

Mérési jegyzőkönyv – Adatbázisok Laboratórium

V. mérés: XML

Név:	Szabó Bence Farkas
Neptun kód:	RF57V5
Feladat kódja:	30 – VASUT
Mérésvezető neve:	Csapó Tamás
Mérés időpontja:	2018-04-23 12:15
Mérés helyszíne:	HSZK N
Megoldott feladatok:	1,2,3,4
Elérhető pontszám (plusz pontok nélkül):	28p

Mérési feladatok megoldása

1. *feladat*

Az első feladatban egy stíluslapot kellett készíteni, mely kiírja az összes állomás adatait (név, város, átlagos utasszám, kapacitás sztrájk esetén) egy html táblázatba.

Magyarázat

A feladat megoldásához a list.xsl file tartalmát kellett módosítani. Itt a mintaprogram alapul véve, átalakítva a templateket juthatunk el a megoldáshoz.

Az első templatet, mely a „/” elemre (gyöker elemre) illeszkedik, csak az apply-template utasításnál kellett módosítani, hogy a „**vasut/allomas**” elemekre illeszkedjen.

A második template-ben, mely az „**allomas**” elemre illeszkedik, létrehozzuk a táblázatot, annak fejlécét, majd a feltöltéshez tovább megyünk a gyerek csomópontokra.

A harmadik template-ben az állomás „record” elemein megyünk végig, ahol a feladatnak megfelelően kiválasztjuk a megfelelő attribútumokat, melyek feldolgozását a mintaprogram feldolgozó template-jei végzik.

Tesztelés menete

A feladat teszteléséhez nyissuk meg a böngészőben a <http://rapid.eik.bme.hu/~rf57v5/xml/list.html> url-t. Az eredményen látható, hogy csak a táblázat kerül kiírásra.

Név	Város	Átlagos utasszám	Kapacitás sztrájk esetén
Budapest-Keleti	Budapest	9000	2000
Budapest-Nyugati	Budapest	9000	2000
Budapest-Déli	Budapest	9000	2000
Budapest-Kelenföld	Budapest	9000	2000
Budapest-Ferencváros	Budapest	9000	2000
Törökbálint	Törökbálint	6000	1200
Biatorbágy	Biatorbágy	6000	1200
Bicske	Bicske	6000	2000
Tatabánya	Tatabánya	9000	3000
Vértesszőlős	Vértesszőlős	1000	200
Tata	Tata	9000	2000
Almásfüzitő	Almásfüzitő	2000	400
Komárom	Komárom	9000	1800
Ács	Ács	1500	300
Győr	Győr	9000	2000
Hegyeshalom	Hegyeshalom	1700	500
Wien-Südbahnhof	Bécs	9000	2400
Enese	Enese	1000	200
Eternitgyár	Lábatlan	2000	400
Csorna	Csorna	1000	200
Kapuvár	Kapuvár	1000	200
Fertőszentmiklós	Fertőszentmiklós	1000	200
Balf	Balf	1000	200
Sopron	Sopron	9000	2000
Dunakeszi	Dunakeszi	3000	600
Göd	Göd	2500	1000
Szödliget	Szödliget	1500	300
Vác	Vác	3000	600
Verőcse	Verőcse	2000	500
Nagymaros	Nagymaros	1500	300
Zebegény	Zebegény	1000	200
Szob	Szob	2300	1000
Debrecen	Debrecen	9000	4000

2. feladat

A második feladatban a filter.xsl stíluslapot kellett megvalósítani, melyben XML formátumban kilistázzuk azokat az állomásokat, melyek Budapesten találhatók (város attribútumuk Budapest).

Magyarázat

A feladat megoldásához szintén jó alapul szolgált a kiadott mintaprogram.

Ebben az első template-t módosítjuk úgy, hogy a „**vasut**” elemre illeszkedjen. Itt a copy paranccsal lemásoljuk a csomópontot a kimenetre. A forrás formátumának megtartása érdekében a „**vasut**” csomópontnak kell egy „**element-type**” attribútum, amely a „**dataset**” értéket kapja. Ehhez a template-ben az xsl:attribute paranccsal hozzacsatoljuk a „**vasut**” elemhez az „**element-type**” attribútumot, melynek értékét az xsl:value-of paranccsal kapjuk meg. Ezek után folytatjuk a feldolgozást az „**allomas**” csomóponton.

A második template-ben az „**allomas**” csomópontokat dolgozzuk fel. Hasonlóan az előzőhöz, itt is hozzacsatoljuk az „**element-type**” attribútumot a csomópontához, majd az xsl:copy-of paranccsal lemásoljuk azokat a „**record**” típusú attribútumokat, melyek „**varos**” gyermeke a „**Budapest**” értéket tartalmazza (másra nincs is szükség, mert a copy-of megtartja az eredeti fastruktúrát a kimeneten).

