

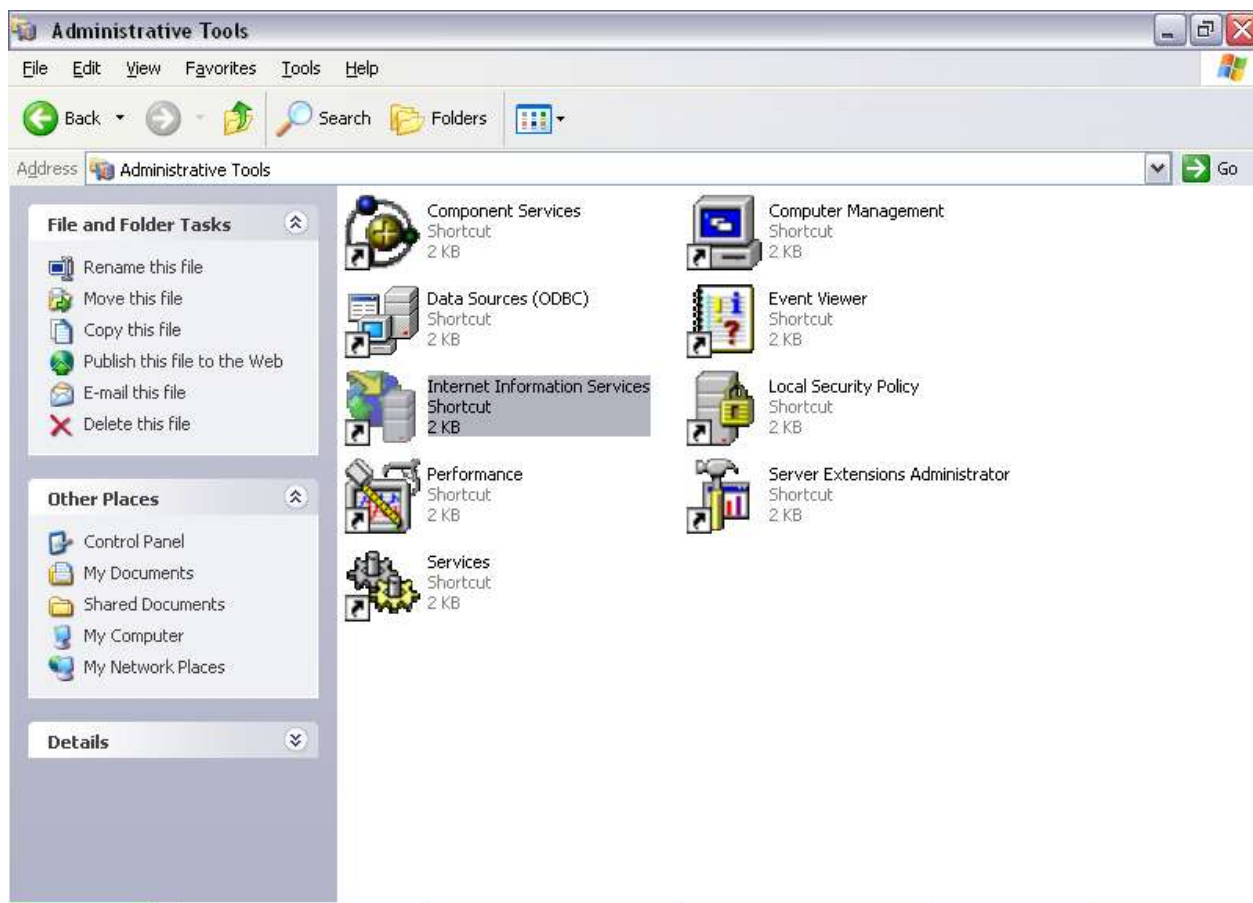
## CUPRINS

Primul pas : instalarea software-ului free.....	3
Al doilea pas : Analiza aplicatiei .....	4
Al treilea pas : Structura Bazei de date.....	6
Al patrulea pas : Importul datelor .....	10

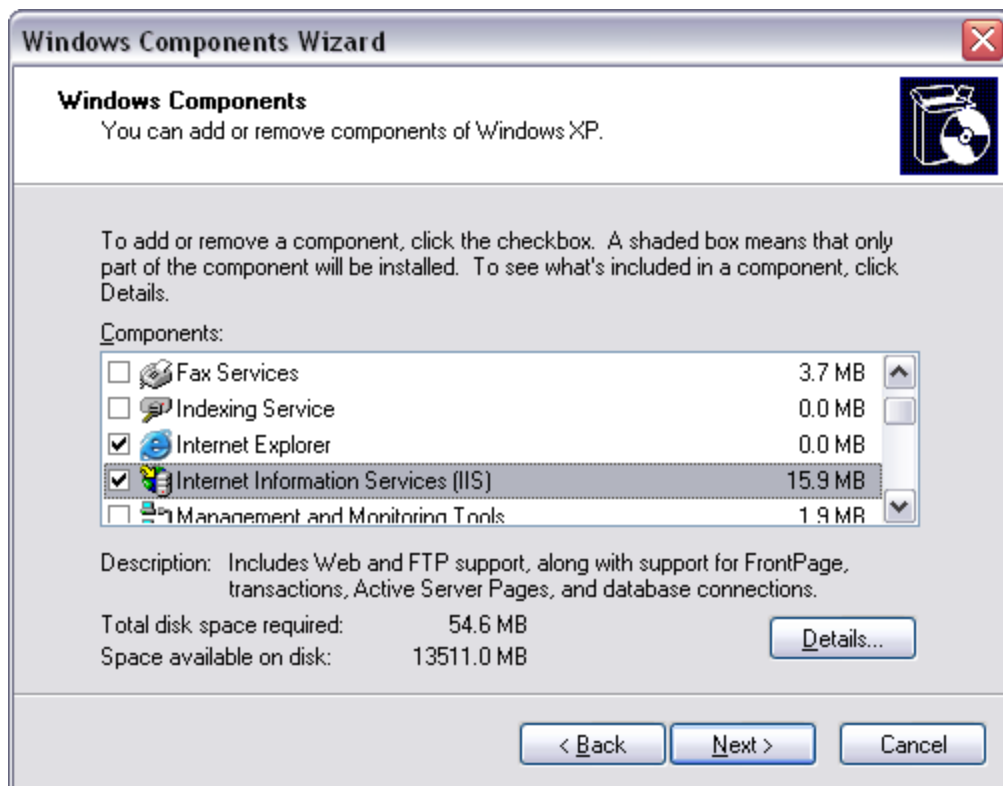
## PRIMUL PAS : INSTALAREA SOFTWARE-ULUI FREE

Acest pas se efectuează o singura dată pe PC. Este vorba despre instalarea server-ului de Web , precum și a Visual Studio Express ca și unealta de dezvoltare. Este cel mai lung pas, dar, precum am spus, se efectuează o singura dată.

