MSSQL

Segédlet az Üzleti intelligencia labor c. tárgyhoz

Ákos Dudás

2018.

Szerzői jogok

Jelen dokumentum a BME Villamosmérnöki és Informatikai Kar hallgatói számára készített elektronikus jegyzet. A dokumentumot az Üzleti intelligencia labor c. tantárgyat felvevő hallgatók jogosultak használni, és saját céljukra 1 példányban kinyomtatni. A dokumentum módosítása, bármely eljárással részben vagy egészben történő másolása tilos, illetve csak a szerző előzetes engedélyével történhet.

# Bevezetés

## Célkitűzés

A labor gyakorlat célja, hogy a hallgató megismerkedjen a Microsoft SQL Server üzleti intelligenciához kapcsolódó termékeivel, a termékek használatával, és a termékekre történő fejlesztéssel.

A labor során a hallgatók egy-egy példán keresztül ismerkednek meg az SQL Server, az Integration Services, és a Reporting Services termékekkel, használatukkal, és a fejlesztés menetével.

## Előfeltételek

A labor elvégzéséhez szükséges eszközök:

* Forrás adatok a tárgy honlapjáról: [MSSQL adatforrás BXBook](https://www.aut.bme.hu/Upload/Course/VIAUMB00/gyakorlat_anyagok/BXBooks-CSV.zip)
* Microsoft SQL Server (LocalBD is megfelelő)
* Microsoft SQL Server Management Studio
* Visual Studio 2017
* Microsoft SQL Server Data Tools

## Amit érdemes átnézned

* Üzleti Intelligencia c. tárgy Microsoft SQL Server és kapcsolódó SDK-k anyaga
* „MSSQL BI termékek használata” c. segédletben található összefoglaló a használt szoftverekről
  + További információk, példák, és tutorial az alábbi oldalakon található
    - <https://msdn.microsoft.com/en-us/library/ms169917(v=sql.120).aspx>
    - <https://msdn.microsoft.com/en-us/library/ms167305(v=sql.120).aspx>
* Az adatok, amiket felhasználunk

## Jegyzőkönyv

A labor feladatairól vezess jegyzőkönyvet. A sablon a tárgy honlapján található. A közösen megoldott feladatoknál elég a végeredményről írni a jegyzőkönyvbe, az önálló feladatoknál részletesebben magyaráz mit és miért csináltál. Minden feladatnál a végleges eredményt mutasd be!

A jegyzőkönyv mellett az elkészült Visual Studio solution-ök könyvtárát is add be.

## Óra menete, értékelés

Az 1-3 feladatot közösen oldjuk meg, a 4-5 önálló munka. Otthon is befejezhetik, de akkor telepítsék a szoftvereket.

Osztályzás: ahány feladatot (jól) megoldott, annyi a jegy.

# Tervezés

A feladat egy könyv webáruház adatainak feldolgozása. Az adatokat CSV dumpok formájában kapjuk. Megtalálható köztük a webshopban regisztrált felhasználók adatai, a könyvek adatai, valamit a vásárlók könyvekre adott értékelése. A feladatunk az adatok feldolgozható formára hozása, és jelentések készítése a könyvekről illetve a vásárlókról. A megoldáshoz az SQL Server termékcsalád relációs adatbázisát, mint adattárházat, az SQL Server Integration Services-t (SSIS), és az SQL Server Reporting Services-t (SSRS) használjuk.

a) Tervezze meg az SQL Server termékeinek használatával a feldolgozási folyamatot. Azonosítsa a részfeladatokat, válassza ki az SQL Server termékek közül az egyes részfeladatok megoldására alkalmasakat. A terv tartalmazza az adatok áramlásának irányát, és hogy mikor melyik terméket használjuk. A tervet szövegesen vagy a Word rajzolási képességeit felhasználva ábra segítségével készítse el (utóbbi a preferált).

b) Töltse le az adatokat, és csomagolja ki a laborvezető által megadott könyvtárba. Vizsgálja meg az adatokat.

c) Tervezze meg a relációs adatbázis táblák sémáját, amibe az adatokat importálni fogja. Az adatok legyenek típusosak a további feldolgozás miatt, azaz ne szövegként importálja, ami nem szöveg. Hozzon létre az SQL Serverben egy új adatbázist (neve legyen a neptun kódja), és készítse el a táblákat. A mezők képezzék le a CSV fájlban található tartalmat, kivéve a BXUsers.csv esetén a Location adatot, amelyet ketté szeretnénk vágni Country és City néven.

Hagyjunk a hallgatónak időt a gondolkodásra. Az a) feladatot beszéljük meg közösen, utána ismét hagyjunk időt az adatok feltérképezéhez. Az adatbázis sémát közösen alakítsuk ki.