Tesztelés menete

Az eredmény ellenőrzéséhez nyissuk meg a <http://rapid.eik.bme.hu/~rf57v5/xml/filter.xml> url-t. A kimeneten láthatjuk a kész XML struktúrát.

```
▼<vasut element-type="dataset">
  ▼<allomas element-type="recordset">
    ▼<record element-type="record" id="1">
      <nev is-null="False">Budapest-Keleti</nev>
      <varos is-null="False">Budapest</varos>
      <atlagutas is-null="False">9000</atlagutas>
      <sztrajkutas is-null="False">2000</sztrajkutas>
    </record>
    ▼<record element-type="record" id="2">
      <nev is-null="False">Budapest-Nyugati</nev>
      <varos is-null="False">Budapest</varos>
      <atlagutas is-null="False">9000</atlagutas>
      <sztrajkutas is-null="False">2000</sztrajkutas>
    </record>
    ▼<record element-type="record" id="3">
      <nev is-null="False">Budapest-Déli</nev>
      <varos is-null="False">Budapest</varos>
      <atlagutas is-null="False">9000</atlagutas>
      <sztrajkutas is-null="False">2000</sztrajkutas>
    </record>
    ▼<record element-type="record" id="4">
      <nev is-null="False">Budapest-Kelenföld</nev>
      <varos is-null="False">Budapest</varos>
      <atlagutas is-null="False">9000</atlagutas>
      <sztrajkutas is-null="False">2000</sztrajkutas>
    </record>
    ▼<record element-type="record" id="5">
      <nev is-null="False">Budapest-Ferencváros</nev>
      <varos is-null="False">Budapest</varos>
      <atlagutas is-null="False">9000</atlagutas>
      <sztrajkutas is-null="False">2000</sztrajkutas>
    </record>
  </allomas>
</vasut>
```

3. *feladat*

Ebben a feladatban az index.xml stíluslapot kellett megvalósítani, mely a list.xml stíluslapot felhasználva, önálló html dokumentumként jeleníti meg az állomásokat táblázatos formában.

Magyarázat

A megoldáshoz módosítani kellett az index.xml-ben található template-et úgy, hogy felüldefiniáljuk a list.xml root elemre illeszkedő template-jét. Ezt a `match="//allomas"` attribútum beállításával tehetjük meg.

A következő lépés a táblázat kiírása. Ehhez meg kell hívnunk a list.xml egyik template-jét. Ezt módosítanunk kell úgy, hogy hivatkozni tudjunk rá (tehát a list.xml-ben a második template-et, mely az „**allomas**” csomópontokra illeszkedik, ellátjuk egy **name** attribútummal). Ezek után az `xsl:call-template` paranccsal meghívjuk a list.xml megfelelő template-jét, amely majd beszúrja a táblázatot.

Tesztelés menete

A megoldást a <http://rapid.eik.bme.hu/~rf57v5/xml/index.html> url-en érhetjük el. Itt látható, hogy az index.xml-ben kialakított html oldalba sikerült beszúrni az állomásokat tartalmazó táblázatot:

Allomasok

Név	Város	Átlagos utasszám	Kapacitás sztrájk esetén
Budapest-Keleti	Budapest	9000	2000
Budapest-Nyugati	Budapest	9000	2000
Budapest-Déli	Budapest	9000	2000
Budapest-Kelenföld	Budapest	9000	2000
Budapest-Ferencváros	Budapest	9000	2000
Törökbálint	Törökbálint	6000	1200
Biatorbágy	Biatorbágy	6000	1200
Bicske	Bicske	6000	2000
Tatabánya	Tatabánya	9000	3000
Vértesszőlős	Vértesszőlős	1000	200
Tata	Tata	9000	2000
Almásfüzitő	Almásfüzitő	2000	400
Komárom	Komárom	9000	1800
Ács	Ács	1500	300
Győr	Győr	9000	2000
Hegyeshalom	Hegyeshalom	1700	500
Wien-Südbahnhof	Bécs	9000	2400
Enese	Enese	1000	200
Eternitgyár	Lábatlan	2000	400
Csorna	Csorna	1000	200
Kapuvár	Kapuvár	1000	200
Fertőszentmiklós	Fertőszentmiklós	1000	200
Balf	Balf	1000	200
Sopron	Sopron	9000	2000
Dunakeszi	Dunakeszi	3000	600
Göd	Göd	2500	1000
Szödliget	Szödliget	1500	300
Vác	Vác	3000	600
Verőcsmaros	Verőcsmaros	2000	500

4. feladat

A negyedik feladatban a join.xsl stílusnapot kellett megvalósítani, mely kiírja az állomások listáját és az állomásokhoz tartozó megállások adatait XML formátumban. Az állomás attribútumként tartalmazza saját, mesterséges kulcsát, gyerek csomópontként pedig a név, város, átlagutas mezőket, illetve a vonatok gyerekben szerepelnek az itt megálló vonat adatai melyek a vonatszám, érkezés és indulás idejét tartalmazzák attribútumként (ha valamely időpont null, mert az állomás kezdő-, vagy végállomás, azt nem kell kiírni).