Să începem: ai Windows (XP, de preferință) și ne asigurăm că ai server-ul de Web (Internet Information Services – IIS de acum înainte) pe mașină .Pentru aceasta te duci în “Control panel” => “Administrative Tools” și verificăți că există.



Dacă nu există, mergeți înapoi la “Control Panel”, apăsați pe “Add or Remove programs” și apăsați pe “Add/Remove Windows Components”. Bifați IIS ca în imagine și dați “Next”(asigurați-vă că aveți CD-ul de instalare Windows prin apropiere)



Bun – acum o sa instalam VS 2008 Express. Acesta este free si îl puteți descărca de la adresa

<http://www.microsoft.com/express/download/offline.aspx>. Acolo găsiți o imagine ISO pe care o puteți downloada . O puteți vedea cu ISObuster, Daemon Tools or Virtual CloneDrive ( căutați-le pe google pentru linkuri de download) sau cu un program făcut de MS, Virtual CD-ROM Control Panel for Windows XP care poate fi downloadat de la adresa:

[http://download.microsoft.com/download/7/b/6/7b6abd84-7841-4978-96f5-bd58df02efa2/winxpvirtualcdcontrolpanel\\_21.exe](http://download.microsoft.com/download/7/b/6/7b6abd84-7841-4978-96f5-bd58df02efa2/winxpvirtualcdcontrolpanel_21.exe)

( da, știu, e o adresa scurta)

De asemenea, instalati SQL Server Express Edition with Advanced Services SP2 de la adresa

<http://www.microsoft.com/express/sql/download/default.aspx>

Si tool-ul de administrare grafica, numit Microsoft SQL Server 2005 Express Edition Toolkit, de la aceeași adresa

<http://www.microsoft.com/express/sql/download/default.aspx>

## AL DOILEA PAS : ANALIZA APLICATIEI

Va sfatuiam sa cititi prima parte, [Primul pas : instalarea software-ului free](#)

Sau tutorialul anterior despre .NET 2.0

<http://serviciipeweb.ro/iafblog/content/binary/tutorial.pdf>

Aplicatia pe care o sa o facem este una de gestionare(management, pe stil nou) a cartilor dintr-o biblioteca publica. Presupunem ca biblioteca deja isi tine o evidenta a cartilor intr-un Excel cu o multitudine de sheet-uri, cam de aceasta forma

Un sheet, numit "Copii" care contine carti pentru copii,cu urmatoarele date:

Titlu	Autor1	Autor2	editura	Pret	Imprumutata de	Data imprumutului
Craiasa zapezii	Andersen		Teora	3	Ignat Andrei	15/01/2008
GREUCEANU SI ALTE POVESTI	ISPIRESCU Petre		All	5		
Capra cu trei iezi	Ion Creanga		Polirom	2		
Ursul păcălit de vulpe	Ion Creanga		All	5		
Zana muntilor	Petre Ispirescu		Teora	2		

Alt sheet, numit SF, cu urmatoarele date

Titlu	Autor1	Autor2	Editura	Pret	Imprumutata de	Data imprumutului
Caverne de otel	Isaac Asimov		Teora	15	Ignat Andrei	15/01/2008
Fundatia	Isaac Asimov		Teora	31		
Fundatia si imperiul	Isaac Asimov		Teora	23		
Fundatia si Pamantul	Isaac Asimov		Teora	21		
Inainte de fundatie	Isaac Asimov		Teora	13		
RAZBOIUL STELELOR	George Lucas		Polirom	54		

Pare destul de clar , nu ? Fiecare carte are cite o fisa in carte, care spuene cine a imprumutat-o si cind.

Puteti downloada fisierul Excel de la adresa <http://serviciipeweb.ro/iafblog/content/binary/carte.xls>

Daca nu aveti Excel (?), puteti downloada [Excel Viewer](#)

Acum vom face o mica analiza a datelor existente, pentru ca cerintele aplicatiei, ca de obicei, sunt vagi: „sa faca o cautare intre datele existente si sa reproduca procesul existent...”

E clar ca avem de a face cu urmatoarele obiecte:

Editura – ca attribute: nume, site, email

Persoana – nume, prenume, data nasterii

Autor – este o Persoana care in plus are ca atribut – site, (una sau mai multe) Carti publicate

Biblioteca – este o Persoana cu drepturi de modificare Carti/Edituri/Autori/Setari. Este angajat de la data x la data y, unde y poate sa nu existe

Client – Este o Persoana care are dreptul sa imprumute un numar (Setare) de Carti pe o perioada data (Setare). Are un document de identitate.

Carti – ca attribute: Nume, data aparitiei, (publicata de) Editura, (unul sau mai multi) Autori, ISBN, pret

Setari – Numar de carti imprumutate, Perioada imprumutului

Nu voi lua in considerare multe alte lucruri, ca de pilda faptul ca un client pierde o carte sau ca preturile pot fi modificate in timp... Ar complica in mod inutil aplicatia – care nu vrea sa fie o aplicatie completa, ci doar un demo.

Data viitoare o sa facem designul Bazei de date.