**a)** Az alábbi részfeladatokat tudjuk azonosítani.

1. A forrás adatokat Integration Services segítésével feldolgozzuk és SQL Server-re relációs adatbázisba helyezzük.
2. A jelentéseket Reporting Services fogja készíteni, az adatokat az SQL Serverről veszi.

Ábrával az alábbi rendszerképet kellene előállítani.

forrás

fájlok

SQL Server

adatbázis

(adattárház)

Reporting

Services

Integration

Services

**b)** A fájlokat a c: vagy d: meghajtó work könvytárába érdemes kicsomagolni. Nézzük meg az adatokat. (Excel is képes megnyitni a fájlokat.) Nagyjából sejtsük, milyen információ van a fájlokban, és a szerkezetét is figyeljük meg: CSV, pontosvessző szeparátor, ” a mezők körül.

**c)** Az alábbi táblaszerkezetet kialakítása lesz célszerű. Az egyszerűség kedvéért nem állítunk be elsődleges, se külső kulcsokat. Ez az importálási folyamatot fogja megkönnyíteni.

BXUser: A Location mezőt kettévágjuk országra és városra. Ez a későbbiekben jól fog jönni.

|  |  |  |
| --- | --- | --- |
| BXUser | oszlop | adattípus |
| UserID | int |
| City | nvarchar(max) |
| Country | nvarchar(max) |
| Age | int |

BXBook: A képek közül csak a legkisebb méretét őrizzünk meg.

|  |  |  |
| --- | --- | --- |
| BXBook | oszlop | adattípus |
| ISBN | nvarchar(13) |
| Title | nvarchar(max) |
| Author | nvarchar(max) |
| Year | int |
| Publisher | nvarchar(max) |
| Image | nvarchar(max) |

BXBookRating: a BookRating értéke 0-10 közötti szám, ezért a tinyint elég lesz.

|  |  |  |
| --- | --- | --- |
| BXBookRating | oszlop | adattípus |
| UserID | int |
| BookISBN | nvarchar(13) |
| BookRating | tinyint |

A létrehozáshoz SQL Server Management Studio-t elindítva kapcsolódjunk a (localdb)\mssqllocaldb címen található szerverhez *Windows Authentication* használatával. Az *Object Explorer*-ben a *Databases*-en jobb egérrel kattintva válasszuk a *New Database*-t és a neptun kódunkkal hozzunk létre egy új adatbázist. A táblákat SQL kódból vagy designerből is létrehozhatjuk. SQL kódból: kattintsunk a New query gombra, és futtassuk az alábbi scriptet.

CREATE TABLE "BXUser" (

"UserID" int,

"Age" int,

"City" nvarchar(1000),

"Country" nvarchar(1000)

)

CREATE TABLE "BXBook" (

"ISBN" nvarchar(13),

"Title" nvarchar(max),

"Author" nvarchar(max),

"Year" int,

"Publisher" nvarchar(max),

"Image" nvarchar(max),

)

CREATE TABLE "BXBookRating" (

"ISBN" nvarchar(13),

"UserID" int,

"BookRating" tinyint

)

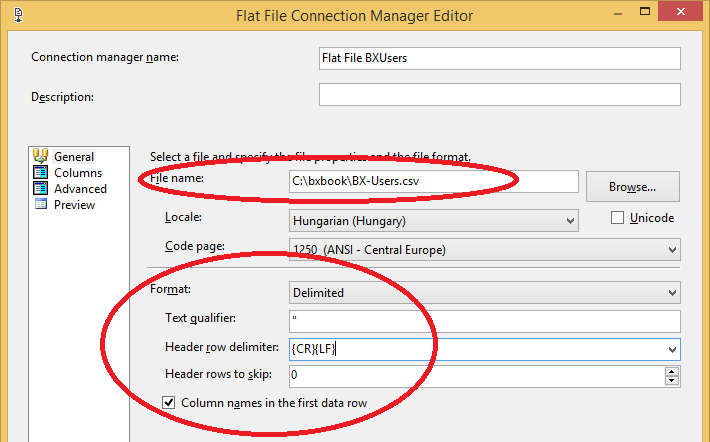
# BX-Users Adatok importálása

Integration Services segítségével tervezzen meg és implementáljon egy ETL folyamatot, amely a „BX-Users.csv” szöveges fájlból relációs adatbázisba emeli az adatokat.