Magyarázat

A megoldáshoz a példaprogram join.xsl stíluslap template-jei módosítására és kiegészítésére volt szükség. Az első template-ben az illeszkedést biztosító match attribútumot kellett a „/vasut” értékre módosítani, illetve az xsl:apply-templates hívást, hogy a „allomas/record” csomópontokon folytassa a feldolgozást.

A második template-ben a mintának megfelelően feltöltjük az állomás adatait. Az állomás id-ját attribútumként, a többi adatot gyerek csomópontként adjuk hozzá. A vonatok lekérdezéséhez létrehozunk egy xsl:variable változót, mely az állomás id-ját tartalmazza. Ez után az xsl:apply-templates utasítással megkeressük a „megall” csomópontok közül azokat, melyeknek „allomas_id” gyerekcsomópontjuk megegyezik az állomásunk id-jával:

```
<xsl:apply-templates
select="/vasut/megall[@element-type='recordset']/record[allomas_id=$all_id]
"/>
```

A harmadik template-ben (módosítva azt, hogy a „megall/record” csomópontokra illeszkedjen) kell létrehozni a vonat adatait. Ehhez egy „vonat” csomópontot hozunk létre, mely az adatokat attribútumok formájában tartalmazza. Az első attribútum a **vonatszám**, melyet az xsl:value-of parancs segítségével kapunk meg.

Ahhoz, hogy az indulás és érkezés ideje csak akkor jelenjen meg, ha létezik is további template-ekre van szükség. Ezekre nagyon jó példa a list.xsl feldolgozó template-jei, melyek figyelik, hogy az adott csomópont értéke null e. A kiíratáshoz az xsl:apply-templates hívást használva folytatjuk a feldolgozást az „erk”, illetve „ind” csomópontokon.

Az érkezés és indulás feldolgozása ugyan azon a módon történik:

```
<xsl:template match="record/erk[@is-null='True']"
priority="0.8"></xsl:template>
```

```
<xsl:template match="record/erk" priority="0.4">
  <xsl:attribute name="erk">
    <xsl:value-of select="." />
  </xsl:attribute>
</xsl:template>
```

Itt az első template-ben megnézzük, hogy az adott csomópont null e, ha igen végrehajtjuk a parancsokat (esetünkben nem csinálunk semmit). Ha nem igaz a feltétel (azaz nem null), haladunk tovább a következő template-re, ahol létrehozunk egy attribútumot és értéket adunk neki. A megoldás ugyan az az indulás illetve érkezés feldolgozásánál is, így összesen 4 db új template-et kellett létrehozni.

Tesztelés menete

A feladat kimenete a <http://rapid.eik.bme.hu/~rf57v5/xml/join.xml> url-en érhető el, ahol látható a felépített XML struktúra:

```
▼<allomas-lista xmlns:funcfx="http://www.funcfx.com" xm
  ▼<allomas id="1">
    <nev>Budapest-Keleti</nev>
    <varos>Budapest</varos>
    <atlagutas>9000</atlagutas>
    ▼<vonatok>
      <vonat vonatszam="213" ind="900"/>
      <vonat vonatszam="214" erk="1650"/>
      <vonat vonatszam="327" ind="1115"/>
      <vonat vonatszam="328" erk="1835"/>
      <vonat vonatszam="469" ind="1115"/>
      <vonat vonatszam="470" erk="1835"/>
      <vonat vonatszam="4581" ind="1315"/>
      <vonat vonatszam="6666" ind="2010"/>
    </vonatok>
  </allomas>
  ▼<allomas id="2">
    <nev>Budapest-Nyugati</nev>
    <varos>Budapest</varos>
    <atlagutas>9000</atlagutas>
    ▼<vonatok>
      <vonat vonatszam="9567" ind="1120"/>
      <vonat vonatszam="2113" ind="1920"/>
    </vonatok>
  </allomas>
  ▼<allomas id="3">
    <nev>Budapest-Déli</nev>
    <varos>Budapest</varos>
    <atlagutas>9000</atlagutas>
    ▼<vonatok>
      <vonat vonatszam="163" ind="1225"/>
      <vonat vonatszam="164" erk="1925"/>
      <vonat vonatszam="553" ind="725"/>
      <vonat vonatszam="554" erk="1425"/>
      <vonat vonatszam="1145" ind="920"/>
      <vonat vonatszam="1146" erk="2100"/>
      <vonat vonatszam="2265" ind="930"/>
      <vonat vonatszam="2266" erk="1605"/>
    </vonatok>
  </allomas>
  ▼<allomas id="4">
    <nev>Budapest-Kelenföld</nev>
    <varos>Budapest</varos>
    <atlagutas>9000</atlagutas>
    ▼<vonatok>
      <vonat vonatszam="213" erk="915" ind="918"/>
      <vonat vonatszam="214" erk="1630" ind="1635"/>
      <vonat vonatszam="327" erk="1130" ind="1137"/>
```