[Surse](#)

De citit:

1. Despre Object oriented [http://en.wikipedia.org/wiki/Object\\_oriented](http://en.wikipedia.org/wiki/Object_oriented)

## AL TREILEA PAS : STRUCTURA BAZEI DE DATE

Va sfatuim sa cititi partile anterioare

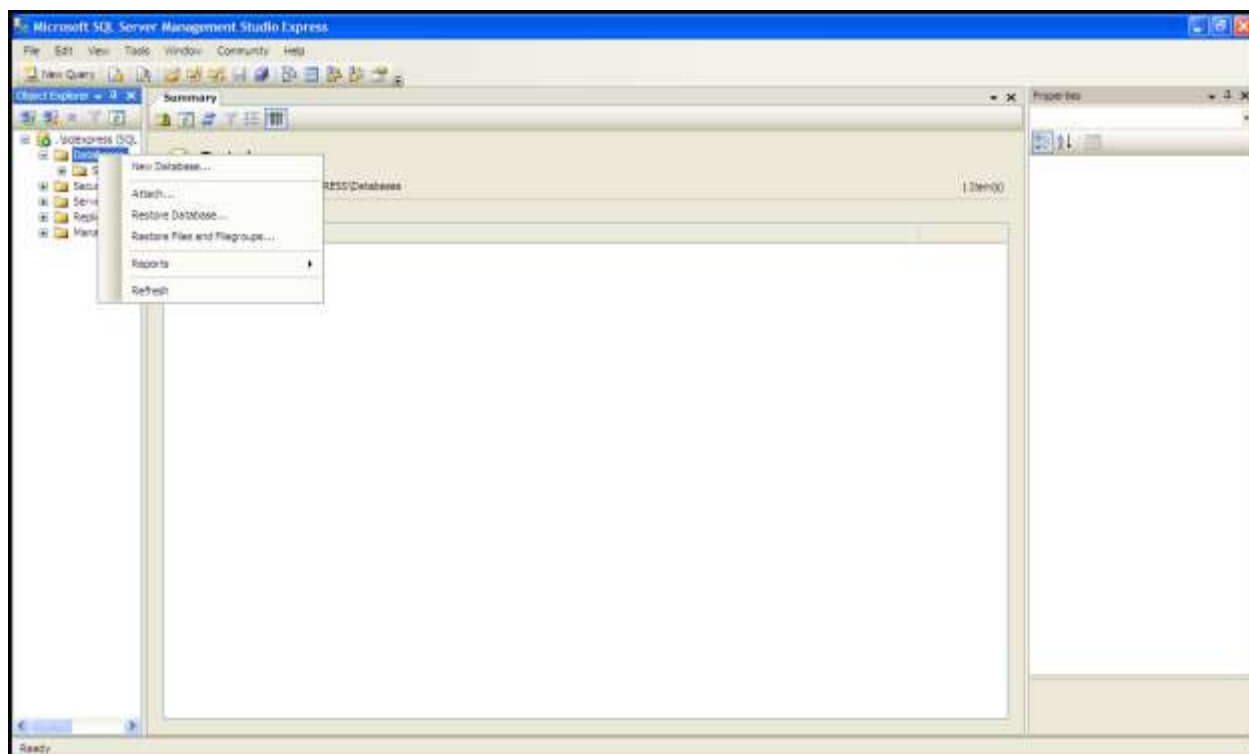
[Primul pas : instalarea software-ului free](#)

[Al doilea pas : Analiza aplicatiei](#)

Sau tutorialul anterior despre .NET 2.0

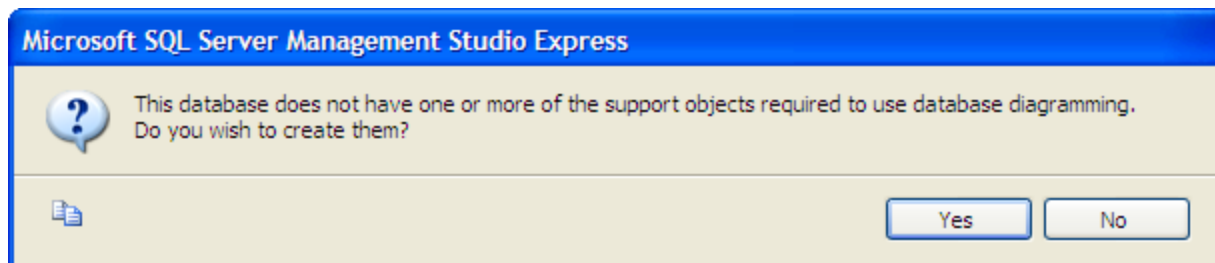
<http://serviciipeweb.ro/iafblog/content/binary/tutorial.pdf>

Facind deja (minima) analiza a aplicatiei, putem acum sa vorbim despre structura bazei de date. Vom face direct o baza de date relationala in cea de a treia forma normala. Pentru aceasta pornim SQL Server Management Studio Express, ne logam la serverul local (./ sau ./sqlxpress, depinde cum ati numit instanta) prin Windows authentication (cea mai simpla metoda) si click dreapta pe Databases => New Database

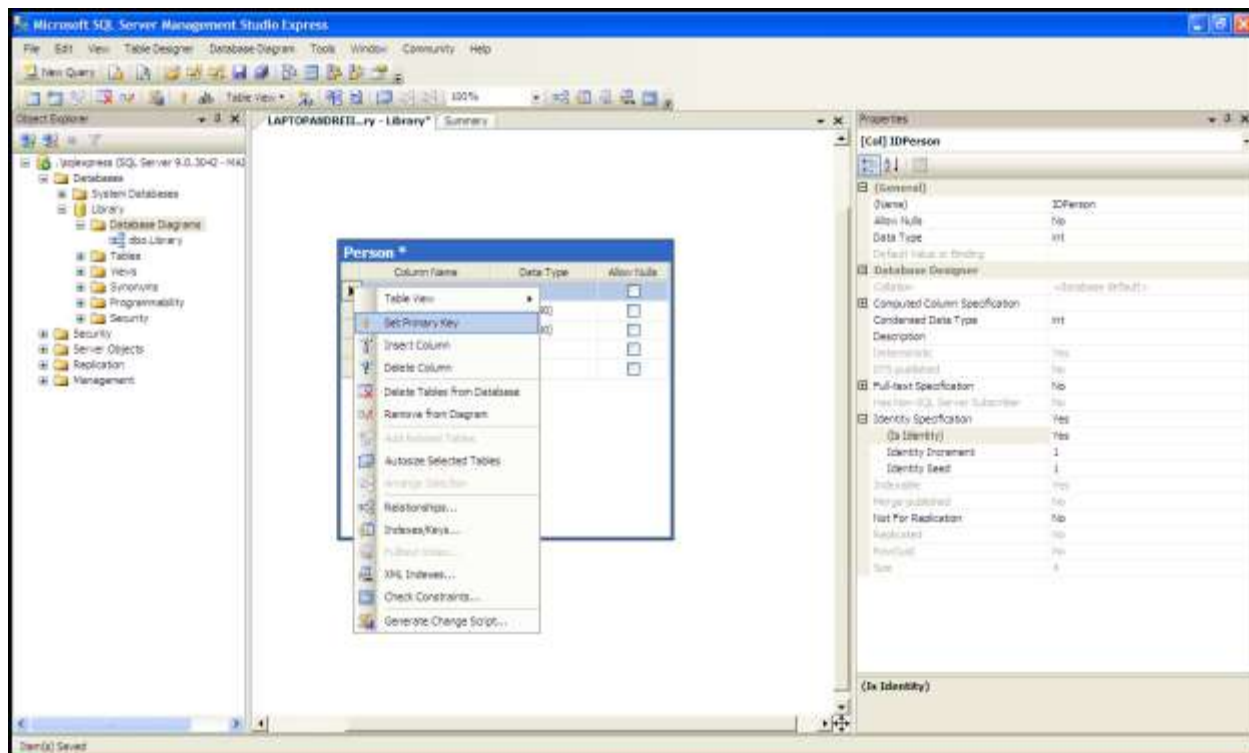


Numele pe care o sa i-l dam o sa fie „Library” si o sa concepem tabelele ca fiind replica exacta a obiectelor.