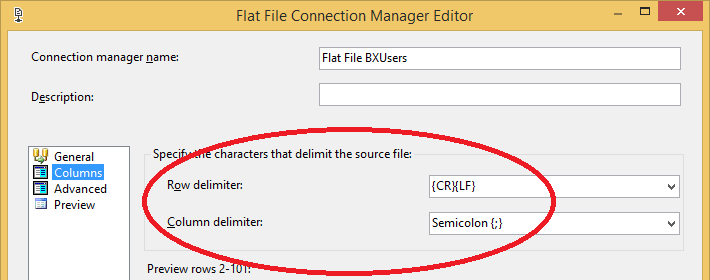
* Ügyeljen az életkor számként történő importálására, a NULL mezőket viszont őrizze meg.
* A Location oszlopban található adatokból a város és ország szétválasztását a betöltési folyamat közbeni transzformációval valósítsa meg. Ehhez használja a Derived Column nevű transzformációt (a szintaktikáról lásd: <https://docs.microsoft.com/en-us/sql/integration-services/expressions/integration-services-ssis-expressions?view=sql-server-2017>).
* A fájlban előfordulhatnak duplikátumok (azonos UserID), ezeket szűrje ki.

Az ETL folyamat létrehozása.

1. Visual Studio-ban hozzunk létre egy új Integration Services projektet.
2. A megnyíló designerben *Control flow* nézetben vagyunk. A fájl importálása egyszerű lesz, csak egy *Data flow task*-ra lesz szükségünk, a toolboxról húzzuk a designerre. A *Properties* ablakban adjuk meg a nevének (Name): „import bxusers file”. Dupla kattintással nyissuk meg, ezzel átváltunk *Data Flow* nézetbe.
   1. A toolboxról vegyünk el az *Other Sources* alól egy *Flat File Source* elemet.
   2. Dupla kattintással nyissuk meg. A *Connection Manager* oldalon válasszuk a *New* gombot egy új kapcsolathoz. Itt adjuk meg a fájlt, amit beolvas.
      * A *File name*-nél adjuk meg a fájl elérési útvonalát (használjuk a Browse gombot)*.*
      * *Locale* és a *Code page* maradjon.
      * *Format* legyen „Delimited”.
      * *Text qualifier* legyen ” (idézőjel). Ezt a fájl megnyitásával tudjuk, ha jegyzettömben nyitjuk meg a fájlt, láthatjuk, hogy a mezők tartalma idézőjelben van.
      * *Column names…* mellett legyen pipa, a fájlban van header sor.

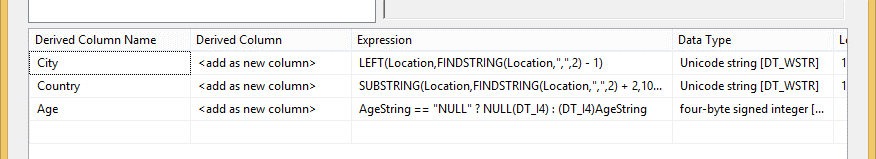


* 1. Váltsunk a *Columns* fülre. Az automatikusan kitöltött adatok jók lesznek:
     + *Row delimiter*: {CR}{LF} (újsor jel)
     + *Column delimiter*: Semicolon

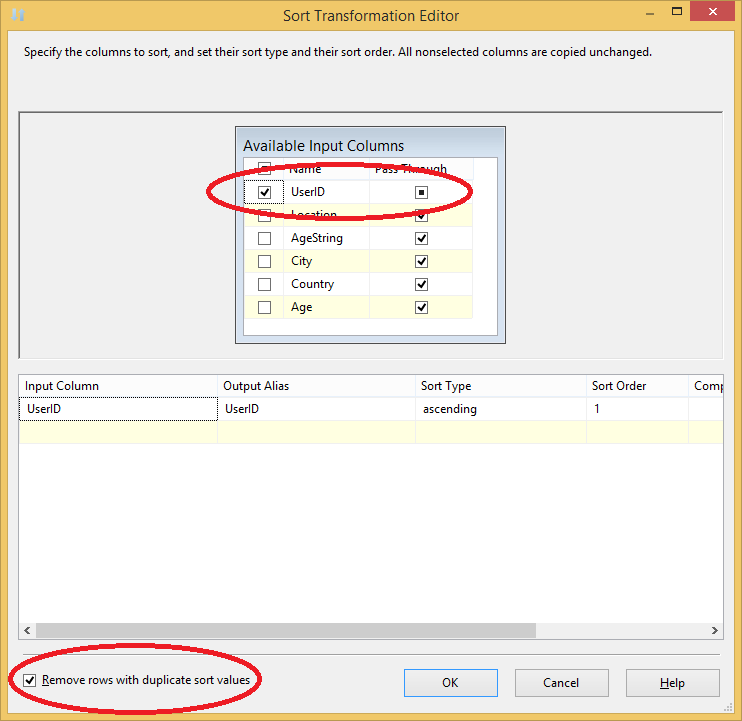


* 1. Váltsunk az *Advanced* fülre. Az oszlopoknak van neve és típusa, de változtassunk rajta kicsit. A névből vegyük ki az idézőjeleket, és adjuk meg a helyes típust (name, data type, outputcolumnwidth). Az age esetén a NULL értékek miatt kell szövegként importálni.
     + UserID, four-byte signed int
     + Location, string, 1000
     + AgeString, string, 4
  2. Zárjuk a dialógust, és a Flat File Source beállításait is.

