-09"/>
    </shipments>
  </order>
  <order id="2017/000003">
    <description>építőanyag</description>
    <quantity>20000</quantity>
    <deadline_date>2017-03-21</deadline_date>
    <shipments>
      <shipment vehicle_id="2" arrival_date="2017-01-15"/>
      <shipment vehicle_id="2" arrival_date="2017-01-16"/>
    </shipments>
  </order>
</orders>
```