Vom crea tabelele direct din „Database Diagrams” Raspundem cu „Yes” la intrebarea despre „Diagram support”

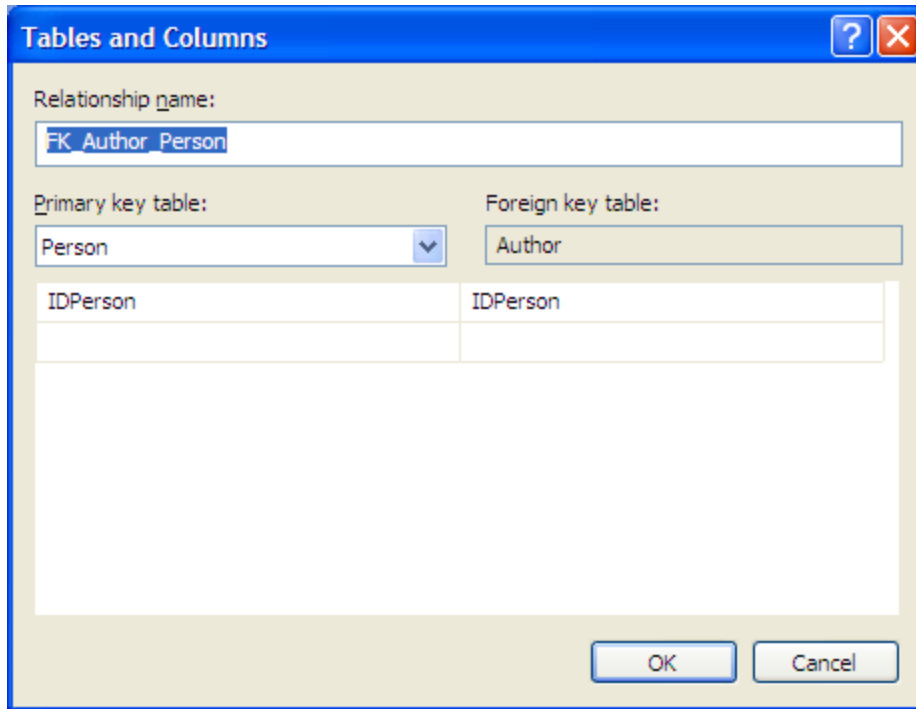


Si cream o diagrama noua, numita "Library". Click dreapta, "New Table", "Person". Adaugam coloanele ca in figura, cu mentiunea ca "IDPerson" o facem "Identity" si Primary Key



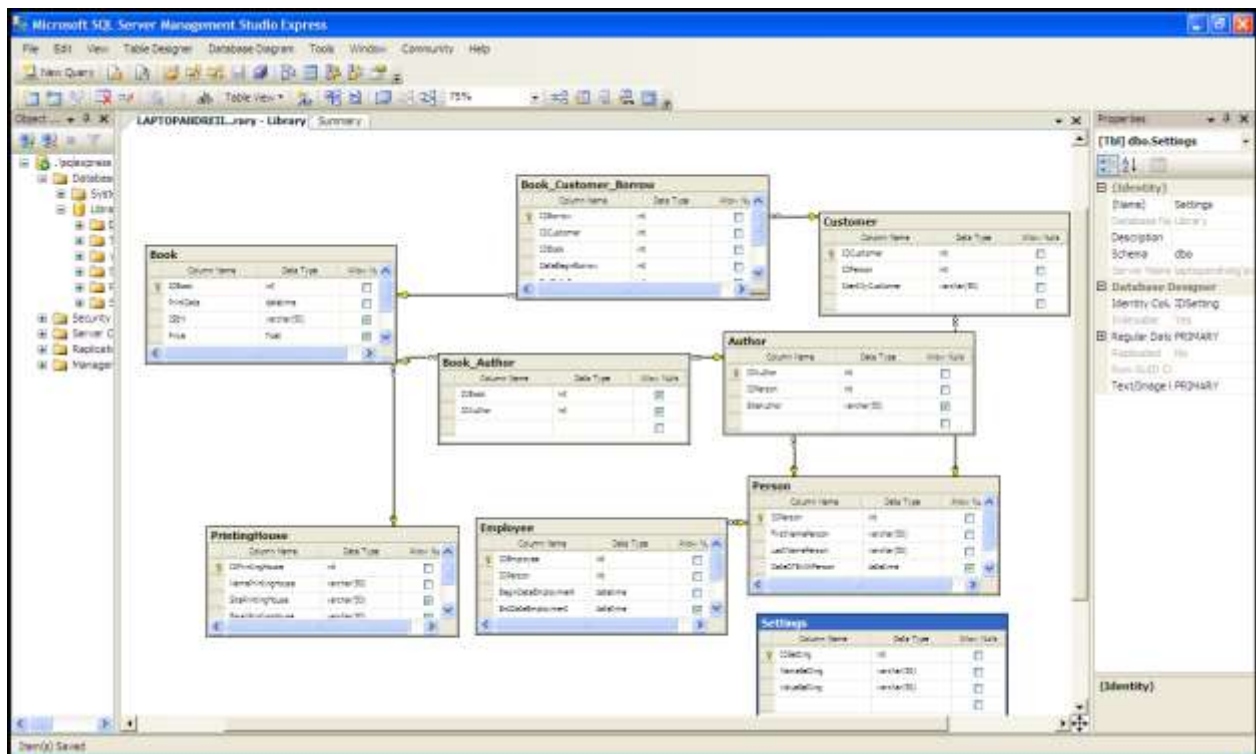
Acum vom crea tabela Author – aceasta va contine o IDPerson – evidentiere a faptului ca orice Autor este si o persoana. La fel, IDAuthor este PK si Identity

Vom "trage" IDPerson din tabela Author peste IDPerson din tabela "Person", obtinind in acest fel legatura intre Autor si Persoana.