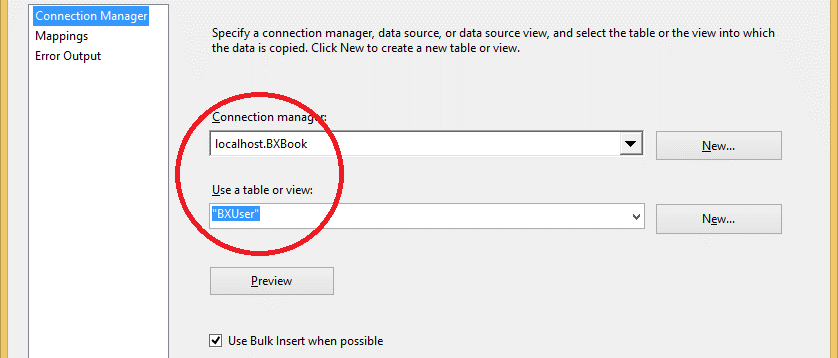
1. A Location mezőt daraboljuk városra és országra, az Age mezőt pedig alakítsuk számmá. Húzzunk egy *Derived Colum* traszformációt a designerre. A Flat File Source kimenetéből a kék nyilat kössük ennek a bemenetére, majd nyissuk meg a Derived Column beállításait dupla kattintással.
   1. A location-t vágjuk ketté. Vegyünk fel az ablak alsó részén két új sort az alábbi beállításokkal.
      * City (add as new column), LEFT(Location,FINDSTRING(Location,",",2) - 1)
      * Country (add as new column), SUBSTRING(Location,FINDSTRING(Location,",",2) + 2,1000)
   2. Az életkort konvertáljuk számmá vagy NULL értékké a szövegből. Vegyünk fel egy új sort az alábbiakkal.
      * Age (add az new column), [AgeString]=="NULL" ? NULL(DT\_I4) : (DT\_I4)[AgeString]



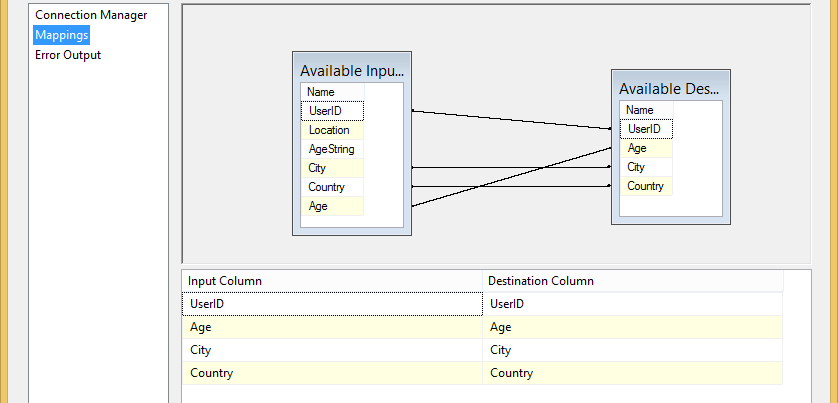
1. Mielőtt az adatbázisba mentenénk az adatokat, a duplikátumokat szűrjük ki. Erre egy *Sort* transzformációt tudunk használni. (Nem ez a leghatékonyabb, sok memóriát igényel, de a célnak most megfelel). Húzzunk egy *Sort* elemet a Data Flow nézetbe, kössük bele a *Derived Column*-ból a kék nyilat.
   1. Nyissuk meg a *Sort* beállításait dupla kattintással.
   2. A dialógus felső részén pipáljuk ki a UserID-t, ezt alapján fogunk szűrni. A többi mezőnél láthatjuk a „Pass Through” beállítást, azok alapján nem sorrendezünk.
   3. A dialógus alján kattintsuk be a „Remove rows with duplicate sort values” elemet.
   4. OK-val zárjuk a dialógus ablakot.



