

## Jegyzőkönyv XSQL mérésről

A mérést végezte:	Páli Márton
Neptun kód:	Z2PWV9
Mérésvezető neve:	Kiss Máté Levente
Mérés időpontja:	2016.05.10. 10:15
Mérés helyszíne:	HSZK B
Kapott feladatsor kódja:	REND-28
Mellékelt forrás:	Z2PWV9-5-K101.zip/Z2PWV9-5-K101/Z2PWV9-5-K101-src.zip
On-line változat elérhető:	<a href="http://rapid.eik.bme.hu:9080/~z2pwv9/index.xsql">http://rapid.eik.bme.hu:9080/~z2pwv9/index.xsql</a>
A megoldott feladatok:	1,2,3,4
Elérhető pontszám (pluszpontok nélkül):	22

## Kötelező feladatok

### 1. Feladat

Az első feladat megoldásához a segédletben található kódrészletekből (XSQL és XSLT résznél) merítettem ihletet, az nagyon jó kiindulópontot adott a feladat megoldásához. Nem is kellett rajta sokat változtatni, csupán a kimenet formátumán. Nekem ebben a feladatban a nehézséget a dátum kezelése okozta, először nem tudtam, hogy hogyan módosítsam az SQL DateTime attribútumomat úgy, hogy csak a dátum jelenjen meg belőle, majd rátaláltam a segédprogramban az xsl:query date-format attribútumára, és ott megcsináltam a módosítást. A másik nehézséget a márka és az eszköznév nagybetűssé tévése okozta, erre csak az interneten találtam megoldást, a megfelelő helyen a forrást is megjelöltem.

A mellékelt forrásban a **feladat\_01/index.xsql** és a **feladat\_01/rend.xml** fájlok vonatkoznak erre a feladatra.

### 2. Feladat

A második feladathoz létrehoztam a bad.xsql fájlt, valamint kicsit módosítottam a rend.xml-t. A

bad.xsql fájl alaphól rosszul van megírva, hogy a tesztelés során látszódjon, jelzi-e a program a hibát. Ez annyiban nyilvánul meg, hogy az SQL lekérdezésben FROM helyett („elgépelve”) FOM van, amely hibát okoz a lekérdezés kiértékelésekor. Ettől az „elgépeléstől” eltekintve a bad.xsql megegyezik az 1. feladatban elkészített index.xsql-lel.

A rend.xsl-en a módosításokat a kiadott példaprogram alapján csináltam: felvettem egy elágazást, amely hiba esetén kiírja a hibaüzenetet, egyébként pedig megjeleníti a táblázatot.

A mellékelt forrásban a **feladat\_02/bad.xsql** és a **feladat\_02/rend.xsl** fájlok vonatkoznak erre a feladatra.

### **3. Feladat**

A harmadik feladathoz csupán a rend.xsl fájlt kellett módosítani. Ehhez a feladatrészhez nem a példaprogramot, hanem a segédletet, illetve a második feladatot hívtam segítségül. A táblázat sorait kiegészítettem egy elágazással, ahol megtörténik az évszám ellenőrzése, valamint a cellákat kiegészítettem az „Akción ár” és a „Megtakarítás” részekkel, amelyeket piros színnel emeltem ki. Az eredeti árat a „strike” tag-gel húztam át. Az adatok 0.95-szörösét valamint 0.9-szeresét úgy írtam ki, hogy az „xsl:value-of” tag „select” attribútumát módosítottam úgy, hogy a megfelelő számmal beszoroztam az adatbázisból kiolvasott értéket.

A mellékelt forrásban csak a **feladat\_03/rend.xsl** fájl vonatkozik erre a feladatra.

*Megjegyzés a feladathoz:* A tesztelés során a „10 évnél idősebb” eszközök helyett csak az 5 évnél idősebbeket néztem, mivel az „eszközök” tábla tartalmaz egy olyan kényszert, hogy az eszköz vásárlásának dátuma nem lehet régebbi 2008.01.01.-nél, így 10 évnél idősebb eszköz még bő másfél évig nem is szerepelhet az adatbázisban.

### **4. Feladat**

A negyedik feladatot a példaprogram alapján kezdtem el megírni, szétbontottam a rend.xsl fájlt több xsl:template-re, hogy a megoldás átláthatóbb legyen. Az első xsl:template a „search\_field”, amely a feladat szövege alapján egy text alapú beviteli mezőt tartalmaz, valamint egy legördülő menüt, amelyből kiválasztható, hogy mire szeretne a felhasználó keresni. A keresési mező neve „search-value” és az értéke az index.xsql fájl „search-value” nevű page-param-jának értékét tartalmazza. A legördülő menü neve „search-key” és az értéke az a 3 attribútum lehet, amelyekre keresni lehet. A második xsl:template, amellyel most frissítettem az oldalt, a „result\_field”, ahol az eredménytáblázat jelenik meg. Ez megegyezik az előző feladatokban használt táblázattal.

A feladat megoldása során még módosítottam az index.xsql fájlt is, mivel az SQL lekérdezést paraméteressé kellett változtatnom. Ezt is a példaprogram alapján írtam meg, felvettem a page-param-okat, azokat a mezőket, amelyekre keresni lehet: nev, típus, vasarlas. Ezeknek a kezdetben

„%” az értéke, hogy a másik paraméterre való keresésnél ezek ne okozzanak gondot. A paraméter az értékét közvetlenül a lekérdezés lefuttatása előtt kapja meg, és csak az a paraméter kap új értéket, amely a legördülő menüből ki lett választva.

A mellékelt forrásban a **feladat\_04/index.xsql** és a **feladat\_04/rend.xsl** fájlok vonatkoznak erre a feladatra.

## Vélemény a mérésről

Alapvetően tetszett a mérés, kicsit összetett volt, de szerintem a kiadott példaprogram rengeteget segített a feladatok megoldásában.

Gondolom, itt lehet átfogó értékelést is adni a tárgyról: szerintem hasznos tárgy, és fontos dolgokat lehet tanulni. A JavaScript, HTML és XML fontos elemei egy informatikus mindennapjainak és eddig nem volt olyan tárgy, amely kellő részletességgel foglalkozott volna ezekkel. A kiadott feladatok megoldása általában legalább 2-3 teljes napot vett igénybe, így a tárgy teljesítéséért kapható 2 kreditet kevésnek érzem, remélem ez a jövőben pozitív irányba módosul.