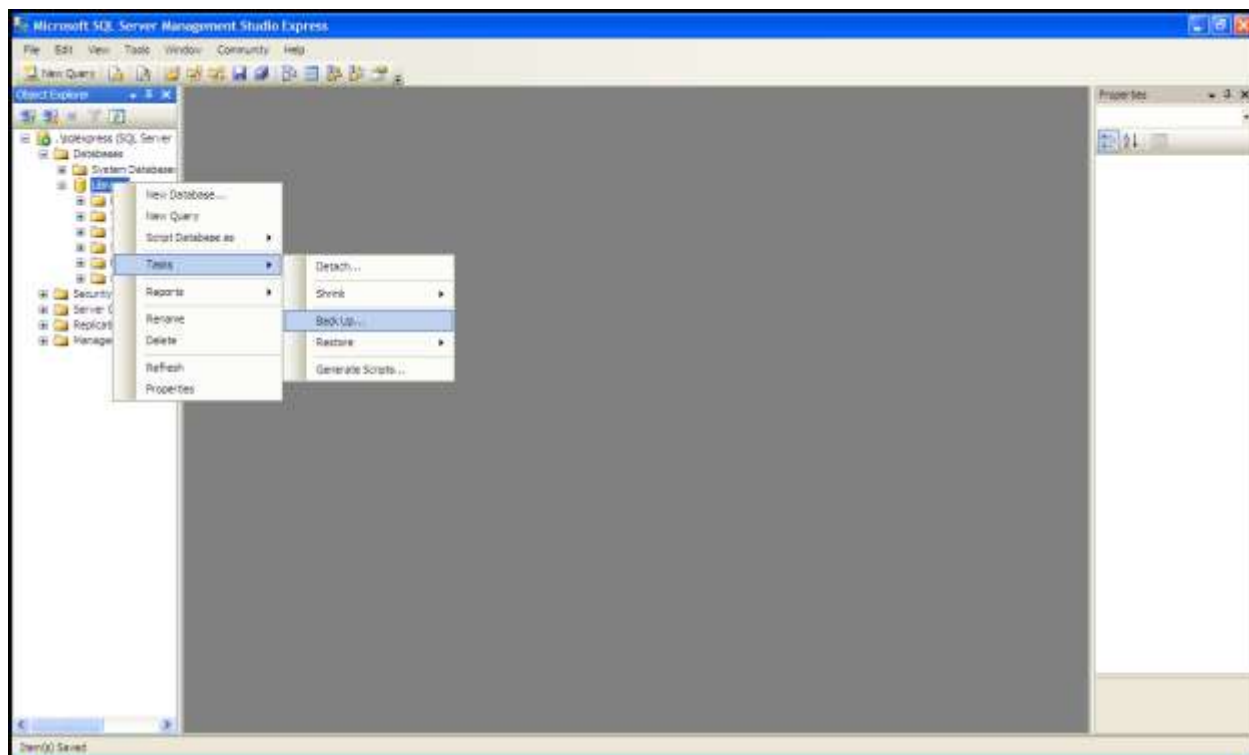


La fel si pentru celelalte tabele.

In final vom avea urmatoare structura :



Vom face un backup al Bazei de date ca in figura:



Puteti crea BD singuri sau puteti face "restore" la ea dupa acest backup , numit lib\_empty.bak

Surse [aici](#)

<http://serviciipeweb.ro/iafblog/content/binary/net3/20080226.zip>

Acest tutorial in intregime [aici](#)

Data viitoare vom importa datele din Excel in SQL Server.

De citit:

1. Despre cele 3 forme normale : [http://en.wikipedia.org/wiki/Database\\_normalization](http://en.wikipedia.org/wiki/Database_normalization)
2. Identity in SQL Server : <http://www.sqlteam.com/article/understanding-identity-columns>
3. PK, FK : [http://en.wikipedia.org/wiki/Foreign\\_key](http://en.wikipedia.org/wiki/Foreign_key)
4. Moduri de a defini mostenirea in cadrul BD :  
<http://blogs.microsoft.co.il/blogs/bursteg/archive/2007/09/30/how-to-model-inheritance-in-databases.aspx>

## AL PATRULEA PAS : IMPORTUL DATELOR

Va sfatuiam sa cititi partile anterioare

[Primul pas : instalarea software-ului free](#)