1. Adjunk egy *ADO .NET Destination* elemet a designerbe. Húzzuk bele az előző *Sort*-ból a kék nyilat. Nyissuk meg az *ADO.NET Destination* tulajdonságait dupla kattintással.
   1. Hozzunk létre egy Új *Connection manager*-t, majd abban egy új *Data Collection*-t. Adjuk meg a szerver nevét (localdb)\mssqllocaldb és a korábban elkészített adatbázis nevét.
   2. A *Use a table or view* alatt válasszuk ki a „BXUser” táblát.

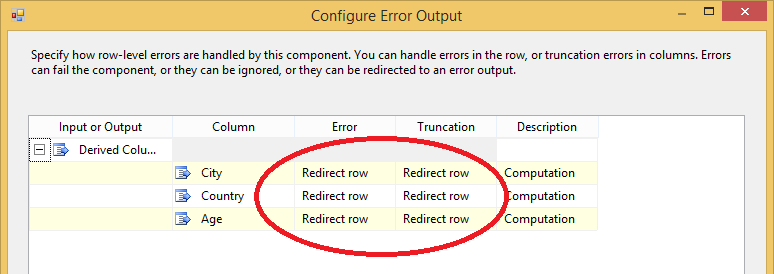


* 1. Ellenőrizzük a *Mapping* fülön, hogy a szükséges mezők le vannak képezve (kivéve az AgeString és a Location).



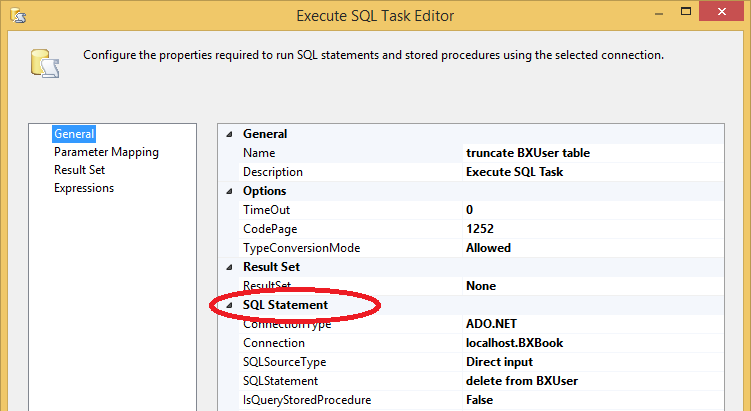
* 1. OK-val zárjuk a dialógus ablakot.

1. A *Derived Column* transzformáció sikertelen lehet, ha nem a jó szintaktika szerinti az adat. Ezeket az adatokat se hagyjuk veszni, vegyük fel az adatbázisba hely és életkor nélkül.
   1. Vegyünk fel egy újabb *ADO .NET Destination* elemet, és a *Derived Column*-ból a piros nyilat húzzuk bele. Ezen az útvonalon jönnek a hibás rekordok. Amikor a nyilat összekötöttük, feljön egy dialógus, ahol a hibák kezelésének módját megadhatjuk. Válasszuk ki az összes elemet a táblázatban, és állítsuk át „Redirect row”-ra.

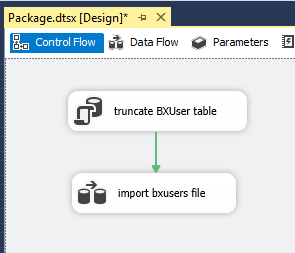
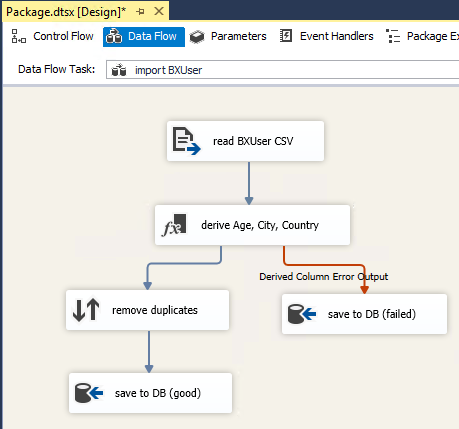


* 1. Az *ADO NET Destination*  beállításanál adjuk meg ugyanazt a kapcsolatot és ugyanazon táblát.
  2. A *Mapping* fülön látható lesz, hogy csak a UserID lesz mappelve.

1. Az importálási folyamatot többször futtatva az adatokat többször importálnánk. Mielőtt a folyamat elindul, töröljük a BXUser tábla tartalmát. Váltsunk vissza *Control Flow* nézetbe, és adjunk egy *Execute SQL Task* elemet az egyetlen data flow task elé.
   1. Az új elemből a zöld nyilat kössük a data flow elembe (ez határozza meg a sorrendet).
   2. Dupla kattintással nyissuk meg a tulajdonságokat. Az SQL Statement alatt adjuk meg a következőket.
      * *ConnectionType*: ADO.NET
      * *Connection*: válasszuk ki a már létező kapcsolatot (localhost.NEPTUN néven szerepelhet)
      * *SQLSourceType*: Direct input
      * *SQLStatement*: „delete from BXUser” (lehetne „truncate table BXUser” is, de ha lenne külső kulcsunk, akkor csak a delete működik)
      * *IsStoredProcedure*: false



A végeredményben a Control flow és a Data flow így néz ki.

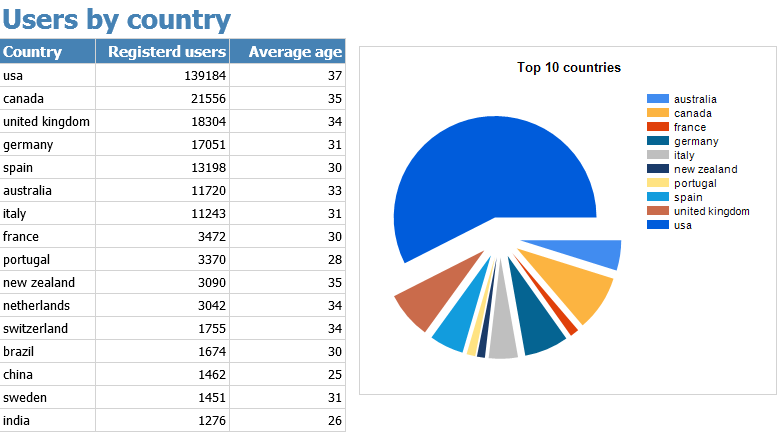
 

Futtassuk le a folyamatot és ellenőrizzük az eredményét. A futtatáshoz a toolbaron levő Debug gombot használjuk.

# Felhasználók országok szerinti listája

Reporting Services segítségével készítsen táblázatos riportot, amely országok szerint csoportosítva megadja a regisztrált felhasználók számát és átlagéletkorukat. Csak azon országokat listázza, ahol van legalább 50 regisztrált felhasználó. A táblázat mellé készítsen kördiagramot arról a 10 országról, ahonnan a legtöbb regisztrált felhasználó származik.

A riport hasonlóan nézzen ki, mint az alább látható.



1. Indítsuk a Visual Studio-t és hozzunk létre egy új Reporting Services Wizard projektet.
2. A varázsló második oldalán találjuk az adatforrás tulajdonságait. Válasszuk a *New data source*-t, adjunk nevet neki, és az alábbi beállításokat használjuk: *Type*: Microsoft SQL Server, *Connection String* mellett *Edit*, *Server name*: (localdb)\mssqllocaldb, *Windows Authentication*, megfelelő adatbázis kiválasztása.
3. Következő oldalon a riport adataiként szolgáló lekérdezés jön.

select Country, count(\*) as Count, avg(Age) as AvgAge

from BXUser

where Country is not NULL and Country<>''

group by Country

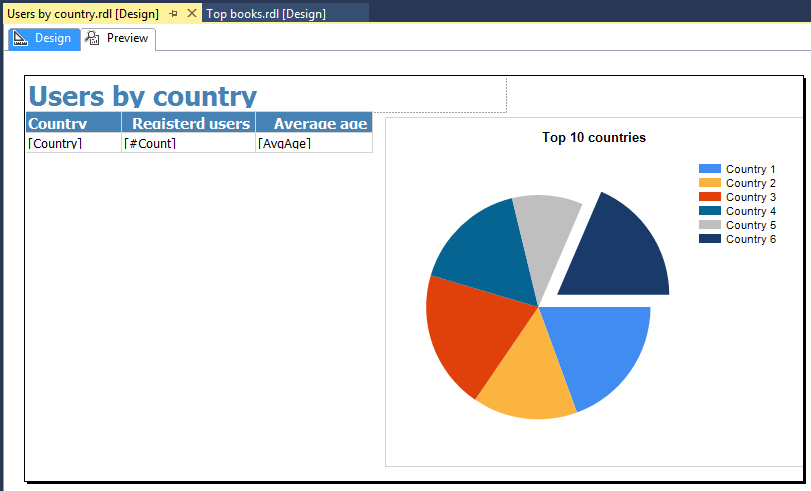
having count(\*) >= 50

order by Count desc

1. A következő oldalon válasszuk a *Tabular* nézetet, a következő oldalon pedig a *Details* csoportba tegyük mindhárom mezőt. Finish-sel zárjuk a varázslót, a végén adjunk nevet a reportnak.