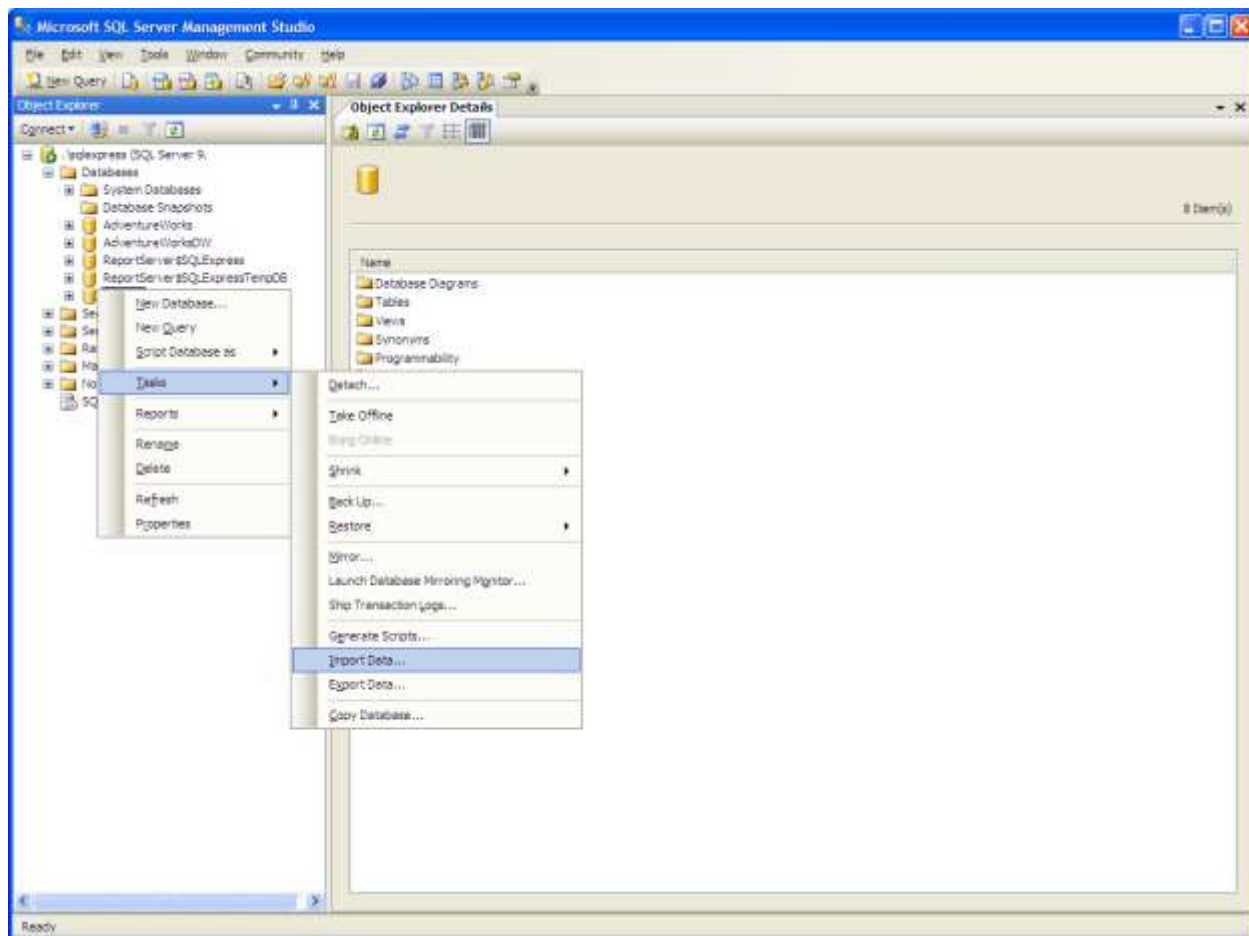
[Al doilea pas : Analiza aplicatiei](#)

[Al treilea pas : Structura Bazei de date](#)

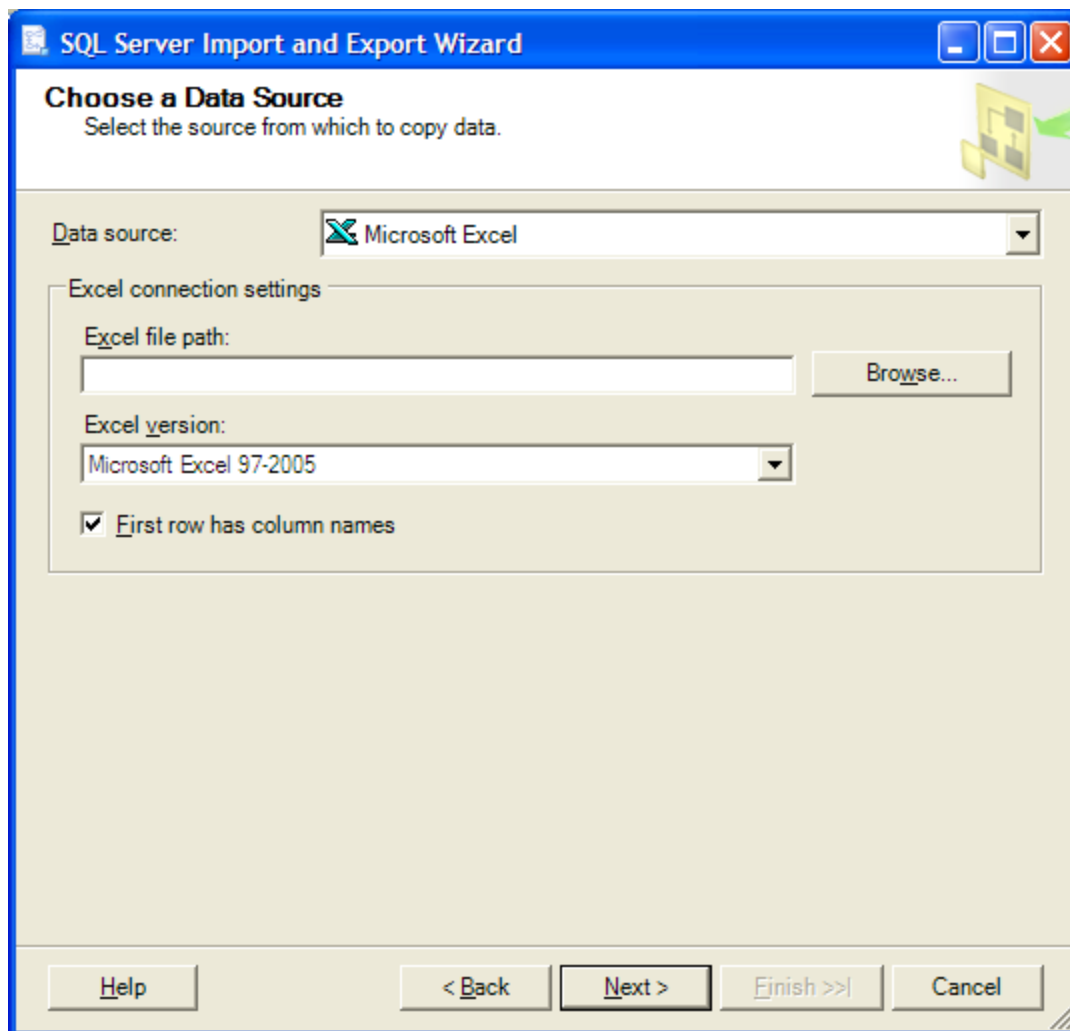
Sau tutorialul anterior despre .NET 2.0

<http://serviciipeweb.ro/iafblog/content/binary/tutorial.pdf>

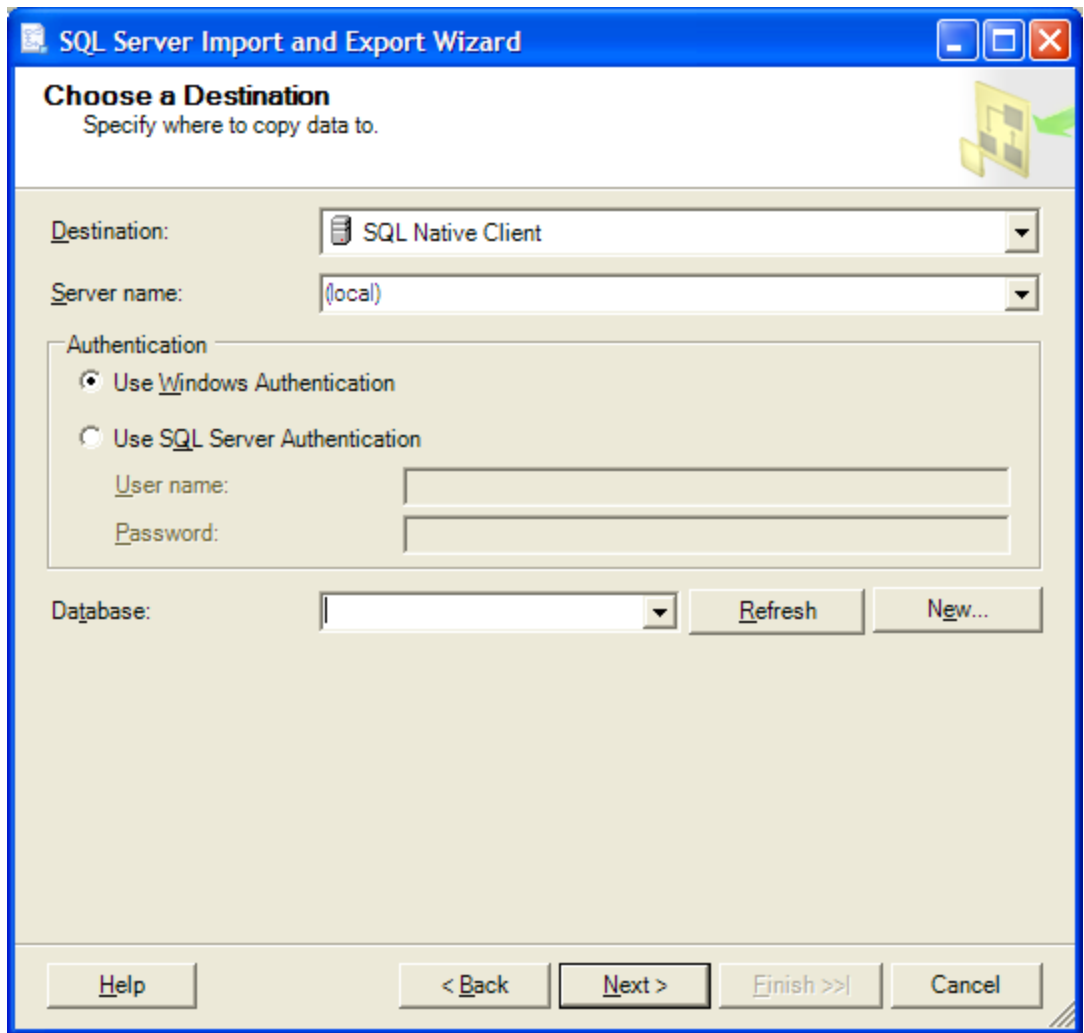
Vom importa datele din fisierul Excel in Baza de date. Daca am avea SQL Server Standard( sau mai mare) am putea importa direct din Excel in SQL Server.Este suficient sa facem click dreapta pe baza noastra de date , Tasks=> ImportData – ca in figura alaturata :



Dupa ce apasam, vom selecta la surse Excel:



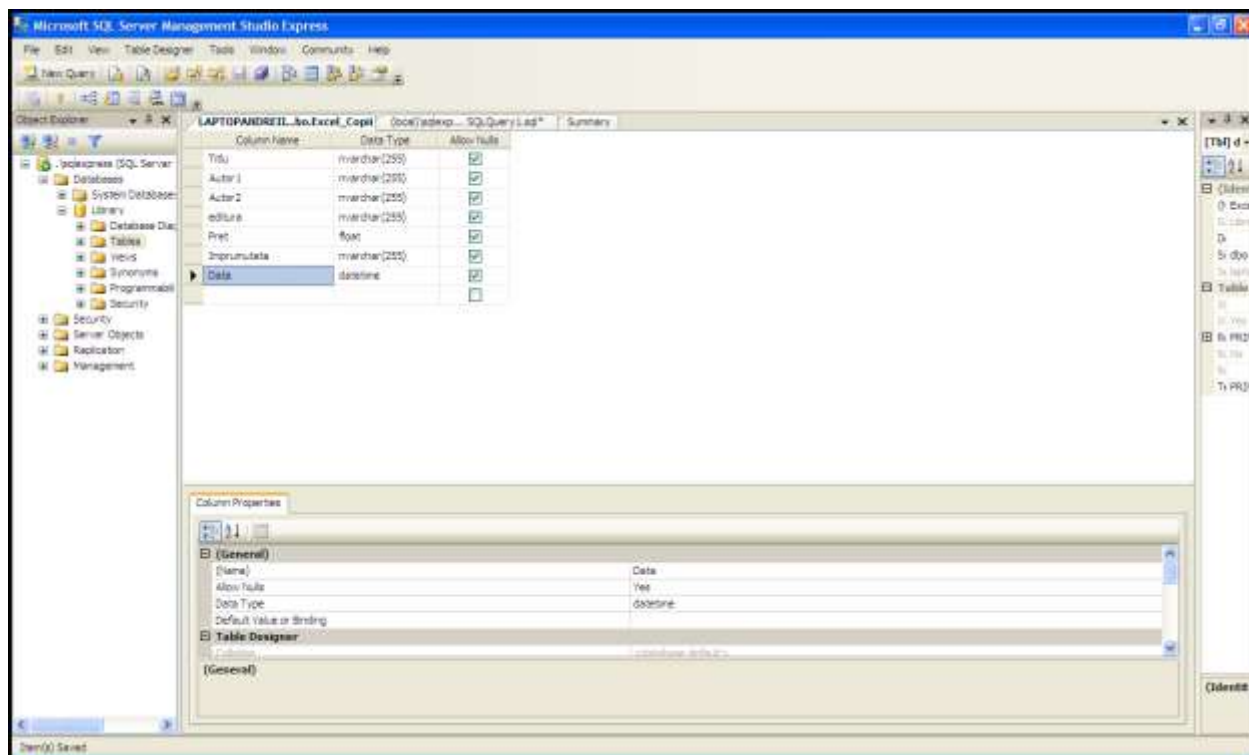
Iar la destinatie serverul local de SQL Server



Cam asta ar fi, daca am avea SQL Server Standard.