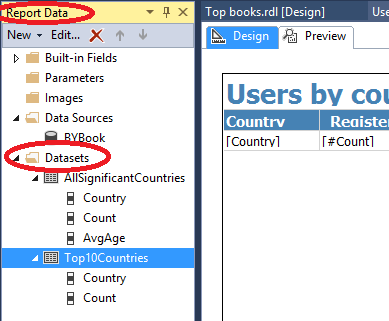
Lehet szerkeszteni a riportot, színezni, átstrukturálni. Ha marad idő, foglalkozhatunk vele. Megnézni a *Preview* fülön vagy a Start gombra kattintva lehet Utóbbi a report szerveren publikálja és úgy mutatja).

A szerkesztés közbeni állapot valahogy így kell kinézzen (a kördiagram nélkül egyelőre).



Kördiagram hozzáadása.

1. A kördiagramhoz egy új adatforrást kell felvennünk. A Report Data ablakot keressük meg, itt tudunk a Datasets-be újat felvenni.



1. A lekérdezés az alábbi.

select top 10 Country, count(\*) as Count

from BXUser

where Country is not NULL and Country<>''

group by Country

order by Count desc

1. A *Toolbox*-ről húzzunk egy *Chart*-ot a designer felületére. Ha kiválasztjuk a chart objektumot, megjelenik mellette a *Chart data* ablak. Ide kell az előbbi *Report Data* ablakból húzni az adatokat. A „Count” a *Values*-hoz, a „Country” pedig a *Category Groups*-hoz megy.

# BX-Books és Bx-Book-Ratings Adatok importálása

A korábban kialakított ETL folyamatot egészítse ki a „BX-Books.csv” és „BX-Book-Ratings.csv” szöveges fájlok importálásával.

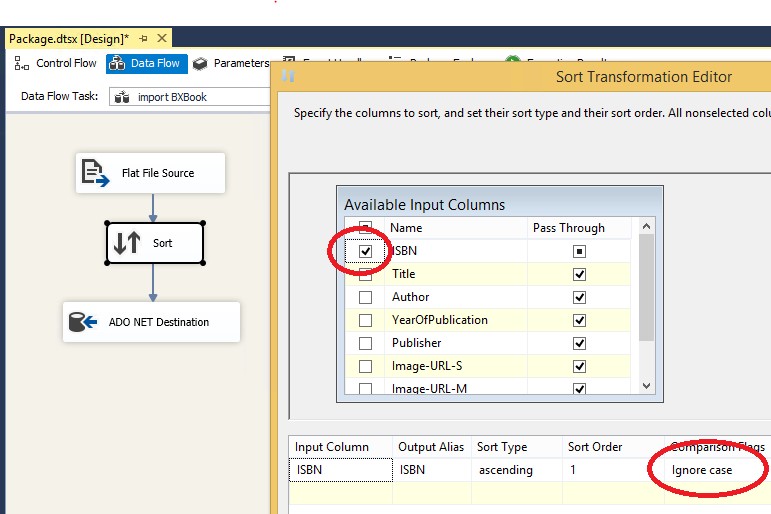
* A BXBooks esetében az ISBN-t szövegként importálja és ügyeljen a duplikátumok kiszűrésére (ISBN szám alapján, és az összehasonlítás case insensitive legyen).
* A BXBookRating importálása során csak az érvényes értékeléseket importálja, azaz ahol a BookRating értéke nem 0 (a 0 értékelés a nem értékeltet jelenti).
* A BXBooks adathalmazból az „M” méretű kép url-t importálja (a többire nincs szükség).
* Ügyeljen a CSV fájlok beolvasása során az oszlopok hosszának megnövelésére (1000 karakter mindegyiknél elégséges lesz).

Az ETL folyamatot a korábban kialakított csomagba vegye fel. Az egyes feladatok sorrendjét Control Flow nézetben tudja meghatározni a „dobozok” közötti nyilak behúzásával. (Érdemes egymás után kötni a folyamatokat – kizárva ezzel a párhuzamos futtatást – a memória használat csökkentése érdekében.)

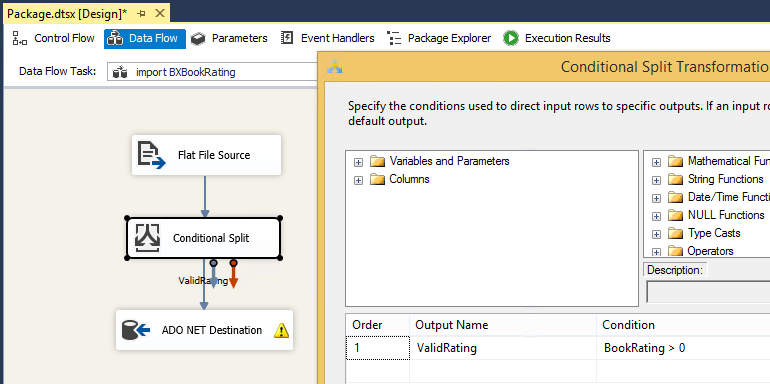
Mindkét forrásfájlhoz hasonlóak a lépések.