Dar ,pentru ca avem SQL Server Express, nu avem o astfel de facilitate incorporata - asa ca va trebui sa ne descurcam importand datele cu un program in C#.

Vom crea tabele in SQL Server asemanatoare cu structura datelor din Excel . Vom crea tabelele asa cu am facut la [pasul 3](#). De pilda tabela cu date despre cartile de copii va arata asa:



Acum vom importa datele. Va trebui sa facem citirea datelor in Excel si apoi scrierea lor in SQL Server.

Sa le luam pe rand:

Cream un nou proiect in C# , intitulat „ImportDate” de tip Consola in folder-ul C:\book3.

Linga toate „using” mai adaugam si un „using System.Data.OleDb;” ca sa putem citi din Excel si using System.Data.SqlClient pentru conectare la SQL Server.

O sa ne folosim de faptul ca DataAdapter stie sa faca modificari de date automat. Ne facem ca citim un DataTable din SQL Server, il umplem apoi cu datele de la Excel si ii spunem lui DataAdapter sa faca insert-urile pentru noi.

Deschidem o conexiune la SQL Server si citim datele din tabela „Excel\_Copii”:

Pentru citirea din Excel vom folosi driverul de OLEDB.Cream o conexiune la Excel si o sa citim datele din tabela „Copii”.

Deschidem o conexiune la Excel

```
Provider = \"Microsoft.Jet.OLEDB.4.0\";Data
Source=\"C:\\book3\\carte.xls\";Extended Properties=\"Excel
8.0;HDR=Yes;IMEX=1\"";
```

(daca vreti sa stiti ce ISAM aveti , vedeti cu regedit cheia

HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Jet\4.0\ISAM Formats)

Deschidem o noua comanda, prin care selectam datele din Worksheetul „Copii”

```
oc.CommandText = "select * from [Copii$]";
```

(de remarcat sintaxa cu \$ si paranteze drepte)

Cod complet:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Data.OleDb;
using System.Data.SqlClient;
using System.Data;
namespace ImportDate
{
    class Program
    {
        static void Main(string[] args)
        {
            using (SqlConnection sco = new SqlConnection())
            {
                sco.ConnectionString = "Data Source=.\sqlExpress;Integrated
Security=true;Initial Catalog=Library";

                sco.Open();
                using (SqlDataAdapter sda = new SqlDataAdapter("select * from
Excel_Copii", sco))
                {
                    // construct insert
                    if (sda.InsertCommand == null)
                    {
                        SqlCommandBuilder scb = new SqlCommandBuilder(sda);
                        sda.InsertCommand = scb.GetInsertCommand(true);
                    }
                    System.Data.DataTable dtTransfer = new
System.Data.DataTable();
                    sda.Fill(dtTransfer);

                    using (OleDbConnection odc = new OleDbConnection())
                    {
                        odc.ConnectionString = "Provider =
Microsoft.Jet.OLEDB.4.0";Data Source=\"C:\\book3\\carte.xls\";Extended
Properties=\"Excel 8.0;HDR=Yes;IMEX=1\"";
```



```
FROM [Library].[dbo].[Excel_SF]
```

Observam urmatoarele date:

NULL

ISPIRESCU Petre

Andersen

George Lucas

Ion Creanga

Isaac Asimov

Petre Ispirescu

Va trebui sa facem 2 lucruri:

1. Consolidarea datelor – in definitiv, Petre Ispirescu = ISPIRESCU Petre
2. Inserarea in tabela Person si Tabela Author

Punctul 1 este destul de usor de facut cu un update ...

```
update Excel_Copii set Autor1= 'Petre Ispirescu' where Autor1 = ' ISPIRESCU  
Petre'
```

Punctul 2 il vom face inserind in tabela de persoane si pe urma in tabela de Autori

```
INSERT INTO [Library].[dbo].[Person]  
    ([FirstNamePerson],[LastNamePerson]  
    )
```

```
select autor, '' from  
(  
SELECT  
    [Autor1] as autor  
    FROM [Library].[dbo].[Excel_Copii]  
union  
select [Autor2]  
FROM [Library].[dbo].[Excel_Copii]  
UNION  
SELECT  
    [Autor1] as autor  
    FROM [Library].[dbo].[Excel_SF]  
union  
select [Autor2]  
    FROM [Library].[dbo].[Excel_SF]
```

```
) a where a.autor is not null
```

Acum separam nume de prenume:

```
UPDATE
    Person
SET
    FirstNamePerson = substring(FirstNamePerson,1,charindex(' ',
,FirstNamePerson)-1),
    LastNamePerson = substring(FirstNamePerson,charindex(' ',
,FirstNamePerson)+1,100)
WHERE charindex(' ',FirstNamePerson)>0
```

Rezultatul este:

IDPerson	FirstNamePerson	LastNamePerson	DateOfBirthPerson
2	Andersen	NULL	
3	George Lucas	NULL	
4	Ion Creanga	NULL	
5	Isaac Asimov	NULL	
6	Petre Ispirescu	NULL	

Acum le vom insera in tabela de Autori:

```
INSERT INTO
    [Author]
    ([IDPerson]
    )
SELECT IDPerson FROM Person
```

La fel inseram cartile si editurile..

Acum trebuie sa refacem legaturile, de pilda, intre autori si carti

```
INSERT INTO [Book_Author]
    ([IDBook]
    , [IDAuthor])

select IDBOOK, IDAuthor from Book b
inner join Excel_SF excel
inner join Person p on excel.Autor1 = p.FirstNamePerson + ' ' +
p.LastNamePerson
inner join Author a on p.IDPerson = a.IDPerson
```

```
on excel.Titlu = b.Title
```

La fel si pentru carti cu edituri :

```
update book
set IDPrintingHouse =
p.IDPrintingHouse from Book b
inner join Excel_SF excel
inner join PrintingHouse p on p.NamePrintingHouse = excel.Editura

on excel.Titlu = b.Title
```

Ramine la latitudinea cititorului exercitiul cu celelalte update-uri.

Backupul la BD il gasiti in folder-ul database si se numeste „lib\_date\_importExcel.bak” . Puteti face restore.

De citit:

1. [Primul pas : instalarea software-ului free](#) , in care downloadati VC# Express si SQL Server Express
2. Stringuri de conexiune : [www.connectionstrings.com](http://www.connectionstrings.com)

[Surse](#)

[Tutorial PDF](#)