1. Új *Execute SQL statement*, ami a delete-et elvégzi.
2. És ez után kötött *Data flow* lépés, ami konverzió nélkül egy *Flat File Source* importot és egy *ADO.NET Destination*-t tartalmaz. Egyedül az adattípusokra érdemes figyelni a korábban kialakított adatbázissémának megfelelően.

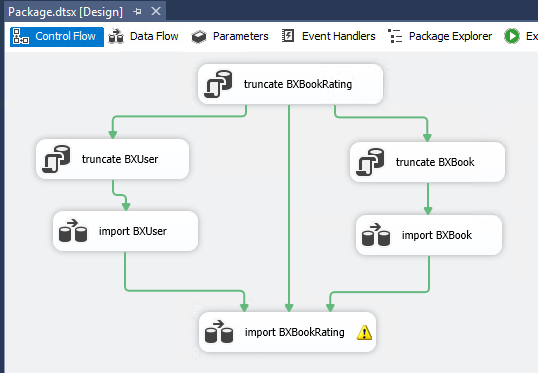
BXBook esetében alakítsunk ki egy szűrést a korábbiakhoz hasonlóan. A szűrés beállításainál meg kell adni, hogy az összehasonlítás „ignore case” beállítással működjön.



A BookRating esetén a 0 értékelések kihagyása. Ehhez egy *Conditial Split* lépést kell felvenni a *Flat File Source* és az *ADO NET Destination* közé, amely a „BookRating>0” feltételt használva állítja elő a kimenetét. A feltételt teljesítő elemeket el kell látni egy névvel, és az ADO.NET Destination bekötésekor meg kell adni ezen kimenet nevét (így csak a feltételt teljesítő elemek kerülnek be az adatbázisba).



A végső csomag így néz ki. A törlések és az importálások sorrendje azért kötött, mert ha lennének külső kulcsok a táblákon, ilyen sorrendben kellene először törölni, majd felvinni az adatokat. (A futtatás a Sort műveletek miatt memória igényes. Ha nem tud lefutni ilyen módon, akkor érdemes az import lépéseket egymás után kötni, ha van köztük zöld nyíl, az sorrendi végrehajtást köt meg.)



# Havi toplista

Készítsen riportot, amiben a 10 legnépszerűbb könyv szerepel (szerző, cím, kiadás éve). A könyv népszerűségét az értékelések átlagával számolja, de csak azon könyveket vegye figyelembe, amelyekre létezik legalább 20 értékelés. Ügyeljen rá, hogy az átlagszámítás matematikailag helyes legyen (ne integer aritmetika szerint számolja)!

Opcionális: a könyv adatai mellett szerepeljen a könyv képe is (a hivatkozást megtalálja az adatokban). Ehhez a táblázatba egy Image elemet kell felvenni, amely „External” forrásból tölti be a képet.

A riport hasonlóan nézzen ki, mint az alább látható.



1. A riportot a varázslóval ugyanúgy tudjuk elkészíteni, mint az előzőt. Felhasználhatjuk a korábbi riport projektet és, és a *Solution Explorer*-ben a *Reports* mappára jobb kattintással tudunk új reportot készíteni.
2. A lekérdezéshez az SQL az alábbi.

select top 10 BXBook.Title, BXBook.Author, BXBook.Year, avg(cast(BookRating as float)) as Rating, BXBook.Image

from BXBookRating join BXBook on BXBook.ISBN = BXBookRating.ISBN

group by BXBook.ISBN, BXBook.Title, BXBook.Author, BXBook.Year, BXBook.Image

having count(BXBookRating.ISBN) > 20

order by Rating desc

1. A *Tabular* módot választva a varázslóban mindegyik oszlopot a *Details*-be vegyük fel. (Így a kép URL-jét fogja egyelőre kiírni.)
2. A táblázatot megformázzuk, a mezőket megfelelő szélességűre állítjuk.
3. A képhez a designer nézetben válasszuk ki a kép oszlopban a képet, és jobb egérrel kattintva az *Image properties*-en állítsuk be az alábbiakat.
   1. *Select the image source*: „External”
   2. *Use this image*: [Image]
   3. *Size* oldalon a méretet *Original size*-ra állítjuk

A designer nézet így néz ki.

